

# 551.I

# SOC Design and Operational Planning



SANS

THE MOST TRUSTED SOURCE FOR INFORMATION SECURITY TRAINING, CERTIFICATION, AND RESEARCH | [sans.org](http://sans.org)



# 551.1: SOC Design and Operational Planning

© 2021 John Hubbard and Mark Orlando | All Rights Reserved | G02\_02

Welcome to SANS MGT551: Building and Leading Security Operations Centers!

<b>TABLE OF CONTENTS</b>	<b>PAGE</b>
Introduction	5
SOC Functions	20
SOC Planning	33
Exercise 1.1: Threat Actor Assessment	53
Team Creation, Hiring, and Training	55
Exercise 1.2 – Attack Path Development	88
Building the SOC	90
SOC Tools and Technology	106
Exercise 1.3 – Structuring, Documenting, and Organizing Use Cases	127
Protecting SOC Data and Capabilities	129
Summary and Cyber42 Simulation – Day I	148



This page intentionally left blank.

## Course Overview

### Day 1:

#### SOC Design and Operational Planning

- Understanding the adversary
- Tools and technology
- Hiring and training

### Day 2:

#### SOC Telemetry and Analysis

- Defense theory
- Data collection and monitoring
- Cyber threat intelligence

### Day 3:

#### Attack Detection, Threat Hunting, and Triage

- Alert triage and capacity planning
- Detection engineering
- Threat hunting and investigative process
- Active defense

### Day 4:

#### Incident Response

- IR planning and process
- IR tools
- Dealing with a breach

### Day 5:

#### Metrics, Automation, and Continuous Improvement

- Staff retention and engagement
- Measurement and prioritization
- Strategic planning
- Automation, analytic testing, and adversary emulation



This page intentionally left blank.

## Day 1 Overview

- **Introduction**
- **SOC Design and Planning**
  - Mapping the SOC Functions
  - SOC Planning
- **Building a Strong Foundation**
  - Team Creation
  - Physical Space and Equipment Planning
  - Core SOC Toolset
  - Protecting SOC Data and Capabilities
- **Exercises:** Threat actor assessment, Attack Path Development, Developing and Implementing SOC Playbooks



## Day 3 Overview

Here is a list of topics we will be discussing throughout the first book of this course.

# Course Roadmap

- *Book 1: SOC Design and Operational Planning*
- Book 2: SOC Telemetry and Analysis
- Book 3: Attack Detection, Threat Hunting, and Triage
- Book 4: Incident Response
- Book 5: Metrics, Automation, and Continuous Improvement

## SECTION I

### Introduction

#### SOC Design and Planning

- SOC Functions
- SOC Planning
  - *Exercise 1.1: Threat Actor Assessment*
  - Team Creation, Hiring, and Training
    - *Exercise 1.2: Attack Path Development*

#### Building a Strong Foundation

- Building the SOC
- SOC Tools and Technology
  - *Exercise 1.3: Developing and Implementing SOC Playbooks*
  - Protecting SOC Data and Capabilities
  - Summary and Cyber42 Simulation – Day 1



This page intentionally left blank.

## Welcome to MGT551!

Designed to help you build the SOC of your dreams

- Focused on the right tools, data, people, and process
  - Pairs with *SEC450 – Blue Team Fundamentals*
- In this course we'll cover:
  - SOC planning and building
  - SOC daily operations and processes
  - Detection and Response
  - Continuous improvement
  - Analytic testing and verification



WELCOME



MGT551 | Building and Leading Security Operations Centers

6

### Welcome to MGT551!

Welcome to MGT551! In this course we'll be covering Security Operations Centers from all angles – the people, process, and technology required to run them, the data we care about collecting, the mindset of a modern defender, and the workflows needed to be our most effective. We've designed this course to convey and demonstrate this information in the most efficient way possible, through visual models, explanations, and hands-on exercises that will reinforce each point.

The course is divided into 3 main sections:

- Planning and building a SOC, which will walk through important considerations for getting your SOC up and running
- Operating the SOC, which will discuss the "everyday" tasks such as data collection, threat detection, triage, investigation and incident response
- Continuous improvement and assessment, which will cover how to ensure your SOC continues to grow and can keep up with the attackers who are constantly modifying their attacks, and improving their own tactics.

At the end of this course, the goal is to ensure you have deep understanding of each of the core components of a SOC, and how to establish everything is working at peak efficiency, let's get started!

## Creating Your First SOC?

- Some of you will be creating your org's first SOC
- This course will cover important considerations
  - Documents and Policies
  - SOC Design (physical layout/hardware required)
  - Monitoring infrastructure
  - Hiring and org structure
  - Daily Operations
  - Strategies for building capabilities



**UNDER  
CONSTRUCTION**

SANS

MGT551 | Building and Leading Security Operations Centers

7

### Creating Your First SOC?

If you are in the stages of forming the first SOC for your organization, you have come to the right place. Throughout the course, we will be guiding you through the most important decisions that need to be made, and how to decide what best fits the needs and budget of your organization. We'll cover the documents and policies that need to be in place to give the SOC authority, the physical design of the SOC and what analysts should have at their disposal, org structure, hiring, processes for daily operations, and strategies for building up from zero.

## Improving the SOC You Already Have

- Some of you are looking to improve your existing SOC
- Improving operations is the other course focus
  - Network Security Monitoring
  - Continuous Security (Host) Monitoring
  - Software and Integration
  - Automation
  - Continuous improvement



SANS

MGT551 | Building and Leading Security Operations Centers

8

### Improving the SOC you Already Have

While some students may not have a SOC yet, many of you are likely in a pre-existing SOC looking to further mature its capabilities or improve its effectiveness. To this end, the course will also heavily focus on the monitoring activities that need to occur for network and host-based data, the software and integrations that make SOC life efficient and painless, automation opportunities, and how to ensure the SOC is continuously improving.

## Key Capability – Knowing Your Enemy

- Intellectual property theft
- Customer information
- Extortion
- Destruction
- Hacktivism
- Geopolitical advantage
- Leverage access to attack someone else



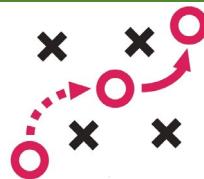
*What all hackers obviously look like*

### Key Capability – Knowing Your Enemy

Attacker motivation is as varied as the groups themselves. Some want your intellectual property, others want to steal your customer data, others may want to extort you, cause destruction, or perhaps only read your email to play the stock market or give them information the government can use to their advantage. There are plenty of times where attackers have even leveraged one company to access another. What they will want from *you*, however, is the question you must answer by doing your own research and producing or acquiring the relevant threat intelligence.

## The Tactics They'll Use to Get It

- (Spear) Phishing
- Watering hole attacks
- Supply chain attacks
- Site cloning
- Zero days
- Weaponized documents
- Ransomware
- Wiper malware
- Rootkits
- Worms
- DDoS
- Scripting / App ctrl. bypass
- Credential Theft
- Lateral Movement
- Encrypted Cmd. & Ctrl.
- Low and slow exfiltration



### The Tactics They'll Use to Get It

Regardless of the goal, you can bet that attackers have no shortage of tricks to get to their goal. For each stage in the kill chain attackers now have amazing open-source tools, let alone what they may create on their own if they are a truly advanced group! Being part of the Blue Team means we must be ready to face any of the challenges above and architect our network, technology, and processes to reject or at least minimize an attempt to disrupt our operations.

## Breaking Down the Average SOC<sup>1</sup>

Micro Focus State of Security Operations category breakdown:

- **Business Alignment** – "Heightened awareness and impact of breaches driving increased understanding of cyber risk by the business."
- **Technology** – "Most organizations continue to focus heavily on technology solutions and tools in their cyber defense program."
- **People** – "Consolidating security operations and related hunt, threat intelligence, and IR functions, ... and attracting, training, and retaining security talent under capable leaders."
- **Process** – "Without a solid foundation of processes and procedures, SOCs become reliant on the 'tribal knowledge' of individuals and less predictable in the results they produce."

### Breaking Down the Average SOC

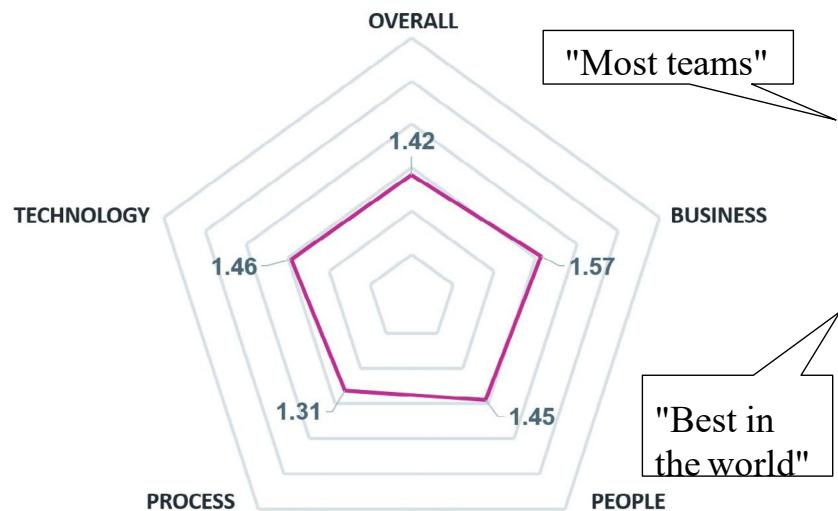
One way of breaking down a SOC is the above categories which Micro Focus<sup>1</sup> uses for their "State of Security Operations" survey. Here's what they have to say about the trends by category:

- Business Alignment – "*Measurements of the Business aspect of the maturity model have increased significantly in the last 3 years, largely due to heightened awareness and impact of breaches driving increased understanding of cyber risk by the business. This has brought increased visibility of and better articulated requirements to the cyber security function.*"
- Technology – "*Technology has traditionally scored high due to the fact that engineering and technology deployment tasks are often the focus in most enterprise security organization's investments in cyber security. Most organizations continue to focus heavily on technology solutions and tools in their cyber defense program.*"
- People – "*The People aspect of the maturity model saw the biggest single-year jump to a 1.45 for the 5-year median, structure of their security teams, consolidating security operations and related hunt, threat intelligence, and incident response functions, developing strategic partnerships with subject matter expert providers, and attracting, training, and retaining security talent under capable leaders.*"
- Process - "*Without a solid foundation of processes and procedures, SOCs become reliant on the 'tribal knowledge' of individuals and less predictable in the results they produce. Turnover of individuals cripples the capability of the SOC that lacks good processes and sets organizations back years. The most capable cyber defense programs around the globe stand out in this area with repeatability, continuous improvement, and metrics that track execution of processes.*"

[1] [https://www.microfocus.com/media/white-paper/state\\_of\\_security\\_operations\\_wp.pdf](https://www.microfocus.com/media/white-paper/state_of_security_operations_wp.pdf)

## Where Is the Average SOC Now?

Overall Median SOMM Score by Dimension Last 5 Years



SOMM Level	Rating
Level 0	Incomplete
Level 1	Initial
Level 2	Managed
Level 3	Defined
Level 4	Measured
Level 5	Optimizing

### Where Is the Average SOC Now?

Where does the average SOC sit in terms of maturity? According to the 2018 Micro Focus, the average measured maturity for security operations teams across five categories was between a 1 and 2, putting them somewhere between the qualitative terms of “Initial” and “Managed”. In practice, this could mean tools that are not well-integrated, a lack of structured process, and people who aren’t trained as well as they could be. Micro Focus positively notes that the averages are trending upwards, but has the following to say about these teams: *“Overall and within each of the areas measured, the cross-industry median capability score continues to fall between a 1 and 2. While SOCs in this range are generally getting the job done, Micro Focus Cyber Security Services continues to observe that a lack of repeatability, metrics, and demonstrated continuous improvement make the effectiveness and sustainability of these cyber defense programs unpredictable across most organizations.”*<sup>1</sup>

Should we be worried that we are nowhere near a score of 5 – “Optimizing?” We will address that question over the next few pages.

[1] [https://www.microfocus.com/media/white-paper/state\\_of\\_security\\_operations\\_wp.pdf](https://www.microfocus.com/media/white-paper/state_of_security_operations_wp.pdf)

## How Do We Enter "Best in the World" Territory?

**Level 3:** "Operations are **well defined, subjectively evaluated, and flexible**. Processes are **defined or modified proactively**.

This is the ideal maturity level for most **enterprise SOCs**."

**Level 4:** "Operations are **quantitatively evaluated, reviewed consistently, and proactively improved utilizing business and performance metrics** to drive the improvements.

This is the ideal maturity level for most **managed service provider SOCs**."

### How Do We Enter "Best in the World" Territory?

So how then do we enter the level 3 and level 4 "best in the world" territory? Looking at how Micro Focus has defined the levels for security operations here's what is viewed as the standard for each level. Notice that level 3 is recommended for enterprise SOCs while level 4 is recommended for managed security service provider (MSSP) SOCs.

- Level 3: "*Operations are well defined, subjectively evaluated, and flexible. Processes are defined or modified proactively. This is the ideal maturity level for most enterprise SOCs.*"<sup>1</sup>
- Level 4: "*Operations are quantitatively evaluated, reviewed consistently, and proactively improved utilizing business and performance metrics to drive the improvements. This is the ideal maturity level for most managed service provider SOCs.*"<sup>2</sup>

This course will focus on the most important tactics and strategies for moving towards a level 3 and 4 type of SOC environment, one where we have defined, repeatable process that is still flexible but is dependable and in a cycle of continuous improvement. What about level 5 though?

1 [https://www.microfocus.com/media/white-paper/state\\_of\\_security\\_operations\\_wp.pdf](https://www.microfocus.com/media/white-paper/state_of_security_operations_wp.pdf)

2 Ibid.

## Taking It Too Far

**Level 5:** "Operational improvement program has been implemented to track any deficiencies and ensure all lessons learned to continually drive improvement. **Processes are rigid and less flexible, and significant overhead is required to manage and maintain this maturity level, outweighing the benefits achieved.**"<sup>1</sup>



### Taking It Too Far

You may be wondering, "Why is level 3-4 the limit, shouldn't we go for full maturity?" Micro Focus has this to say on the top level 5 rating: "*The ideal composite maturity score for a modern enterprise cyber defense team remains a level 3—where the capability is “defined.” This is achieved with a complimentary mixture of agility for certain processes and high maturity for others. Micro Focus SIOC has observed that levels of aggregate maturity above a 3 are more costly to achieve and should be reserved for organizations looking to protect subsets of applications, data, systems, or users.*"<sup>1</sup>

Within the definition of level 5 is the reason that it is not appropriate for many organizations:

Level 5 – "*Operational improvement program has been implemented to track any deficiencies and ensure all lessons learned to continually drive improvement. Processes are rigid and less flexible, and significant overhead is required to manage and maintain this maturity level, outweighing the benefits achieved.*"<sup>2</sup>

The reason is simply that the rigid processes and overhead required to maintain them start to become counter-productive. Also, as we'll discuss later in the class, rigid processes can create a less engaged workforce and may contribute to higher turnover in the SOC, another issue we don't want to create for ourselves.

1 [https://www.microfocus.com/media/white-paper/state\\_of\\_security\\_operations\\_wp.pdf](https://www.microfocus.com/media/white-paper/state_of_security_operations_wp.pdf)

2 Ibid.

## What Top-Performing SOCs Have In Common<sup>1</sup> (I)

Micro Focus State of the SOC 2019<sup>1</sup>, found the best teams have:

- **Clarity** of mission for the cyber defense program
- **Board-level support** and **visibility** into the KPIs of security programs
- **Insight** into the **applications, data, systems, and users** most likely to impact customers
- Immediate **security investments** following direct financial loss
- The **continuity** and **retention** of key security **personnel**
- A **narrower focus** to protect specific assets for the organization

### What Top-Performing SOCs Have In Common (1)

What then makes a fantastic SOC? From the Micro Focus report, these are the items listed as contributing to creating a “top-performing” SOC. Looking through the items on this page and the next will likely not be a surprise, but yet many SOCs struggle to establish these goals. Things like a clarity of mission, board-level support, insight into key data, and retention are some of the most common items that a SOC must cultivate but often struggle with.

[1] [https://www.microfocus.com/media/white-paper/state\\_of\\_security\\_operations\\_wp.pdf](https://www.microfocus.com/media/white-paper/state_of_security_operations_wp.pdf)

## What Top-Performing SOCs Have In Common<sup>1</sup> (2)

- **Strategic partnerships** with niche security service providers
- The use of **automation** for repeatable tasks
- **Leveraging expert consulting** organizations to guide design, development & optimization
- A **willingness to transform** IT culture
- **Advanced integrations** that provide better end-to-end visibility
- **Tools** that are **faster** and **easier** to use

### What Top-Performing SOCs Have In Common (2)

Here is the second set of characteristics that “top-performing” SOCs have in common. From this slide, one of the most prominent and important is automation. Automation can solve many problems on nearly every security team, but many do not have the tools, time, or talent to implement it where needed. Advanced integrations is another spot that can drastically improve the lives of the security team, but getting the integrations figured out and established often takes a back seat to more immediate seeming tasks like alerts and incidents. Expert consulting from both internal and external teams (such as Red Teams) can also make a significant difference in the SOCs ability to assess their defenses. We will discuss these topics in much more detail throughout the course.

[1] [https://www.microfocus.com/media/white-paper/state\\_of\\_security\\_operations\\_wp.pdf](https://www.microfocus.com/media/white-paper/state_of_security_operations_wp.pdf)

## Common Security Operations Center Issues/Trends

### Additional State of the SOC

2020 Findings<sup>1</sup>:

#### Technology

- Tools galore – wide use of 11 common tools with 80%+ adoption
- Widespread (93%+) AI and ML usage used for improving threat detection (not false pos. reduction)

#### Process

- 90% of orgs now using ATT&CK
- "Protect" seen as the biggest challenge from NIST CSF

#### People

- 90% of organizations still say they have shortages in Sec Ops teams
- Most teams looking for **attack detection and analysis** talent

#### Perceptions

- 96% of orgs now using cloud svcs.
- COVID-19 brings serious challenges for security operations teams – increased attack volume and more
- MSSPs used in 87% of orgs for at least one security function
- MSSPs used for 3 functions on average



## Common Security Operations Center Issues/Trends

Some of the key additional findings and trends from the 2020 version of the Microfocus State of Security Operations Survey are listed on the slide above in the categories of technology, process, people, and perceptions. In short, organizations continue the trend of moving to the cloud, buying more tools, and wanting to grow their cyber team's talent, but are still struggling. In trends, MITRE ATT&CK has picked up usage as a standard in 9 of 10 organizations surveyed, MSSPs remain a common way to outsource IT security related processes as is done so for at least one function in 87% of organizations. Check out the report for some great additional detail and discussion of each of these trends and findings.

[1] <https://www.microfocus.com/media/report/2020-state-of-security-operations-report.pdf>

## Goals for This Class

- For new SOCs: Building a solid starting foundation
  - Develop a strategy to build out the SOC
  - Learn from others to avoid common pitfalls
- For all: Move toward "level 3-4" maturity level
  - Focus on reliable detection with systematic approach
  - Implement continuous improvement and useful metrics
  - Utilize automation / orchestration to "do more with less"
  - Develop and maintain talent and an engaged workforce

### Goals For This Class

In this class, we will set our sights towards closing the gap between our current level, and a "level 3-4" SOC. For those who are just starting their journey, we will lay out the path forward with an eye towards developing a solid strategy and avoiding the common pitfalls that new organizations often run into. For everyone, we will drive towards developing a reliable detection capability that can be done in an automated, repeatable, and dependable fashion. This will be done using continuous improvement tactics to ensure we are always growing, and metrics to create a feedback loop to validate we are being as effective as we hope to be. Achieving these goals will require the use of automation and orchestration where possible to "do more with less" as Micro Focus puts it, and implementing strategies to grow and maintain talent, as well as a happy and engaged workforce.

## In This Class

The class is broken into three main sections:

1. Creating your SOC
  - Plan, define, design, build, and hire
2. Execution
  - Collect, detect, triage, investigate, and respond
3. Continuous Improvement
  - Staff retention, metrics and effective execution, analytic assessment, adversary emulation and SOC testing



## In This Class

Over the course of this class, we'll be tackling all the common SOC problems using current best practices and evidence to back up our approach. Security operations can be tough, but you don't need to, and shouldn't, figure it out from scratch. Taking the lessons from the road already traveled by others we'll save you time and money and guide you in building the best security operations center in the least amount of time! Without further ado, let's dive in!

# Course Roadmap

- *Book 1: SOC Design and Operational Planning*
- Book 2: SOC Telemetry and Analysis
- Book 3: Attack Detection, Threat Hunting, and Triage
- Book 4: Incident Response
- Book 5: Metrics, Automation, and Continuous Improvement

## SECTION I

Introduction

### SOC Design and Planning

- **SOC Functions**

- SOC Planning

- *Exercise 1.1: Threat Actor Assessment*

- Team Creation, Hiring, and Training

- *Exercise 1.2: Attack Path Development*

### Building a Strong Foundation

- Building the SOC

- SOC Tools and Technology

- *Exercise 1.3: Developing and Implementing SOC Playbooks*

- Protecting SOC Data and Capabilities

- Summary and Cyber42 Simulation – Day 1



This page intentionally left blank.

## In This Module

Painting the SOC at an abstracted high-level view:

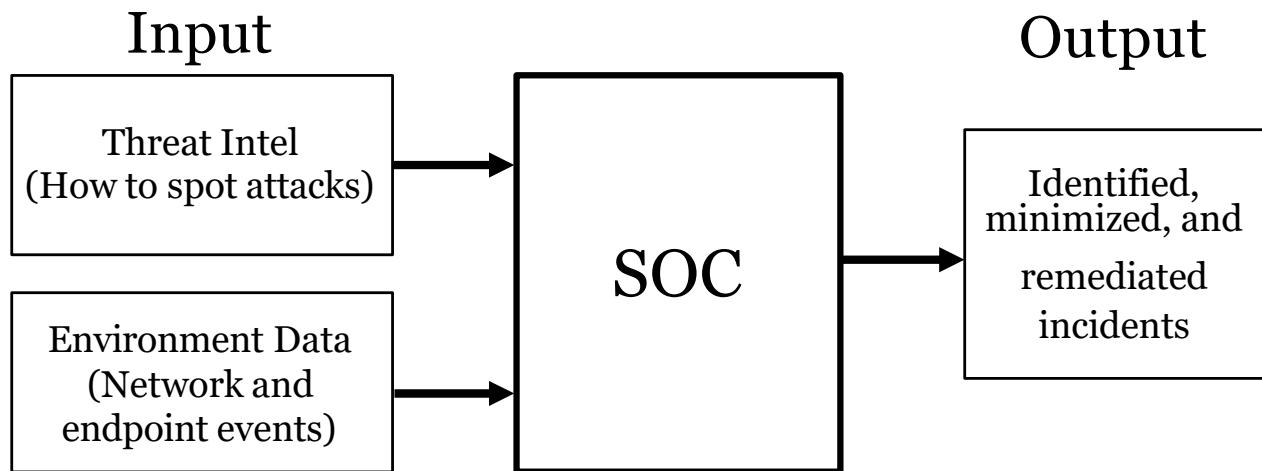
- The functions of a SOC
  - Core SOC activities
  - Interaction with auxiliary groups (threat intel, forensics, ...)
- How the SOC functions interact with each other
  - Inputs
  - Outputs
  - Who is typically responsible
  - Goals



## In This Module

We'll start out in this module doing an overview of the SOC with a simple mental model that breaks down the SOC into an abstracted set of processes with inputs and outputs. As we "zoom in", we'll see each abstracted box can be further broken down into additional functions, each with their own inputs and outputs. Looking at a SOC in this way helps us understand how data flows, and how the inputs from one stage will drive the quality and effectiveness of functions further down the line. In addition, breaking down the SOC from a high-level view helps put everyone in the same mindset and gives us a common terminology we can use to discuss the functions throughout the rest of the course. The goal is to ensure everyone has a clear model in their head of the core functions of a SOC, how they interact, and how they tie in to auxiliary functions that are often associated with the SOC, but may not be directly part of the group (threat intelligence, forensics, penetration testers, etc.).

## The SOC at the Highest Level



Better output requires better input, “*garbage in, garbage out*”

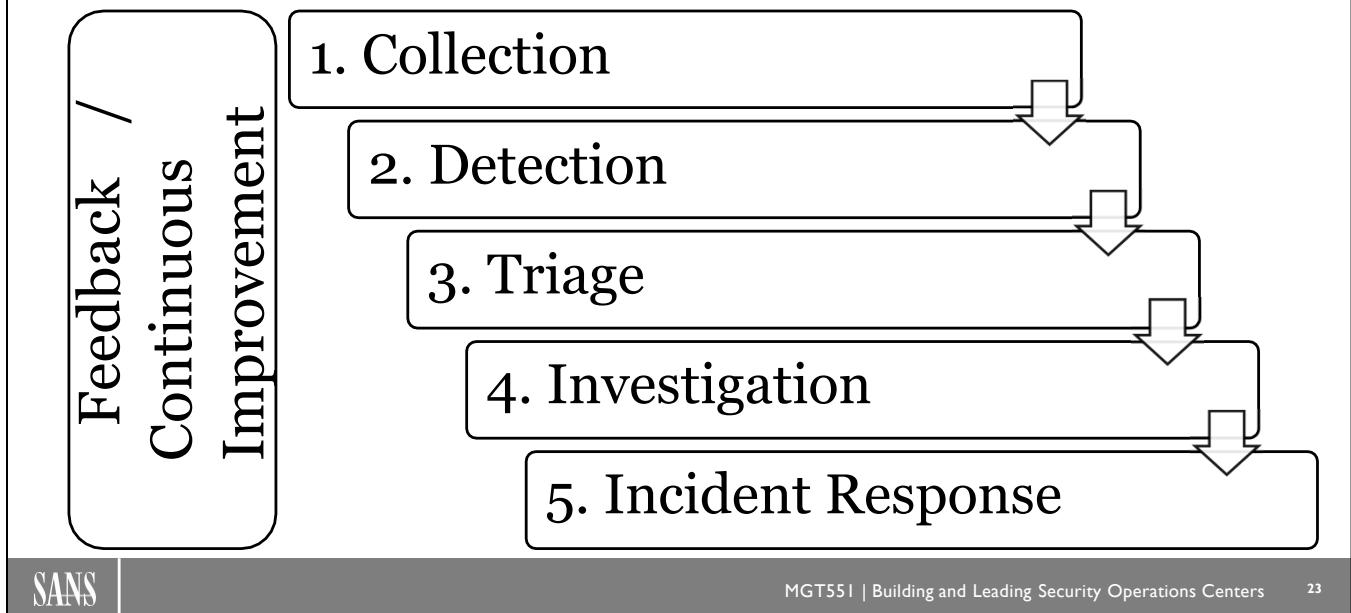


### The SOC At the Highest Level

If we take the SOC and zoom all the way out, consider what is happening at the most basic level. As the main *input*, the SOC takes information about what is occurring in the environment – network transactions, file downloads, authentications, running processes, patch levels, configurations, and more, and collects it. The data that is collected is then compared against another set of data that may be considered the second input, threat intelligence. In simplistic terms, threat intelligence can be considered "what an attack looks like" (of course it is much more than this, but at this level of abstraction, one key role it plays is helping use initially identify attacks). This may be done at an atomic level, such as looking for specific file hashes or domain name access, or at a higher level of abstraction, such as patterns of login attempts or user file access activity. Given what an attack may look like, and what is actually happening in the environment, in theory the SOC should be able to identify any potentially malicious activity occurring in the environment and can act on it.

The *output* of the SOC is the remediation of all identified problems. Specifically, we can think of it as identified, minimized, and remediated incidents. Therefore, the SOC at this abstracted level is a function that takes environment data and attack signatures, identifies attacks, and fixes the issues such that they ideally cause minimal disruption to the organization. Fixing in this sense may include both at the individual incident level, as well as initiating communication to stakeholders of the affected systems to help solve the root cause. To understand how it does this in more detail, we must zoom in on the SOC piece of the picture to understand its inner workings.

## High Level SOC Functions



SANS

MGT551 | Building and Leading Security Operations Centers

23

### High Level SOC Functions

The activity a SOC must perform in order to successfully operate can be logically broken down into the steps shown above: collection, detection, triage, investigation, and incident response, (plus a conceptual continuous improvement piece). These steps cover the SOC taking the things that are happening in the environment (collection) and monitoring them for suspicious activity (detection). Any identified activity should be put into a queue (triaged) for closer analysis and verification (investigation). Any confirmed malicious activity is then passed off to the incident response function for remediation.

Since the output from each step is the input of the next, each function cannot work effectively unless it is being fed good data – "garbage in, garbage out" as the saying goes. Detection cannot occur if items are not collected for example. Since attackers change tactics daily however, none of these are "set it and forget it" type activities, therefore, continuous improvement and adjustment is required for all steps, which is why it is listed as well. Each function needs to be subjected to a constant stream of feedback from the other stages to continuously improve and adjust to the shifting requirements required for attack identification. Looking at the process of security operations this way helps us deconstruct each individual piece and show the factors that are required for success, and the inputs, outputs, and roles required for each item. Let's take a look at how these steps appear in a process diagram format and how the auxiliary functions tie in.

## Core SOC Activities vs. Auxiliary Functions

### Core SOC Activities

- **Data Collection:** What's happening on the network / devices
- **Detection:** Identifying items of interest from data collected
- **Triage and Investigation:** Confirming and prioritizing detected issues
- **Incident Response:** Responding to and minimizing the impact of attacks

### Specialty / Auxiliary Functions

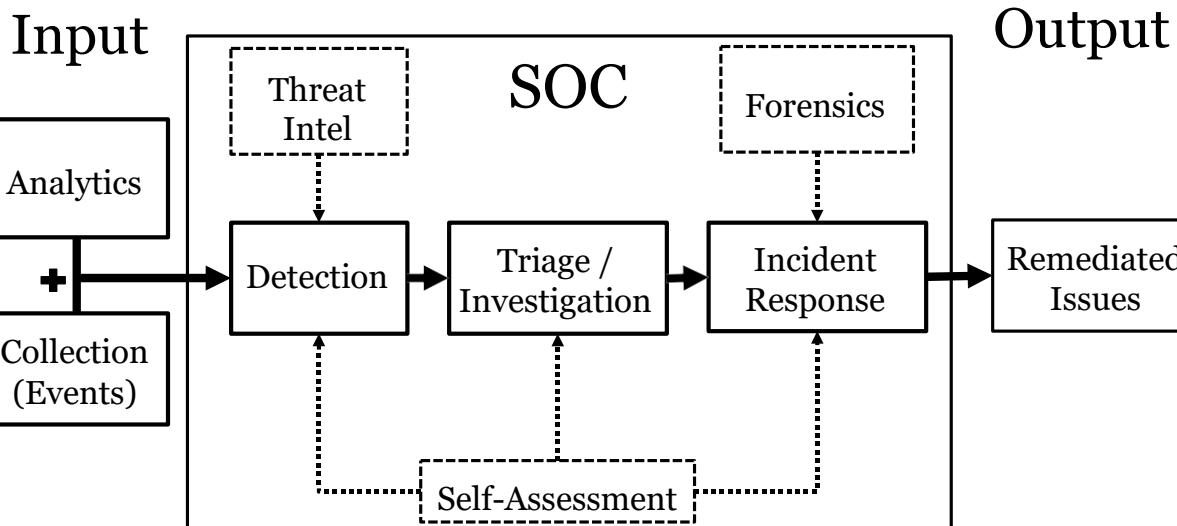
- **Threat Intelligence:** Collecting information to improve attack detection
- **Forensics:** Supporting IR with deep research and reverse engineering
- **Self-Assessment:** Vulnerability assessment, Penetration Testing, Red Teaming, Inventory, etc.

### Core SOC Activities vs. Auxiliary Functions

Considering the breakdown of responsibility of the functions listed on the previous page, most of the time the items listed under the SOC functions are performed by SOC engineers, analysts, and incident responders (which may or may not directly fall under the SOC org), which is why on this page they are listed under the "core SOC" category.

What about the other specialty functions that need to feed and support these activities, however? In this slide, we have a breakdown of the core SOC activities as listed in the previous slide (the ones typically done by those inside the SOC organization), and the auxiliary functions that ensure they can do the best possible job. This includes things like **threat intelligence**, which in larger organizations is often a dedicated group considering the significantly different nature of the job vs. being a SOC analyst. In addition, there are functions like **forensics**, which is often a specialty category that can help incident responders deep-dive into malware or extract the history of activities that occurred from an imaged hard drive. Then there is the **self-assessment** function, which is an umbrella term for all of the items that help give feedback to the SOC on how they're doing, the status of the environment, and ensuring they stay on their toes. This can be thought of to include everything from vulnerability assessments to penetration testing to Red Teaming, configuration monitoring, asset inventory, and more. All of these play a vital role in feeding the SOC information about what's on the network, and how it can be better.

## Core SOC and Auxiliary Activities Diagram



### Core SOC and Auxiliary Activities Diagram

This page shows the internals of what we put in the simple "SOC" box on the earlier high-level SOC slide and how each function connects to each other, as well as interfaces with the dashed lines of the auxiliary functions. Data is collected and compared with any signatures and analytics (tactical threat intelligence) that have been created based on threat intelligence. This information, along with any internally generated threat intelligence, drives the detection capability. All detected items are then passed to triage and investigation where they are verified and handed off to incident response. The incident response function is assisted by any specialty forensics capacity that is required, and ultimately remediated or prevented issues are the output. The self-assessment box is shown here as an input to detection, triage, investigation, and incident response since it continuously feeds and tests these capacities in various ways.

Thinking of a SOC in this way makes it clearer how the output of one function will affect the ability of the next step to function. If detection is failing for instance, triage and investigation will not produce good results, and if our investigations are done poorly, incident response may never see the data they need to act on to stop an attack in progress. It also lays out in a clear form the auxiliary functions that have inputs to different pieces of the core SOC functions (this is not to imply these functions are necessarily part of the SOC organization however). Of course, since this is a simplification not everything can be perfectly represented. There is nuance here that is not depicted, but for keeping a common model in our head we can work with throughout the class, this picture of how a SOC operates will be useful. Over the next few pages, we'll walk through each of the core SOC functions and talk about their specific inputs and outputs.

## Collection

- **Input:** Device activity
- **Output:** Events (logs, network traffic, metadata, etc.)
- **Responsible:** Data/Infrastructure/Endpoint Engineers
- **Goal:** Collection of activity of interest



### Collection

The first step in the process is collection. This step involves taking the activity that is occurring in the environment - everything from logins to web transactions and more and converting it into a recorded event. The output of this collection is events which may be logs, network traffic, metadata, or other derived information about what occurred on a given device or network segment. Data turned into events is typically generated either on an endpoint device or gathered from a tap on network traffic. This data is then centrally collected via an endpoint agent or sensor that can parse the traffic and metadata about it to the SIEM for indexing. It is the output events (logs and traffic metadata) that are then analyzed against the employed set of detection rules. The goal of this stage is thorough collection of all *security relevant* data that can then be used in the next stage for detection analytics.

The type of data you can record is determined by data, infrastructure, or endpoint engineers as its collection requires setting up device auditing and collection points throughout the enterprise. Windows logs that are generated, for example, are decided by the audit policy set in Windows Group Policy Objects, network taps will likely require cooperation with the network operations team. Which specific data *is* collected should be informed by your threat intelligence function letting you know what data is required to detect attacks. In most SOCs, a thorough collection strategy will require cooperation across multiple teams and budget for the hardware and software needed to meet visibility requirements.

## Detection

- **Input:** Events (network and endpoint) + threat intel
- **Output:** Alerts
- **Responsible:** Detection/Content Engineering, SOC Analysts, Threat Hunters
- **Goal:** Identification of potentially malicious events—either manually or automatically generated



### Detection

From the collected data, all items of interest that indicate potential attacks must be identified. This happens two main ways, either by automated analytics engines in the SIEM, network, or endpoint sensors, or manually by analysts searching through the data. In larger organizations, these analytics or correlation rules are typically made by a dedicated detection engineering team, in smaller teams, analysts will be required to do this role in tandem with the rest of their duties. The goal of this stage is to find all malicious activity and get an alert related to it into the triage queue for action with a minimum number of false positives.

The effectiveness of a SOC in this stage is correlated with the quality of the tools employed as well as the strength of the SOC's threat hunting capabilities, threat intelligence information (both feeds and internally generated intelligence), and detection engineering functions. Successful detection naturally relies on the data being available from the collection stage, the better your collection function the better your detection function can be and the same is true of the quality of signatures produced by threat intel and analytic engineers.

## Triage

- **Input:** Alerts
- **Output:** Ranked alerts / highest priority
- **Responsible:** SOC Analysts
- **Goal:** Identification of most important/dangerous alerts, management of queue size



### Triage

Once the detection stage has generated alerts on the events of interest, these alerts are all forwarded to one or more queues for triage. This is where the job of the typical SOC analyst starts. In this stage, analysts must sort through all the potentially malicious activity that has been detected and make an intelligent determination of which alert seems to be the more pressing at the moment. These decisions are often based on factors such as how far the attack may have already progressed, the criticality of the system being attacked, the privilege of the account that may be compromised, and/or whether it appears to be a unique or targeted attack. Similar to an emergency room, the analyst's goal in this stage is to correctly queue up items to be investigated in a logical priority order given the data they are presented. Effective analysts do this combining their knowledge of concepts such as the Lockheed Martin Cyber Kill Chain, attacker TTPs such as those in the MITRE ATT&CK framework tactics and techniques, and their previous defense experience. Once alerts are prioritized, the top item is chosen for investigation and validation in the next stage.

## Investigation

- **Input:** Chosen highest priority alert
- **Output:** Confirmed attack/incident or false positive identification
- **Responsible:** SOC Analysts
- **Goal:** Accurate and fast verification of alert content as true or false positive, escalation of



### Investigation

Once the most seemingly dangerous item is selected, SOC analysts will dive in and investigate the alert to see if it truly is something bad going on. As many SOCs suffer from overactive and untuned alerts, this can be a step that often leads to a false positive determination and dismissal of the alert. To get to the truth of what occurred, an analyst must often take the data available from the alert and put it into context, perform additional research, and see if it's what the alert claims to have found is indeed occurring. This may involve gathering data from other sensors, logs from other sources in the SIEM, or open-source intelligence research. The goal of this stage is the accurate verification of whether an alert is a true or false positive. When false positives do occur, it's wise for analysts to dig into the cause of the error and feed that information back to the detection engineering team to correct the alerting rules.

Analysts should be trained to perform the investigation stage in a rigorous way, free of cognitive bias and errors in analysis. While this may seem straightforward and intuitive, there are many traps that new analysts fall into at this stage, the main one being confirmation bias. Many analysts like to match the alert with what they think is happening, then go off to gather the data to show they are correct. This is the wrong way to approach the problem as it ignores the multitude of other scenarios that could have produced the same evidence, and thus can lead to incorrect diagnosis. To verify a theory, analysts should attempt to *disprove* what they believe is going on. If they are unable to, it's a better reason to believe they are correct than online finding confirming evidence. Doing this step well, like all others, takes thorough training and improves with experience. Those interested in training analysts on the details on these steps are encouraged to check out SANS SEC450: Blue Team Fundamentals – Security Operations and Analysis, where thorough investigation is one of the main topics of the class.

Once the investigation is complete, if necessary, analysts must spin up the incident response capacity to take over if remediation, forensics, and cleanup are necessary. In large organizations, incident response may be a separate team, in smaller orgs, like detection, this may fall under the responsibilities of the analysts as well.

## Incident Response

- **Input:** Confirmed attack/incident or false positive identification
- **Output:** Remediated incidents, feedback to collection/detection process
- **Responsible:** Incident Response / SOC Analysts
- **Goal:** Fast, complete remediation and recovery from incident, root cause identified



### Incident Response

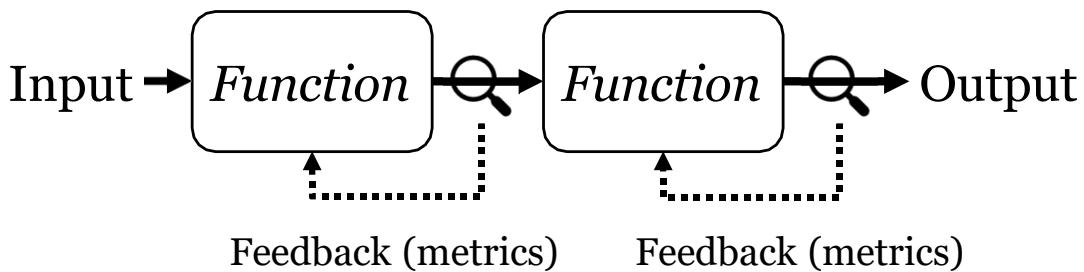
Incident response (either as a dedicated group in larger organizations or as a logical function in smaller ones) receives alerts that have been investigated and shown to be a true problem. The goal of this stage is to, as quickly as possible, scope the problem, contain, eradicate, and ultimately recover from the issue. Solutions such as EDR and centralized SIEM logging help analysts and incident responders quickly query events from endpoints and the network and put together a picture of what has occurred and the extent and timeline of the issue.

The outputs of incident response are both the remediation of the incident and the lessons learned. This step, although frequently skipped, provides crucial feedback to the collection, detection engineering, and threat intelligence functions. The goal of the feedback is to ensure a similar incident never happens again and is also detected immediately if an attempt is made. Output of details to the threat intelligence function tracks the adversary behind the attack (if known) and the tactics used. Tracking these details over the long term helps give the SOC a tactical and strategic advantage in subsequent attacks.

## Visibility For Continuous Improvement

Each function must strive to continuously improve

- Feedback loops sample the output of **each function**
- Uses findings to continuously adjust the input



### Visibility For Continuous Improvement

The final piece of the puzzle that was not depicted in the overall diagram is the need for continuous improvement implemented through feedback. Conceptually, feedback is information from the output of a function feeding back to modify the input, adjusting it as required. This conceptual feedback, in reality, is provided by the metrics we produce within the SOC. The more metrics (output measurements) we have from each point throughout the process, the more understanding we will have about the health of the process as a whole, and the better we will be at adjusting the correct pieces of it. When we get back to metrics later in the course keep this in mind as it is a key concept.

Here's an example to drive this idea home. Imagine a car manufacturer where the only measured metric was the output of the whole manufacturing process - "the percentage of cars that roll off the production line in working order". Would this be a good idea? Of course not, they'd have terrible results, and finding issues would be impossible. Instead, for good manufacturing, each individual step on the assembly line is measured and sampled, and each step is individually optimized to ultimately produce the highest yield possible from the whole process. When it comes to metrics and the SOC process, we must think of the SOC functions in the same way. Producing and collecting metrics from each stage will help us modify each function individually. Therefore, one of the goals of useful metrics collection will be to sample and understand each individual function so that each piece can be independently adjusted and optimized, producing the best overall output for the system.

## SOC Functions Summary

- **Core SOC** activities can be broken down into five functions
  - Collection, detection, triage, investigation, and incident response + continuous improvement at each stage
- Each activity is assisted by **auxiliary groups**
  - Threat intelligence, forensics, self-assessment
- **Metrics** provide **feedback** that helps us optimize each individual step

### SOC Functions Summary

In this section we've laid out a mental model for the SOC process including the core functions as well as the interactions with the auxiliary groups that help measure, improve, and support the SOC. If you've never thought of the SOC in this way before, hopefully, it helps clarify some of the individual processes that will need to be attended to and forms a useful framework for you to consider your operations.

Through the rest of the course, we will be discussing the components of each of these functions in more detail and laying out how we can measure their individual outputs with KPIs and metrics that help ensure that each step in the process is working to its highest capacity. As mentioned in the beginning, a process can only perform well when provided high-quality inputs. Therefore, in an abstract sense, driving high-quality and efficient SOC operations is all about considering each step in the process and optimizing the inputs such that the best possible output can be provided to the next function.

# Course Roadmap

- *Book 1: SOC Design and Operational Planning*
- Book 2: SOC Telemetry and Analysis
- Book 3: Attack Detection, Threat Hunting, and Triage
- Book 4: Incident Response
- Book 5: Metrics, Automation, and Continuous Improvement

## SECTION I

### Introduction

#### SOC Design and Planning

- SOC Functions
- **SOC Planning**
  - *Exercise 1.1: Threat Actor Assessment*
  - Team Creation, Hiring, and Training
    - *Exercise 1.2: Attack Path Development*

#### Building a Strong Foundation

- Building the SOC
- SOC Tools and Technology
  - *Exercise 1.3: Developing and Implementing SOC Playbooks*
- Protecting SOC Data and Capabilities
- Summary and Cyber42 Simulation – Day 1



This page intentionally left blank.

## In This Module (1)

- Do you need a SOC?
- SOC Mission, Requirements, and Goals
- SOC Standards and Policies
- SOC Roles and Staffing Levels
- SOC Constituency and Steering Committee
- Defining Services and Capabilities
- SOC Charter



### In This Module (1)

In this module, we'll cover some of the main planning and preparation items you will need to consider when either building your SOC or improving the one you have. We'll first start with a discussion of whether you are most likely to find the greatest value in a full internal SOC, or whether going with an MSSP or hybrid model might help you get what you're looking for more easily. We'll also cover what should and shouldn't be considered a SOC, as occasionally the term "SOC" is thrown around in inappropriate ways.

After the discussion of what a SOC is and what type of model you might want to pursue, we'll dive into the discussion on defining the mission, constituents, and capabilities of your new SOC. This includes defining standards, policies, services offered, SOC roles and staffing levels, as well as making a charter and steering committee.

## Do You Need a Dedicated Internal SOC?

How many endpoints/users/assets must you protect?

Users/Endpoints	Common Solution
0 – 1,000	MSSP + non-dedicated internal security team
1000 – 10,000	MSSP Hybrid with some functions in-house
10,000 – 100,000	Full internal SOC with possible outsourcing of specialty functions
100,000+	Full-fledged internal SOC with auxiliary/specialty services

### Do You Need a Dedicated Internal SOC?

One initial question to consider when setting out to form a SOC is the model you'd like to pursue. While any company *could* create a dedicated internal security group, the number of employees your organization has and assets they must protect will determine if this is the right move, or if you could be more cost-effective with another model.

In general, the smaller the organization, the less likely it is there will be a need for a dedicated security group with the sole job of monitoring and defending the network. The headcount and hardware, and, therefore, budget required to start such an operation for a small company can make the dedicated SOC option cost-prohibitive for the smallest companies. The emergence of excellent open-source tools have alleviated the cost somewhat, but in an organization with only a few hundred people, having a dedicated security team can still be difficult to justify. Of course, if your organization deals with regulatory compliance issues, health, or payment card transaction data, you may have no choice. Therefore, at the <1000 people or asset level, companies often utilize managed security service providers to help split the cost of a full security team with other organizations, knowing that they are small enough that they shouldn't need a dedicated full-time team. In these organizations, you may have some internal employees who will take on a security role when needed, but that may be as far as it goes.

For larger organizations with roughly 1k-10k employees, it starts to make sense to have dedicated security personnel, although a whole team may not be needed. At this level, you may consider using a hybrid model where an MSSP does the bulk of the alert monitoring and operates security tools for you but escalates dangerous items to in-house specialists for incident response and remediation.

Above 10k people and assets, you get into the realm where a full-fledged SOC is possible with most roles covered in house, and an MSSP may no longer be cost-effective. At this stage, you *may* still consider outsourcing certain things like hard drive forensics, malware reversing, specialized Red Teaming, or other niche tasks, but are likely to be taking on security fully on your own. At 100k+ sizes, companies are often fully self-sufficient and have the budget and headcount to justify having core SOC functions as well as the auxiliary functions performed in-house. Of course, these numbers are very rough estimations, but reflect sizes the author has seen throughout the industry. The real cutoff point to determine which model is most appropriate will depend on the nature of your business and the risk appetite and budget of your organization.

## Planning a SOC Overview

- Defining the **Mission and Goals** (What/Why)
  - Why are we making a SOC? What are our goals?
- Threat Modeling: Starting to understand your adversaries
- **Requirements:** Regulatory, standards, policies, etc.
- Defining the **Constituency** (Who/Where)
  - What is our scope?
  - Which assets, accounts, and environments are we in charge of protecting?
- Defining the **Capabilities** (How)
  - What specific **services** will we offer?

### Planning a SOC Overview

Over the next few pages, we'll discuss some of the main pieces of building a SOC.

1. Mission: Defining the goals of the SOC
2. Threat Modeling: Understanding who your adversaries are, and what they might do to cause harm to your organization
3. Requirements: What external regulatory requirements will be imposed on your operation? Are there any standards frameworks you will be audited against? Does your company have any specific policies you must adhere to?
4. Constituency: What is the scope? Which users, assets, networks, environments, or business units it will serve? Is this an internal or external SOC?
5. Capabilities: What specific services will the SOC offer to the constituency, and what will be outsourced to an MSSP or specialists?

## Defining Your Purpose

- Gain clarity before you begin...
  - **Define:** Why are you creating a SOC? What are the goals?
  - Is this a **security SOC** or a **compliance SOC**?
  - What types of attackers and attacker goals are we aiming to stop?
- Most security-focused SOCs have similar goals
  - Maintaining **situational awareness** of threat landscape
  - **Monitoring events** network and endpoint
  - **Preventing or minimizing impact** to your organization due to cyber security incidents

### Defining Your Purpose

Before making plans, let's take a step back and consider what the mission of your SOC will be. Undoubtedly there will be some specifics related to your organization based on the size, budget, and capacity of your SOC, but the mission of the average security-focused SOC is similar. Being clear on what exactly we are looking to accomplish will help clarify the vision in future steps. As Albert Einstein supposedly said - " If I had only one hour to save the world, I would spend fifty-five minutes defining the problem, and only five minutes finding the solution."

So, what then should the average SOC be looking to do? While producing your list of goals, here are a few items you will likely want to include:

- Preventing or minimizing damage and disruption due to cyber incidents – Ideally, a SOC prevents every attack that it can. When bad things happen, however, and we must assume they will, the SOC's job is to run down the ground truth of what occurred, help clean up the problem, and return the affected people and assets to working order as quickly as possible.
- Monitoring network and endpoint events for suspicious activity – This is the "detection" piece of the SOC process, one of the biggest focuses of any SOC.

Maintaining situational awareness of the threat landscape – SOCs exist to keep the company safe and minimize damage, and you can't do that without keeping your finger on the pulse of the security industry. Therefore, situational awareness of both the internal and external security environment should be a key goal of any SOC.

The output of this phase can be considered a "mission statement" of sorts for your SOC.

## Threat Modeling

Do you know the answers to the following questions<sup>1</sup>:

1. *What do I have that is worth protecting?*
2. *Who do I want to protect it from?*
3. *How likely is it that I will need to protect it?*
4. *How bad are the consequences if I fail?*
5. *How much trouble am I willing to go through to prevent these consequences?*



*"Every battle is won before it is fought."*  
- Sun Tzu

Where do most of these answers come from?

- Threat Intelligence

### Threat Modeling

When operating your SOC it's easy to get caught up in the whirlwind of the day-to-day fight and lose track of the big picture. You have likely seen the Sun Tzu quote before, but that's for good reason – it's true! Giving yourself the best chance of winning a battle is all about preparation, understanding your enemy, and optimizing the resources you have at your disposal to face that threat. If you have not thought very carefully about this, you're putting yourself at a disadvantage from the start.

To be truly effective at defense requires starting with the best possible threat model you can produce.

- Who is the enemy?
- What do they want?
- How will they get it? Etc.

These are questions *everyone* in your SOC should think about daily so that we do not "miss the forest for the trees". These questions on threat modeling from the Electronic Frontier Foundation<sup>1</sup> inspire big-picture thinking and help ground and remind us of why we're doing what we're doing. They also direct us towards developing a strategic and tactical advantage over your adversary, which just so happens to be the entire point of threat intelligence, the function that can help us answer these questions as thoroughly as possible.

[1] <https://www.eff.org/document/surveillance-self-defense-threat-modeling>

## SOC Requirements

### What parameters must your SOC work within?

- Compliance frameworks
- Standards and frameworks
- Company policies
- Service levels
  - Monitoring – 24 x 7 x 365? 9 x 5? Somewhere between?
  - SLOs – Have you defined them?
  - SLAs – Will you have them?

SANS

MGT551 | Building and Leading Security Operations Centers

39

#### SOC Requirements

Once you have a broad sense of the overall mission it's time to look at the specific requirements we may be bound to following due to the nature of your organization or the data you work with. This may include regulatory compliance frameworks that will drive specifics of your monitoring or data handling, standards or frameworks you will be audited against that will compel certain types of reporting, or company policies that affect how you operate. In light of these specifics, consider the level of service you hope to achieve. Will you be running a 24 x 7 x 365 operation or just a 9 x 5? Within that timeframe, will you want to set up a service level agreement, have you considered service level objectives? Over the next few pages, we'll discuss compliance, standards, and policies in more detail to help you flesh out what you might need.

## Regulatory Compliance Frameworks

Are you bound by any regulatory compliance frameworks?

- Consider:
  - What are the monitoring requirements?
  - What are the data privacy and storage requirements?
- Common compliance frameworks that affect infosec:
  - Data privacy – **GDPR / Privacy Shield**
  - Payment Card Industry Data Security Standard (**PCI-DSS**)
  - Health Insurance Portability and Accountability Act (**HIPAA**)
  - US Government Contractors – NIST **SP800-171**
  - Sarbanes-Oxley (**SOX**), Gramm-Leach-Bliley Act (**GLBA**)

### Regulatory Compliance Frameworks

Is your organization bound by any of the listed compliance frameworks? (Hint: The answer is almost certainly yes given the ever-growing list of national and state-level privacy laws in the US and beyond). If so, staying within the law will be one of the primary items the SOC must be concerned with. Each of these compliance frameworks comes with their own mandates on how data, monitoring, and breaches must be handled and a failure to comply can potentially result in hefty fines or even jail time! If you will be held to these standards you should seek the advice of experts, legal counsel, and anyone else required to ensure you are meeting what is required.

Some of the most commonly encountered compliance frameworks are listed below:

- Privacy Related
  - General Data Protection Regulation (GDPR) – Requires businesses protect EU citizen personal data
  - Privacy Shield – Regulations for trans-border data movement between the European Union, United States, and Switzerland.
- Payment Card Industry Data Security Standard (**PCI-DSS**) – Created to protect credit cardholder data, PCI-DSS requirements will apply to most any systems this data touches and specifies policies, controls, devices, and monitoring that must take place to secure the data from theft.
- Health Insurance Portability and Accountability Act (**HIPAA**) – Developed in 1996 in the United States for protecting anyone dealing with health care information, this regulation contains privacy and security rules that must be applied to anyone considered a "covered entity". Most items relevant to information security are found under Title II under the "Security Rule" and "Privacy Rule".
- Protecting Controlled Unclassified Information in Nonfederal Systems and Organizations (**NIST 800-171**) – For those dealing with US government Controlled Unclassified Information (CUI). This standard lists recommended security requirements for protecting CUI resident on nonfederal systems

- Sarbanes-Oxley Act of 2002 (**SOX**) – SOX dictates a set of compliance mandates for publicly traded companies which contain requirements around security controls and risk assessment
- Gramm-Leach-Bliley Act (**GLBA**) – Applies to financial institutions and mandates consumer data confidentiality and integrity protection as well as requires informing customers of privacy policy information

## Controls Frameworks and More

### Control Frameworks – Identify/assess security controls

- CIS Top 20 Critical Security Controls (V7.1+ "implementation groups")
- NIST 800-53

### Program Frameworks – Structure the security program

- NIST Cyber Security Framework (CSF)
- ISO 27001

### Risk Frameworks – Approaches for assessing risk

- NIST SP800-30, SP800-37, SP800-39
- ISO 27005, CIS RAM, FAIR
- **SOC Assessment – SOC-CMM**

Awesome resource!

### Controls Frameworks and More

There are numerous cyber security frameworks you can leverage as additional guidance for building out your SOC or security program at large. The most common ones break down into controls frameworks, program frameworks, and risk assessment frameworks.

Controls frameworks exist to give you a list of baseline controls and items you should be implementing as a best practice in order to mitigate cyber intrusions and a great way to start off planning what your SOC will need. For established SOCs, they assess where you fall in terms of technical capability and help prioritize where you should use your budget. Controls frameworks include lists like the Center for Internet Security Top 20 Critical Security Controls<sup>1</sup> and NIST SP800-53<sup>2</sup>, both of which provide a list of controls that can help you deploy the right type of protection for the priorities your organization has, and the threats you expect to face. The CIS version 7.1 now includes guidance for controls required broken down into three "implementation groups"—recommendations on which controls you should feasibly be able to implement, based on the size of your organization.

If you are building out your SOC as part of a new security program at large, you also may want to investigate the other listed program and control frameworks listed below. These can help give you best practice for structuring and operating your security program, as well as give a standard way of assessing risk throughout your organization.

Program Frameworks:

- NIST Cyber Security Framework – <https://www.nist.gov/cyberframework>
- ISO 27001 – Information Security Management - <https://www.iso.org/isoiec-27001-information-security.html>

Risk Frameworks:

- NIST SP800-30 – Guide for Conducting Risk Assessments - <https://csrc.nist.gov/publications/detail/sp/800-30/rev-1/final>
- NIST SP800-37 – DoD Risk Management Framework - <https://csrc.nist.gov/publications/detail/sp/800-37/rev-2/final>

- NIST SP800-39 – Managing Information Security Risk: Organization, Mission, and Information System View - <https://csrc.nist.gov/publications/detail/sp/800-39/final>
- ISO270005 – Risk Management Standard - <https://www.iso27001security.com/html/27005.html>
- CIS Risk Assessment Method - <https://learn.cisecurity.org/cis-ram>
- FAIR (Factor Analysis of Information Risk) - <https://www.fairinstitute.org/fair-risk-management>

SOC Assessment:

- SOC-CMM - <https://www.soc-cmm.com>

Since you are building out a SOC, the *controls*-centric frameworks will likely be the most useful initially, unless you are simultaneously building the entire security program. Familiarize yourself with these frameworks and for the ones that match your goals, use them to guide your SOC roadmap. Using a well-established framework and list of controls based on industry consensus is a sure way to start off on solid footing. If you're looking for additional guidance and tools for SOC maturity assessment, technologies, and processes, the SOC-CMM by Rob van Os is another excellent free resource<sup>3</sup>. Rob has created an in-depth tool for assessing your SOC capabilities in many different areas and can help you get a grasp on critical gaps in your defense capabilities, it is a "do not miss" resource for anyone managing security operations.

- 1 <https://www.cisecurity.org/controls/cis-controls-list/>
- 2 <https://csrc.nist.gov/publications/detail/sp/800-53/rev-4/final>
- 3 <https://www.soc-cmm.com>

## Cyber Security Policies

- First, gather and analyze all pre-existing org. policies
- Develop any newly required policies
  - Guide with your org's **risk appetite**
- Pre-existing policies / templates you can leverage:
  - **SANS**<sup>1</sup> (<https://www.sans.org/information-security-policy/>)
  - **Information Security Policy Made Easy**<sup>2</sup>
  - **Universities:** Virginia Tech<sup>3</sup> (<https://it.vt.edu/resources/policies.html>)



### Cyber Security Policies

Beyond regulatory requirements and standards, your organization itself may have policies you must adhere to, or you might need to create them on your own. The first step in this stage is a review of current company policies to see which will apply to the SOC and which the SOC may take over enforcement of in its new position. If you are creating the first SOC for your organization, you may find there is a need to create new policies around cyber security that don't yet exist. A suggestion that can help make this easier is to leverage policies that have already been created and can be used as templates. The footnote references are some resources that contain pre-made IT policy templates that may be of use.

1 <https://www.sans.org/information-security-policy/>

2 <https://informationshield.com/products/information-security-policies-made-easy/>

3 <https://it.vt.edu/resources/policies.html>

## Defining the SOC Constituency

Gain clarity on exactly who will your SOC be protecting:

- Potential groupings
  - Users
  - Assets
  - Networks
  - Environments
  - Geographies
  - Business units
  - Organizations



### Defining the SOC Constituency

The SOC's constituency is the set of users, assets, networks, or organizations your team will be responsible for securing. This group should be well defined so that everyone in the SOC can clearly understand what does and does not fall under their scope of protection. Consider the size and capacity of your team. The services that you select (discussed on the next page) are what your SOC will be responsible for providing to the constituents you have defined.

## Steering Committee

- Group of key stakeholders from the constituency
- Provides a two-way communication channel
- Directs the SOC towards the right activities
- Ensures business alignment
- Good for initially forming the SOC and for regular check-ins



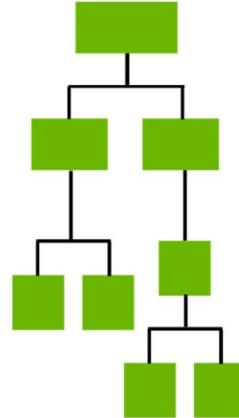
### Steering Committee

Once the constituency is defined, key stakeholders from within it should be invited to be part of a SOC steering committee. This committee can gather to take input from different groups, executive SOC sponsors, and business areas to ensure the SOC truly understands the systems it will be protecting, and the highest risk scenarios associated with them.

Through meetings, while planning the SOC as well as periodic check-ins, the steering committee provides a key two-way communication channel between the SOC and the rest of the business. Meetings with the steering committee should be held regularly, or, at a minimum, called when there is any significant change in the business, new large project, or new cyber security-related risk that is introduced. These will ensure expectations are correctly set and that the SOC is recognized for the value that it brings to the table. A SOC that is not visible to the rest of the business and does not make its contributions clear is a SOC that may not exist much longer.

## SOC Roles

- What roles do you plan to staff? When?
  - Analysts – Tiers?
  - Manager
  - Engineer
  - Architect
  - System administrators
- Which roles are needed immediately
  - Which can be hired in a “phase 2”?
- Consider your planned reporting structure



### SOC Roles

Considering the SOC you plan to create, which roles will be required, and how many people will be in each? If you run a small operation, the SOC may consist of only a few analysts with the rest of the functions being covered by other groups or outsources, but once you grow large enough the SOC may require its own engineers, architects, sysadmins, and more. We'll discuss staffing levels on the next page but as a starting point, you should know what roles will make up your initial team, and what level of talent you will need to search and budget for in each. Salaries in the US for these types of roles can range easily from \$35,000-\$200,000+ / year depending on the geography and experience. Since your strategy should involve doing a few things well at first, instead of trying to take on everything at once, consider what your minimum viable set of people would be to get a service going, and who can be added in a secondary capacity.

## Do You Need 24 x 7?

Consider when your SOC will need to operate:

- 24 x 7 x 365
  - Staffing constant monitoring can be expensive
  - Requires FIVE people per role ( $7 \times 24 / 40 = 4.2$ )
  - Potential turnover implications. "Follow the sun" model with multiple geographies can help
- 9 x 5 only
  - Off-hours alerts may sit longer
  - Can be supplemented with on-call weekends



### Do You Need 24 x 7?

What level of staffing do you plan to support for all of the previously determined roles? In most cases, non-analyst jobs will be 9 x 5 on weekdays with on-call support for emergency fixes from sysadmins, but analyst jobs are the bigger decision. Do you have the need and the budget to staff for a security team that is present and working 24 x 7 x 365? Most organizations desire this level of monitoring given that attackers may be more likely to strike when they think the company is not watching, and that is not a bad assumption. Funding around-the-clock operation, however, can be cost-prohibitive. Consider for example, how many people you will have to hire for every 24 x 7 x 365 seat you want to fill:  $24 \times 7 = 168$  hours per week. Assuming each person will work 40 hours,  $168 / 40 = 4.2$  people. Since those people will sometimes be on vacation, sick, or otherwise out of the office, that means you will need roughly 5 people to create a single seat of round the clock monitoring! That's a 5x increase over a single 9 x 5 seat! If you are an MSSP providing security for external customers, you may have no choice but to staff at this level, but many organizations find that they can do a good enough job with less coverage.

Considering the costs, a 9 x 5 SOC may seem like a better and more realistic option. While coverage is lower in off-hour times, most organizations will compensate for this by having one of the analysts be in an on-call position over the weekend. These on-call positions can be additionally paid to make it more appealing to analysts, and cover for the lack of active monitoring by still alerting staff when something important happens that needs to be looked at. Organizations following this model will designate specific use cases and alarms that are worthy of sending alerts to the on-call analyst who can then VPN in from home and see if anything needs to be attended to at that time. The additional side benefit of this model is that it potentially avoids needing to rotate staff into 2<sup>nd</sup> and 3<sup>rd</sup> shift, which tends to be life-disrupting and increases turnover.

## Defining SOC Capabilities

What **services** will your SOC offer?

- Log Management
- Security Monitoring and Threat Detection
- Threat Hunting
- Incident Response
- Threat Intelligence
- Penetration Testing / Red Teaming
- Vulnerability Management
- Forensics
- Audit/Assessment
- Policy and Procedure Support



### Defining SOC Capabilities

As part of the business value chain created by the SOC, you must decide which specific *services* your SOC will provide the business. For small organizations with few to zero dedicated personnel, the list will likely be very small and focused around monitoring and detection. For large organizations, the list will likely include all of the core SOC functions as well as many of the auxiliary capabilities as well – threat intel, forensics, and more. The assumption is that anything listed will be covered in-house while services outside the list will either be covered by another internal group, outsourced, or otherwise not needed.

When it comes to choosing services, it's ok and encouraged to not try to boil the ocean from the start. A smart and realistic plan may break service offerings down into phases that are rolled out as the group is built and matured.

## SOC Charter

- Use requirements to write it into a **SOC charter**
- The charter should be signed-off by management
- Charter contents:
  - SOC goals and mission statement
  - How the SOC operates (organization, governance, and rules)
  - Scope of operation
  - Permission to collect and monitor sensitive data
  - Actions the SOC has authority to perform for incident management



### SOC Charter

Once you have established the requirements, mission, constituency, capabilities, and services the SOC will need to offer your organization you can take these pieces of information and put them together to form a SOC charter. The SOC charter will be the document that authorizes the build and operation of the SOC, so it is important to think it through carefully.

The charter document should cover all the necessary items listed above that clearly explains what the SOC does, why it exists, what it has the authority to do, and how in general it will carry out that mission – including the permission to do so, and take action when necessary. Once complete, it should be signed off by management so that constituents of the SOC know what they can and should expect, and what is allowed.

## Keeping a Seat at the Table for Security

- Companies are racing toward "digital transformation"
- Cloud migration is happening *very* fast
  - Often without regard for security
- **Your goal:** Ensure security has a seat at the table
  - Be an integral part of all new initiatives
  - "Embed security in the process"
  - "Shift left" (DevSecOps)
  - It's MUCH easier to add security from the start than figuring it out after the fact



### Keeping a Seat at the Table for Security

Special consideration for the SOC in the planning phase is how to ensure all new major IT initiatives will include security. While nearly all companies are running towards cloud migrations and "digital transformation" unfortunately many times this comes with a disregard for proper lockdown. In the rush to get things working and provide some sort of value security easily becomes an afterthought. If unchecked, prepare to try to unravel the 10 different accounts across 3 different cloud providers that have been created by multiple people in different business units, all attempting to move as quickly as possible. Much better is to do your best to ensure ahead of time that all off-premises IaaS Paas, SaaS, etc. solutions go through a required step where security is notified of their usage, and gives them a chance to build in secure processes and monitoring from the start. Constant engagement with IT, Legal, and Finance is a good way to ensure security stays aware of and represented in these initiatives.

## SOC Planning Summary

- Define your mission and specific needs
- Define constituency, stakeholders
- Identify compliance and regulatory requirements
- Find or create policies, standards, and frameworks to use
- Gathering steering committee input
- SOC services offered
- Consider staffing roles required and order of hiring
- Codify it all in the SOC Charter

### Planning the SOC Summary

As the saying goes “if you fail to plan, plan to fail”, and it is as true as ever when it comes to the SOC. A security operation center is a complex endeavor with many moving pieces that need to fall into place at the right time. To make that happen, the more thoroughly you can plan what you will need, the better a chance you will have of a smooth and effective launch that can build into the mature SOC you want it to be in the future.

# Course Roadmap

- *Book 1: SOC Design and Operational Planning*
- Book 2: SOC Telemetry and Analysis
- Book 3: Attack Detection, Threat Hunting, and Triage
- Book 4: Incident Response
- Book 5: Metrics, Automation, and Continuous Improvement

## SECTION I

### Introduction

#### SOC Design and Planning

- SOC Functions
- SOC Planning
- *Exercise 1.1: Threat Actor Assessment*
- Team Creation, Hiring, and Training
- *Exercise 1.2: Attack Path Development*

#### Building a Strong Foundation

- Building the SOC
- SOC Tools and Technology
- *Exercise 1.3: Developing and Implementing SOC Playbooks*
- Protecting SOC Data and Capabilities
- Summary and Cyber42 Simulation – Day 1



This page intentionally left blank.

## EXERCISE 1.1

# Exercise 1.1: Threat Actor Assessment

### OBJECTIVES

- Identify specific threat groups applicable to your industry
- Create a basic profile of each threat group
- Gather high-level tactics and techniques used by those groups
- Organize threat data into an easily usable form
- Convert threat intelligence into guiding information for building your SOC



#### **Exercise 1.1: Defining Your Assets and Adversaries**

Please go to Exercise 1.1 in the MGT551 Workbook or virtual wiki.

# Course Roadmap

- *Book 1: SOC Design and Operational Planning*
- Book 2: SOC Telemetry and Analysis
- Book 3: Attack Detection, Threat Hunting, and Triage
- Book 4: Incident Response
- Book 5: Metrics, Automation, and Continuous Improvement

## SECTION I

### Introduction

#### SOC Design and Planning

- SOC Functions
- SOC Planning
  - *Exercise 1.1: Threat Actor Assessment*
- **Team Creation, Hiring, and Training**
  - *Exercise 1.2: Attack Path Development*

#### Building a Strong Foundation

- Building the SOC
- SOC Tools and Technology
  - *Exercise 1.3: Developing and Implementing SOC Playbooks*
- Protecting SOC Data and Capabilities
- Summary and Cyber42 Simulation – Day 1



This page intentionally left blank.

## Team Creation, Hiring, and Training Overview

- Team Creation
  - Org. charts
  - Tiered vs. Tierless SOCs
- Hiring
  - Recruitment tips
  - Personality traits and mindset to look for
  - Interviewing techniques
- Training
  - Plans and goals
  - Certifications and assessment

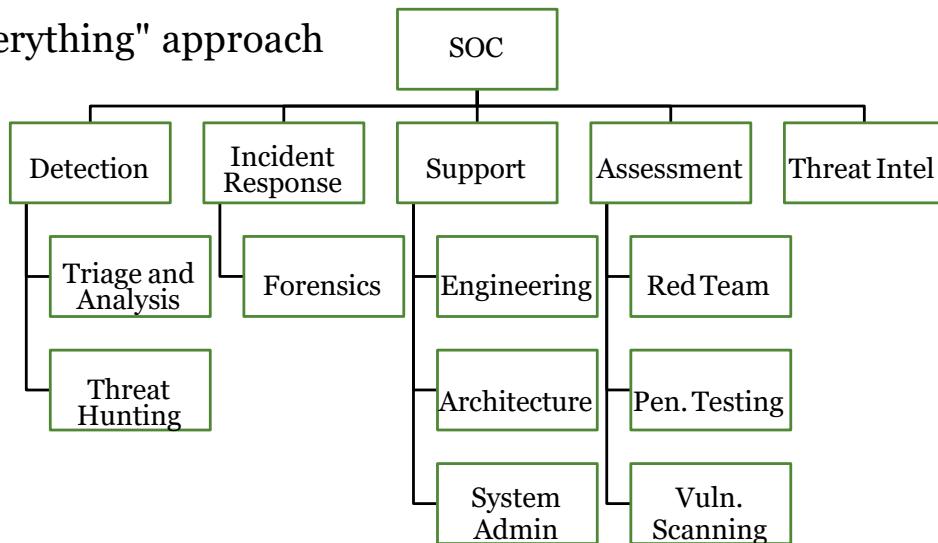


### Team Creation, Hiring, and Training Overview

In this section, we'll discuss the people-focused portions of creating a SOC. Specifically, we'll cover creating your team, the org chart options you should consider, and the pros and cons of a tiered vs. tierless SOC model. We'll then step into tips and advice on hiring your dream team – personality traits to look for, interviewing methods and guidance, and how to reduce bias to find the truly best individuals. We'll then wrap it up with a discussion on training your new or current employees, both how to get them started as quickly as possible as well as training plans and certifications they may want to aim for in the medium to long term.

## Wide Scope SOC Functional Org Chart

The "everything" approach



### Wide Scope SOC Functional Org Chart

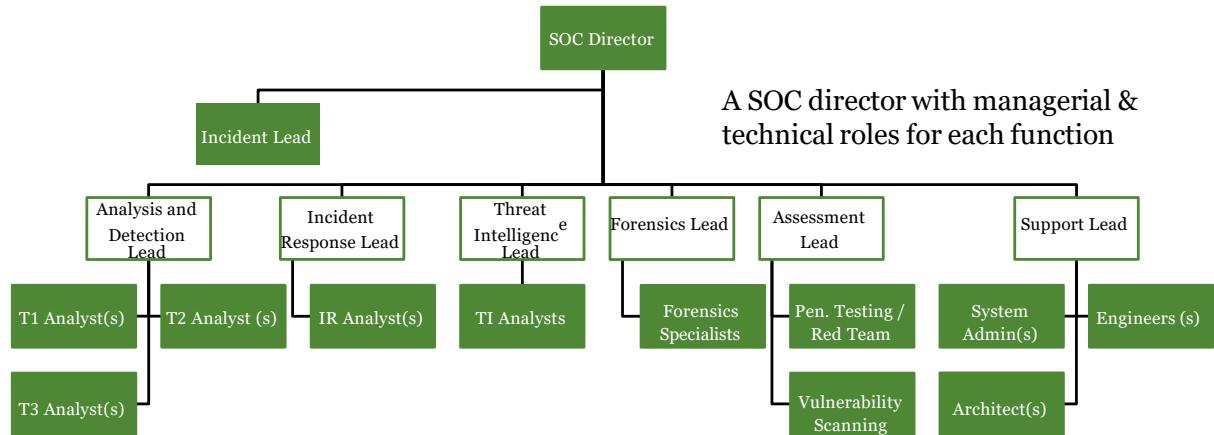
Over the next few pages, we'll look at various common SOC org charts. Remember that as we look at options over the next few pages, there is no universal "right" or "best" way to do it, and that you can mix and match anything you see here. Each approach has its merits and optimizes for different things, and one option may work better with the pre-existing structure at your organization.

If you want to take the "put everything in the SOC" approach to building a SOC and put all core and auxiliary functions under the same management at the "SOC" level, this page shows an org chart of what the functional areas might look like.

- **Detection** – All SOC analysts from all tiers (if using tiers) doing triage and analysis and threat hunting (detection and attack identification focused work)
- **Incident Response** – A dedicated group of IR specialists that work closely with the analysts once an incident has been verified. The IR group is not always broken up into a separate group but is done so for clean separation of duties at many organizations. This group may contain the analysts that do the actual response and cleanup as well as a forensics capacity that helps them understand what must be done through hard drive, mobile phone, or network traffic forensic investigation.
- **Support** – The functions that support the SOC can go under one large umbrella or can be broken out separately. This group will consist of architects designing and implementing new services for the SOC, system administrators that keep the current infrastructure running and healthy, and engineers that may tweak, improve, or add analytics to your existing set of appliances. Again, not all SOCs have a dedicated group for new analytic writing and let analysts handle it instead; if you have the capacity to do either, the choice is yours, a good argument can be made for either arrangement.
- **Assessment** – The assessment group may be within the SOC or as a peer group to the SOC. In this diagram we have penetration testing/Red Teaming and vuln scanning all under the SOC org. Since the mission of the assessment team (making sure the company is secured from cyber-attacks) is closely tied with the core SOC activities (monitoring for and stopping cyber-attacks), my personal preference is to keep this group as closely integrated as possible to maximize communication and collaboration.

- **Threat Intel** – Threat intel is another group that is sometimes placed adjacent to the SOC org but may be better utilized if situated within it. Like the assessment team, the goal of the threat intel team is largely to support what the SOC is trying to do, and therefore any setup that maximizes communication and collaboration will benefit both teams. Since the output of the incident response team needs to go to the Threat Intel team, and the output of the Threat Intel Team needs to go to analysts and IR, it makes sense that they may function best under shared management.

## Large Tiered SOC Org Chart

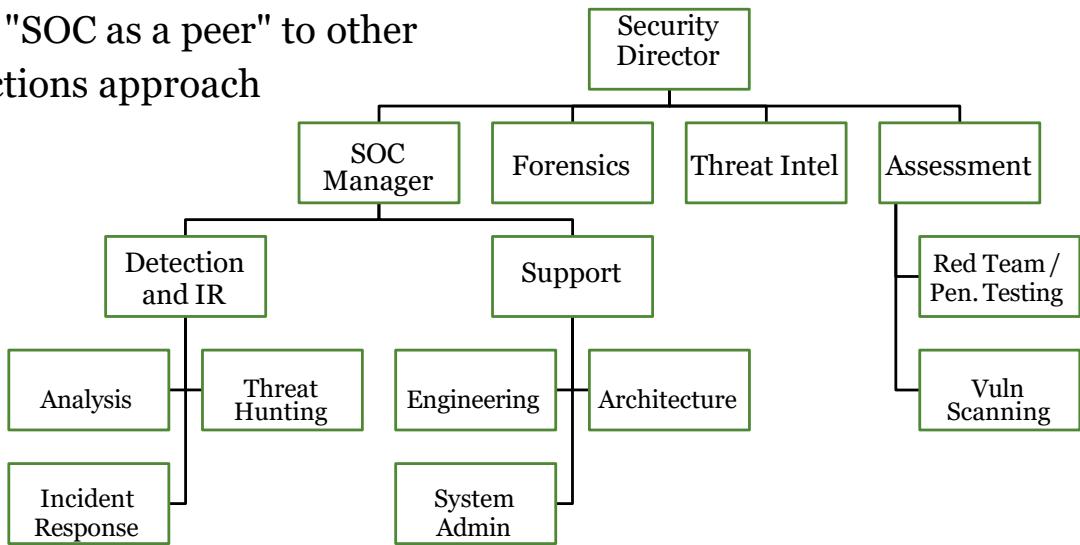


### Large Tiered SOC Org Chart

If you have dedicated roles for the "everything inside the SOC" approach, you also likely have a large organization with multiple people doing each specialty. In this situation it is often wise to appoint a lead of each subdivision. In this org, the role at the top of the chart may be director level with the leads doing both managerial and technical duties.

## Condensed Scope SOC Functional Org Chart

The "SOC as a peer" to other functions approach



### Condensed Scope SOC Functional Org Chart

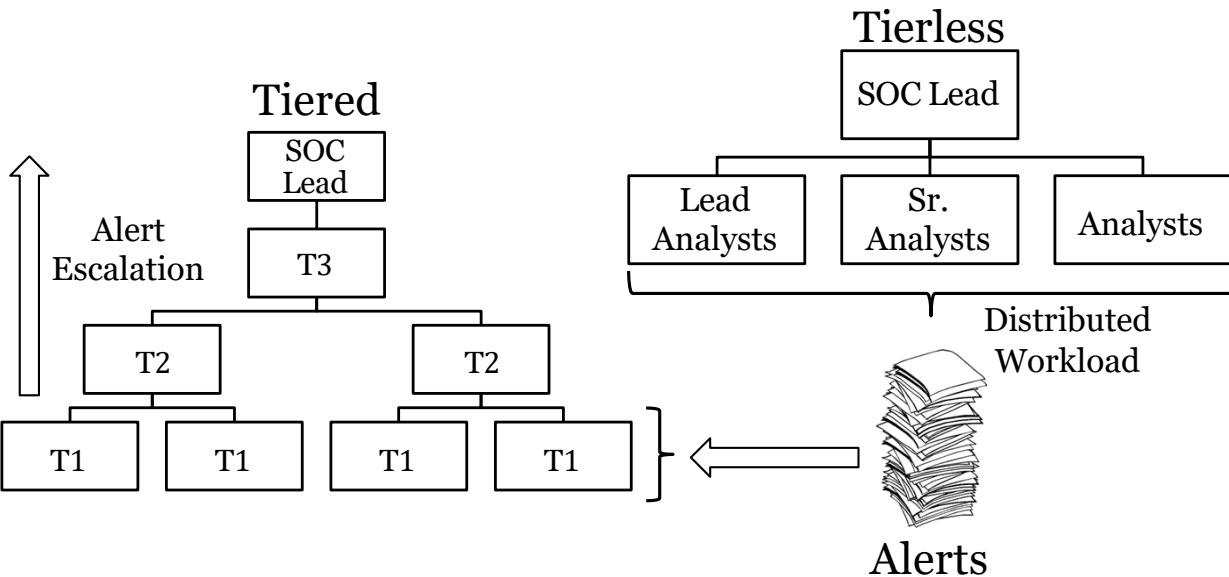
An alternative to the previous slide is this condensed SOC org with all components still present, but a different reporting structure.

The SOC still stays as a group, but management of the SOC only includes watching over more "core" SOC activities such as detection, triage, analysis, and IR, plus the supporting functions that create and keep the tools running that the SOC uses daily.

Forensics in this org is separate from the SOC and this can make sense in some organizations. Since forensics knowledge is specialized and different enough from analyst's knowledge that the two don't usually overlap enough to have one individual doing both (like is often the case with incident response). The SOC may also not be the only group that uses forensic specialties – many orgs for example have separate (outside the SOC) groups for internal corporate investigations and insider threat-like situations. This capacity very often requires forensics as well, therefore having a generalized "forensics" service used by all may make more sense in this case.

You'll also notice in this chart the SOC is separate from the Threat Intelligence and Assessment functions. Like forensics, these job roles are considerably different than what SOC analysts and incident responders will be doing in their day to day lives, but in this case, they are placed outside of SOC management. This is another configuration that many organizations, especially larger ones, choose to use. The key to ensuring this works is keeping lines of communication between the two teams open. Things like co-location and regular cross-team meetings can help compensate for the natural likelihood to speak less the further your management is separated.

## Tiered vs. Tierless SOC



SANS

MGT551 | Building and Leading Security Operations Centers

61

### Tiered vs. Tierless SOC

Speaking of tiered SOCs, while the traditional org structure for analysts has been on a two- or three-tier model, there are many organizations that have moved away from tiers.

In the traditional tiered analyst model, Tier 1 is the front-line soldiers receiving and triaging alerts generated from the security appliances. Those alerts that analysts either cannot make a determination for, or require a more complex solution than they are capable of performing, get escalated to Tier 2. Tier 2 then takes what it can, calling in Tier 3, as specialty knowledge is required. The job breakdown of Tier 1-3 is somewhat dependent on the organization. As we saw on the previous pages, SOCs with dedicated incident response teams may use tiers as an "increasing difficulty of triage and investigation" solution. Others without an incident response group may dedicate Tier 1 to triage and alert investigation/validation, Tier 2 to incident response, and Tier 3 for hunting, malware reversing, and other specialty tasks. If you are going with the tiered model, consider what is most important to you and how you plan to structure your SOC to guide you in this decision (there are also some potential retention factors to consider as well that will be discussed on the next page).

In tierless SOCs, all analyst levels participate in triaging and investigating alerts and the analysis function is much flatter. While there may be job title differences that establish salary jumps and acknowledgment of advancement, each person more or less has access to the same tools and does the same thing (within their capability). This means everyone collectively ensures that alerts are taken care of in a timely matter, and those analysts that hit a road bump in their investigations must be comfortable and willing to ask others for help in how to proceed. This model may seem counter to what you have heard about in the past, but there are *many* SOCs run by large organizations that work perfectly well with some version of this model.

## Tiered vs. Tierless Pros and Cons

### Tiered

#### Pros:

- Repeatable process
- Clear separation of work tasks
- Defined escalation path
- More optimized use of time

#### Cons:

- Limits analyst growth
- Analysts feel more like robots
- Potential retention issues

### Tierless

#### Pros:

- More varied and creative work
- Uncapped analyst talent growth
- Happier analysts

#### Cons:

- Requires a responsible team
- Less defined process
- Can be riskier if expectations are not properly set



### Tiered vs. Tierless Pros and Cons

While we will cover the reasons in more depth later on in the course when we discuss retention, the short story is that tiers have both up and downsides. While tiers can be great for repeatable processes, they can also cause analysts to stick on a narrow track of repetitive tasks that can ultimately drive them to lose enthusiasm for the job. Tiering, in a very real sense, means a smaller job scope, and that lack of freedom can quickly cause analysts to reach a ceiling in growth. If they aren't able to quickly jump to one of the next tiers, they may start looking for another job. If you run an MSSP or are in a similar situation, tiered SOCs may be the right choice if for no other reason than your customers will expect you to use what has been seen for many years as best practice. If you're creating an internal SOC however, there may be some very real benefits to going with the tierless model, long-term retention, and analyst growth being some of the big ones.

To choose which model is right for you, consider:

- How important is a specifically defined, repeatable process to me?
- Will I have enough analysts to make a meaningful distinction?
- Do we have the budget to hire the roles desired?
- How will I divide tasks amongst analysts?
- Will there be tools and data that one tier can use, but lower ones cannot?
- What is the expected skill level of each tier?
- How important is *long* term SOC analyst retention?

For additional detail and the author's personal view on the issue (who came from a tierless SOC), check out the 2019 SOC Summit talk – "Virtuous Cycles: Rethinking the SOC for Long-Term Success".<sup>1</sup>

[1] <https://www.youtube.com/watch?v=G5lj7M2ZuT0>

## Staffing for Continuous Coverage (1)

- Think hard about the need for 24/7 coverage:
  - Headcount dramatically increases
  - Premiums for off-hours work is the norm
  - Most activity you'll want to monitor happens when users are active
  - Retention will be difficult
  - Consider off-hours escalation path and response times



### Staffing for Continuous Coverage (1)

Many organizations assume that a security operations center must provide 24/7 coverage, but this is far from true. In fact, running a 24/7 SOC can add significant cost and management overhead that may not be worth it for the value an off-hours operation provides. You can expect to pay a premium for analysts to work nights and weekends, and retention will be difficult with ample opportunity elsewhere for analysis work during normal business hours.

If your first instinct is to build a 24/7 operation, ask yourself:

1. When are my users active? While it's true that a persistent attacker may be making moves outside of normal work hours to avoid detection, many attackers will need or want your users to be active to either facilitate their activities or hide among the noise.
2. What is my escalation path for incident response? If you rely on IT engineering and leadership to execute a response, make sure they are also available outside of normal working hours. Identifying threats at 2AM has little value if you can't do anything about it until 9AM (or days later if it's a weekend).

## Staffing for Continuous Coverage (2)

- Covering 24/7 ops:
  - Minimum 10-12 people
  - Common configurations:
    - 12-hour shifts, 4 on 3 off
    - 10-hour shifts, 4 on 3 off
    - 8-hour shifts, 7 on 3 off (on a rotation) or 5 on 2 off
  - Other considerations: breaks, shift overlap for turnover, team meetings

### Staffing for Continuous Coverage (2)

There are many different possible shift configurations to cover a 24-hour schedule and you'll have to select one that best fits your team and organization. Whatever configuration you choose, you'll need a minimum of 10-12 people to make it work. While it's technically possible to do it with less – this author has with less-than-optimal results – you will need at least some extra staffing to cover when people get sick or take vacation.

The most common rotations are 12-hour, 10-hour, and 8-hour shifts, with smaller teams normally requiring longer shifts to maintain the requisite coverage. The author recommends against 12 hour shifts for a few reasons: burnout is common, efficiency falls off a cliff after about hour 9, and the rotation makes it harder for team members to plan and balance their personal lives. Other considerations for shift planning include how much overlap you want to have for busier periods and shift turnover, when team events like meetings or one-on-ones will occur, and accommodations for analyst breaks.

## Staffing for Continuous Coverage (3)

- Night and weekend coverage
  - **Permanent staff for each shift** – better for team cohesion and work-life balance, but hierarchy may form between shifts (analysts will want to “move up” to days)
  - **Rotation** – no one is “stuck” on nights or weekends, hiring may be easier, but burn out is common and work-life balance suffers



### Staffing for Continuous Coverage (3)

Night and weekend coverage can be achieved by permanent staffing for each shift or by filling those shifts with rotating assignments. Each approach has positives and negatives: permanently assigned shifts can be better for team cohesion and camaraderie as well as work-life balance. A consequence of this approach can also be an unofficial hierarchy between shifts, where off-hours staff looks to “advance” to preferable hours. A shift rotation can be an easy way to avoid hiring challenges for off-hours coverage but burn out and work-life balance issues caused by this approach can result in retention problems.

## Alternative Coverage Approaches

- Extended hours
  - 7am to 7pm or similar
  - Easier to escalate and engage IT, engineering, and management for containment
- Follow-the-sun
  - 2 to 3 teams around the world with coverage windows aligned to their business hours
  - Expensive but avoids many of the recruiting and retention challenges of 24/7 ops

### Alternative Coverage Approaches

SOC coverage is by no means an all-or-nothing proposition. If there is value in extending coverage – for example, you have users across multiple time zones but mostly in the same hemisphere – then a 7am-7pm plan might be a good fit. Alternatively, if your organization is global or you can support it through outsourcing, a “follow-the-sun” coverage model can be a good alternative to staffing 24/7 operations in a single location. In this model, teams across multiple time zones provide complete coverage around the clock by each staffing their own day shift local time. This can make hiring and retention easier but can be slightly more challenging to manage. Each approach has its trade-offs; what is most important is realizing that there are a few different options available when extended coverage is required – especially in the current remote-friendly climate.

## MSSP Coverage Considerations

- Service provider contract and SLAs should reflect your expectations
  - “How can I measure the value you’re providing?”
  - “What happens if I get breached?”
  - “What happens to my capability when you go away?”
- The myth of “eyes on glass”
  - MSSP business model is to support lots of customers by doing one or more of these:
    - Merging customer data into a single platform
    - Automating alert triage and escalation
    - Investigative time sharing
  - Make sure you understand how your MSSP scales!

### MSSP Coverage Considerations

When filling out coverage using a third-party service provider, it's important to make sure that their service aligns to your needs. The service contract and/or statement of work must clearly state what is within their scope versus yours down to the tactical level. What alerts or queue or data are they responsible for analyzing? What are your expectations for alert validation? How should they escalate to your team? What work can you hand off to them? Establishing these guidelines up front can minimize wasted cycles and poor collaboration later.

These three questions from the Bionic blog establish useful reference points for your SOW and SLAs:

1. How can I measure the value you’re providing?
2. What happens if I get breached as a result of an oversight in your work?
3. What happens when you go away?

The answers to these questions will inform your expectations of your third-party service provider. The key is ensuring that the services they deliver meet your coverage expectations. Many third-party service providers will tell you that they provide “eyes on glass” monitoring or other sustained attention for the entirety of their coverage window. This may be a dubious claim since the business model of any service provider depends on providing services to many clients at once! In most cases, the service provider scales by doing one or more of the following:

- Consolidating customer data into a single platform where they can monitor everyone at once
- Automating some or all triage and escalation functions
- Dividing up analyst time across multiple customer data

Regardless of which approach your service provider takes, it is unlikely they can deliver full-time, dedicated monitoring of your security data *by human eyes* within a given coverage window. If they promise this but aren't charging staff augmentation prices, it's time to ask more questions.

<https://www.bioniccyber.com/blog/2019/11/25/3-questions-to-ask-your-mssp>

## Onboarding Service Providers

- MSS should be considered an augmentation of your internal team
- Expect 30-60 days minimum
- Onboarding normally includes some or all of the following:
  - Technology deployment
  - Credential sharing
  - Knowledge transfer
  - Escalation planning

Value   time →	During on-boarding / before service	During service consumption
To enable service delivery (MUST)	Deploy sensors, share network diagrams and access credentials, provide contacts	Notify on asset and network changes, access changes, contact info
To enable maximum value from the <u>MSSP</u> (SHOULD)	Refine & share a security policy, have <u>IR</u> plans, provide detailed asset and context info	Respond to alerts (!), remediate systems, declare incidents and run <u>IR</u> , jointly tune the alerts, communicate changing security priorities

### Onboarding Service Providers

Deciding to outsource a coverage window or SOC function(s) is only the first step in what should be a close working relationship with your managed service provider. You should see your MSS not as a separate capability, but as an augmentation of your team. In his blog post "MSSP Onboarding – A Critical Process!," Anton Chuvakin writes about four activities common to most MSS onboarding processes: technology deployment, credential sharing, knowledge transfer, and escalation planning. Technology deployment could be its own major project, such as a SIEM or sensor installation, or it might be something as basic as installing log collection agents for your existing data sources.

Regardless of how technology factors into service delivery, it is critically important to treat vendor-provided hardware and software as you would any new addition to your environment – meaning, they are subject to vulnerability assessment, software updates, and other activities required to maintain a defensible network. Credential sharing encompasses a wide range of activities that may involve creating domain accounts or establishing VPN connections with your service provider. We'll discuss supply chain risk management later in this class, but for now, keep in mind that establishing these trust relationships – even with a security vendor – increases your attack surface and presents an extremely attractive attack vector to a sophisticated threat actor. There have been many data breaches traced back to initial access via these trusted connections!

Knowledge transfer is another area we don't want to overlook in onboarding a new service provider. As an analyst, consider everything you would need to properly validate alerts and investigate anomalies: network diagrams, asset lists, IP ranges, and other contextual information should provide your MSS with the means to minimize escalation of false positives. Finally, work with your MSS to develop a communications plan that aligns best to your organization. Will you be the one getting the phone call at 3AM for critical incident notifications? What if you aren't available? Are there other points of contact for questions about the environment or administrative discussions? What about billing and contract management? Even a well-defined escalation plan usually needs 30-60 days to work out all of the bugs, so being as complete and specific as possible up front can help you avoid headaches in those first few months.

Finally, many of these are not "one and done" activities. MSS relationships require active engagement and ongoing management in order to keep the service provider up to date on network changes, personnel departures, process updates, and other key information. SOC teams often become unhappy with their MSSPs, complaining that they get little to no value from the service when they have done very little to enable success. That being said, if your MSS does not seem interested in detailed contextual information or this level of ongoing engagement, you may need to question the value you expect from the offering.

## Recruitment and Hiring Overview

- Job Postings
  - Where to post
  - What job recruits are looking for
- Interviewing
  - Characteristics to seek
  - Effective interviewing tips
  - Questions **not** to ask
  - Reducing bias
  - Assessing technical depth



SANS

MGT551 | Building and Leading Security Operations Centers

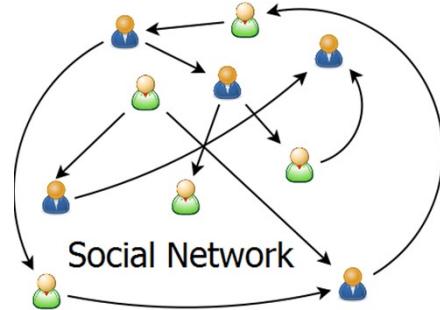
69

### Recruitment and Hiring Overview

Over the next few slides, we'll cover one of the most important activities you do as a manager – recruitment and hiring. You can't hire the best without a solid recruiting effort since you'll never find those A-players in the first place, so knowing where to post your jobs and how to word them is crucial. Once you get their attention, you still need to hurdle the problem of finding the best applicant through a relatively short interview. How you perform those interviews can easily steer you in the wrong direction or fail to highlight red flags that you may have wished you saw before extending an offer, so a thorough behavioral and technical interview can be your best bet at succeeding.

## Advertising Your SOC Jobs

- You can't hire awesome people if they don't know about your job!
  - Do NOT only advertise your job on your organization's website
- **Pound the pavement!** Go where infosec people meet and **recruit!**
  - Local / National Conferences – BSides, SANS Summits, BlackHat, DEF CON, etc.
  - Infosec / Meetup Groups – ISACA, ISSA, DEF CON groups, local groups
  - College clubs / infosec programs / internships
- **Online**
  - Reddit / netsec hiring thread
  - Twitter
  - Infosec Slack / Discord channels
  - SANS Advisory Board and other email lists
  - CTF competitions



### Advertising Your SOC Jobs

If you want the best of the best talent, even at an entry-level, you need to think outside the recruiting box a bit. Don't just give your job description to HR and hope the resumes roll in from your company websites. Even if you post the job to Indeed and similar job marketplaces you've virtually guaranteed to get a torrent of questionably matching resumes from around the world that you'll need to spend time sorting through. Sure, you can find nuggets of gold, but considering what your time is worth, there is a better way.

The first method is to get out into the world and go meet people in person! There are *tons* of infosec conferences all over the world of both local and national scope and many of them have hiring villages. The beauty of doing it this way is the people that attend are self-selected as willing to give up their weekend to go learn more about information security – exactly the type of attitude you're looking for. Whether it's a conference, local infosec group, or college club, the signal to noise ratio is almost guaranteed to be higher recruiting at these types of events. Don't forget, especially for the college club and program group that you can always pre-select ungraduated candidates as interns and get a good chance to know them before bringing them on full time!

In addition, there's plenty of non-traditional online locations to job hunt. Again, think "if I were REALLY into information security, where would I be hanging out?" The answers to that question are forums, email lists, infosec Slack and Discord channels, and anywhere else enthusiasts gather.

Want to get really out of the box? Many online CTFs such as the yearly SANS holiday hack challenges have RPG video game style boards where all players can walk around together and converse with each other. There's nothing to stop you from talking to the other contestants and feeling out if someone has the particular talent you're looking for. Any opportunity to tap into a group of people who have self-selected to spend their time on challenges and learning can be a great place to pick up new employees.

## Building the Dream Team

To pull in top talent, ISC2 suggests<sup>1</sup> your job posts highlight:

1. An **appealing team** to join
2. A **rewarding** place to stay



What candidates said they are looking for:

- Relevant on-the-job work experiences
- Robust training and professional development opportunities
- Access to career advancement opportunities
- Work in a continuously evolving field with stability and job security
- Engaging and challenging work
- Help with typical career challenges such as training costs, career progression, etc.

### Building the Dream Team

What about the content of your job postings? How do you stick out against the myriad of other SOC jobs available around the world? The 2019 ISC<sup>2</sup> Cybersecurity Workforce Study and ISC<sup>2</sup> Hiring and Retaining Top Cybersecurity Talent Guide<sup>2</sup> suggest that to pull in the best talent you need to focus on getting across that you have an appealing team to join and that your organization is a rewarding place to stay. According to the most popular answers from the survey, the way to do this is to highlight that your workplace provides

- Relevant on-the-job work experiences
- Robust training and professional development opportunities
- Access to career advancement opportunities
- The chance to work in a continuously evolving field with stability and job security
- Engaging and challenging work
- Addressing typical career challenges such as
  - Contributions / coverage of training costs
  - A clear career progression path for security roles
  - Opportunities to increase security awareness with end-users
  - A way to keep up on cyber security trends

1 <https://www.isc2.org/Research/2019-Cybersecurity-Workforce-Study>

2 <https://www.isc2.org/Research/Hiring-Top-Cybersecurity-Talent>

## Additional Hiring Advice

Additionally, ISC<sup>2</sup> suggests the following<sup>1</sup>:

- For **upper-level roles** – focus on relevant experience
  - Look to consultants, contractors, vendors, industry peers, MSSPs
- Use **realistic** expected qualifications and experience
  - **Five years of cybersecurity**
  - **Nine years** IT was ISC<sup>2</sup> survey **average**
- For **new hires** – look for degrees relevant to cyber security
  - Computer and information sciences, engineering, etc.
- Leverage **internal talent** when possible
  - Non-security IT roles (networking, endpoint), legal, finance, HR

## Additional Hiring Advice

Some other useful advice based on the findings from the ISC<sup>2</sup> Global Cybersecurity Workforce study:

- For hiring upper-level roles, put the heaviest focus on relevant experience. An easy place to find people with the most relevant experience might be considering your past vendors, contractors, industry peers, or MSSPs you've worked with. Anyone that has some idea about your environment already would have a hand up in jumping into your team.
- When hiring either advanced or entry-level roles, use *realistic* expectations for job qualifications and experience. We've all seen the job descriptions that ask for 10 years' experience in a technology that came out 10 years ago – don't be that company. If you're hiring for a lower-level position, don't demand the world of candidates, someone who is very capable and needs a minimum of training might slip by.
- For new hires, one of the largest sources of hiring is graduates from security or related college programs (see below for workforce makeup details).
- Leveraging capable internal candidates can be a huge boon for security teams. If someone from network engineering, the helpdesk, endpoints wants to jump into the security team, the transition will be all the easier. There are many jobs in an organization, even beyond IT that have transferrable skills. With some baseline training on security work, they may be able to hit the ground running much quicker than an external candidate.

To give you an idea of what the workforce looks like the survey found that within cyber security 12% have a high school diploma, 11% have attained an associate's degree, 38% have a bachelor's, 28% have a master's and 10% have a doctorate or post-doctoral level of training. The breakdown of these applicants were 40% computer and information sciences, 19% engineering, and 10% business.<sup>2</sup>

1 <https://www.isc2.org/Research/2019-Cybersecurity-Workforce-Study>

2 Ibid.

## Interviewing

- Goal: Predict candidate success as accurately as possible within a reasonable amount of time
- Assess candidate in a **bias-free way** to find
  - Fit for the team
  - Talent, capabilities, curiosity, passion
  - Upward potential
  - Short- and long-term goals
  - Unique skills not already present on the team



### Interviewing

Looking at the interview process from a high level, what is the goal? Overall, the whole point of an interview is to get a good feel if the candidate is likely to be successful in your organization and job role within the shortest amount of time. More specifically, we often try to evaluate fit for the team, talents, personality traits, upward potential, and ambitions, and what unique skills the candidate might bring to the table.

While there are some straightforward ways to test a match for desired technical requirements (discussed shortly), assessing personality is a bit more difficult. The trouble with interviewing is that it's very easy to let unconscious bias sneak in, and that unconscious bias can lead us to make a sub-optimal decision. Throughout this section, we'll cover some evidence-backed ways to identify the best possible candidates from your SOC on both a technical and personality trait level while simultaneously avoiding unconscious bias.

## What to Look For in a Candidate

### Good Signs

- Curiosity
- Motivation
- Persistence / grit
- Positive attitude
- Analytical thinking capability
- Familiar with current events
- Self-learning, projects, blogs, research
- Unique background

### Red Flags

- Misrepresentation
- Suspicious number of excuses
- Ego
- Trying to bluff answers instead admitting they don't know
- Disparaging former employers
- Unable/unwilling to give references
- Job hopping



### What to Look For in a Candidate

Beyond technical aptitude, there are many personality characteristics that you can screen for in your behavioral interview questions to try to understand what kind of person the candidate is. Asking for examples of research they've done, blog posts or code they've written, projects they've struggled particularly hard to complete, how they react under pressure, and more can bring to light some of the positive characteristics you would hope to find in a team member. In a SOC the work is tough, ever-changing, and employees must be willing to take on the challenge. Any questions that highlight a candidate's persistence to solve hard problems, motivation and curiosity to continuously learn, and show analytical thinking capability can be a very useful addition to an interview.

On the flip side, there are many red flags you may run into in an interview as well. If you encounter any of the items on this list proceed with caution.

## Technical Interviewing Tests

### Technical tests should closely approximate job tasks:

- Alert triage – “Which of these seems most important and why?”
- Logs – “Can you describe what is occurring on this system?”
- Incident Response - “Can you find evidence of a compromise in these endpoint logs?”
- PCAP interpretation – “Is there an infection / exploit occurring here?”
- OSINT – “Can you find information on this malware?” / “Tell me if this URL is malicious or not, and why you think so.”
- Email – “Is this email malicious? If the user clicked it, what would happen?”

### Technical Interviewing Tests

If you want to include a hands-on technical portion of the interview, the tests should focus on tasks that approximate the normal work the employee will need to handle. This slide has some suggestions of tests that could be asked of a potential SOC analyst during an interview that would closely match their likely day to day work.

For the implementation of these tests, the easiest way to manage this would be to develop data sets with real logs, PCAP files, real spam email, and other as realistic as possible samples that can be used. Then, place all this information into a virtual machine along with standard industry tools that would be used to analyze it (Wireshark, etc.) that can be presented to the analyst during the interview. The virtual machine should have a snapshot taken to make rollback and setup for the next candidate effortless and to ensure each applicant is presented with an identical environment.

## Probing for Technical Depth

- Open-ended, situational questions
  - "Someone thinks they may have become infected from visiting a malicious website, what data might you use to confirm or deny that, and how?"
  - "How would you set up an internet-available service at your house?"
- Questions across multiple domains that get progressively harder
  - Start with an easy question....
  - If they get it correct, go slightly deeper
  - Continue with harder questions until you find their limit
  - Score candidates in your notes as you go – helps eliminate biases from poor memory<sup>1</sup>
- Benefits:
  - Allows candidate to display technical depth and capability with less pressure
  - Tests what happens when they don't know the answer – will they admit it / bluff / freeze up

### Probing for Technical Depth

While testing your candidate in various domains can be great, it can lead to someone shutting down and a very awkward interview ending if the questions are all beyond their capability. One way to solve this is to provide technical tests that ramp up in difficulty across different domains and scenarios. The idea is to give them some easy wins while seeing how far they can get down the set of questions (make them hard enough that most people won't finish them all). You can even tell the candidate that this is how the test is structured so it's not a surprise to them that they weren't able to answer everything possible and aren't hard on themselves about it. The benefit of doing this is that it won't throw them off their game for the rest of the interview when they don't finish them all (and if they do, you know they're exceptional), and also gives you an idea per domain how much a candidate may know. It will also show you how an applicant deals with not knowing the answer – will they admit it, freeze up, try to bluff? These issues can be telling as well.

Another effective technique is to use situational, open-ended questions that allow applicants to show their own depth of knowledge. Things like "a user hands you their laptop and says it might be infected, how might you handle this situation and confirm/deny the presence of a virus?" Having used questions like this in an interview before, the answers were everything from "I would run antivirus to check" to "I would enumerate running processes, check out Sysmon logs, look at the Windows prefetch, run Sysmon Autoruns to look for persistence and investigate the DNS cache." This very quickly shows you if the candidate "talks the talk" and what you might be able to expect from them on the job.

## Things You Should NOT Do In an Interview

### 1. Falling for the "similar to me" bias

- Don't confuse "possible best friend" with "possible best employee"
- One of the biggest biases in interviewing



### 2. Easy to predict/practice interview questions

- Many, new candidates will be Googling for infosec interview questions
- If you do this too and use them, both of you learn nothing
- Create your own, unique interview questions

### 3. Puzzle / lateral thinking questions

- Google found these are “a complete waste of time,” and “don’t predict anything” when it comes to candidate success<sup>1</sup>

### Things You Should NOT Do In an Interview

There are some interview pitfalls that, with knowledge of them, are easy to avoid. The first and foremost is questions that screen for people similar to you. The whole idea of the interview is to find the candidate that seems best suited for the job, not the one most likely to be your best friend, but your brain will sabotage you on this. It's undeniably easy to think someone is the best candidate simply because you had the easiest flowing conversation, similar interests or background, or go to the same gym – that doesn't mean they will be the best hire. Using a structured interviewing process and knowing exactly what traits and talents you need *before* going into the interview can help clear away this unconscious tendency.<sup>1</sup>

The second thing you should avoid in an interview is not considering which questions to ask and using the first Google hit for suggestions. Any candidate can, and many candidates *will*, probably do the same thing, especially if they have never done an information security interview before. In this scenario, you will be unable to gauge what the candidate truly knows and you may have a false impression because they effectively saw the answers before taking the test. Take the time to make up your own custom interview questions, and of course, base them as closely on the job at hand as possible.

The final type of question not to ask in an interview are those famous lateral thinking and puzzle questions such as "how many ping pong balls fit in a 747?" Why do these not belong in a job interview? While companies like Google and Microsoft were the genesis of many of these mind-benders, since their introduction they have been not only abandoned as not useful, but even banned<sup>2,3</sup> since they were shown not to be predictive of anything useful.

1 <https://blog.psionline.com/talent/interviewing-tip-stop-the-similar-to-me-bias>

2 <https://www.newyorker.com/tech/annals-of-technology/why-brainteasers-dont-belong-in-job-interviews>

3 <https://io9.gizmodo.com/why-google-banned-brainteasers-from-their-interview-pro-576334070>

## Two Key Items for Bias-Free Interviewing

Two of the most important tips for reducing bias:

### 1. Use a **standardized process** for all candidates

- Use the same questions in the same order
- Score interview in your notes as you proceed
- Do **not** come back to it later and try to remember
- Explain what you're doing to the candidate and why

### 2. Focus on **relevant** past / future **behavioral measures**

- Past: "Describe a situation where..."
- Future: "How would you ... / What would you do if..."

*[Read: mgt551.com/bias](http://mgt551.com/bias)*



### Two Key Items for Bias-Free Interviewing

When interviewing, the most important thing to remember is to fight any type of bias, conscious or unconscious, from creeping into the process which could lead to suboptimal hiring decisions. Working with your HR team to review potential interview questions is a great first step towards making sure you aren't asking any questions that could open your organization up to legal issues down the road. Using this approved list as a standardized question set and a focus on past and future behavioral questions are often given as two of the most effective ways of ensuring interviews stay fair and even across candidates<sup>[1]</sup>. This means developing a set of questions / tests, and giving them identically to every candidate, even in the same order. Doing this ensures every person is given the exact same chance to demonstrate their competency in the chosen tasks and questions. While many companies use free-flowing, conversational-style interviews which may feel more natural, this method, unfortunately, does not give each candidate a fair chance and may lead to unintentional bias based on the unique direction every interview might take. The better move is to pre-determine what types of skills and knowledge the ideal candidate needs, and ensure those points are hit across all applicants.

In addition, behavioral measures have also been consistently shown to be one of the best types of questions to focus on throughout an interview. These questions can be in the form of past behaviors, as well as future-based behavioral questions. Past-related questions are often in the form of "Describe a situation where...", and the candidate has a chance to demonstrate how they handled a difficult or sensitive situation. In addition, future-based questions in the form of "How would you ..." or "What would you do if ..." are often used to check and predict the candidate's behavior in a complicated, hypothetical situation they haven't yet encountered.

[1] [mgt551.com/bias](http://mgt551.com/bias)

## Reducing Bias in Your Hiring Process

Additional ways to reduce unconscious hiring bias:

- Discuss with the team, and **build awareness** around hiring bias
- Ensure the **job description language** is neutral and wide-ranging
- Use a **blind resume pre-screening** process
- **Collaborative** hiring decisions, (not interviewing)
- Beware of **confirmation bias** – seeking info that supports what we have already concluded before meeting someone
- Diversity in **job posting locations** leads to diversity of candidates

### Reducing Bias in Your Hiring Process

Beyond the previous suggestions, there are plenty of other techniques to ensure a more objective hiring process.

- Provide awareness to the team of the potential for hiring bias and how it can easily unconsciously sneak in *before* starting the recruitment and interviewing process
- Verify the job description does not contain any language that hints at gender or age, also consider whether you have written the requirements so specifically that you may be limiting who would apply
- Use a blind resume reviewing process to ensure that irrelevant factors beyond the past experience and qualifications are not considered
- Use a team of coworkers for hiring decisions instead of just one person, this can help balance out "similar to me" bias that can unconsciously appear. The previously linked HBR article however suggests that collaborative/group *interviewing* shows no improved effectiveness and may actually be worse than individual interviewing. Why? With individual interviews you will get multiple conversations worth of data points on a candidate instead of multiple people seeing only one conversation—in short, you get more data the individual way.
- Inform the team about, and be careful of falling prey to confirmation bias – candidates resume may influence what you expect the interview to be like, and you may fall into the trap of confirming a decision you have already come to before meeting the person
- Arrange to have your job postings sent far and wide, not just in one location where the applicant pool may be of a certain demographic

## Virtual Interviews

- **The upside** of remote SOC hiring:
  - Your candidate pool just became *enormous!*
- **The downside:**
  - Remote interviewing can be more difficult, time consuming
- Consider
  - How will you conduct hands-on technical tests remotely?
  - Will latency and audio issues affect your experience / scoring?
  - How can you convey company culture and the office environment without them being there?
- **Pre-screening** to help limit candidate pool



### Virtual Interviews

While interviewing can be awkward in person, adding the layer of complexity for virtual interviewing can add further difficulty to the situation. At their best, virtual interviews are nearly the same experience as an in-person interview. At their worst, they can be terrible, filled with distractions, poor audio quality, echo, blurry cameras, and other technical issues.

To ensure the best experience with virtual interviews, give interviewees instructions on how to test their microphone and camera setup before the meeting and encourage them to do so. Suggest having headphones ready as well to eliminate potential echo issues. Then, when time comes for the actual interview, let the candidate know that if the connection is poor or there are other technical obstacles that prevent an otherwise clear communication experience, your preferred fix is falling back to phone calls, or a reschedule, or otherwise. As someone who has sat through blurry interviews with difficult to comprehend sound, trust me, it's *much* better to just call it off until the interview can be done in a clearly understandable way. It's not a good use of you or the candidates' time to try to fight through these problems, neither of you will get a real, fair impression of each other.

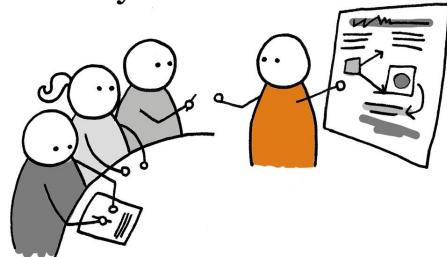
How to deal with the potentially enormous new candidate pool? Additional pre-screening can help. Consider crafting a pre-interview assessment that can be done in a self-guided fashion by the interviewee that can help screen out unqualified candidates before spending a longer in-person interview with them. This could utilize one of the many online services for such tests or be as simple as a set of questions and files that are emailed to the candidate with a requested time limit for returning them. Included in these questions could be conceptual as well as technical questions with log and PCAP samples. Keep in mind that with extra time, interviewees will have the ability to Google and work their way through answers, so try to select questions that will help differentiate candidates given that it will effectively be an "open book" test.

## Training Mindset

**Common Concern:** "What if we train them and they leave?"

Yes, training can be:

- Time-consuming
- Hard to show a direct ROI
- Potentially expensive



**Alternative phrasing:** "What if you don't train them and they stay?"

*"Train people well enough so they can leave,  
treat them well enough so they don't want to."*

— Richard Branson

## Training Mindset

When it comes to training, the number one worry of managers and organizations is "What if we train them and they leave?" This is a somewhat rational fear, training is time-consuming, hard to directly measure an ROI on, and potentially expensive to top it off. Consider though, would you rather have a set of well-trained employees that *might* leave, or a group of untrained employees that stay and drag down your operation with inaccuracy and inefficiency? I think you know the answer. Therefore, the best move is to attempt to find the sweet spot of providing training to your employees as much as possible while not going over budget to reduce the likelihood that they will leave afterward.

Many companies compensate for the risk of employee flight after training by having payback agreements for costly training, and that can be one way of minimizing risk if costs are important. Others anchor funding for the class to the understanding that the employee will attain a certification afterward. This, in theory, should guarantee the employee has internalized the lessons of the course that was paid for and brings that back to the workplace. Regardless of how your organization chooses to approach this topic, overall, the attitude of Sir Richard Branson (the billionaire owner of the Virgin Group which controls over 400 companies) is probably the right one "Train people well enough so they can leave, treat them well enough so they don't want to." If employees take their training and walk out the door, it's likely your organization should do some self-reflection and consider what would make them want to do so.

## Learning and Development Models – 70:20:10

Can we optimize learning and development?

- The commonly quoted "**70:20:10 model**"<sup>1</sup> is **dubious**
  - Based on self-reported survey answers of successful executives in 1988 study
  - The original studies did not identify this split, it came from a later book<sup>2</sup>
  - Reviews found a lack of empirical data to support it<sup>3</sup>
- What we *can* take from the study:
  - Both **formal and informal** training are important
  - Employees are always learning via multiple methods
  - Training on **relevant** issues at the **right time** was found to be very impactful
  - We cannot specifically quantify the impact of training because performance is a factor of both **capability and environment**

### Learning and Development Models – 70:20:10

How *much* training and what *type* should be aimed for? While researching an answer to this question I, unfortunately, came up short on concrete specifics but did run into some interesting generalizations as well as a refutation of a popular training framework. The optimum training type split is often said to be the magic 70:20:10 combination<sup>1</sup>, a model that has taken hold in numerous organizations. This model was popularized based on data from studies at the Center for Creative Leadership summarized in 1988 in "*Lessons of Experience: How Successful Executives Develop on the Job*" by McCall, Lombardo, and Morrison<sup>2</sup>, that asked successful executives to self-report where they spent their time learning.

Unfortunately, when looked at critically, this model apparently does not hold up to scientific scrutiny for multiple reasons<sup>3</sup>. The model seems to have been a generalization that was made in passing by the authors of the original study in their book originally published in 1996 called "*Career Architect Planner*" where they stated:

"Lessons learned by successful and effective managers are roughly:

- 70 percent from tough jobs
- 20 percent from people (mostly the boss)
- 10 percent from courses and reading."<sup>4</sup>

From this statement, the 70:20:10 rule apparently took hold as gospel. When later researchers went looking for the origin and to study the truth of the matter, they concluded: "From our review it is clear that there is a lack of empirical data supporting 70:20:10 and, while the above-mentioned sources are frequently credited, there is also a lack of certainty about the origin."<sup>5</sup>

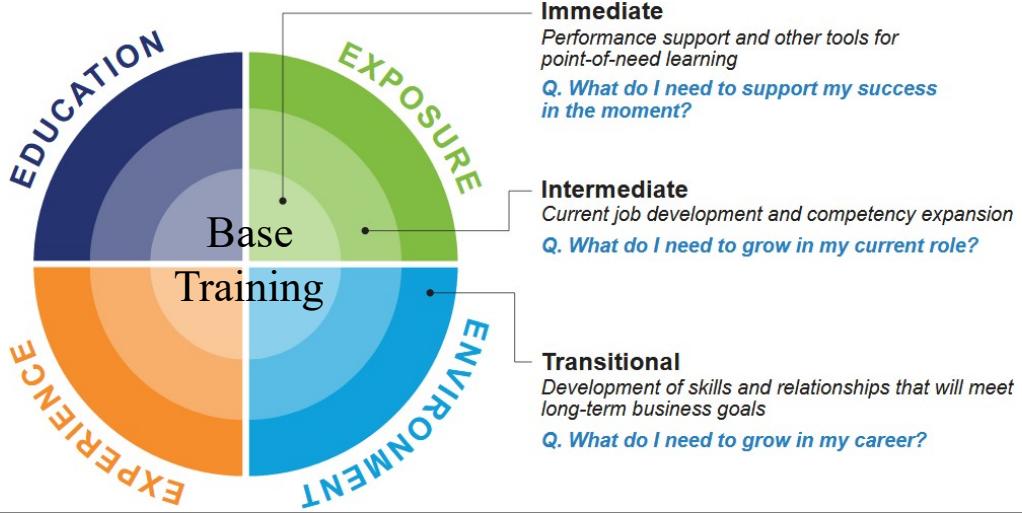
Is there anything we *can* say about learning and development from this study? According to the Association for Talent Development there are some key lessons from the original 1988 studies that can be generalized<sup>3</sup>. They are that formal and informal learning methods are both important. Executives reported that formal coursework was included as an event that "made a significant difference to them" and that this coursework had the properties in common of being both relevant and being administered at the appropriate time for that individual.

Unfortunately, they are not able to isolate and quantify the impact since performance, as we will discuss later in the course, is a factor of both individual capabilities and that person's work environment.

- 1 [https://en.wikipedia.org/wiki/70/20/10\\_Model\\_\(Learning\\_and\\_Development\)](https://en.wikipedia.org/wiki/70/20/10_Model_(Learning_and_Development))
- 2 <https://www.amazon.com/Lessons-Experience-Successful-Executives-Develop/dp/0669180955>
- 3 <https://www.td.org/insights/70-20-10-where-is-the-evidence>
- 4 <https://www.amazon.com/Career-Architect-Development-Planner-5th/dp/193357822X>
- 5 [https://www.deakinco.com/uploads/Whitepaper/dc\\_70-20-10wp\\_v02\\_FA.pdf](https://www.deakinco.com/uploads/Whitepaper/dc_70-20-10wp_v02_FA.pdf)

## Continuous Learning Model<sup>1</sup>

- Describes needs and contexts in which employees learn



### Continuous Learning Model

An alternative and evolution to the 70:20:10 model is the Continuous Learning Model by Enterprise Learning and Talent Development firm Bersin<sup>1</sup>. This model that still acknowledges the importance of the formal/informal split but picks up where 70:20:10 left off and further defines important pieces of an employee training plan to consider.

One thing the Continuous Learning Model brings to the table is the idea of the Immediate, Intermediate, and Transitional split of training needs. **Immediate** training needs are those which are required at the current moment. For example, when new analysts start the job, what will be the "baseline" training. **Intermediate** training considers further development for the *current* role into more advanced territory and a deeper understanding of the skill. Finally, **Transitional** training is the long-term training an employee needs to advance their career into their next role or promotion. Breaking down your training plan for each person into these items can help clarify and prioritize what is most important and when it is needed.

Secondarily, the Continuous Learning Model usefully breaks up learning into four different contexts in which employees learn, represented by 4 "E's":

**Education** – Formal education with a defined beginning and end

**Exposure** – Learning that involves interaction and relationships with others – building connections with other professionals and thought leaders

**Environment** – Tools, systems, and other infrastructure employees use on the job to learn

**Experience** – things that occur while employees are in the workplace including stretch assignments, job rotations, and special projects<sup>2</sup>

This is a more detailed and useful mental model that helps us not only understand the different formal and informal ways employees will develop, but also gives a framework for making a training plan for each individual.

1 <https://mkto.cisco.com/rs/cisco/images/Bersin-Continuous-Learning-Cisco-Collaborative-Knowledge.pdf>

2 <https://www2.deloitte.com/content/dam/Deloitte/at/Documents/human-capital/research-bulletin-2014.pdf>

## Training Plan

An example SOC-based formal training plan based on SANS curriculum:

Immediate	Intermediate	Transitional
<ul style="list-style-type: none"><li><b>SEC450:</b> BlueTeam Fundamentals – Security Operations and Analysis (GSOC)</li><li><b>SEC401:</b> Security Essentials (GSEC)</li></ul>	<ul style="list-style-type: none"><li><b>SEC511:</b> Continuous Monitoring and Security Operations (GMON)</li><li><b>SEC586:</b> Blue Team Ops – Defensive PowerShell</li><li><b>SEC504:</b> Hacker Tools, Techniques, Exploits, and Incident Handling (GCIH)</li><li><b>SEC503:</b> Intrusion Detection In-Depth (GCIA)</li><li><b>SEC595:</b> Applied Data Science</li><li><b>FOR610:</b> Reverse Engineering Malware (GREM)</li></ul>	<ul style="list-style-type: none"><li><b>SEC555:</b> SIEM with Tactical Analytics</li><li><b>SEC530:</b> Defensible Security Architecture</li><li><b>FOR500:</b> Windows Forensic Analysis</li><li><b>FOR578:</b> Cyber Threat Intelligence</li><li><b>MGT512:</b> Security Leadership Essentials for Managers</li></ul>



### Training Plan

This slide shows an example of a SANS-based formal training plan that might be appropriate for a new SOC analyst. The idea isn't to take all of these classes, but to pick which ones would be most appropriate based on the new employee's experience, specialization interests, and career development plan. Of course, your organization may have training requirements such as US DoD 8140 that mandate certification of analysts under certain training programs before they can be approved to do the job.

In the Immediate column, we have courses that are often used as "baseline training" for everyone to know how to do the job they were just hired for. In the Intermediate column, we have specialty focus classes that teach analysts details of advanced topics that would still be useful in their analyst role (perhaps a Tier 2-3 as well). Finally, we have the Transitional training which are courses that are often used for SOC-adjacent architect, forensic, threat intel, or management jobs that an analyst may be interested in pursuing in the long term.

## Certifications

- Should you have your team pursue certifications?
- **Yes!** Why?
  - Most analysts will desire the training that goes with them
  - Employees can be proud in the achievement of a major accomplishment
  - It commits the training you paid for them to attend into long term memory, brings better ROI
  - Management can brag about the qualifications of the team
  - You can use it to justify funding the training cost
- 2019 ISC2 Global Information Security Workforce Study:  
*"those working for organizations that pay for their certifications are significantly more satisfied in their role than those working for organizations that don't"*



### Certifications

Once you have decided on a training plan for your new and current employees, consider your ability to push them toward certification. If it's within your budget to fund certification attempts for your employees, should you do it? While some organizations worry that will make their staff more marketable and likely to leave, again, remember the Richard Branson quote from earlier. if someone is staying only because they have to, you have different problems that need to be addressed.

Assuming you are able to provide certification attempts and are interested in doing so, what are the benefits? The biggest one in my personal experience and what many SANS students echo as well is that you get more value out of the training when you know you'll be taking the associated certification test. With the added pressure to perform, students tend to pay closer attention, take better notes, and internalize the material much more than they would if they knew they won't be asked about it again. Not only that, those who have taken a GIAC test know that passing means performing a *thorough* review of the material and committing it to long term memory, exactly what you want out of the training in the first place!

Employees who are funded to take classes and the associated certificate exam will not only appreciate the offer, but it will send them the message that they are worth putting an investment into and they will no doubt be proud of their ability to overcome the challenge. Once they are finished, you can use their accomplishment to tell upper management about how awesome of a team you have and the progress you've made as a result of their new skills. Be sure to close the loop on any new initiatives, detection techniques, or anything else that comes as a result of training. This feedback on ROI to those who pay the bills will help to ensure the training budget keeps coming.

## Team Creation, Hiring, and Training Summary

- Team Org chart
  - Which functions will be offered and "under" the SOC
  - Tiered or Tierless
  - Continuous coverage
- Recruitment
  - Work your network!
  - Ask for recommendations
  - Put time into the job description
  - Go where infosec lovers go
- Interviewing
  - Standardized structure
  - Use bias-reduction techniques
  - Real-time scoring
- Training
  - Informal and formal sources are important
  - A big selling point for new recruits
  - Break into Immediate, Intermediate, and Transitional
  - Add certifications!

### Team Creation, Hiring, and Training Summary

In this module, we covered the pieces of building a strong team from designing the org chart, to recruitment and interviewing, to getting people onboarded and set up with the right training to help them be successful. Whether you have a SOC already or are building one from scratch, these key items to remember will help you avoid the biggest pitfalls when it comes to this stage of setting up an all-star team.

# Course Roadmap

- *Book 1: SOC Design and Operational Planning*
- Book 2: SOC Telemetry and Analysis
- Book 3: Attack Detection, Threat Hunting, and Triage
- Book 4: Incident Response
- Book 5: Metrics, Automation, and Continuous Improvement

## SECTION I

### Introduction

#### SOC Design and Planning

- SOC Functions
- SOC Planning
  - *Exercise 1.1: Threat Actor Assessment*
  - Team Creation, Hiring, and Training
    - *Exercise 1.2: Attack Path Development*

#### Building a Strong Foundation

- Building the SOC
- SOC Tools and Technology
  - *Exercise 1.3: Developing and Implementing SOC Playbooks*
  - Protecting SOC Data and Capabilities
  - Summary and Cyber42 Simulation – Day 1



This page intentionally left blank.

## EXERCISE 1.2

# Exercise 1.2: Attack Path Development

### OBJECTIVES

- Take an "attacker's eye" view of your organization
- Pre-meditate attacks and how they might occur
- Identify key assets, users and systems
- Assess your capability to protect and defend against high impact cyber attacks
- Gain an improved understanding of your defensive posture



#### **Exercise 1.2: Defining Your Assets and Adversaries**

Please go to Exercise 1.2 in the MGT551 Workbook or virtual wiki.

# Course Roadmap

- *Book 1: SOC Design and Operational Planning*
- Book 2: SOC Telemetry and Analysis
- Book 3: Attack Detection, Threat Hunting, and Triage
- Book 4: Incident Response
- Book 5: Metrics, Automation, and Continuous Improvement

## SECTION I

### Introduction

#### SOC Design and Planning

- SOC Functions
- SOC Planning
  - *Exercise 1.1: Threat Actor Assessment*
  - Team Creation, Hiring, and Training
    - *Exercise 1.2: Attack Path Development*

#### Building a Strong Foundation

#### **• Building the SOC**

- SOC Tools and Technology
  - *Exercise 1.3: Developing and Implementing SOC Playbooks*
  - Protecting SOC Data and Capabilities
  - Summary and Cyber42 Simulation – Day 1



This page intentionally left blank.

## In This Module (2)

- Once the SOC is planned, time to build!
- Physical space considerations
  - Desk layout and space planning
  - Work area specifics
- Virtual/Remote SOC
  - Onboarding
  - Managing
  - Collaborating

### In This Module (2)

In this module, we'll walk through some of the considerations for the physical build of the SOC as well as SOC IT requirements. Whether you are building your first SOC or already have one, what follows are several tips and considerations that will keep your analysts working as efficiently, safely, and comfortably as possible. We'll start out covering the physical SOC space and analyst work areas then shift to discussing SOC private networks. The SOC has unique needs in accessing the internet as well as in how it works with the rest of the constituent environment and careful design of a private network for the SOC is often required.

## SOC Physical Space Overview

### Room-Level Considerations

- Room Location
- Desk layout
- Video walls
- Conferencing
- White boards
- Conference Rooms
- Lighting

### Individual-Level Considerations

- Computer Requirements
- Monitors
- Desks
- Chairs
- Keyboards / Mice
- Ergonomics
- Storage



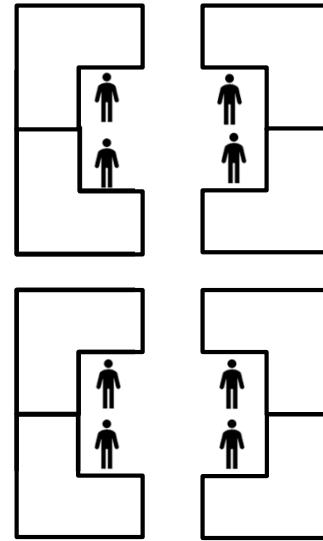
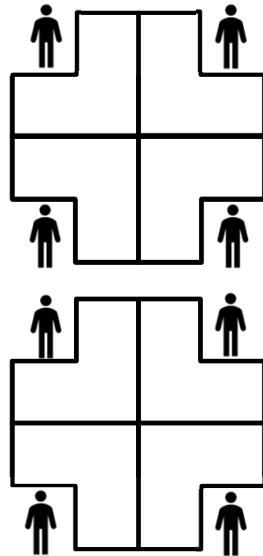
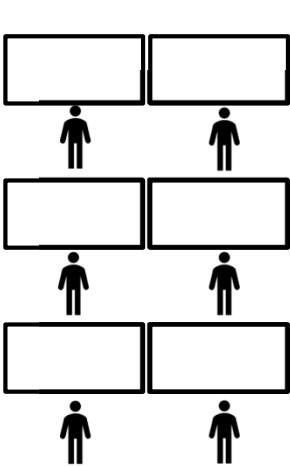
### SOC Physical Space Overview

When building out the physical space used for your SOC there are lots of considerations to keep in mind both at the room level and at the individual desk / work area level. In the next section we'll touch on some specifics you should consider in each of these categories. You probably already have a vision of what a SOC room looks like in your head, and it's probably mostly correct, but there are some nuances that can make life easier if they are planned for ahead of time and do not need to be added on later when it would be more disruptive and the cost is higher to modify the space.

For the room, some things you want to consider will be the location within your building, the desk layout within the room, ports, and power available at each desk, video walls, whole-room conferencing capabilities, wall-size whiteboards, lighting, and dedicated conference rooms.

For each analyst desk you should consider the requirements for how many computers will be needed, the size of the desks, how many monitors each person will have, ergonomics, storage available, and even keyboards and mice.

## Room Layout



SANS

MGT551 | Building and Leading Security Operations Centers

93

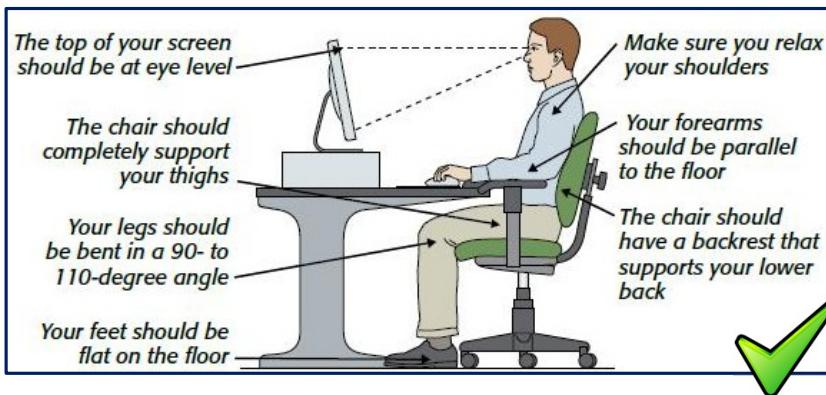
### Room Layout

When it comes to SOC room layout, there are a few popular options – rows of desks all facing the same direction, pods of desks where people are more or less facing each other, and the opposite where people sit in groups with their backs to each other. Each layout has its strengths and weaknesses.

- Rows
  - Pro - Good for collaboration with people in your row, everyone faces the same direction (video wall).
  - Con – Hard to talk to people in other rows, may be hard to get in and out of rows, people in front might feel like everyone behind is watching their screen.
- Facing each other
  - Pro – Easy to talk to people, often easier to navigate to desk than rows, no "looking over the shoulder" feel
  - Con – Collaboration is more difficult since you can't always wheel over and see someone else's screen nearby, not everyone faces the video wall.
- Facing away from each other
  - Pro – Easiest to collaborate by just turning around, no "looking over the shoulder" feel
  - Con – While each group can work well together, not everyone faces the video wall, collaborating with *other* groups can be more difficult unless you keep an extra chair in each pod.

## Chairs and Desks

Adjustable (standing?) desks and chairs seem like minor points, but make a **major** difference over time



### Chairs and Desks

It should go without saying, but having an ergonomic workspace is actually a very serious issue, and it is one that is unfortunately often ignored. Being forced to sit in a chair or at a desk that's even a little too high or low for thousands of hours a year can easily add up to unanticipated health consequences. While nice chairs and desks can be expensive, they last an incredibly long time, and the flexibility is likely a tiny fraction of what the medical bills associated with poor posture, eye strain, and neck and back problems may cost. The new style "standing" desks can even be purchased for a relatively low cost if you buy the hand-cranked versions. Don't forget to consider the actual size of the desktop as well. As someone who has sat at a desk that was not nearly wide enough, I can tell you it is not fun. As a manager of people whose job will be sitting at a desk year-round, it's worth it for you to fight for the budget needed to get the right workspace.

## Keyboards and Mice

- Human interface devices are a highly personal choice
  - Analysts will spend 2000 hrs. per year using it
  - SPEND SOME MONEY TO GET NICE THINGS – It's worth it!
  - Ideally, provide a budget, let analysts choose their own
- The difference in crappy vs. good mice/keyboards is huge
  - The cost difference is ~\$20/year...you can do this
  - They help prevent RSI
  - Connect to multiple devices
  - Productivity is higher



### Keyboards and Mice

While you might think discussing keyboards and mice seems silly, they are items that can be a *significant* quality of life improvement for analysts with only a slightly higher expense. Again, analysts will be literally touching these items for thousands of hours a year, don't you think it's worth spending a bit more to get something that's enjoyable to use, especially considering the potential productivity-enhancing features of upscale keyboards and mice?

For mice, I am a big fan of the Logitech MX Anywhere and Master series<sup>1</sup>. They have a scroll wheel that can click into frictionless mode to allow high-speed precision scrolling, I've found this to be invaluable when dealing with large web pages, documents, and log files. They also allow simultaneous connections to multiple devices; wouldn't it be nice to use a single mouse for every PC on your desk?

As for keyboards, there are plenty of outstanding options, Logitech again makes a line of wireless, slim keyboards that support connecting to and switching between multiple devices easily. I find these, combined with multi-device mice to be a great combination for SOC analysts with multiple machines they might need to frequently switch back and forth between.

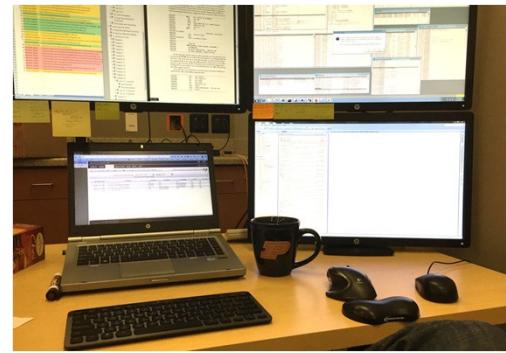
An alternative option is mechanical-switch keyboards, which although are not typically wireless, some people are fanatical about. They allow customization of key pressure and types and are built to last much longer than the average keyboard. If you haven't typed on one before, they're worth trying out as the experience is *much* nicer than the cheap membrane keyboards that come with the average PC. The thing to consider with mechanical keyboards however is the associated noise, some key types are louder than others – Cherry MX Brown and Blue key types are the most recommended key types for general typing. Both the Brown and Blue key types have a noticeable click feel on the way down, but Blue keys are more audible. A room full of people typing furiously on Cherry MX Blue<sup>2</sup> keys (which is what this class was written on) might be distracting to some, but sure would sound impressive to visitors. :)

1 <https://www.logitech.com/en-us/mice>

2 <https://www.youtube.com/watch?v=3a4IfFO9KIs>

## Monitors

- Working with multiple windows is a *constant* in the SOC
- Analysts MUST have enough screen real estate
- Options:
  - 1 *large* 4k screen (to minimize scaling)
  - Ultra-wide monitors
  - 3 smaller (~24") monitors
- Consider pixel density, scaling
  - 23" 1080p = 96ppi – no scaling required
  - 32" 4K UHD = 137ppi – scaling required



Author's actual workspace in 2015

## Monitors

Monitor space is another workspace requirement that many analysts are fanatical about, and with good reason. Keeping watch of a SIEM, web browser, email, ticketing system, and other security tools all at once is a lot of windows to attend to. Having to alt-tab between windows all day it can be an enormous impediment to productivity, as well as a nag that chips away at your sanity. Nowadays additional monitors are very inexpensive (a 24" 1080p screen is roughly \$100 on Amazon) and each analyst should have at least 2, if not more, at their disposal. This becomes even more important if your analysts will be utilizing multiple PCs at once.

Options for screen real estate include multiple monitors, but newer large 4K UHD and ultra-wide screens are viable options as well, although they may come with a slightly higher cost. If you're planning on purchasing *large* screens with high resolution, do not forget to calculate the pixel density. Buying a 4K 24" screen instead of a 1080p screen will not increase usable screen estate for analysts, they'll have to use 200% scaling effectively making the picture nicer and sharper, but not expanding workable room. Consider that a 23" 1080p screen is 96 pixels per inch (ppi) and doesn't need scaling, when going larger, calculate using that as a reference point. Sites like pxcalc.com can do the math for you.

## Other Details to Consider

- TVs in the front of the SOC
  - Ability for any analyst to share video to front TV
- Wall-size whiteboards
- Wall space for posters / reference material
- In-SOC conference room for meetings
  - Speakerphones and room webcams for conferencing
- Lighting / ventilation / outside windows
- Shared workspaces? Consider health implications
- Private networking, servers, test labs

### Other Details to Consider

Some other details to consider for the SOC physical space:

- Video wall – Most SOCs have some sort of whole-room orient display up at the front. These can be used for situational awareness such as watching incoming events and ticket counts, spam waves, live information sources like Twitter, or any other data that should be acted upon immediately. It's also a nice feature to have the ability to cast the screen of any team member up onto the screen for the room to see.
- Wall-size whiteboards – Having a dedicated place for team members to collaborate and draw out ideas is great for productivity.
- Wall space for posters and reference material – Analysts in a SOC have a lot of information to remember and having a place it can be displayed and found with a glance can assist in day to day work.
- In-SOC conference rooms – If you have the space, ask for a dedicated conference room to be created *inside* the SOC. This way you won't have to fight for room capacity with other teams and can have a private space to talk whenever it's needed.
- Lighting / ventilation / outside windows – Do you want your SOC to have lots of natural light, or be deep within the belly of your company's basement? The choice is yours but remember most people enjoy natural light and a view. Obviously, ensure the HVAC in the room you plan on using will be able to keep up with the demands of many people sitting in it all day.
- Shared workspaces – If you're going to run a 24 x 7 x 365 operation where analysts will need to use the same desk as someone else on the previous shift, make it easy for each person to bring their own mouse, keyboard, and headset. You don't want one person who is sick affecting 3-4 more who share the same desk.

## An Example SOC



SANS

MGT551 | Building and Leading Security Operations Centers

98

### An Example SOC

Given all this discussion on physical space, it might be interesting to get a peek into other organizations' physical SOC layout. This slide shows a picture of a small SOC that implements many of these ideas. As seen in the photo, there's a video wall, multiple monitors, nice chairs (Steelcase Leap with headrests), better keyboards and mice being used, world clocks, and more. In this configuration, analysts can easily turn around and collaborate with each other yet have plenty of space at their "L" shaped desks. One whole wall of the room was a whiteboard for drawing and collaboration, while the other side was a pin-up board used for posters and other decorations.

## Additional SOCs



Target's Cyber Fusion Center (2015):

(<https://corporate.target.com/article/2015/07/cyber-fusion-center>)



MITRE's CSOC (2010):

(<https://www.mitre.org/publications/project-stories/mitres-cyber-security-operations-center-helps-sponsors-keep-networks-secure>)

## Additional SOCs

For reference, here are a few other SOCs that had publicly available photos. The Target photo shows what a large SOC may look like with the desks arranged for people to face each other in each group, while the MITRE SOC shows the row-based setup in a smaller SOC area.

## Virtual SOC

COVID-19 changed the working world VERY quickly

- **Working from home** becomes the norm
- **Digital transformation** on fast forward
- **Online communication** becomes the norm
- Most organizations considering **more WFH in the future**
- For the SOC
  - Many previous students report their SOC is now moving increasingly to the remote/ work from home model
  - The previous principles still apply when at home!
  - Infrastructure required to manage it changes



### Virtual SOC

While having a physical SOC has been the norm for decades, in 2020, COVID shifted the entire world to a work-from-home default. Not only did working from home become the new normal, but companies had to *quickly* adopt the tools, services, and online communication platforms to support it, and the SOC is no different. No longer can we restrict access to security tools to those on-premise. Most organizations must now consider how those on the security team can remotely, and yet still securely, access all the same data, services, and security appliances they could within the office, on the outside as well.

While many orgs may have already moved in this direction and supported this kind of setup pre-covid, it was still likely not the default mode of operation. No longer can that be the case. Experience teaching in 2020 and polling students in class has shown in large part that many SOCs are now interesting in a more permanent work from home setup, with a remote as default way of working. The principles previously explained do not go away in this new world, in fact, if anything, things in some ways become more complex. Fortunately, some of that complexity can be balanced out with the availability of cloud-based systems and networks that can be created in ways similar to what may have been done on-premise in the previous paradigm. Over the next few slides, we'll walk through some of the considerations for a work from home as default SOC.

## Extra Considerations

### Extra considerations for a virtual SOC

- **Communication** software becomes *much* more important
- Additional complexities of remote **team management**
- **Virtual collaboration** for day-to-day work and incidents
- Secure **connectivity** to SOC specific network and SOC tools
- Interviewing and **onboarding new team members**
- **Ergonomics** of home working



### Extra Considerations

For those teams who do not have a physical space, there are a few special considerations, as well as modifications to the previously mentioned issues.

- Communication – In a virtual SOC, the capability to simply turn to your left or right and ask a question disappears and the team becomes reliant on digital communication methods. Whether your team uses Slack, Skype for Business, Microsoft Teams, or any other software, ensure team members feel comfortable chatting with each other on a frequent basis. Schedule a yearly and regular physical team meetup, if possible, so everyone can meet each other in-person. In past teams I have been on, there has been a marked difference in communication after a physical team meetup where everyone got to see each other get to know one another in an "outside of work" environment.
- Team Management – Managing a remote team takes extra dedication as the clear signs of struggle or interpersonal conflict may become hidden when an employee is at home on the other end of a keyboard.
- Remote secure connectivity – Without a physical SOC network or desk to be physically present and connected to, remote connection technologies and cloud virtual desktops can be leveraged as an alternative. Though the team may be physically dispersed, this doesn't change the requirement to both secure analysts' role as privileged user and separate that from their role as malicious file investigator.
- Interviewing and Onboarding – Building your team virtually and getting them set up needs extra consideration, given that you will not be able to be physically present to help them through the process.
- Ergonomics – While having a remote team does mean people are free to design their workspace as they please, that does not mean people will necessarily want to use their money to improve their work area. To ensure your team is still getting the benefits of a productive and healthy work environment, a stipend for monitors, chairs, or other approved workspace related items is an alternative solution for remote workers. This type of investment can easily pay for itself over time through minimized doctor's visits for work related repetitive stress injuries or working with bad posture in an improper chair.

## Onboarding in a Virtual SOC

Planning onboarding of new team members is a **major** consideration for virtual teams

- Common Issues:

- How will the new team member get to know the rest of the team?
- How will they be trained on the SOC tools?
- Who will be their "go-to" person for questions, company culture, and more?
  - Will they be comfortable asking numerous, small questions?
- How will accounts be requested / created?
- How will you encourage an ergonomic work from home setup?

### Onboarding in a Virtual SOC

In person onboarding for new employees can be daunting enough – piles of information to read, training to take, people to meet, and tools to get access to, getting it done virtually adds additional layers of complexity. Some aspects of onboarding in a virtual SOC that deserve extra planning and consideration include:

- Introductions to the rest of the team
- SOC tool training – while vendor training will likely be possible in a self-guided way, organization specific usage and training should be planned for with an assigned mentor
- New account registration for on-premise tools and systems, as well as cloud services
- Ergonomics – With employees working using whatever furniture is available within their homes, you don't want them working long term from a couch or in a lawn chair. Provide resources, suggestions, and perhaps even a stipend where for setup of a proper workstation. SOC analysts spend *lots* of time on their PCs, so a sustainable, ergonomic setup is a must.
- Company culture, general questions and more – beyond SOC related and technical questions, there will likely be numerous other smaller questions that new employees have. In person, turning around and asking a question wouldn't seem like a big deal, but new employees, unfamiliar with people on the team, may be hesitant to ask what they might fear asking what they think will be too many small nagging questions via online chat. If these issues stack up over days, it can lead to frustration. As a manager, make it abundantly clear that you're available for any and all questions and that they shouldn't feel bad about asking for directions.

The Wall Street Journal recently suggests<sup>[1]</sup> pairing employees with someone that can help new employees navigate not just the daily ins and outs of working the job, but also as a "culture buddy", to help them figure out what is expected of new employees, the tone and levels of formality in communication and more.

[1] <https://www.wsj.com/articles/its-not-just-working-remotely-hiring-and-onboarding-go-virtual-too-11586963419>

## Managing a Virtual SOC Team

- As a manager of a remote team...
  - You will not know your employees as well
  - Will not see their day-to-day experience and interaction with others
  - Might miss signs of struggle
- Suggested Tactics for Remote Team Management<sup>1</sup>
  - Establishing daily structured check-ins
  - Provide and use multiple communication options (video, audio, chat, email)
  - Establish "rules of engagement" / expectations
  - Encourage and provide time for social interaction

### Managing a Virtual SOC Team

What about the purely team management aspects of working with a virtual team? Well, consider some of the issues listed above – you need to ensure the team has guidance, stays on track, and has the help they need, all without being able to physically work with them. It would be *much* easier to miss signs of a team member struggling in a remote environment since you don't see them most of the day and cannot overhear their discussions with the rest of the team as you might in a SOC.

What to do? The Harvard Business Review article below suggests the following strategies<sup>1</sup>:

- To overcome communication issues, creating a regular, daily check-in meeting with team members can help make sure you're aware of any problem that arise as quickly as possible, and keeping those lines of communication open with team members helps ensure they will be comfortable telling you about problems.
- Provide and use multiple modes of communication to stay in touch with your team. Email is not a very personal medium, even if you send a lot of them, body language and tone is unreadable from an email, therefore video, calls, chat and more should all be utilized as appropriate.
- Setting rules for engagement and expectations around the frequency, nature, and type of communications that will be used within the team. For example, you will likely want to have a daily stand-up or shift changeover meeting with all members of the team. But what if something urgent pops up (as it often does in the SOC), declaring that audio calls should be used as the default to ensure fast, efficient communication can be helpful, and letting messaging tools used for casual conversation. Methods for communication off-hours should also be spelled out, and when and why to use them.
- Social interaction about topics beyond work – Encouraging team members to know each other as people, and not just coworkers helps build bonds within the team. A planned 5-10 minutes of "structured unstructured time" at the beginning of daily shift change meetings can be used to give team members space to catch up about whatever is going on in the news, their lives, or whatever else they'd like to talk about to stay connected.

[1] <https://hbr.org/2020/03/a-guide-to-managing-your-newly-remote-workers>

## Virtual Team Collaboration

- More than just meetings – consider all types of communication
  - Chat – one on one, small group, team wide
  - Screen sharing and remote assistance
  - Document writing collaboration / Shared files storage
  - Whiteboard-style and collaboration
  - Video conferencing
  - On-call notifications
- Popular Solutions
  - **Whiteboards** –Microsoft OneNote, Microsoft Whiteboard, Google Jamboard, AWW (A Web Whiteboard – [awwapp.com](http://awwapp.com))
  - **Document Collaboration** –SharePoint, Google Docs, OneNote
  - **Chat** – Slack, Teams, Google Chat, or self-hosted (Rocket.Chat, Mattermost)

### Virtual Team Building and Communications

When trying to stay in constant communication with a remote team, your suite of communication tools needs to not only assist you but encourage and improve communications to make up for the lack of face-to-face interaction. Consider the multiple methods in which people work together and consider if you have an acceptable tool for each that works for the team. Yes, one on one / group chat and video conferencing may be the majority of interactions, but don't forget about other methods of collaboration and communication. Collaborative document creation, incident response tasks that require visual diagrams, and being able to reliably video conference/call those via PC and mobile device are important as well. The goal is to find the right mix of tools that allow your employees to communicate whether they are sitting at their machine, need to call in to a meeting, or see a document while on the go.

There are numerous options for virtual whiteboards that can be shared across multiple users, many of which support multiple platforms and touchscreen input from a stylus or Apple Pencil as well. Popular options include Google's Jamboard (which requires a G-Suite business account), another is AWW or A Web Whiteboard (free tier available), finally Microsoft's own free OneNote, which has whiteboard like drawing functionality combined with text and more, and finally Microsoft's dedicated Whiteboard application available free in the Microsoft Store.

For chat, the obvious first choice may be whatever communication suite your organization offers, whether that's Microsoft / Google tool based or otherwise. But what if you don't want to have your private team communications owned by a third party? In these cases, tools like Rocket.Chat and Mattermost, which are privately hosted, open-source Slack alternatives, can be utilized.

## Summary / The Keys to Great (Virtual) Teamwork

- HBR describes important goals for the ideal "4-D" team<sup>1</sup>
  - **4-D = Diverse, Dispersed, Digital, and Dynamic**
- Key factors in 4-D team success<sup>1</sup>:
  - **Direction** - explicit goals that are challenging, consequential, but achievable
  - **Strong structure** - members and processes that discourage destructive behavior and promote positive dynamics
  - **Supportive context** – Training, rewards for performance, access to required data, and resources required to do a good job
  - **Shared mindset** – ensuring all members have same info and mindset, seeing the team as a whole instead of multiple sub-groups

### Summary / The Keys to Great (Virtual) Teamwork

The keys to success in virtual (and any teams) have been studied over and over. While there are many factors that go into team success, HBR has a great article summarizing the research and discussing four key items that have the higher correlation with likelihood of success for teams that meet the description of "4-D"<sup>1</sup>. A 4-D team is described as one that is diverse, dispersed, digital, and dynamic, and the listed factors that are the highest predictors of success are as follows:

- **Direction** – The team must have a direction all understand the goals they are trying to meet. Those goals should be challenging, but doable, and meaningful as well (no one wants a challenging but ultimately pointless project).
- **Strong structure** – The right structure, composition, and balance of skills to encourage collaboration and discourage destructive behaviors; for example, each team member may not have the desired combination of technical and social skills to foster effective collaboration, but the team overall should have the right mix of skills.
- **Supportive context** – A work environment that supports the team members in getting the job done in all factors. Training, access to the required data, adequate resources and funding to get the job done, and rewards and clear recognition of performance.
- **Shared mindset** – One of the most important, in the authors opinion. The article gives the example of a team that is physically located in two different cities. When the group does have a chance to do an in person full-team meeting, both groups choose different hotels, and stick to different sides of the table at meals. This type of behavior and sub-divisions within a group is a predictor of poor performance, as it is a symptom of dividing the one group into "us" vs. "them", which leads to problems. Teams should ideally feel, regardless of physical location, background, or other divisions within the group, that they are acting as one team.

How can you evaluate your meeting of these goals? The article suggests looking at 3 criteria – team output, collaborative ability, and members' individual development. *"The ideal approach combines regular light-touch monitoring for preventive maintenance and less frequent but deeper checks when problems arise.*

*For ongoing monitoring, we recommend a simple and quick temperature check: Every few months, rate your team on each of the four enabling conditions and also on the three criteria of team effectiveness. Look in particular at the lowest-scored condition and lowest-scored effectiveness criteria, and consider how they're connected. The results will show where your team is on track as well as where problems may be brewing."<sup>1</sup>*

[1] <https://hbr.org/2016/06/the-secrets-of-great-teamwork>

# Course Roadmap

- *Book 1: SOC Design and Operational Planning*
- Book 2: SOC Telemetry and Analysis
- Book 3: Attack Detection, Threat Hunting, and Triage
- Book 4: Incident Response
- Book 5: Metrics, Automation, and Continuous Improvement

## SECTION I

### Introduction

#### SOC Design and Planning

- SOC Functions
- SOC Planning
  - *Exercise 1.1: Threat Actor Assessment*
  - Team Creation, Hiring, and Training
    - *Exercise 1.2: Attack Path Development*

#### Building a Strong Foundation

- Building the SOC
- **SOC Tools and Technology**
  - *Exercise 1.3: Developing and Implementing SOC Playbooks*
  - Protecting SOC Data and Capabilities
  - Summary and Cyber42 Simulation – Day 1



This page intentionally left blank.

## SOC Tools and Technology

In this module:

- Attack prevention principles
- Collection and detection:
  - Foundational SOC capabilities
  - "Next-Gen" SOC capabilities and technologies
- Triage, investigation, and response
  - SIEM, Threat Intelligence Platforms
  - Incident Management Systems
  - Information Organization - Playbooks, Use Cases Databases, Software Repositories, SOC Knowledgebase



### SOC Tools and Technology

In this module, we'll be stepping through some of the key capabilities in terms of monitoring, data collection, and analytics, that you will need to run an effective SOC. We'll also cover the supporting analysis tools, incident management systems, playbooks, and documentation and software repositories. Each of these systems plays a key role in the collection, detection, triage, investigation, and I.R. process. They also play the auxiliary roles of organizing playbook, use case, and analytic data, custom software and tools written by the team, and store any documentation and guides that will have to be kept easily accessible for analysts.

## Zero Trust Principles for Prevention

Domains of consideration<sup>1</sup>:

- **Identity** – Strong authentication for verified, least privilege access
- **Endpoints** – Monitoring and enforce device health and compliance checks and asset ownership verification
- **Data** – Encrypt, restricting access based on classification, labeling
- **Apps** – Monitoring for shadow IT, anomalous activity, and unexpected / unauthorized permission and configuration changes
- **Infrastructure** – Monitoring for configs, versions, and access, provide just in time access, auto-block and detect anomalous behavior
- **Networking** – Limit attack blast radius using software defined perimeters, micro-segmentation, etc.
- **Visibility, Automation and Orchestration** – For monitoring and management of generated data



### Zero Trust Principles for Prevention

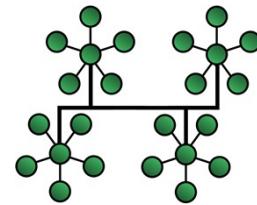
Before diving into detection technologies, a solid prevention strategy should be conceived to stop as many attacks as possible before they become a problem. In a modern network, this means continuous moving further towards "zero trust" networking concepts. While we will not take a detailed dive into these concepts in this class, it is worthwhile to consider your capabilities in each of the above domains. A network infrastructure filled with users, data, and endpoints that each can verify and authenticate interactions, and moving towards least privilege for each creates a *much* more defensible setup for your organization, giving the security team time to focus on the relatively fewer intrusions that do succeed.

For an excellent resource on Zero Trust principles, especially when it comes to applying them in an active directory environment, see the footnote below for Microsoft's Zero Trust Deployment advice in the above domains.

[1] <https://docs.microsoft.com/en-us/security/zero-trust/>

## Foundational Network Collection and Detection Technologies

- Network Flow Records
  - Network Transaction Data
  - Full Packet Capture
  - Next-Gen Firewalls
  - Network IDS/IPS
  - Email Filtering
- Event Focused
- Alert Focused



### Foundational Network Collection and Detection Technologies

Building out the foundation of a collection and detection program will involve implementation of the fundamental technologies that SOCs have been using for years. These will need to cover both network visibility and endpoint visibility so that you can see both data in motion on the network as well as what is happening on the endpoints. We'll cover both of these items over the next two slides

On the network side, it's important to be able to see what is talking to what, which protocols they are using, and ideally, have a full packet capture of the transaction. This type of network visibility can be thought of as falling under the "collection" function of the SOC as its goal is recording all events that happen on the network, whether they are known to be malicious or not. Supplementing these capabilities are the threat-centric, alert focus detection capabilities such as next-gen firewalls (which can also function as an event collection device), network-based IDS/IPS, and dedicated appliances for email filtering and file sandboxing, all of which can help bolster this capability by identifying any malicious files as they travel across the network.

## Foundational Endpoint Collection and Detection Technologies

- Centralized Log Collection (SIEM)
  - Authentication, System, Application, Process Creation, Network Connections, Scripting
- Anti-Exploitation
- Host Firewall
- Application Control
- Anti-virus
- Host IDS/IPS

Event  
Focused



Alert  
Focused



### Foundational Endpoint Collection and Detection Technologies

On the host side, centralized log collection of key system events will form the basis of your endpoint collection capability. In addition to the events and visibility gained through log collection, malicious event detection facilitated through anti-virus, anti-exploitation tools, application control, host IDS/IPS, and application control can help identify anomalies and potential threats the same way network IDS/IPS and next-gen firewalls focus on detection malicious files on the network.

As new protocols such as TLS 1.3 and DNS over HTTPS (DoH) become more prevalent, focus on the endpoint data piece of the puzzle is going to become more and more crucial as network data will become increasingly encrypted. Without the ability to decrypt traffic, Blue Teams must double down on endpoint monitoring where the data is still accessible.

## Newer "Must-Have" SOC Capabilities / Technologies

Capabilities to implement when possible:

- Threat hunting
  - Proactive hunting of in-progress compromise
- Endpoint detection and response (or XDR)
- Automation
  - Security Orchestration Automation and Response (SOAR)
  - In-house developed scripts
- Malware sandbox
- "Zero trust" security design where possible



### Newer "Must-Have" SOC Capabilities / Technologies

In combination with the foundational technologies that have been around for a long time, many established and mature SOCs build and utilize some of the newer capabilities and tools listed on the slide above. These items are listed in approximate order of necessity and effort required to generate value for security operations. While things like threat hunting and EDR are easy to see quick returns from, things like big data analytics are much more difficult and require higher levels of skill.

- Threat Hunting, although newer to security teams than some of the foundational items, should still be considered a "must-do" once your team is capable. Assuming your network is already compromised and proactively looking for evidence of that compromise is a cornerstone item of a modern defense mindset. We'll discuss threat hunting later on in the course.
- EDR is one capability that helps teams immensely in the "visibility" area as it acts as a sort of "flight data recorder" for your endpoints. EDR implements the collection of events combined with detection of malicious activity in one easy to install agent and gives SOCs the ability to collect events they might otherwise miss due to gaps in log collection capability. XDR (extended detection and response) is a new product category that covers not just endpoints but adds in network data as well.
- Security Automation, Orchestration, and Response (SOAR) and other automation tools, while not strictly about detection, can assist with shortening response times as well as making life in the SOC more pleasant in significant ways. Automation and Orchestration capability is another "must-have", whether it is implemented through a specific SOAR tool or otherwise, we'll discuss how this plays into keeping the SOC happy and functional later in the course.
- A malware sandbox can be an enormous benefit to any security team. It both removes the barrier to basic malware analysis so that even new analysts can extract indicators from potentially malicious files and also ups your capacity to perform more analysis at once.
- Zero trust design, while an enormous topic, is another capability you will certainly hear more and more about, and hopefully one that is already creeping into some of your (likely cloud) services. Note that this is not "rip and replace" of everything you have and then one day, boom, you have zero trust. This is the journey of, piece by piece, slowly integrating more zero trust-like principles into the way your organization designs its network services, and how employees interact with your assets and network.

[1] <https://cuckoosandbox.org/>

## Advanced SOC Detection Capabilities / Technologies

Once you have a solid implementation of the basics, consider:

- User and Entity Behavior Analysis (UEBA)
- Artificial intelligence & machine learning analytics
- Data lakes / "big data" analytics and processing



**Beware** – Micro Focus found "*for the majority of organizations assessed such investments continue to be a science experiment with an uncertain future...adopters have found that significant gains made in simplified collection of unformatted machine logs are quickly lost in the labor required to maintain these systems and post-processing the data collected*"<sup>1</sup>

### Advanced SOC Detection Capabilities / Technologies

Beyond the clearer "must-have" items on the previous page, we also have advanced technologies such as UEBA, AI and ML, and data lakes implementing "big data" analytics. Teams looking for an edge are starting to move in this direction. From talking to students across many organizations, the author finds the general consensus at this time is that getting value from solutions based around data science will potentially take significant effort to derive value from. While it can sound impressive to those outside the SOC to say you're implementing these types of tools, the temptation to jump into them should *not* be indulged unless you have maxed out your capabilities based on more foundational technologies first. In nearly every case, even advanced attackers can be caught with the endpoint and network visibility technologies already discussed.

Since these technologies are inherently often based around anomaly detection, the cleanliness of your network will inherently be a key driver in the usefulness of these types of solutions. A "flat" network where all users are allowed to talk to everything and download whatever tools they want will, obviously, be more difficult to find malicious interactions on compared to a strictly segmented and locked-down environment. The quote on the slide, from the Micro Focus State of Security Operations Report, reflects the apparent difficulty organizations have in finding ROI with advanced technologies.

[1] [https://www.microfocus.com/media/white-paper/state\\_of\\_security\\_operations\\_wp.pdf](https://www.microfocus.com/media/white-paper/state_of_security_operations_wp.pdf)

## Analyst Core Toolset and Reference Systems

The main tools your analysts will use daily:

- Security Information and Event Management (SIEM)
- Threat Intelligence Platforms (TIP)
- Incident Management Systems (IMS)
- Use Case Databases
- Unstructured Information Knowledgebase



### Analyst Core Tools and Reference Systems

In the SOC there are several systems that will receive near-constant use. The systems listed on this slide are the items that your analysts will be referring to, using, or searching for in nearly every alert they investigate or incident they cover. Note that these are logical functions separated for discussion here, in your environment they may be integrated into a single system – many SIEMs have incident ticketing (and IMS) built-in for example.

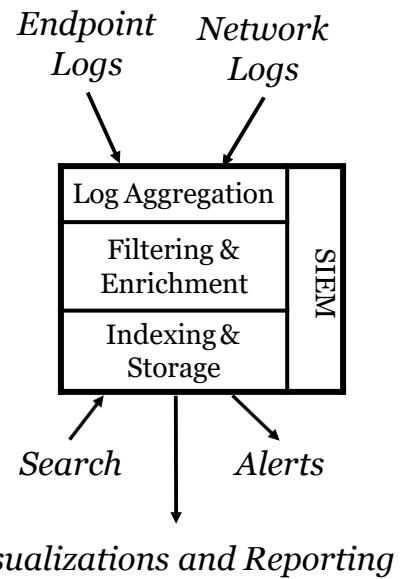
The analyst core toolset consists of:

- SIEM – The nexus of all the log data collected throughout the environment.
- Threat Intelligence Platform – Whether a dedicated solution or built in to one of your other products, a threat intelligence platform (ideally) gives analysts the context around any matched Indicators of Compromise (IOCs) found in the environment and the adversaries behind them.
- Incident Management System – The ticketing system analysts will use for working incidents, writing up reports on what happened, and ultimately closing out finished investigations.
- Use Case Databases / Playbooks / SOPs – A supporting system that informs analysts what to do when a given alert is triggered. This system documents the use cases implemented with the security tools and stores the related metadata.
- Unstructured Information Knowledgebase – Another core SOC tool and the place you store any additional reference data analysts may need to have close by. Teams often use OneNote, SharePoint, or similar systems that allow the team to easily and collaboratively add and edit data. This may not strictly be used every day but something fulfilling this function should be present in any environment.

Over the next few pages, we'll step through these items and describe in further the role these critical systems play in your day to day life in the SOC.

## SIEM

- One of the *most* important SOC tools!
  - Centralized log search
  - Data correlation and alerting
  - Visualization and reporting
- SIEM duties:
  - **Receive** all log data, **parse** it correctly
  - **Enrich** useful events with additional data
  - **Index** parsed for quick **search**
  - **Visualization** and **dashboard** creation



*Visualizations and Reporting*

## SIEM

If there was a single tool that it's most important for the security team to get right, the SIEM might be it. The SIEM is in the best position to see and correlate nearly all the data from throughout the environment. It can collect asset information, vulnerability scan info, and all the events and alerts that are centralized from endpoints and network traffic and use it for detection purposes. No other single tool in the environment has this scope of data to work with, meaning the SIEM is in a uniquely powerful position in your security stack.

The SIEM's main job is to faithfully receive all logs and parse them correctly into the fields of interest, potentially enriching and correlating the information in the process. Afterward, the parsed fields are (ideally) indexed into some sort of fast searching database for quick retrieval. It is this data your analysts can then quickly search through, alert on, or make visualizations and reports with. It should go without saying that the more high-quality, tactical data you can put into your SIEM, the better your chances are for using it for successful detection. The same goes for enrichment and visualization. Enrichment takes a single log's content and makes it better with data from external info or another data source. Visualization options help you take that information and use it to visually identify anomalies, an invaluable tactic for analysts, especially threat hunters.

## Threat Intelligence Platform

Stores tactical, operational, and strategic threat info

- IOCs to watch for in the environment
- Context for those IOCs (timeframe, use, campaign)
- The threat actors behind them and their goals, TTPs



### Threat Intelligence Platforms

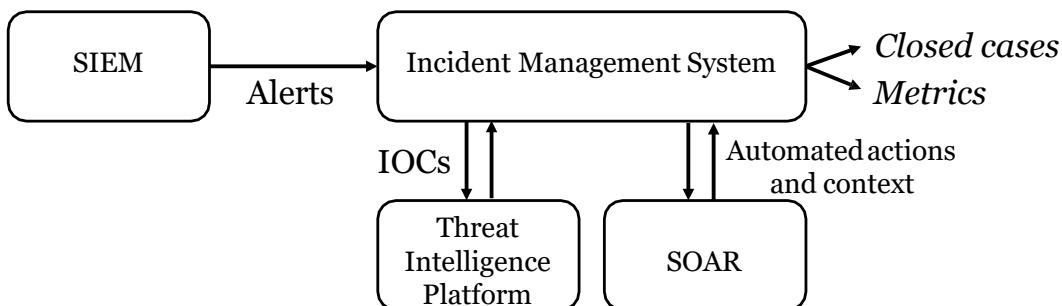
Another key SOC tool is the threat intelligence platform (TIP). In every SOC there must be a master list of all the known-bad domains, hashes, IP addresses, etc. that should be matched against all network and endpoint events. In many cases, the TIP is where this information is stored, and the TIP serves as the "source of truth" for IOCs to match against. To do so, all atomic indicators must be exported periodically to the SIEM, IDS, or any other tools being used to match the data so that an up-to-date watch list can be applied to the events that are observed. At the same time, they will likely be taking in info from external threat intel vendors, as well as any information produced within the threat intel team or the SOC fed back from actual incidents.

While it is tempting to stop there with the threat intelligence platform, consider what an analyst must do when these indicators are matched – if an alert goes off that says "Threat Intel Match – Malicious IP Contacted", what now? The analyst will need to find *why* that IP was marked bad, and if the TIP is merely a set of atomic "bad" IPs with no context, they will have no direction to take the investigation. The TIP, therefore, ideally should contain information related to each atomic indicator that can inform an analyst *how*, *when*, and *why* each atomic item was marked as an IOC. If they can access the platform and see "This IP was related to APT1234 and was used for malware X in campaign Y on July 4<sup>th</sup>, 2020", they now have a clear next move to validate the alert. For a great extended take on what high-quality CTI should include, see the referenced medium post by Andy Piazza below.<sup>1</sup>

[1] <http://mgt551.com/ioc>

## Incident Management System

- Your incident tracking and ticketing system
- Where your analysts will spend a *lot* of time
- Ideally integrated with threat intel and automation platforms



### Incident Management Systems

Incident management systems (IMS) may be bundled as part of one of your other systems (such as your SIEM), or you may have a dedicated external solution. Regardless of the literal implementation, the function that it plays as well as the inputs and outputs are shown on the page above. For the IMS, the main source of incoming data is alerts sent to it from the environment. Analysts log in to the IMS and triage these alerts and qualify them as either false positives or true incidents as they walk through and investigate each one. To do this as quickly as possible, integrations with your TIP and SOAR platform can help make additional information lookup and correlation quick and painless.

Once the incidents are investigated, remediated, and closed, the analysts close the associated ticket. In doing so, one key item that should not be ignored is the categorization of the incident for metrics purposes. This is the perfect time to note in a structured way the nature of the attack, the delivery vector, impact, systems involved, and more. These metrics can then be ideally automatically pulled at the end of each week and SOC managers can use the observed trends to feedback to where additional budget can be best spent to protect the company.

Since your analysts will spend nearly all day every day in your IMS, it is one of the most important pieces of software to choose carefully. Before selecting (or changing) IMS, the contenders should be *thoroughly* tested against each other with multiple real-world scenarios. Analysts should drive the testing (since they're the ones using it after all), and the opinions should be taken seriously on which one they would like to work with.

## Playbooks

Common incident analysis steps lead to **playbooks**:

- Investigation steps to do before that alert can be closed
- Opportunities for SOAR to help make you more efficient



**Playbooks must be used carefully**

- **Bad**

- No playbooks – Process is not stable or repeatable, different results from different analysts
- Strict playbooks – Limit flexibility to allow for varying context
- Too many playbooks – Management nightmare, never-ending hole

- **Good**

- Middle ground – enough to cover common situations, give flexibility

### Playbooks

When using incident management systems to investigate and respond to issues, analysts are likely to find themselves taking very similar actions across different cases. Items like "check for suspicious processes" and "check for contact with suspicious domains" are highly common to nearly every potential attack scenario. As these common actions are identified, the types of alerts they apply to can also be grouped to help identify these commonalities. For example, an infection of nearly any type - whether a virus, worm, backdoor, or otherwise, will almost always involve validation of what was running on the host, the context of why, and the identification of the source of the file. When these higher-level commonalities are found, these can be made into a generic "playbook" for investigating the situation. These playbooks can be codified into procedure and turned into investigative steps enforced through the IMS during the investigation phase. The goal is to ensure no obvious analysis steps are missed, even when an alert may be picked up by someone who is unfamiliar with the situation. They also have the benefit of giving you potential points for automation. If one of the activities is automatable through an API using a SOAR platform or script, all the better!

## Example Investigation Playbook

Standardizes actions the team takes

- What should be done before escalation / closure as false pos.
- Each item may be **mandatory** or **optional**
- Balance by defining **what** but not necessarily **how**
- Manual investigative steps handled by analysts
- Automatable actions handled by **SOAR**



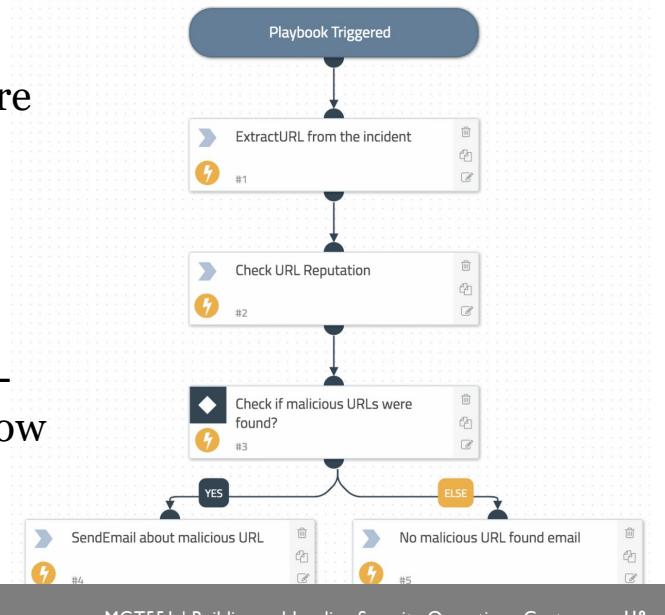
### Example Investigation Playbook

This page shows an example of the type of activities you may want to guide analysts to perform using a playbook system. When a common event occurs, such as a phishing email wave, the playbook lays out the steps to ensure the situation is thoroughly investigated before closure. These playbooks may have a mix of mandatory and optional steps, different directions based on the previous step's outcomes, and more. Manual steps should be taken by analysts while steps requiring simple data gathering and automated actions can be performed with a SOAR platform.

Be aware that playbooks, while great at creating a repeatable process, can be taken too far. Some common errors when diving into making playbooks are making too many of them or making them too strict. Trying to make a playbook for every scenario you might encounter will lead to a never-ending rabbit hole of impossible to define or predict situations. Making playbooks that are too strict will quickly fail as analysts encounter situations you didn't anticipate and are forced to work around the playbook. To avoid these issues, try to make high-level generic playbooks for common situations such as "suspected command and control", "malware infection", etc., and construct them with steps that focus on *what* to do without necessarily mandating the *how* so that analysts are not forced to follow an investigation path that doesn't make sense.

## SOAR-Based Playbooks

- If you have SOAR, playbook-centric alert work may occur there
- Alert triggers SOAR playbook
  - Enriches data automatically
  - Makes decisions on responses
  - Takes fast action if possible
- Eliminates repetitive, non-value-added steps from analyst workflow
- Leaves the fun, human-required analysis tasks



### SOAR-Based Playbooks

If you have a SOAR platform, nearly all of your alerts may be worked down a defined playbook-style workflow. An example of these types of playbooks is shown on the right side of the page above. In these systems, certain alerts are connected to the relevant playbook and automatically trigger as soon as the alert occurs, getting analysts a head start on some of the actions. Facilitating playbook use through a SOAR system can be the obvious choice for those who own this type of software. For those that don't, it's still possible to use steps during an alert investigation with most ticketing systems. TheHive incident management system, for example, offers what is called "Case templates" which are essentially the same exact thing – a list of tasks that must be completed before a case can be closed.

## Use Cases and Use Case Tracking

- The specific conditions or events to be detected
- Additional metadata and justification
  - Who owns it?
  - Stakeholders
  - Why is it interesting?
  - Logic
  - Playbook for response
  - Tool used for implementation
  - Age
  - Lifecycle
  - Known false positives
  - Data required
  - Priority
  - Action to take

### Use Cases and Use Case Tracking

Given the primary goal of detection, the SOC will no doubt want to have a list of the specific conditions it is trying to detect as well as why and how they plan to detect that condition. These lists of conditions of interest are often referred to as the "use cases" you will be implementing with your security tools.

While detection conditions and playbook steps may be a key *part* of a use case, that is not all that needs to be documented. A use case should be a fully filled-out documentation set that explains why you want to detect that condition, how it is detected, who owns the use case, the data required, and more. Tracking detailed use cases is one important way the SOC can fight the tendency to build up "tribal knowledge" and build continuity across time and team members. If nothing is documented about detection conditions, if key personnel leave, no one may know what an alarm means or what action to take when it occurs. Use case documentation and tracking alleviate this potential issue. Therefore, tracking your use cases and putting them into a use case tracking system or database of some sort (described in much more detail later in this course) will also be a required SOC activity.

The use case database or tracking system is the repository where you will keep the information on what action to take when any of your key alerts fire. Usability, customization, flexibility, and metrics reporting capability are key items you should be looking for in a solution. Above all else, the tool should be easy to enter new data into and allow you to track the lifecycle and any metadata required about all your live and retired use cases. We will dive deeper into use cases, and you will see a demonstration of a fully implemented use case tracking system in an exercise later in this course.

## SOC Knowledgebase

- A centralized knowledge base for team members
- Storage unstructured documents / text
  - New team member instructions
  - Tool and process reference
  - Points of contact
  - Network diagrams / Host config information
  - “What to do if x breaks” instructions
- Bonus points for co-location with use cases!



### SOC Knowledgebase

Each SOC should also have a repository used for unstructured documents, reference info, and general storage. There are many types of software available for this type of task, but a system like Microsoft OneNote or SharePoint is likely one of the easiest and most accessible.

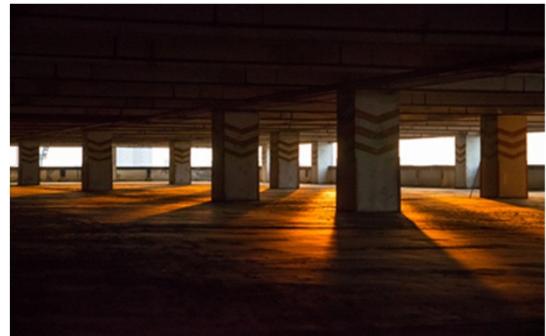
What we are looking for in the knowledgebase is a *quick and easy* (this is important so people actually use it) place that analysts can go when they need guidance on how to do things like get access to a system, identify points of contact, find policy and procedure, network diagrams, or instructions for less-common tasks. You get extra points if you can use the same software as used for your use case database to create this knowledge store as co-located data is more likely to be referenced.

Having something that is simultaneous readily available to everyone, easy to edit, and ideally with real-time collaboration capability for documents makes a system that analysts will actually *want* to use and keep up to date. The knowledge base is only as useful as the data inside it, and if no one wants to update that data because it's too painful, then it will quickly become ignored and useless. Usability is key here, pick whatever system will actually be maintained, first and foremost. A sample implementation for SOC information storage will also be shown in our use case database exercise later in this course.

## Emerging Technology

### The SOC as a Platform

- Governance for evolving capabilities and focus
- Minimizes “pet projects” and reactive solutions
- Encourages continuous growth and innovation



#### Emerging Technology

One common pitfall in security operations is accepting the SOC as “done”. Too often, teams put in the work of building out their core technology stack and integrations, processes, and playbooks, only to go into “maintenance mode” and neglect to keep those elements updated. Treating the SOC as a platform of baseline people, technology, and processes that is updated regularly through new or updated capabilities can help us avoid the complacency that sometimes accompanies ongoing operations.

A SOC-as-a-platform mentality has the added benefits of reducing pet projects and ad hoc solutions likely to become shelfware. Fostering individual projects and experimentation can result in great new things for our SOC, but if we want to institutionalize those new elements, we must train the rest of team, incorporate the new tool or process into our overall management approach, and update leadership on the new capability. Incorporating new tools and technologies into releases for our SOC platform can formalize these changes and help us capitalize on the improvements we’re making.

## Introducing New Technology

### Opportunities:

- Gaps in visibility or capability
- Existing tools unable to bring SLAs into desired range
- Additional *measurable* gains that will positively impact metrics or defenses



### Analysis of alternatives:

*From MITRE: “An analytical comparison of the operational effectiveness, suitability, risk, and life cycle cost (or total ownership cost, if applicable) of alternatives that satisfy validated capability needs.”*

### Introducing New Technology

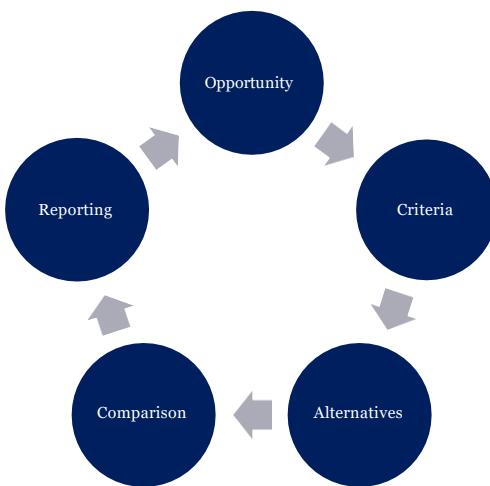
While you may or may not want to formalize SOC release planning, it's important to maintain some governance over changing and improving SOC technologies. Whether you are filling visibility gaps, automating SOC processes, or shrink your time to respond to an incident, you must quantify the return on the investment you're making – even if that investment is engineering time. How much time will the new technology save your team? What new threats will you be able to identify with the new tool? And just as importantly, how do you justify the selection of one solution over another?

Mitre has created a formal analysis of alternatives (AoA) process that is outlined here:

<https://www.mitre.org/publications/systems-engineering-guide/acquisition-systems-engineering/acquisition-program-planning/performing-analyses-of-alternatives>. As the article states, “AoAs provide a framework to consistently evaluate and compare the value of different solutions for providing a needed capability to specific end users.” Like any framework, it's important to make the process your own; you may not need to get as detailed or as rigid as the process described at the link above. But using the AoA process as a reference can help you answer those fundamental questions and keep your decision data-driven and objective.

## Analysis of Alternatives

1. Identify the opportunity
  - What problem are you solving?
2. Define analysis criteria
  - What features or measurable improvements will tell you you're solving that problem?
3. Identify alternatives
  - What tools or features will provide roughly the same solution?
4. Compare features & functionality
  - For the cost, which tools or features best solve our problem?
5. Report results



### Analysis of Alternatives

A 2006 report by the US Government Accountability Office found that the Department of Defense caused hundreds of millions of dollars in budget overruns by failing to evaluate possible alternatives as part of its acquisition programs. While most of us probably aren't spending that kind of money in the SOC, it's common to see SOC tools that have failed to deliver on vendor promises or have far exceeded their anticipated total cost of ownership. We can't always anticipate or avoid these challenges but conducting a proper evaluation when selecting new tools or functionality can help reduce the cost and operational risks inherent in any technology change.

The AoA process consists of five phases: (1) Identify the opportunity, (2) define your analysis criteria, (3) identify alternatives that may meet that criteria, (4) compare features and functionality, and (5) report your results. Though these guidelines were developed with the US Federal Government in mind and may be a bit more extensive than what is needed, the AoA framework is an incredibly useful reference for conducting (and documenting) an objective analysis of various tools and technologies.

AoA "handbook": <https://afacpo.com/AQDocs/AoAHandbook.pdf>

## AoA in Action

### AoA best practices:

- Make a plan
- Take your time
- Pick the right team
- Be objective
- Fully understand what you have already and why it doesn't work
- Evaluate many alternatives



### AoA in Action

There's no single right way to conduct an AoA and your approach will likely depend on the time and resources you have available to run it. But there are some best practices we should keep in mind when preparing to conduct a formal AoA based on Mitre's guidance:

- Use an AoA process as part of an over-arching project management methodology – use sound processes and techniques, avoid bias or subjectivity
- Create at least an informal "study plan" that defines how you will perform the analysis, who will do it, and why there is a need for it
- Budget sufficient resources, understanding that conducting a thorough AoA will require people and time
- Make sure that you understand the existing solution or capability, including its limitations – if additional analysis is required to fully understand what you currently have, build that time into the AoA plan
- Pick the right people to conduct the AoA based on their technical skills and experience
- Ensure that your analysis is objective and not just a "box checking" exercise
- Ensure the right number of alternatives – select a large number of alternatives at the outset, brainstorming with your team if necessary, in order to do so
- Anticipate problems – not all technology and capabilities will be easy to compare, and there are likely to be a lot of nuance in your analysis that isn't easy tabulated and analyzed. Take the time to define good baselines and prepare your data before finalizing your analysis.

## SOC Tools and Technology Summary

SOCs utilize many tools to get the job done:

- Foundational SOC Technology – AV, IPS, HIPS, etc.
- Analyst Core Toolsets for analysis and data management
  - SIEM
  - Threat Intelligence Platform
  - Incident Management System
  - Playbooks, Use case databases, SOC info knowledgebase
- "Next-gen" SOC tools – EDR, SOAR, Malware Sandboxes
- Advanced-Data Analytics – UEBA, AI/ML-based analytics
- Identifying and introducing new technology using an AoA
- **Goal:** Understand which are best for you, and how they integrate

### SOC Tools and Technology Summary

In this section, we briefly discussed some of the main tools that should be considered required when building a SOC as well as some that may be more aspirational. Throughout the rest of the course, we will dive into more detail on key features and deployment considerations, but for now, the goal was to understand what is the most important, and what can be held off on purchasing until later. While it may be tempting to jump to the fancy data science analytics, be aware that *most* intrusions can be caught with a masterful deployment of the basics. A well-monitored network combined with tactical data collection from endpoints can collect and detect nearly any attack out there if done correctly, yet many companies, unfortunately, struggle to perform the basics right and jump ahead to the advanced tools before they are ready.

# Course Roadmap

- *Book 1: SOC Design and Operational Planning*
- Book 2: SOC Telemetry and Analysis
- Book 3: Attack Detection, Threat Hunting, and Triage
- Book 4: Incident Response
- Book 5: Metrics, Automation, and Continuous Improvement

## SECTION I

### Introduction

#### SOC Design and Planning

- SOC Functions
- SOC Planning
  - *Exercise 1.1: Threat Actor Assessment*
  - Team Creation, Hiring, and Training
    - *Exercise 1.2: Attack Path Development*

#### Building a Strong Foundation

- Building the SOC
- SOC Tools and Technology
  - ***Exercise 1.3: Developing and Implementing SOC Playbooks***
  - Protecting SOC Data and Capabilities
  - Summary and Cyber42 Simulation – Day 1



This page intentionally left blank.

## EXERCISE 1.3

# Exercise 1.3: **Developing and Implementing SOC Playbooks**

### OBJECTIVES

- Develop a list of common scenarios appropriate for standardized playbooks
- Create a playbook for a commonly encountered scenario
- Define task categories for each step in the playbook
- Implement the playbook in TheHive incident management system
- Work an example case to see playbook workflow for analysts



### **Exercise 1.3: Developing and Implementing SOC Playbooks**

Please go to Exercise 1.3 in the MGT551 Workbook or virtual wiki.

# Course Roadmap

- *Book 1: SOC Design and Operational Planning*
- Book 2: SOC Telemetry and Analysis
- Book 3: Attack Detection, Threat Hunting, and Triage
- Book 4: Incident Response
- Book 5: Metrics, Automation, and Continuous Improvement

## SECTION I

### Introduction

#### SOC Design and Planning

- SOC Functions
- SOC Planning
  - *Exercise 1.1: Threat Actor Assessment*
  - Team Creation, Hiring, and Training
    - *Exercise 1.2: Attack Path Development*

#### Building a Strong Foundation

- Building the SOC
- SOC Tools and Technology
  - *Exercise 1.3: Developing and Implementing SOC Playbooks*
  - **Protecting SOC Data and Capabilities**
  - Summary and Cyber42 Simulation – Day 1



This page intentionally left blank.

## SOC IT Overview

Some IT considerations for the SOC:

- Keeping analyst accounts and computers safe
  - Analyst privileged accounts and analysis computers
- Keeping SOC data secure
  - Securing the data collection process
  - Securing the SOC data and tools
- SOC External / Unattributable connections
  - Keeping yourself and your investigations anonymous



### SOC IT Overview

Over the next few slides, we'll consider some important pieces of keeping the SOC safe. Specifically, we'll cover the multiple roles that analysts have to play in their daily jobs, the separate accounts and IT infrastructure that will be necessary to separate SOC data and accounts from the constituency at large, and the different types of network setups and connections to consider when creating a SOC.

## What Could Go Wrong?

Here's the nightmare scenario:



1. Attackers compromise machines on your network
2. Use those machines to gather highly-privileged credentials
3. Network access allows attackers to use gained credentials to...
  - Read the security team's email, monitor, if caught, change tactics
  - Delete or tamper with SOC data
  - Use security team accounts for additional persistence and data access
  - Phish other employees as the security team

How do we reduce this risk? The answers follow...

### What Could Go Wrong?

Before we dive into the problems, let's set the stage on the issue we're trying to avoid. Imagine as an attacker, you gain access to an organization and are able to establish some level of highly-privileged credentials. This could be domain admin, or something lower but still powerful such as a desktop, server, or email admin. If the SOC data and analyst machines can be accessed from the constituent network at large, it is highly likely as an attacker you would be interested in doing so. With these credentials you'd be able to monitor the security teams' email to look for your detection, interfere with data collection, erase or wipe critical data, or even pose as the security team to gain further persistence within the environment!

As a SOC manager and security team, this is a nightmare scenario, and don't brush it off as unlikely, attackers have repeatedly engaged in these types of activities! Having an attacker leverage your own infrastructure and security team to make things worse is not only adding insult to injury, but makes your team look incompetent and destroys any trust the organization had in you to do the job they had assigned to you. This scenario, therefore, must be avoided as much as possible. While smaller teams may need to dial back some of the suggestions here due to the extra complication they add, take the spirit of the discussion and do your best to separate the SOC data and accounts to prevent a similar disaster from playing out.

## Keeping Analysts Safe

Consider the roles a SOC analyst must play

1. **Normal employee** that answers email / browses web
2. **Privileged user** that can access servers, laptops, sensitive data
3. **Investigator** of malicious emails, files, and sites

To safely perform all these **separation of accounts and assets is a must**, this means:

- Privileged accounts
- Privileged access workstations
- Malicious content analysis workstations



### Keeping Analysts Safe

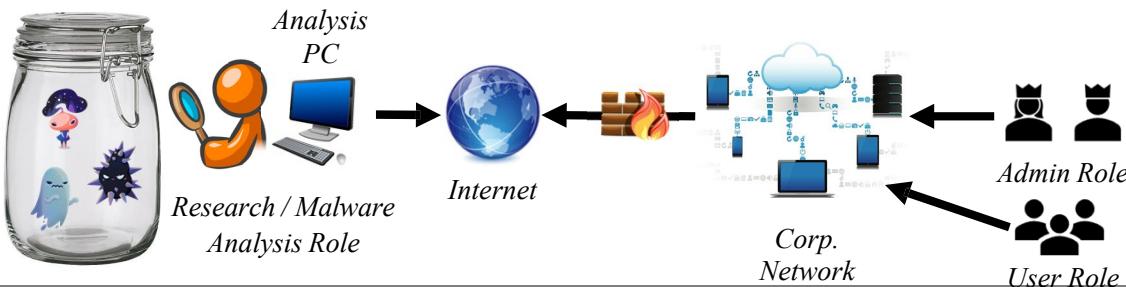
As a first step, consider the "hats" an analyst must wear in their day-to-day role. Analysts will not only have to do typical (and potentially dangerous) work tasks such as reading email, downloading files, and browsing the internet, but also may use administrative level credentials to access constituent systems for investigation or incident response, as well as diagnose potentially malicious content. They will also likely have access to sensitive data, plus the power to make changes to the environment that could cause outages or other damage if wielded incorrectly. Given this breadth of activity and higher than normal risk, we must consider how they can operate within the three different roles without endangering themselves. The last thing you would want is for an attacker to take over an analyst account and wreak havoc throughout your environment using their privileges.

To mitigate this risk, the best solution is to take the three roles an analyst plays – normal employee, privileged access user, and malicious content investigator, and break them up as much as possible. This means different accounts and/or computers where separation of these duties can be enforced, and where a mistake in one role won't lead to access as another. For example, we wouldn't want an analyst falling for a phishing email causing exposure of credentials that would let an attacker log into the SIEM or EDR tool. Similarly, we wouldn't want a malware reversing experiment to endanger the company network if a virus was able to get out. To achieve this kind of protection analysts should be properly set up with separate privileged accounts, privileged access workstations, and some sort of isolated analysis environment. Over the next few slides, we'll cover these concepts as they are incredibly important to maintaining the security of your SOC and trust in your operation.

## Requirements for SOC Connectivity

The SOC needs several forms of access:

- Access to the corporate network as a "user"
- Access to security tools as an analyst
- Privileged access to devices on the network for incident response
- Corporate and unfiltered internet access



## Requirements for SOC Connectivity

To maintain the needed separation between analyst roles, multiple types of access to different data will be required as well. In an ideal scenario, strict separation of tools and data access would be maintained. Here are the 3 scenarios to consider:

- Your analysts will need normal network connectivity into the organization's constituency network for email and everything else the typical employee uses – analysts should NOT use the username they use for this role, or the machine (ideally) to access SOC data unless necessary. Doing so can be a privilege escalation opportunity for attackers.
- Your analysts will have *privileged* user access. Access for live incident response on machines within the network, facilitated in a safe way that will not compromise the credentials used to access the potentially infected machines. Analysts will also need to be able to access security tools and data using this role.
- Your analysts will need unfettered internet access so that they can perform OSINT, malware research and more, without being subjected to the constraints of the corporate security tools (this should be done from a separate computer), on a separate, non-org-attached internet connection to provide strict isolation where possible.

These three scenarios and their very different risk profiles, uses, and requirements for security are the driving idea behind the following proposed solutions. If your setup or needs are different for this, follow the guiding principles of role-based separation of accounts and data access.

## Analyst Privileged Accounts and Workstations

Many analysts have dangerous levels of access:

- Endpoint / server data
- Networking data and equipment
- Ability to implement blocking actions
- How do we stop attackers from abusing this?
  - **Privileged access workstations (PAWS)**
  - **Separate privileged accounts!**
  - Microsoft guidance: <https://aka.ms/privsec>



### Analyst Privileged Accounts and Workstations

The first consideration is breaking up the access that analysts have into two accounts in Active Directory:

- A normal account where everyday business can be conducted – email, meetings, web browsing, etc.
- A privileged access account that can be used to access secure systems, sensitive data, etc.

We know advanced attackers can and do go after security teams during incidents. The separation of these duties into two different accounts reduces the risk that a simple phish or exploit kit that hits a user operating as their normal account will also give the attacker access to the privileged account. Note the normal user should NOT have local administrative privileges on the laptop where it is used. Why? Because if they do, this allows easy credential dumping with tools like Mimikatz that would potentially expose the password for both accounts and ruin the whole point of this account separation exercise.

Even better than having two accounts is strictly separating the use of those two accounts to two different *machines*. This could be two physical laptops or one laptop with virtual machines on it for "everyday business use". The reason for this is that even if an analyst's daily machine becomes completely owned by an attacker, it still will have no chance at exposing the high privileged credentials that would let attackers into the security team tools.

Privileged access is a deep and complicated topic, which is why I *highly* recommend you read the link in this slide for Microsoft's guidance on privileged access administration. It is applicable to the SOC and all IT administrators and guides you towards best practice to disrupt attacker attempts at lateral movement. It includes how to set up PAWS – privileged access workstations using physical machines or virtual machines, and helps you avoid the non-obvious pitfalls that many organizations fall prey to.

## SOC Analysis Workstations

- In addition to PAWS, analysts must handle malicious items
  - Potentially malicious programs, scripts, links, email, ...
  - Do *not* want to risk them accidentally infecting themselves!
- Solution:
  - Linux-based (cloud-deployed?) **analysis workstations**
  - Virtual machines with clean snapshots for dangerous analysis
  - NO internal network access + failsafe to prevent accidents



SANS

MGT551 | Building and Leading Security Operations Centers

135

### SOC Analysis Workstations

What about the "examine potentially malicious content" analyst role? For analysts that will be regularly engaging in tasks that could jeopardize their machine, they should have a 3rd machine that will allow them to safely inspect files, links, and other sites without fear of letting an attacker into the network. The most common solution I've seen to this is a dedicated *off-network* or even cloud-based analyst workstation.

This workstation should be running Linux (or something non-Windows) since most viruses you may encounter simply won't work in that environment. Within the host running Linux however, the analysis should be done within virtual machines where a series of snapshots has been strategically taken for just this purpose. Here's how it works – an analyst starts by taking on investigation of a supposedly malicious file download alert and they want to inspect it to see if the alert was correct. To do so in a safe way, they can bring a sample of the file into a fresh, ready for analysis virtual machine on their analysis workstation. (REMnux<sup>1</sup> a Linux distribution made for malware reverse engineering created by Lenny Zeltser, author of SANS "FOR610 – Malware Reverse Engineering" is a great choice for the analysis virtual machine.) Once the analysis is complete, analysts can revert the REMnux snapshot back to before the virus was uploaded, ensuring nothing is left over to pose a risk down the line.

[1] <https://remnux.org/>

## Protecting SOC Data

- Protecting SOC data means:
  1. Protecting the data collection process
  2. Protecting data access and storage
- **Data collection process**
  - Out-of-band network data collection where possible
  - Authenticated, encrypted, and monitored log collection
- **Data access and storage**
  - If an attacker can't access the SOC, they can't tamper with it
  - How do we prevent attacker access? A separate SOC enclave!



### Protecting SOC Data

Protecting SOC *data* means protecting both the transport and collection of that data as well as protecting the appliances and locations where it is stored. Over the next several pages, we'll cover considerations for the protection of data collection as well as for accessing and storing that data. The goal is to prevent adversaries from being able to abuse our security monitoring infrastructure to help them achieve their goals and also prevent them from getting access to any of the data or resources used or collected by the security team.

## Protecting SOC Data Collection

Data collection for logs has two primary concerns:

### 1. Authentication and encryption for log transport

- **Mutual TLS** authentication for log sources / collectors where possible
- **Encryption** so logs cannot be seen in transit
- Prevents a rogue malicious device from flooding SIEM with bogus logs
- Prevents sending logs to unintended places

### 2. Dependable network data collection

- **Out-of-band** collection
- Collection that cannot be disrupted
- **Secure transport** to the SOC



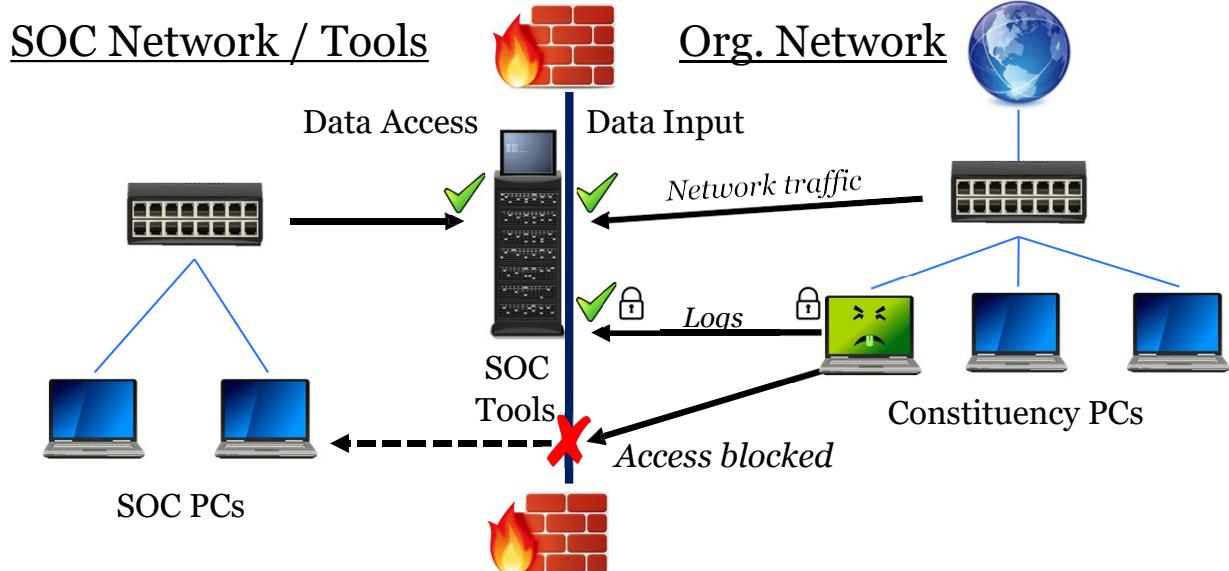
### Protecting SOC Data Collection

For protecting log collection, the first goal is to ensure all the data we receive is both encrypted in transit as well as validated to come from a source we intend to be collecting logs from and is sent to a validated log collection point. This means using some sort of authentication and encryption mechanism. Most SIEM agents and other log collection methods support TLS for mutual authentication and encryption. Setting up mutual TLS will ensure logs are coming *from* an expected source and being sent *to* a trusted source. Why mutual TLS? You wouldn't want an employee traveling to a public network and attempting to send logs to a device that happens to share the same IP as your log collection server, for example. You also wouldn't want a rogue device joining the network and attempting to flood the SIEM with bogus logs. Mutual TLS minimizes this risk.

In addition, for network data that is captured via taps or SPAN ports, you want to make sure that data travels to your collection devices as securely as possible and you minimize the chance adversaries have to interfere with the collection. As Rob Joyce, NSA Tailored Access Operations Chief (an offensively focused team of the US Government), said in his 2016 Usenix talk on Disrupting Nation State Hackers<sup>[1]</sup> like his own team - the thing that keeps him up at night is an out-of-band network tap collecting data, that someone is actually watching. Why? Because there's nothing he can do about it. Attackers must move across the network, and if data collection is inescapable, they're necessarily potentially exposed. You *will* have evidence of any attack that uses the network (nearly all of them), all that's left is to detect it! Ideally, the copy of this traffic that is being made is either sent directly to a sensor or is sent over a dedicated channel, making it more difficult to intercept in transit.

[1] <https://www.youtube.com/watch?v=bDJb8WOJYdA>

## SOC Data Collection Diagram



### SOC Data Collection Diagram

This page shows a conceptual diagram of how a SOC data collection operation should operate. On the right side, we have a typical network environment; on the left side, we have the dedicated SOC systems and tools such as the SIEM, PCAP capture tools, Network IDS consoles, etc.

In the center, this picture highlights that the SOC tools themselves should be exposed to the constituent environment only as much as needed to receive the traffic they are trying to collect. This means the data input interfaces should ideally be separate from the management and login/access interfaces analysts will use to view data collected by the appliance. On the constituency network side each device has the ability to authenticate and send logs and network taps and SPAN ports have the ability to pick and send data. Beyond that, the monitored environment should *not* be able to reach out to or interact with the systems inside the SOC (depicted in the lowermost arrow). The SOC analyst machines and tools should only be accessible in the minimum required ways. If the constituency PC on the left became infected and the attackers gained highly privileged credentials, we wouldn't want them able to immediately use them to log in or even connect to SOC tools or analyst PCs.

## Protecting SOC Data Access and Storage

The SOC should also have (where SOC size permits):

- Dedicated **SOC "enclave"** (private network) for
  - Accessing security appliances
  - Analysis machines
  - SOC specific services / services
  - For lab environment and testing
- Dedicated **internet connection**
  - Separate DSL, Cable, or Cellular modem
  - Used for sensitive investigation, cannot be tied back to your org.

### Protecting SOC Data Access and Storage

The second piece of protecting SOC data is securing the locations where it is stored and the rest of the machines in the SOC itself. The solution often implemented to solve this problem is creating a private SOC network as shown briefly on the previous page. It is from this private network that collected data can be accessed on the security appliances, analysis machines can be used, and SOC specific services can be safely run, isolated from the rest of the main network. In the following slides, we'll further explain how to implement such an environment.

As we discuss the SOC private network, one final useful item to have available is a dedicated external internet with a connection that has an IP unassociated with your organization, or barring that, a VPN connection to a VPS where your traffic can emerge. Sometimes analysts may want to download files from exploit sites or other sensitive places that you do not want to be traced back to your organization. Having an additional connection available to the SOC can help ensure you can do this in an operationally secure way and won't have to send potentially malicious traffic through your normal organization connection. We'll also discuss how this can be implemented and the types of things you might want to make the connection available for.

## SOC Private Network

- **Goals:**
  - Prevent the attacker tampering with SOC members and SOC data/assets
  - Securely offer additional services required for SOC use only
- **Details for private network implementation**
  - Incoming connections from constituent network should be blocked via firewall
  - Access to security tools allowed via SOC machines and IP addresses only
  - Systems inside should not trust or use constituent domain credentials
- **The SOC network should support secure use of any special services you don't want available to the network at large**

### SOC Private Network

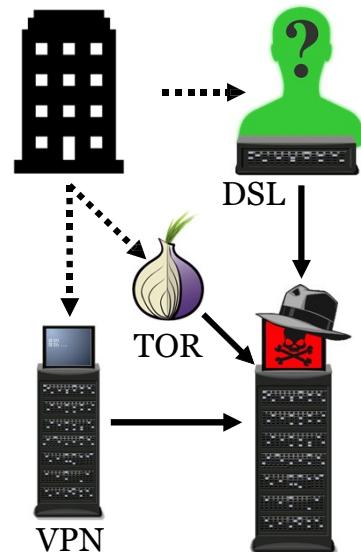
The SOC private network can act as a safe haven for security personnel computer resources. This is a cordoned off subnet where no traffic will be allowed except where strictly necessary. It should effectively otherwise appear invisible to the rest of the network. The goal is that attackers that succeed in gaining a foothold on the constituent network should have no capability to connect to anything the SOC controls or ideally even see it exists.

Given your own subnet with strict firewalling (and potentially a second route to the internet as discussed next), this also gives the SOC great flexibility to run any specialty services it needs without risking them affecting the rest of the network – a malware sandbox, for example.

To implement this secure enclave, the basics are that it should be firewalled off from the constituent network for traffic in both directions except where necessary, and SOC tools on the SOC network should only be reachable by SOC analyst PCs. This is what ensures an attacker that achieves domain admin privilege will not just log into SOC tools and modify or clear data. Since we assume at some point the constituent network will be compromised, even highly privileged credentials ideally should not work for tools and systems inside the SOC network (this can be achieved with Active Directory using one-way trusts and other mechanisms). Implemented correctly, this setup will create a very strong security border isolating the security team from the rest of the environment. After setup, it is worth threat modeling how an attacker might affect the SOC and ensure any additional detections are in place for items that cannot be covered.

## Dedicated Unattributed Internet Connections

- You will need to conduct investigations where
  - You don't want **security appliances interfering**
  - You want to stay **anonymous**
- **Solution:** A dedicated SOC internet connection
  - LTE, Cable modem, DSL for SOC use only
  - No corporate security controls filtering traffic
  - IP is not traceable to your org
- Can't set that up? Alternative options:
  - Use a router with all traffic forwarded to external VPN
  - TOR allowed from *specific machines only*

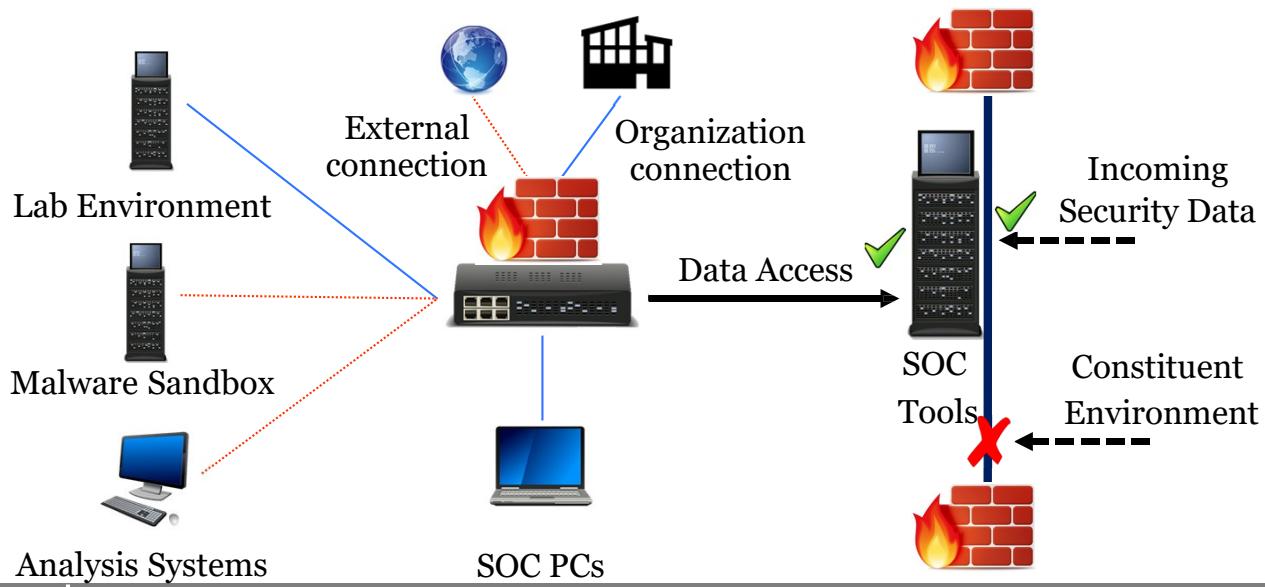


### Dedicated Unattributed Internet Connections

There will be numerous occasions where analysts in your SOC may want to investigate a potentially malicious site, IP address, or domain name in an anonymous way. In addition to wishing to stay anonymous, it's likely you will not want to potentially trigger all your security protections by using your normal company network to do this research. What is the solution? One of the ways you can help avoid giving up your identity as well as avoid subjecting your malicious content research to your normal network security appliances is purchasing some type of external, unattributed internet connection for the sole use of the SOC. This could take the form of an LTE hotspot, a dedicated DSL cable modem, or, if nothing else, a connection that forces an external VPN route for connectivity (TOR could be utilized as well).

Any of these methods will allow your analyst to investigate on the internet unimpeded and (assuming they follow other good operational security practice) without giving up their anonymity. Since this is an unprotected route to the internet, of course, you'll want to lock it down so only certain dedicated systems and virtual machines can use it—analyst workstations, the SOC enclave in general, and perhaps malware detonation appliances are the obvious choice for connections to this line, NOT typical corporate PCs.

## SOC Private Network Diagram



### SOC Private Network Diagram

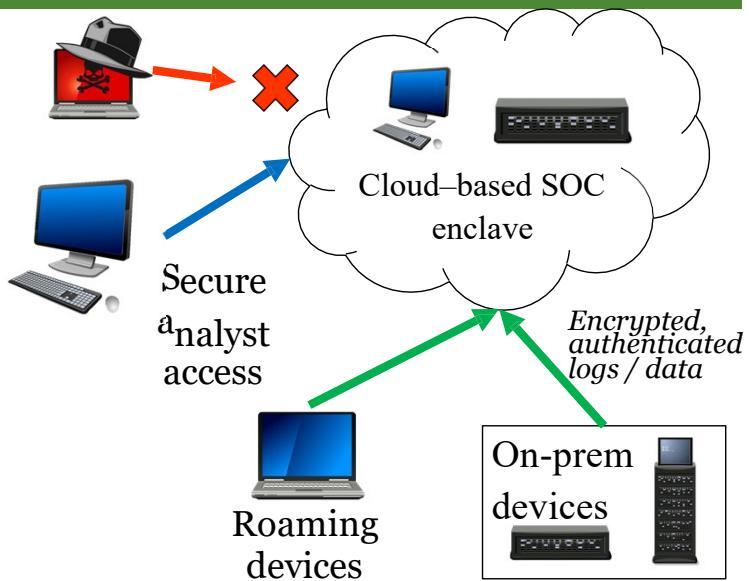
This page has a conceptual network diagram for a private SOC network with lots of optional different services available. Along with the normal PCs the analysts would use for their daily job the SOC private network also has analysis systems, malware sandboxes, a lab environment, and could support anything else required by the team. The two internet connections depicted can be carefully setup per source to route traffic through either the constituent organization or the external unattributed line (depicted in dashed lines). Systems such as the analysis PCs and malware sandbox analysis would likely use the direct line while other systems used to access the constituent network would likely be steered the normal way.

The key to making this all work in a safe way is the firewall in the middle that constrains the connections to the minimum required, even within the SOC. As always in security "least privilege" is the name of the game. You wouldn't want an infection that somehow exploited one of the servers or malware sandbox to be able to reach back out and attack the rest of the devices within the enclave or constituent network. The firewall is the device that will prevent that. Even within the SOC, good network design including segmentation must be used since you will undoubtedly be handling risky files. Think of the private SOC network as another miniature network nested within the organization. While it may be implemented with a minimum of equipment, it still must logically act as several, properly isolated networks.

## Cloud-Based SOC Data Flow and Access

New collection and storage point adds challenges

- Secure data transfer from *anywhere* to the cloud
- Secure access to cloud resources for analysts
- Rejection of data access from internet in general



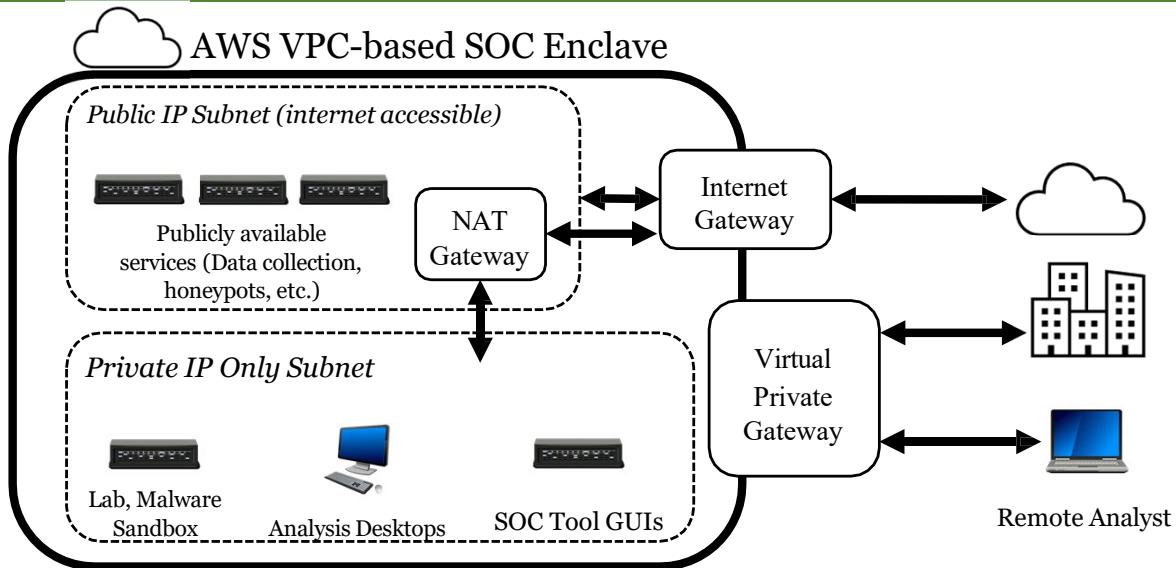
### Cloud-Based SOC Data Flow and Access

Moving your SOC to the cloud vs. on-premise adds some additional considerations for security of data collection and storage. Putting information in the cloud means you now must:

- Consider how all company owned devices and servers, regardless of location, can *securely* send in logs and data to the cloud without chance of interception or tampering
- Consider how analysts will access highly sensitive resources that are now "on the internet"
- Consider how you will keep *all* other attempts to access that data out. In other words, how will you secure your SOC tools from attackers, and even from other people in your organization

The goal is to hit the same type of isolation you would have when creating an on-premise SOC-enclave network and data storage area, but without the benefits of having control of the hardware or network yourself. In practice, this means authentication and encryption of data in transit, and strong authentication for analysts accessing the information. The more requirements of a device accessing information in the cloud, the better the isolation. If an analyst machine must have VPN access utilizing a certificate, be in the correct (separate from general AD) groups, have a privileged account username/password, and 2FA token to access cloud-based data for example, attackers would be hard pressed to meet those requirements to tamper with SOC information and accounts. Good cloud architecture design is key to success here, as it is with any cloud service deployment.

## AWS-Based SOC Enclave Architecture



### AWS-Based SOC Enclave Architecture

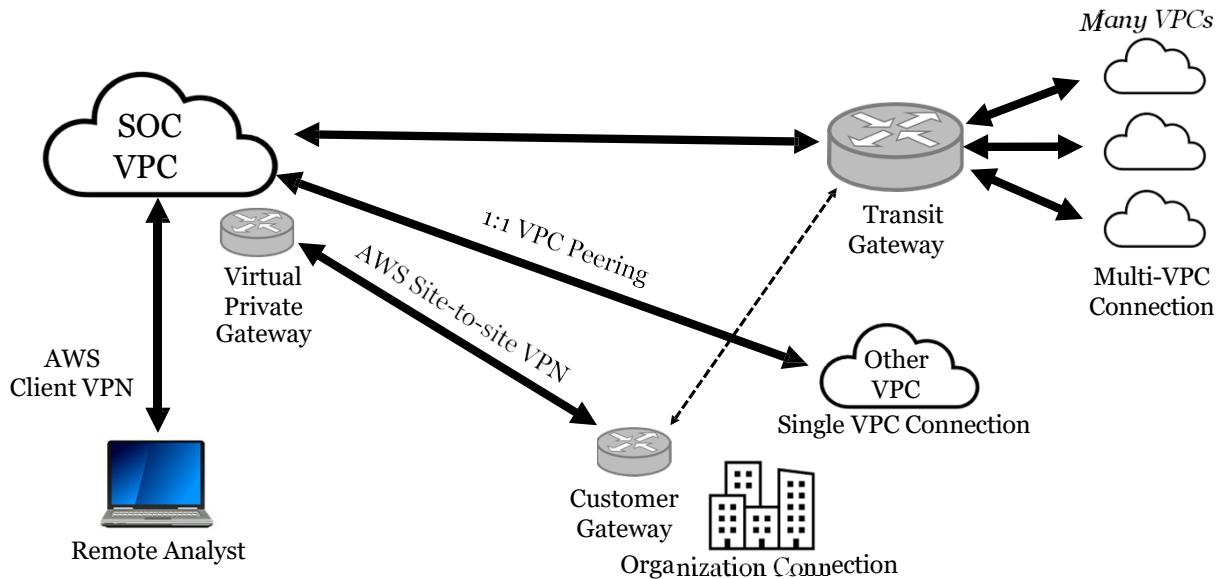
For your reference, here is a sample high-level architecture for an AWS-based SOC enclave.

First, you would create a dedicated VPC (virtual private cloud) inside AWS that will contain all the resources for your SOC enclave. Within that VPC you will want certain items to be internet accessible (such as for data collection from org-owned resources that aren't always on the corporate network, if used), as well as many items that don't need to be accessible directly from the internet. Inside your VPC, you would make at least 2 subnets, one that will contain items assigned a public IP (publicly accessible services) and one that is a private IP only subnet. Resources in both sections will be *able* to access the internet, but items in the private IP subnet would use a dedicated NAT translation gateway to initiate outbound connections only while blocking inbound traffic. Security groups and NACLs should then be defined for each item within the VPC allowing traffic *only as strictly necessary* (details on this are beyond the scope of this course but follow best practice by utilizing both). Finally, route to an assigned internet gateway needs to be added for the VPC so internet-bound traffic has a path to the internet. For direct connections, virtual private gateways can be defined for traffic directly to and from corporate resources, or for analysts to connect to the VPC directly (covered in more detail on the next page).

With this setup, you effectively have created an isolated network in the cloud with both public and private services that can be accessed directly via VPN to working from home analysts, via your company network from site-to-site VPN, and from the internet as required. For monitoring of this SOC enclave, VPC Flow logs can be generated and collected at the VPC level, subnet level, or instance level, and VPC traffic mirror can be utilized for any service you'd like full traffic capture from.

For additional information, see the video "AWS Networking Fundamentals" here:  
<https://www.youtube.com/watch?v=hiKPPy584Mg>

## Connecting to the SOC Enclave



### Connecting to an AWS-Based SOC Enclave

For connection to the SOC enclave in an AWS VPC you have multiple options:

1. For connections directly from within your on-premise network, a site-to-site VPN connection can be built by adding a "virtual private gateway" to your VPC and defining a "customer gateway" on your organizations network. This connection will create an IPSEC based VPN tunnel for direct, secure communication with items inside the VPC from inside the corporate network.
2. If your org has a limited number of existing VPCs in AWS you would like to allow direct traffic from, direct VPC peering can be used to allow traffic to flow directly between the SOC VPC and others
3. If your SOC enclave needs a connection to *many* VPCs because your company is highly invested in cloud infrastructure in AWS, connecting the SOC VPC to a transit gateway is the best solution as it connects the SOC VPC to all other VPCs also connected to the transit gateway. (You may also use this model with a site-to-site VPN connection directly to a transit gateway as shown in the dashed line above)
4. For analysts that want to connect directly to the VPC to work, AWS Client VPN connections can be made to the VPC, regardless of the analyst location, and provide secure access to resources within it.

## Is All of This Really Necessary?

Make no mistake – attackers WILL attempt to monitor and interfere with your security team<sup>1</sup>!

- Yes, this is sort of a pain
- No, it doesn't have to be expensive
  - Utilize virtual machines running in a locked down host
- You do NOT want to be the cause of a virus outbreak in your organization, the extra security is worth it!



Matthew Dunwoody  
@matthewdunwoody

Next we discussed email. #APT29 consistently stole email throughout the intrusion. In addition to stealing mail from VIPs, they targeted the IR team to monitor the investigation. This made for some interesting opportunities for counter-intel (e.g. OOO msg during remediation) 41/n

### Is All of This Really Necessary?

Is this level of complication necessary for analysts? In many cases, yes, this level of paranoia is indeed justified. Is it a 100% requirement? Strictly, no, but understand the risks for any part of this guidance you cannot meet and put in compensating controls as a backup. There are many documented attacks where the attacker accessed files, passwords, accounts, tools, and email of the security team and leveraged it to give them a heads up against them.<sup>1,2</sup> Since it's of utmost importance for the security team to maintain trust within the organization, the last thing you would want to have to explain is how the security team themselves either got compromised, or even worse, was the cause of an outbreak that brought down critical services within the company.

Fortunately, most of the multiple-machine suggestions can be implemented within a single higher-spec laptop running multiple virtual machines and the segmentation for the SOC private network can likely be handled by one or two cleverly managed firewalls. For the accounts, be sure to follow Microsoft's guidance in keeping the host OS the most secure and running less secure virtual machines inside it (see the previous Microsoft links on PAWS for details). For analysis systems, even lower-power systems are plenty capable of running simple malware analysis tasks and should not present a large cost burden, decommissioned employee laptops could even be used for the job.

1 <https://www.fireeye.com/blog/products-and-services/2019/02/state-of-the-hack-no-easy-breach-revisited.html>

2 <https://twitter.com/matthewdunwoody/status/1091785981366333441>

## Building the SOC Summary

- SOC physical space planning is key for efficiency
  - Consider **room** and **desk** details and layout before building
- Securing SOC analysts and data is critical
  - Analyst **accounts** should be separated into different roles
  - Analyst **machines** should follow the same split
  - **Private networks** keep attackers out, enable SOC services
  - **External connections** make OPSEC and investigation easy

### Building the SOC Summary

When building the SOC, especially a large and complex one, there are a multitude of considerations for both the physical and desk space, as well as securing the IT infrastructure that will support it. In this module, we have covered some of the major ideas and best practices seen implemented throughout the industry. Of course, there's plenty more detail than there, unfortunately, isn't time to cover here. For additional info, check out the excellent *free* MITRE book "Top 10 Strategies of a World Class CSOC" for details<sup>[1]</sup>. Both Chapter 10 and Appendix F contain great additional information on these topics.

[1] <https://www.mitre.org/sites/default/files/publications/pr-13-1028-mitre-10-strategies-cyber-ops-center.pdf>

# Course Roadmap

- *Book 1: SOC Design and Operational Planning*
- Book 2: SOC Telemetry and Analysis
- Book 3: Attack Detection, Threat Hunting, and Triage
- Book 4: Incident Response
- Book 5: Metrics, Automation, and Continuous Improvement

## SECTION I

### Introduction

#### SOC Design and Planning

- SOC Functions
- SOC Planning
  - *Exercise 1.1: Threat Actor Assessment*
  - Team Creation, Hiring, and Training
    - *Exercise 1.2: Attack Path Development*

#### Building a Strong Foundation

- Building the SOC
- SOC Tools and Technology
  - *Exercise 1.3: Developing and Implementing SOC Playbooks*
- SOC Enclave and Networking
- **Summary and Cyber42 Simulation – Day 1**



This page intentionally left blank.

## Day 1 Summary

- In this day, we covered:
  - Defining the SOC functions
  - Planning and building the SOC
  - Finding, hiring, and training our team
  - Physical space considerations
  - Core SOC tools
  - Protecting sensitive SOC systems and data
  - Assessing threat actors likely to target us, identifying the ways in which they are likely to do that, and plans to investigate and respond when it happens

### Day 1 Summary

In this book, we covered the core concepts and models for building a SOC. From learning how all the systems work together to gather data and detect attacks, to planning the requirements and physical build of the SOC, to hiring and training, there is a lot to plan to create a successful SOC and team. We also spent some time using various tools and frameworks to tailor our defenses and our processes to the adversaries and attacks likely to impact our organization. If you already have a SOC, hopefully, you picked some additional ideas on how to continue to grow, and if you're forming a team, you should now be well on your way to making a roadmap for success.

Tomorrow we'll continue into the operation section of the course and discuss the rest of the SOC core functions covering how detection, triage, investigation, and incident response work, and finishing the day with how to continuously test and improve your team to ensure the SOC stays performing at the best possible level.



# Cyber42 Simulation

## Day 1

### **Cyber 42**

Your instructor will now give you instructions on how to access the Cyber42 game. OnDemand students should refer to their supplemental documentation for instructions for access.

# 551.2

# SOC Telemetry and Analysis



The SANS logo consists of the word "SANS" in a bold, serif font, with each letter "S", "A", "N", and "S" stacked vertically.

THE MOST TRUSTED SOURCE FOR INFORMATION SECURITY TRAINING, CERTIFICATION, AND RESEARCH | [sans.org](https://sans.org)



# 551.2: SOC Telemetry and Analysis

© 2021 John Hubbard and Mark Orlando | All Rights Reserved | G02\_02

Welcome to book two of SANS MGT551: Building and Leading Security Operations Centers!

<b>TABLE OF CONTENTS</b>	<b>PAGE</b>
Introduction	of Class <sup>1</sup>
Cyber Defense Theory and Mental Models	4
SOC Data Collection	24
Other Monitoring Use Cases	43
Exercise 2.1 – Attack Path and Data Source Assessment	76
Using MITRE ATT&CK to Plan Collection	78
Exercise 2.2 – Prioritizing and Visualizing Attack Techniques and Security Controls	87
Cyber Threat Intelligence	89
Exercise 2.3 – Writing Priority Intelligence Requirements	111
Practical Collection Concerns	113
Prevention and the Future of Security	126
Summary and Cyber42 Day 2	146



This page intentionally left blank.

## Day 2 Overview

### Mindset and Preparation

- Cyber Defense Theory and Mental Models
- SOC Data Collection
- Other Monitoring Use Cases

### Collection and Monitoring

- Using MITRE ATT&CK to Plan Collection
- Cyber Threat Intelligence
- Practical Collection Concerns
- Prevention and the Future of Security

**Exercises:** Attack Path and Data Source Assessment, Prioritize and Visualizing Attack Techniques and Security Controls, Writing Priority Intelligence Requirements



### Day 2 Overview

Here is a list of topics we will be discussing throughout the second book of this course.

# Course Roadmap

- Book 1: SOC Design and Operational Planning
- *Book 2: SOC Telemetry and Analysis*
- Book 3: Attack Detection, Threat Hunting, and Triage
- Book 4: Incident Response
- Book 5: Metrics, Automation, and Continuous Improvement

## SECTION I

Introduction

Mindset and Preparation

- **Cyber Defense Theory and Mental Models**

- SOC Data Collection

- Other Monitoring Use Cases

- *Exercise 2.1: Attack Path and Data Source Assessment*

Collection and Monitoring

- Using MITRE ATT&CK to Plan Collection

- *Exercise 2.2: Prioritizing and Visualizing Attack Trees*

- Cyber Threat Intelligence

- *Exercise 2.3: Writing Priority Intelligence Requirements*

- Practical Collection Concerns

- Prevention and the Future of Security

- Summary and Cyber42 Day 2



MGT551 | Building and Leading Security Operations Centers

4

This page intentionally left blank.

## SOC Operations

We will now shift the class focus to SOC *operations*

- Theory and Mindset
- Understanding the SOC Systems
- Collection – What do you need for success?
- Detection – Organizing, implementing, and tuning
- Triage and Investigation – Speed and accuracy
- Incident Response – Options, tools and automation

### SOC Operations

Now that we've covered the considerations for SOC planning, building, and staffing, it's time to turn our attention to SOC operations. Security operations, as we will focus on, is the day to day work you will need to perform to keep your organization secure. This will include ensuring all of the SOC functions working to their best ability and communicating well with the auxiliary functions and external teams. Over the next few modules, we will cover more specifics about collection, detection, triage, investigation, and incident response, modeling the steps of each. We will pay particular attention to keys for success in each of these fields and build up more detailed mental models that will ensure we understand drivers of these functions' performance.

## SOC Theory and Mental Models

Some concepts and models to build the foundation of our SOC upon:

### Technical

- Modern defense mindset
- Threat models
- Threat intelligence
- Attack cycle models
- Defense in depth models
- Detection maturity levels
- Incident response stages

### Managerial

- The OODA loop and operations tempo
- Leadership vs. Management
- Building the SOC as an "Infinite Game"

### SOC Theory and Mental Models

As a lead in to the SOC function detail, we will first level set the overarching mindset and mental models that anyone in a SOC – manager, analyst, or otherwise, should think about on a daily basis. Information security and management literature are rich with mental models that can be relied upon to guide us in the right direction and help us understand and frame the situations we encounter. These models point us in the right direction when performing technical tasks such as assessing a potential attack and guide us in effective leadership and management of the SOC as well. The models covered here are the ones that have been most useful throughout my career.

## Modern Defense Mindset

A modern cyber defense requires...

1. Presumption of Compromise
2. Detection-Oriented Defense
3. Proactive Detection: Hunt Teams
4. Post-Exploitation Focus
5. Response-Driven Teams
6. Risk-Informed Strategy



"Prevention is ideal, detection is a must"

### Modern Defense Mindset

As a Blue Team, we need to make sure everyone is on the same page as to what constitutes a "modern defense mindset" and what it takes to defend a network against modern cyber attackers. SANS SEC511: "Continuous Monitoring and Security Operations"<sup>1</sup> course (a great course focused on operationalizing a modern defense, targeted towards Sr. analysts and architects) by Eric Conrad and Seth Misenar has a very well thought out rundown of some of the most important concepts, which are listed here.

- Presumption of compromise – The mindset of "we're probably fine" is dangerous and leads to a team that is more reactive than proactive. Modern cyber defense teams need to be constantly looking for evidence of intrusion.
- Detection-oriented defense – The "detection" here means "focusing on where prevention has failed" and follows directly from the previous item. If prevention has failed that means the adversary is already on the internal network, and we must turn our attention to what we previously might have considered "internal, trusted traffic". For a modern defense mindset, the internal network should never be considered "trusted", just "slightly more trusted".
- Proactive Detection: Hunt teams – Who will be the group proactively assuming compromise? Our SOCs hunt team. An alert driven SOC is limited by only the items they detect, a data-driven SOC uses team members to constantly search for what is missed.
- Post-exploitation focus – When deciding what to hunt for, post-exploitation stage attacks should be considered the priority. Attackers at this stage already have a foothold in the environment and once they get that foothold, things are likely to ramp up quickly in cost and complexity. Finding any successful exploitation should be considered the number one job of the hunt team.
- Response-Driven – When the hunt team locates something, they should be empowered to take care of that issue as quickly and thoroughly as possible. To do so, they must be provided with the tools and authority that will help them act quickly under pressure.

- Risk-Informed – Since most teams will never have an infinite security budget, the SOC should carefully consider their organization's threat model and how they can best align defenses to focus on preventing the most damaging scenarios.

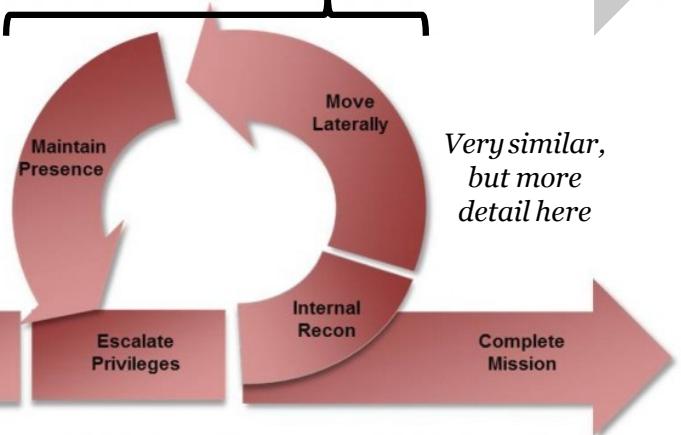
[1] SEC511: <https://www.sans.org/cyber-security-courses/continuous-monitoring-security-operations/>

## Chronological Attack Models

Lockheed Martin Cyber Kill Chain



- Meant to model APTs ONLY
- Often misused by applying to all alerts



SANS

MGT551 | Building and Leading Security Operations Centers

9

### Chronological Attack Models

You've almost certainly run into both the Lockheed Martin Cyber Kill Chain™ and Mandiant Attack Cycle in numerous reports and presentations throughout your time in infosec. We're bringing them up briefly in this class to make a couple of points.

The first item to address is the common misunderstanding. I now often hear objections saying the kill chain is "out of date" or "not as good as new models like MITRE ATT&CK". I don't think either of these statements is true. Remember that these are just models and are meant to serve a specific purpose. It's not that the Kill Chain isn't useful, it's that the goal is different than that of ATT&CK. Additionally, as any model does, the Kill Chain condenses reality into a simplified abstraction that leaves out detail. As the saying goes "All models are wrong, some are useful", and this applies here as well. The Kill Chain and the Mandiant Attack Cycle *are* useful to use as a mental model to visualize the progress and necessary steps for an attack. They are *not* meant to show all the options for an attack, cover every conceivable attack, or be applied to things like adware and other nuisances.

When used for their intended purpose and not beyond, these models continue to bring value in conceptualizing an attack. But remember the Kill Chain white paper *specifically* states that the model is for advanced persistent threats only<sup>1</sup>, if you use it beyond that, you're going to have a bad time. This is mentioned specifically because more than once I've seen tools and SOCs try to categorize every alert with kill chain steps and similar such endeavors, while this may be appropriate for *some* alerts, using it for all will undoubtedly end in confusion.

[1] <https://www.lockheedmartin.com/content/dam/lockheed-martin/rms/documents/cyber/LM-White-Paper-Intel-Driven-Defense.pdf>

## Defense in Depth and the Cyber Kill Chain



### Defense in Depth and the Cyber Kill Chain

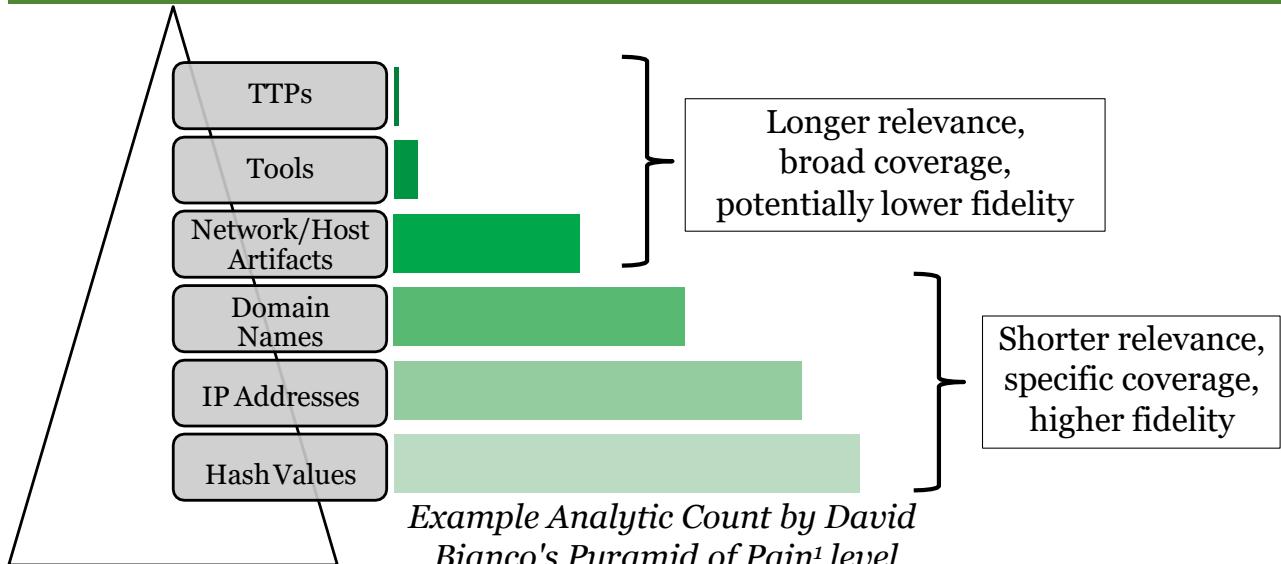
When it comes to designing and testing a defense in depth strategy, the kill chain and any other organized model can help us understand if we are doing a *complete* job by testing controls at each stage.

Consider the two organizations above, org 1 and org 2. Each has assessed how their detection capabilities work for each of the respective stages of the kill chain and plotted the results. Which organization would you rather go up against as an attacker? Probably organization 1! Why? Because while they have great coverage on the Delivery and Exploit stages, everything before and after that is poorly covered. If you have two solid tricks in your bag for those two stages, in all likelihood you will be able to pass their defenses.

Org 2, on the other hand, has a much better "defense in depth" posture. If an attack sneaks past the recon and weaponization stage, they still have an equally tough time across the rest of the kill chain. No stages are perfect, but they all provide decent coverage. This means the defending organization has made it so they must hurdle detections at all stages of the kill chain, and a single mistake in any of them on the attacker's part means they are kicked out and sent back to square one!

If you haven't attempted to test your SOC and organize the data in this fashion, you will be given a tool that will produce exactly this chart for you at the end of the course and learn how to use it in the final exercise!

## The Pyramid of Pain and Analytic Types



### The Pyramid of Pain and Analytic Types

Organizing testing by kill chain stage is one enlightening way to assess the breadth of your ruleset, but there are other ways that are useful as well. An alternative method is to look at your analytics for the *type* of detection they implement. You've undoubtedly seen David Bianco's "Pyramid of Pain" at some point in your infosec life, it's now referenced nearly as often as the cyber kill chain. However, have you ever thought to assess your analytics with the pyramid levels as the axis? As we previously mentioned, IOCs in many cases are highly perishable, or at least some are. If you are trying to cultivate a robust set of analytics, you want detection capabilities to cover each level of abstraction as well.

While it's harder to say what a "good" distribution is in this case, you should have some analytics working at *every* level, not only on the bottom. If you find *all* of your analytics are based on bottom layer items (hashes, IPs, domains), this means all your analytic capability is built off atomic IOCs and lists of bad websites. As the pyramid of pain describes<sup>2</sup>, these items are very easy to change on a whim, which means you have very easy to bypass detection capability. What you want is a spread, again, giving defense in "depth" across the pyramid. This means using atomic IOCs, artifact-based detections, tool-based detections, and the ability to catch the TTPs attackers use (such as those in the MITRE ATT&CK framework). You need both because while the items at the top are longer living and provide broad coverage, fidelity may be lower since they are looking for techniques, which tend to be harder to write an analytic for. On the bottom of the pyramid the indicators are very short-lived, but it's easy once a website is flagged as bad to identify an attack, someone either went there or they didn't, there's no gray area. That means these two types of analytics team up well together to provide depth of coverage in different scenarios. Consider how your analytics would look on this chart right now, and if you find their distribution to be lacking, make it a priority for the team to build up the piece that is missing.

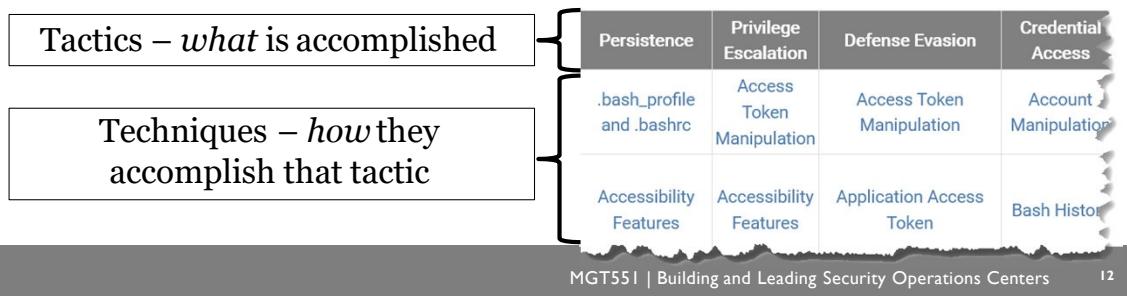
1 <http://detect-respond.blogspot.com/2013/03/the-pyramid-of-pain.html>

2 Ibid.

## TTP-Focused Attack Models – MITRE ATT&CK

Uses for MITRE ATT&CK:

- **Understand attacker TTPs**, how to mitigate / detect them
- **Prioritize** use cases
- **Track coverage** against requirements from threat intel
- Describe intrusions via a standardized vocabulary



SANS

MGT551 | Building and Leading Security Operations Centers

12

### TTP Focused Attack Models – MITRE ATT&CK

A different view on cyber attacks is the now highly popular MITRE ATT&CK<sup>1</sup> framework. This model went from relatively unknown to one of the biggest things in cyber security relatively overnight, and it's a good thing it did, we needed it! MITRE ATT&CK sets out not to describe an attack chronologically, but instead build a vocabulary of standardized *tactics and techniques* that attackers use once they are inside an environment.

Tactics describe some of the goals that attackers may need to accomplish to be successful in their intrusion and the ways they do so are listed below as techniques. The idea is the model will be constantly kept up to date by the team at MITRE. Any time a new attack technique is found, it will be added under one or more of the tactics. If you, as the cyber defense teams, can detect the items relevant to your environment on the MITRE ATT&CK framework, give yourself a pat on the back, because you're off to a great start!

While even the staff at MITRE admit that ATT&CK is not the perfect model (and remember, no model is), do not let minor imperfections get in the way of the abundant usefulness. It is a *highly* useful tool for a number of reasons:

- Understanding attacker TTPs – Especially for newer analysts, the ATT&CK matrix provides an amazing learning opportunity to see how attackers, Red Teamers, and pen testers will be attempting to bypass security.
- Prioritization – The ATT&CK framework gives us an inventory of things we may or may not need to be able to detect and information about each to help decide how important they are. Even better than that, each technique is also tracked along with the groups that are known to use it. So, if you have no threat intel for example, but know that APT X traditionally attacks organizations in your industry, you can look at the MITRE ATT&CK matrix and see which techniques are used by APT X and prioritize them for implementation and testing.
- Track Coverage – MITRE ATT&CK has been used successfully as a way to objectively measure defensive team improvement in many organizations. Once you have a prioritized list of techniques you must catch, turn the coverage of those techniques into a numeric metric that is regularly reported

upon. A rise in either the percentage of techniques that are covered or the absolute count (remember either can change since techniques will be added to the list over time) means the Blue Team is measurably improving by adding new detection capabilities! Is it perfect, not exactly, but it's *way* better than nothing, and this is why MITRE ATT&CK has caught on like wildfire!

[1] <https://attack.mitre.org/>

## ATT&CK Sub-Techniques

In 2020 ATT&CK changed...

- Introduction of sub-techniques
- Further and better organization
- Parent/child layout
  - Top-level technique name
  - Child-level-specific method
- Great for overall usefulness
  - Modification required for tools/metrics using old system



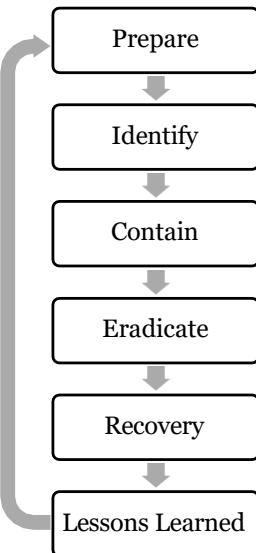
### ATT&CK Sub-Techniques

While you may be plenty familiar with the MITRE ATT&CK framework as it has existed throughout the last several years, in 2020 it underwent a monumental change. The MITRE team, as of July 2020 has released the version of the MITRE ATT&CK Matrix which includes a full reconsideration and refactoring of techniques into "sub-techniques". The techniques will now include higher-level techniques like "Command and Scripting Interpreter" with sub-techniques of things like "PowerShell" and "AppleScript", further specifying the methods of performing the parent-level technique. This is likely to be one of the most important and large-scale shifts the ATT&CK framework has undergone since its inception. As a result of this change, any tools your vendors provide with ATT&CK mappings and use case alignment you've performed against the previous version will need to be revisited to make it compatible with the new format. Despite the short-term pain this may cause, in the long run, this will undoubtedly be a huge boon for the framework and its organization.

The new organization is announced and explained in the blog post referenced below<sup>1</sup>. If you haven't yet gotten a chance to look at the new version in detail, you will get to see it in an upcoming lab exercise.

[1] <https://medium.com/mitre-attack/attack-subs-what-you-need-to-know-99bce414ae0b>

## Incident Response Cycle



### The Incident Response Cycle<sup>1</sup> – "PICERL"

- Kill chain from the **defender's perspective**
- Based off NIST SP800-61
- Covers **detection, response, and improvement**
  - **Detection** – Prepare, Identify
  - **Response** – Contain, Eradicate, Recovery
  - **Improvement** – Lessons Learned
- Highlights the importance of **feedback** from incidents

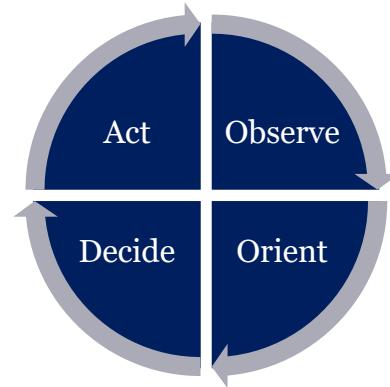
### Incident Response Cycle

A prominent model for understanding how incident detection and response should work is the PICERL model or "incident response cycle", based on the information in NIST800-61<sup>1</sup> "Computer Security Incident Handling Guide". This cycle can be viewed as the kill chain from the defender's perspective because it enumerates what the SOC should be doing in order to identify and respond to an attack. This model is useful in a number of ways. One way is that it highlights the link between each completed incident and how it should feed back into better detection for the next time (similar to the F3EAD cycle). It also can serve as a "what do I do next?" guide for analysts that may be unsure of what the right next move should be in a given situation. Since most analysts will drop in at the "Identify" stage where an alert has just been set off, once they confirm it, if they can remember PICERL, they can remember the next step should be to contain the incident and stop the bleeding.

[1] <https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-61r2.pdf>

## The OODA Loop

- A goal for all SOCs is to keep **operations tempo** high
- The “**OODA loop**” concept helps conceptualize this
  - Observe
  - Orient
  - Decide
  - Act
- How fast can you observe and react?
  - In any head-to-head competition, the faster opponent will win – John Boyd
  - **Key to success: The orient step**



### The OODA Loop

Designed by fighter pilot and military strategist John Boyd in the 1960s, the OODA or "Observe, Orient, Decide, Act" loop is a model to show how even with the disadvantage of poorer technical capabilities and with imperfect information, a fighter pilot could still win in a dogfight using quick, decisive action.

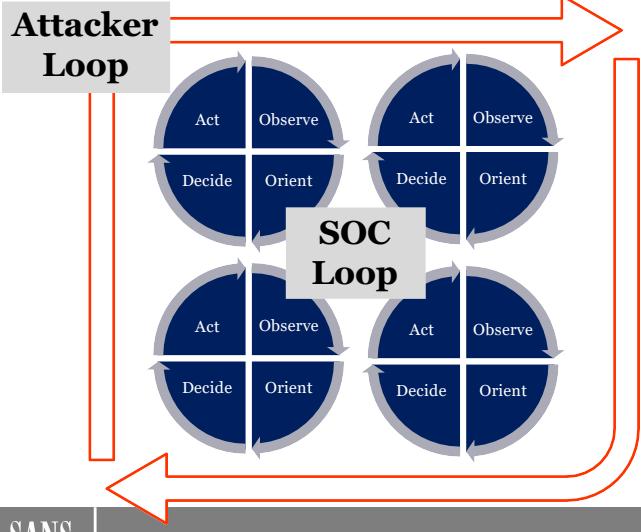
For a SOC, the OODA loop represents the stages that your team must go through when going head-to-head against the adversary. We've previously mentioned that operations tempo is a key factor in your ability to successfully remove an adversary from your network. The OODA loop is a mental model of the steps that must be iterated through that drive the operations tempo you are capable of achieving. The idea is the faster you can go through the loop, the quicker you will be to react to changing conditions and new surprise tactics or attacks from your adversary. The caveat and most important step is the "Orient" phase, in which you assess how to interpret what you are observing, and having an accurate assessment is required for the rest of the activities to go as planned. Mental models and this stage are further explored in the analyst-focused SEC450 – Blue Team Fundamentals.

In essence, the takeaway from the OODA loop model is that the entity that has the faster (and more accurate) running loop in any given head-to-head situation is likely to win due to their ability to assess the situation and react more quickly. If you'd like to do a deeper dive on this topic, there is a long-form writeup on the history of OODA loops and a deeper dive on its meaning and application to everyday life in the article "The Tao of Boyd: How to Master the OODA Loop" from Bret and Kate McKay.<sup>1</sup>

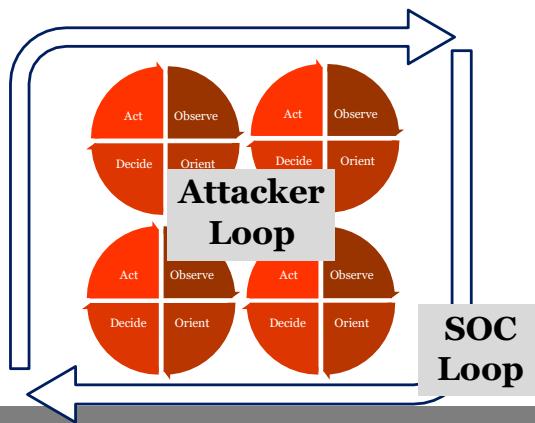
[1] <https://www.artofmanliness.com/articles/ooda-loop/>

## Which SOC Are you?

### Trained and ready SOC



Untrained, slow moving SOC without permission and tools to act quickly



SANS

MGT551 | Building and Leading Security Operations Centers

17

### Which SOC Are You?

Here's how to think about how the OODA loop affects your daily life in the SOC

- If you are the SOC on the left, as soon as you detect an adversary, you can organize and move quickly, your people are trained, and you react fast will well trained analysts who know how to contain and stop an attack dead in its tracks
- If you are a SOC without the proper tools necessary, no visibility, and lacking the permission required to view the data you need and take decisive action during an attack, your enemy will run circles around you. They will be able to make decisions and alter their course of actions at a much faster pace, making it nearly impossible to keep up with them and kick them out.

## Management vs. Leadership

Running a SOC takes both **management** and **leadership**:

- “*The difference between a manager and a leader is that a **manager** focuses on doing **things right**, while a **leader** focuses on doing the **right things***” – Peter Drucker<sup>1</sup>
- **Leadership** – Identifying *what* to do, choosing a vision, strategy, and direction
- **Management** – Ensuring quality execution of the chosen plan, getting the best possible outcome from your team
- Management in the wrong direction will **not** produce good results



Leadership



Management

### Management vs. Leadership

One important concept to remember is that running a SOC, or any successful team, requires both leadership *and* management. While these terms are often used interchangeably, Peter Drucker, one of the biggest names in the philosophy of management, made an interesting differentiation between the two that is useful to remember. In his book.... He says “The difference between a manager and a leader is that a **manager** focuses on doing **things right**, while a **leader** focuses on doing the **right things**” [emphasis added]. This quote should remind you that in order to be successful you must not only effectively execute on your chosen work but pick the *right* work to be doing in the first place. In other words, both your strategy and execution must be correct in order to get the results we are looking for.

To fulfill your role as leader of a SOC it falls upon you to keep your ear open to the news and trends of the industry, and to be aware of what is possible given your environment and resources. In such a rapidly changing field, the solution you need may not exist one day and show up the next, so it is your responsibility for one, to understand how to be successful, the tools required to get you there, and know how to choose the best one for a given situation.

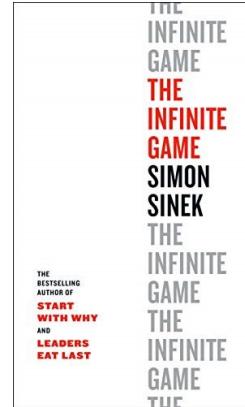
Secondarily, you must also manage the SOC in its day to day operations. Ensuring the workload is balanced, people are following process, and that you ultimately are fulfilling the job that those above you have trusted you to do – secure the organization. This is where metrics, training, procedure, and technology step in and enable you to execute with efficiency on your chosen mission.

## The Two Types of Games

In “The Infinite Game”, Simon Sinek describes **two types of games:**

### 1. Finite Games

- Defined rules, a beginning and end, known players
- In finite games, it is clear who wins**
- Examples: Sports, board games*



### 2. Infinite Games

- Minimal rules, no known end, players come and go
- In infinite games, there are no winners**
- The goal: Stay in the game as long as possible**
- Examples: Business, marriage, education, life*

## The Two Types of Games

In the book “*The Infinite Game*” by Simon Sinek (which is based on the ideas in James P. Carse’s 1986 book “*Finite and Infinite Games: A Vision of Life as Play and Possibility*”), Sinek describes the two types of “games” – “Finite” and “Infinite”. While the book is focused on business strategy, the idea described provides an interesting mental model for the mindset required for a successful and sustainable, long-term operation of any given endeavor, Security Operations Centers included.

As Sinek describes it, finite "games" are played by known players, have fixed rules, an agreed-upon objective, and a defined ending. Activities such as sports fall into this category, the game has a beginning and end, the players are defined, and the rules are well-known. The goal of the finite game is clear – to beat out your opponent.

The secondary type of game is the infinite game. These games have players that can come and go, both known and unknown, and the rules are not necessarily exact or agreed upon. Examples of these types of games are politics, policing, business, education, marriage, and in this course author’s view, running a security operations center. These types of activities have much poorer defined limitations, are more complex, and the strategies used to play are much different.

The primary difference between finite and infinite games is how you “win”. While finite games always have a clear outcome at the ending time, infinite games are much different. In infinite games, no one necessarily “wins” because an infinite game, by definition, has no end. Sure, in business, for example, there are short-term items and goals that you can end up victorious in achieving, but tomorrow the struggle will repeat itself again, and a new goal will always be waiting. Which business is “best” then depends on how you are measuring, and thus the concept is sort of meaningless and undefined. We can see evidence of this in how nearly every company somehow claims to be the best or favorite in their own commercials—how they measured this, though, is often cherry-picked or arbitrary.

While finite games have a single metric that determines the winner, infinite games, in contrast, have multiple metrics and often no single or best way to compare players. **Given that there is no such thing as “winning” in infinite games, the goal, instead, is to stay in the game as a player as long as possible and to continue to play and perpetuate the game.<sup>1</sup>**

[1] <https://simonsinek.com/product/the-infinite-game/>

## Sinek's Main Points

- Playing an **infinite game** with a **finite strategy** leads to **failure**
- Finite and Infinite games require VERY different approaches
- You must use the right strategy for **long-term success**
  - “Infinite games require infinite strategies”
- Businesses, managers, and employees often fail to realize this
  - Prioritizing short-term “wins” before long-term sustainability
  - Focusing on observable items while ignoring second-order effects
  - Placing personal reputation over team success

Consider how this translates to a SOC...

### Sinek's Main Points

Breaking the world down into these two types of games is useful in that it can drive a shift in mindset for the strategy of a game as both finite and infinite games require incredibly different approaches and priorities, which is the focus of Sinek's book. In it, Sinek successfully demonstrates with many different examples of leaders trading short-sighted near-term “wins” (often incorrectly looking at business as a finite game) at the expense of long-term stability and sustainability. In other words, playing an infinite game with a finite game mindset, a strategy he shows is doomed for sub-optimal results at best, and failure at worst.

As individual people, Sinek says finite-minded players tend to over-focus on achieving short-term personal goals. As finite-minded businesses, organizations may, for example, rely too heavily on a single product, while missing the bigger picture. Finite-minded players in a finite game tend to miss, or even purposefully overlook, the second order effects their short-term actions may have due to the individual and immediate benefits these actions may have (think of those who profited, but were also responsible for the 2008 financial crisis). Human nature may drive us to look at the personal rewards that can be gained while brushing aside the long-term problems an action may cause on our team or organization. In an infinite game, however, to succeed, we must focus on the long term, by definition, and this type of temptation and finite-minded thinking is exactly what ultimately leads to the failure of those players.

## Which Game Are You Playing?

What does your SOC reward / optimize for?

- Ticket numbers closed?
- Time-based analyst performance goals?
- Metrics dictated from above?
- Following rigid process?
- Individual wins?
- Solving the problems of *now* and ignoring possible second-order effects?



If so, consider which strategy you might be playing with...

### Which Game Are You Playing?

While it may not initially be easy to think of how this could manifest, consider whether, in your SOC, you are more focused on the short term or long term. The page above lists some metrics and items that are commonly focused on in the SOC. While they may seem normal initially, step back and consider how they might play out on the psychology of employees over the long term, and whether *that* will have an impact on your retention.

Through teaching for SANS for years I've talked to many a SOC analyst and am always curious about what their pain points are and where they feel boxed in. These items are often the theme of those conversations. Though these items may be common in the industry, anecdotally at least, they also seem to be found in teams that are less happy with their work environment and don't plan on staying in their positions long term. Could it be because many teams implement them and follow them with finite game thinking?

## Playing with an Infinite Strategy

- A better approach:
  - Use a strategy that keeps you playing the infinite security game
  - Continuing to play the game requires A-players
  - Training and retaining A-players requires a great place to work
- **Therefore: Optimize to run a human-focused SOC!**
- This means optimizing for
  - Employee engagement
  - Job satisfaction and retention
  - Challenging employees to grow, providing training
  - Giving analysts space for freedom, flexibility, and creativity
  - What works in the *long term* and not just now



### Playing with an Infinite Strategy

Not only does Sinek's book hit the mark in giving advice on operation of a business in a way that will cultivate dedicated employees and happy customers, but one can see how this applies to a SOC as well. Since security is never "solved" or "done", by nature, the SOC is playing what Sinek would call an infinite game, and that means we must operate according to the "infinite game" set of rules too. If our SOC is short-term focused on finite-game minded strategy, expect poor performance, and impact on team morale.

If, however, we operate with Sinek's "infinite game" strategy, we work with an understanding that the problem must be solved in the long term as a team, and that grinding people down for short-term wins and metrics will actually work to our detriment in the long term. Therefore, if following an infinite game strategy, instead of going for short-term wins, we should take the slow and steady pace of focusing on developing people and keeping them around for the long term. In other words, running a "human-focused SOC". To me, this means above all running your SOC with the ideals that will keep employees constantly challenged, growing, creative, and ultimately around for the long term. When *this* strategy is followed, the business, management *and* the employee win, and burnout is avoided. We won't dive further into this yet, but later on, in the course, we will revisit this idea backed with scientific research and show the specific factors we can cultivate to ensure our SOC is on track with having an infinite strategy.

## SOC Theory and Models Summary

- Cyber security mental models are numerous
  - Help analysts contextualize and understand complex events
  - Give us a shared standard to reference
  - Help new analysts frame what they might be seeing
- Understand:
  - What each model is intended to explain
  - How they benefit analysts in understanding a situation
  - Where they can be applied
  - The limits and common misapplications



### SOC Theory and Models Summary

Having a strong grasp of cybersecurity relies upon understanding and being able to draw quickly upon mental models. Not only the ones we've seen here, but of every experience you've seen repetitively throughout your career. These formal models will no doubt be thought of several times a day as you try to gauge incident severity and direct what may need to happen next. The goal of this chapter was not to introduce them, as at a manager level we can assume you're already familiar with most of them. The goal was to discuss their proper place in the SOC, how to avoid misapplying them, and how they can help operations in a day to day capacity. Ensure your analysts are familiar especially with the attack and defense models, both chronological and attack tactic-focused models will help them triage and diagnose alerts more accurately.

# Course Roadmap

- Book 1: SOC Design and Operational Planning
- *Book 2: SOC Telemetry and Analysis*
- Book 3: Attack Detection, Threat Hunting, and Triage
- Book 4: Incident Response
- Book 5: Metrics, Automation, and Continuous Improvement

## SECTION I

Introduction

Mindset and Preparation

- Cyber Defense Theory and Mental Models

**SOC Data Collection**

- Other Monitoring Use Cases

*• Exercise 2.1: Attack Path and Data Source Assessment*

Collection and Monitoring

- Using MITRE ATT&CK to Plan Collection

*• Exercise 2.2: Prioritizing and Visualizing Attack Trees*

- Cyber Threat Intelligence

*• Exercise 2.3: Writing Priority Intelligence Requirements*

- Practical Collection Concerns

- Prevention and the Future of Security

*• Summary and Cyber42 Day 2*



This page intentionally left blank.

## In This Module

- Modeling the collection function
  - What gets written, what gets collected?
- Collection topics
  - Collection goals
  - Collection Tools
  - What to collect
  - New encryption protocols that disrupt collection
  - Cloud data collection



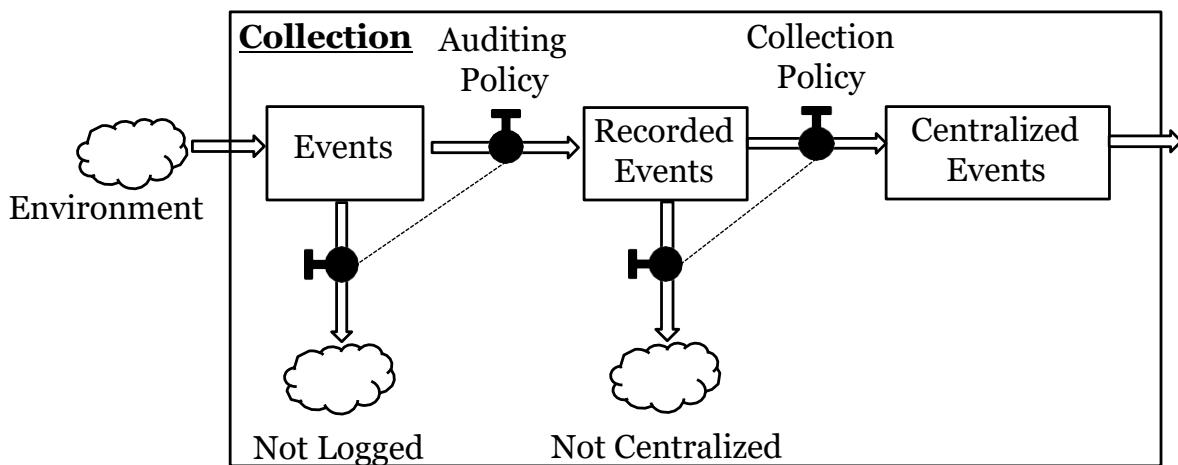
## In This Module

In this section of the course, we'll talk about the first step in the core SOC process – data collection. We'll review a systems-level model of the collection function, discuss what the most important sources of log data are (and how to determine them for your organization), and discuss some of the practical considerations you will need for creating an effective collection function. We'll also discuss some new encryption protocols and how they complicate the data that was previously very simple to collect.

Remember that in the SOC process, what happens upstream will echo all the way down the line of processes that occur later on.

## The Collection System

### How collection works:



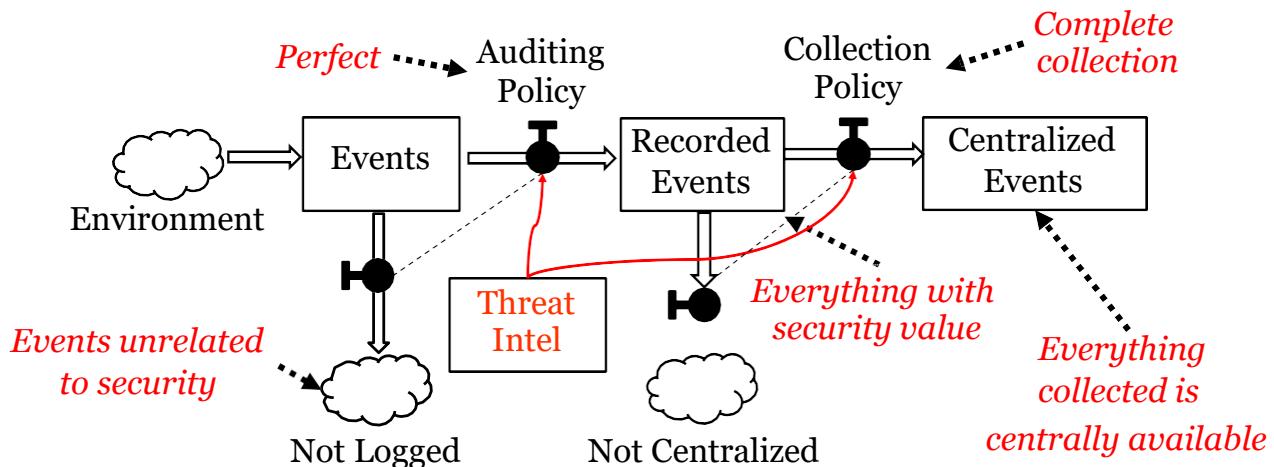
### The Collection System

To further dive into each core SOC function, throughout the next few modules we will take the basic input/output diagrams we saw in the SOC Functions section of the course and expand them out to extract the detail. Doing so will help us tease out the nuance of each function, understand how it operates, and zero-in on how we can make it better.

The first of the core functions to discuss is your data collection capability. This is the infrastructure, policy, and people that enable you to keep a finger on the pulse of the network and the devices on it. Collection in our description will be everything that happens between events being generated in the environment, to those events being recorded, and ultimately collected. As shown above, in the systems-style diagram, everything that enters your collection "system" is either logged / seen on the network, or not. Those things that are logged or recorded in some way are then either centralized to a SIEM (or at least recorded on a centrally available sensor such as an IDS) or not. The drivers of whether these two steps occur for any given event are first up to your auditing policy – do you record that event in a local log or can you see it via a network sensor? If so, is that data stored locally on the device, or is it brought to a central location where it can be viewed and correlated with other data?

## The Ideal Collection System

The ideal security data collection system:



### The Ideal Collection System

One way to learn how we might be able to improve our own SOC functions is to theorize what the system would look like if it were working ideally. Doing so gives us an understanding of the goal for each individual component of the system and helps us decompose which specific piece of our own collection pipeline would bring the best improvements to the whole system.

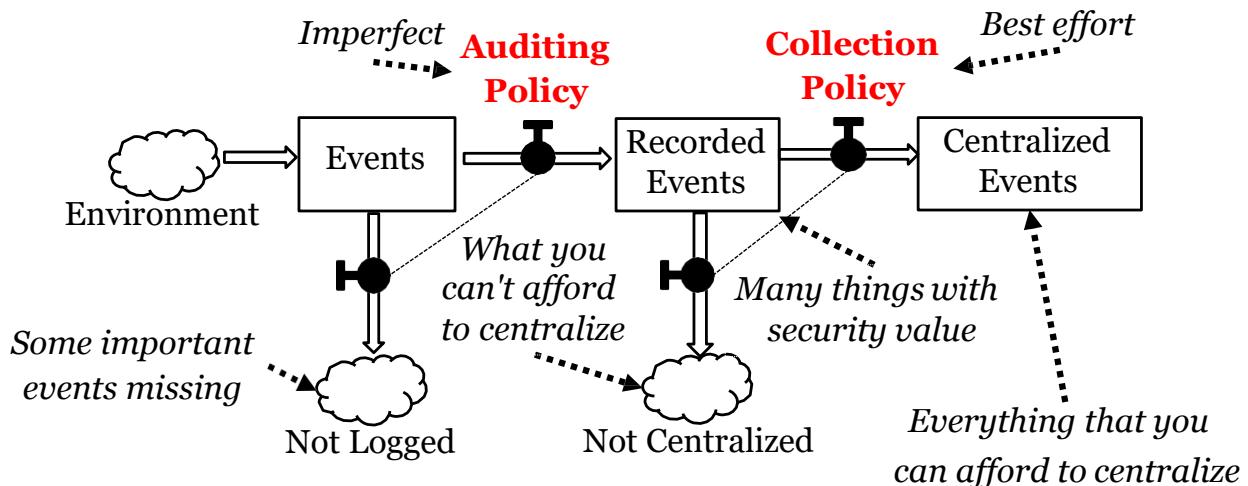
For the collection system, an ideal set up would involve the components shown above in the state noted.

- The events that come in would only be audited or recorded locally if they were of potential security value. In other words, we wouldn't waste any capacity even producing logs for things that were not potentially of interest.
- Of the recorded events of security value, ideally, *everything* would be able to be centralized to a system that could use it for correlation and enrichment.
- The auditing policy that controls the writing of the events, and the collection policy that dictates whether the data is centralized or not would, therefore, have to be perfectly in tune with what is required for attack detection. What feeds this capability? One of the major inputs is threat intelligence. Whether you have a dedicated threat intel team or not, this is one of the major inputs where threat intel will have an influence on your ability to detect attacks.

Next, let's compare this with what is more common in the average organization.

## Realistic Collection System

Realistic security data collection system:



### Realistic Collection System

In your average organization, it looks a bit more like this:

- Many important events are recorded locally to PCs or may be recorded on network sensors, but not all of them due to gaps in what is known, what is affordable, or what is visible.
- Of what is recorded, the collection policy is a best attempt job to gather the most practical items and send them in centrally. Again, volume, access, and threat intel may drive this being a best effort solution that is not always guaranteed to have everything the security team requires. If it is recorded but not centrally collected, the security team can still access the individual device to grab the data.

Given this state of affairs we can then assume that most organizations are not meeting the requirements for a sure attack detection capability, they may be partially covered, but not completely.

If we don't set up an auditing policy and network infrastructure that supports recording the events of interest, nothing further downstream can help us since it is the first step in the system. Assuming we do have thorough collection, that must then be matched with a solid centralization and collection strategy as well. No amount of SIEM analytics and threat hunting will be able to pick up attacks in data that just isn't available, so starting off right by focusing on your auditing and collection policy gives you the best fighting chance of attack detection downstream.

## NSM and CSM

The two components of complete collection:

- **Network Security Monitoring (NSM)**
  - Captured network data "off the wire"
  - Full data – PCAP
  - Network Metadata – NetFlow / Transaction data
- **Continuous Security Monitoring (CSM)**
  - Endpoint /device-generated data
  - Processes, authentications, services, autoruns, vulns, etc.
- BOTH are required for success



### NSM and CSM

Conceptually the collection function can be divided into two major types of data – information that comes by capturing network data off the wire, and data generated from endpoints. While some of the same data could conceivably be gathered from either location (such as firewall logs from the network or host firewall), thinking about it in this way helps us understand the role of each data set.

The network data, often pulled off the wire with a network tap or switch mirror port, tells us the undeniable truth about who is talking to whom on the network, which protocols they are using, and what the content of that conversation is (if it isn't encrypted). Nearly all attacks must leverage the network, therefore if you have a recording of transactions, the challenge is only in identifying the malicious items within that data.

## Open-Source Network Data Collection Tools

Don't have the NSM solution you'd like?

Check out these awesome free options!



*Metadata, IDS, and PCAP capture*



*Pre-made NSM distros*



MGT551 | Building and Leading Security Operations Centers

30

### Open Source Network Data Collection Tools

If you don't have NSM capabilities yet, or are looking to improve what you do have, no need to spend any extra money! The open source toolset available for Blue Teams to perform network security monitoring is incredibly well developed and offers tools for metadata capture (Zeek security, Suricata) as well as intrusion detection (Suricata, Snort), and full PCAP capture as well! (Moloch / SecurityOnion). There are even full Linux distributions put together ready for you to deploy a fully functioning system with a minimum of work. SecurityOnion<sup>1</sup> is the most popular, followed by RockNSM<sup>2</sup>. If you're in to jumping into data science with Apache Spark and Jupyter notebooks, Roberto Rodriguez has created Hunting ELK or "HELK"<sup>3</sup>.

1 <https://securityonionsolutions.com>

2 <https://rocknsm.io/>

3 <https://github.com/Cyb3rWard0g/HELK>

## Open-Source Host Data Collection Tools

Excellent free CSM data sources exist as well!



*Supplemental event generation*

*Data shippers*

*Data search and presentation*

SANS

MGT551 | Building and Leading Security Operations Centers

31

### Open Source Host Data Collection Tools

For CSM data collection there is also a wealth of outstanding free and open-source tools that can be used to supplement or create data you may not have. These can be divided into tools that actually generate the logs themselves based on system events, the tools that ship those events to a centralized system, and the tools that present the data to analysts.

In the data generation category, we have OSSEC HIDS, Sysinternals Sysmon (for Windows), osquery, and auditd for Linux. These tools can function together to record nearly anything happening on an endpoint that you might be interested in from a security perspective and give you visibility very similar to commercial EDR solutions.

The next step is taking this newly generated data and shipping it to a central location. Some tools have their own centralization mechanism built-in, while others just generate logs and let you pick them up however you please. If you need a highly flexible and free log agent, NXLog's Community Edition can likely provide what you need. Other tools like Beats are available as well if you use the Elastic Stack ecosystem. Fluentd is another popular open-source data collector and shipper.

Finally, we have the tools that present the collected data to analysts. Kolide is a GUI front end to make browsing the data captured by osquery easier, Wazuh is data collection and presentation plus more, it provides data collection and presentation (Elastic stack-based) as well as take action based on potential attacks similar to EDR. It is a great full security monitoring and incident response solution.

## Most Important Network-Based Data

### Monitoring Goals:

- Layer 1 & 2
  - What is plugged in?
- Layer 3
  - Source and destination talking
- Layer 4
  - What protocol being used?
- Layer 7
  - Application specifics
  - Conversation metadata
  - Conversation Content
- IOC matches at any layer

### Highest-value sources:

- Network service logs
  - Proxy / web logs
  - DNS (external lookups)
  - DHCP, NAC (802.1X)
  - Lateral movement protocols – SSH, SMB, PS Remoting, VNC, RDP
- Devices / Security Tools
  - NIDS/NIPS
  - Firewall service logs
  - **Cloud** & on-Prem – NetFlow
- Packet capture – cloud / on-prem



### Most Important Network-Based Data

For your network-based log sources consider the questions you're trying to answer with the data collection. Since the network must be used in some capacity by nearly all attacks, there will necessarily be evidence if the right information is captured. To record evidence of an attack in progress we can break the network down into layers and questions those layers can answer as shown above.

The key data we can collect to answer these questions is shown on the right. Obviously, output from our security and other network appliances is a great start – firewalls, NetFlow from switches and routers, and output from a network intrusion detection and prevention system makes a great start. We need more than that, however, because security appliances typically can only catch attacks we already know about. To cover the rest, we need network service logs. Network service logs are the type of data produced by Zeek (formerly Bro), Suricata, or any other sensor that watches protocols and records every transaction at the application level. These tools record every event that occurs and allow threat hunters to identify anomalies during threat hunting that might otherwise have gone unnoticed. Key network service logs such as proxy and weblogs, DNS, and DHCP help us find what is on the network and where those devices are going. Internal tracking of lateral movement protocols like SSH, SMB, PowerShell Remoting, VNC, and RDP helps us catch potential lateral movement.

## Most Important Host-Based Data

### Monitoring Goals:

- Who is logging in?
- What is running?
- What services, autoruns, and scheduled tasks are present?
- IOC matches for all

### Highest-value sources:

- Host events: OS, Sysmon, Auditd
  - Authentication success/failures
  - Process creation & arguments
  - Services, autoruns, scheduled tasks
- Security Tools
  - Anti-Virus / EDR
  - HIDS/HIPS
  - Host firewall (heavily filtered)



## Most Important Host-Based Log Sources

The same logic we use for network devices applies to recording information from hosts as well. We need the output of security tools such as antivirus and host intrusion detection and prevention systems because these match known IOCs and give us high-fidelity attack detections. Beyond the realm of the known, we need further coverage of all activity on the system. The events on the right above show some of the best value events you can collect.

- Authentication events are commonly collected and make a great starting point for looking for malicious activity. While most SOCs will look for things like brute force attempts, consider looking deeper at the context of logins and where they come from. If you can identify where highly privileged accounts such as domain admin for example, are used anywhere other than their expected location, this makes a great fast-acting detection.
- Host process creation events tell us what ran, when, where it ran from, its hash and signature, and the arguments used when it started up. Comparing this information across a large fleet of computers makes it easy to surface anomalies.
- Techniques used for malware persistence are another high-value item to monitor. This includes autorun keys and programs, any installed services, scheduled tasks, and more. Since most malware needs to have some sort of persistence mechanism to start up, if you can compare the programs that automatically start up across your enterprise, the malicious items can be found by stack ranking the most commonly autorun items.

## New Challenges for Network-Based Data Collection

- New standards are making NSM more difficult
  - **TLS1.3**
  - DNS over HTTPS (**DoH**) / DNS over TLS (**DoT**)
  - **HTTP/2 & HTTP/3** add additional complexity to protocol analysis
- Effects:
  - Encryption is becoming mandatory in many cases
  - Ability to read certificate details is disappearing
  - Preventing rogue DNS server usage is MUCH harder
  - Interception of traffic is becoming more complicated

### New Challenges for Network-Based Data Collection

While protocols like DNS and HTTP(S) were rather simple in years past, making collection of metadata and even content simple, new standards are introducing massive changes. With the advent and adoption of newer protocols such as TLS1.3, DoH/DoT for DNS requests, and HTTP/2 and 3, monitoring our own networks is rapidly becoming much more difficult. Make no mistake, these protocols are outstanding in their ability to shield the user from malicious traffic manipulation and snooping, unfortunately, that also means that blue teamers will also have a much harder time viewing what is going on in their own enterprise. Let's take a quick dive into how these new standards will affect our visibility in the SOC, and what options you have to deal with it.

## TLS1.3

### What is it?

- A new TLS encryption standard, the successor to TLS1.2

### The NSM-relevant details:

Difference from previous TLS standards	What it means for us
Enforces "perfect forward secrecy" for traffic	<b>For inbound traffic</b> – you can't just use the private key of your webserver to decrypt all inbound traffic <b>For outbound traffic</b> – you must capture data from endpoint, or perform <i>active</i> interception
Encrypted certificate details	We can no longer passively record certificate info. Leaves (at best) only the domain the user is connecting to visible in Server Name Indicator (SNI) field
Supports encrypted SNI (eSNI)	Can hide domain too, leaving us only able to see encrypted traffic to a destination IP, no other useful details. (Currently this is not used in most cases)

## TLS1.3

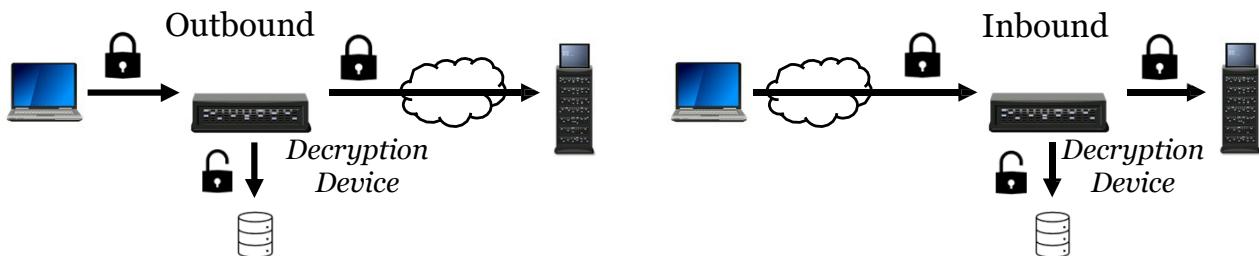
What is TLS1.3? It's a new TLS encryption standard, released in 2018 that is replacing the previous version, TLS1.2 that was released in 2008. As part of the new standard, many of the issues that were present in TLS1.2 are fixed or mistake proofed, meaning it *is* a much more secure standard, but also one that is much harder for a security operations team to perform inspection on as well. The slide and the bullets below summarize the key impact it will have on network security operations.

- TLS 1.3 only allows cipher suites that provide "perfect forward secrecy" (PFS). What this means in practice is that you must collect unique information generated in *every single TLS connection* in order to decrypt that connection. In older standards, interception for all traffic inbound to your own website could be decrypted with only knowledge of that server's private key (assuming PFS, which was possible in TLS1.2, was purposefully disabled) this is no longer possible as non-PFS cipher suites are not supported with TLS1.3. For outbound traffic, this means that you will either need to perform active interception with a proxy, or for passive inspection, collect information from every endpoint about every TLS connection. In addition, once a connection is made to a TLS1.3 site through an interception proxy, that proxy must be present for the entire conversation (not true of previous versions).
- Certificate details for the site the user is connecting to, previously available in TLS1.2 and earlier, are no longer visible. This means it will be harder for you to determine whether the connection is legitimate, or malware. There *is* still potential to detect the domain name the user is connecting to via the "SNI" field, however.
- Encrypted SNI "eSNI" is supported, which may eventually become standard and remove the ability to see which domain the user is connecting to (or block sites based on domain name). This is done via the site first publishing their public key in a known DNS record, then the client does a (potentially encrypted with DoT/DoH) DNS lookup for that public key record *before* connecting to the website. Fortunately, at least for now, this is not commonly used. To check your browser's capability to use eSNI, visit Cloudflare's [encryptedsni.com](https://encryptedsni.com).

## TLS1.3 Active Interception

### Option 1: Active interception

- Proxy all traffic by breaking and remaking the connection
  - Requires installing a root CA certificate on all devices
  - Now more complex to set up, requires bypass for certain site categories
  - Almost certainly your best option for TLS1.3 data capture



### TLS1.3 Active Interception

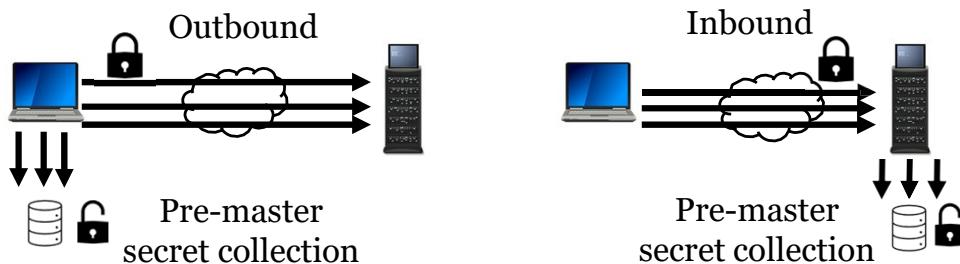
One option you continue to have is to use active interception to effectively put a machine in the middle of each connection, both outbound and inbound, that will terminate the connection on one side of the proxy and reestablish it on the other. The downside to active interception is that it requires you to generate a root certificate authority certificate that must be pre-installed on each client device for interception before the connection is reestablished to the final endpoint using the real certificate. This can cause issues with applications that use certificate pinning and may require special exception rules for such cases. From TLS1.2 to TLS1.3, this method largely is unchanged except for the fact that the proxy cannot drop out of the connection after the session is established, meaning you may need more hardware resources for this method.

Active interception will likely become the chosen solution for most companies, not because active interception is different with TLS1.3, but because the alternatives (passive decryption) have become much more difficult for inbound traffic, which will push orgs in this direction. Since active interception is already supported by many proxy and next-gen firewall vendors, the good news is that it is likely you already have the *technical* capability to do this.

## TLS1.3 Interception

### Option 2: Passive decryption

- Used to be easy for inbound, now is more difficult
- Outbound passive decryption is also more complex
- Requires gathering "pre-master secret" from one side of every individual TLS connection you want to decrypt



### TLS1.3 Passive Interception

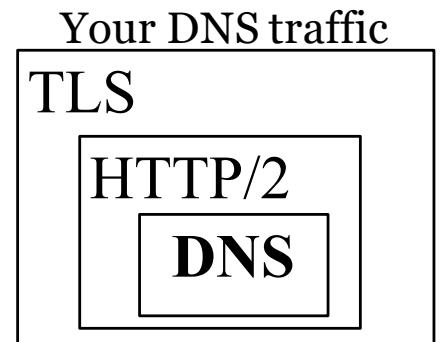
The second option for dealing with TLS1.3 is passive interception. The biggest change here is the ease of which decrypt inbound traffic to your servers can be done. In previous TLS versions, simply giving the decryption device a copy of your private key and disabling PFS was enough to capture all traffic. In TLS1.3 this is not possible due to PFS enforcement. Instead of just using the private key, TLS1.3 will require the collection and storage of a unique key, the "pre-master secrets", from *every single TLS connection* for decryption (which yes, does then negate the whole point of perfect forward secrecy).

For many organizations, inbound traffic will be the big change as collecting these pre-master secrets may be unsupported by the applications in use. Where applications do not support pre-master secret collection, organizations will likely revert to active interception, requiring additional hardware and a new configuration.

For outbound traffic, passive interception is feasible, but again pre-master secrets would have to be collected from every single TLS connection that was established. Since operating systems do not support doing this on the whole, and even more applications do not support it, solutions for passive interception require either the cooperation of the application making the connection (Firefox for example, can write these keys to a text file), or a third-party solution that can otherwise harvest these keys as they are made – a much more technically complex feat.

## DNS over HTTPS (DoH)

- What is it?
  - DNS over port 443 using TLS/HTTPS
  - In other words: **DNS now looks like TLS/HTTP/2 on your network, because it is**
  - The "winning standard" compared to DoT
- Ramifications:
  - DNS is now encrypted (by default in some cases)
  - To log requests, you must be able to intercept TLS or provide your own DoH server for clients
  - Stopping rogue DNS server usage is *much* harder
  - Applications may ignore system DNS settings



### DNS over HTTPS

Another incredibly important source of data required for SOC success is DNS. Unfortunately, the DoH standard is already here and even in use by default in some cases! DoH is effectively taking a normal DNS packet and placing it in the body of an HTTP POST request and sending it to a webserver using HTTP/2 with TLS1.2+. What does that mean? It means that *your DNS traffic now becomes nearly indistinguishable from your web traffic*. That's not good – one of the tenets of good security is blocking all DNS servers that aren't controlled by your organization, lest malware can perform DNS tunneling and other nefarious activity unimpeded. With the oncoming switch to DoH, your team will not only struggle to identify DNS queries, but even logging legitimate DNS traffic will require more work, as you will need to run your own DoH compatible server, or intercept and decrypt the DoH traffic that is being created. Not only that, but certain applications like Firefox are also happy to use DoH by default and bypass your Windows DNS settings, going instead directly to services like Cloudflare! As you can imagine, DoH has had a contentious history for reasons like this, but none of that matters now, the standard is here, and you need to understand what it means for your visibility.

## Recognizing DNS Using DoH

Is this DNS? HTTPS? Something else?

1	0.000000...	192.168.42.129	104.16.248.249	TLSV1.2	110	Application Data
2	0.000151...	192.168.42.129	104.16.248.249	TLSV1.2	136	Application Data
3	0.000287...	104.16.248.249	192.168.42.129	TCP	60	443 → 36880 [ACK]
4	0.000494...	104.16.248.249	192.168.42.129	TCP	60	443 → 36880 [ACK]
5	0.064105...	104.16.248.249	192.168.42.129	TLSV1.2	261	Application Data,
6	0.064196...	192.168.42.129	104.16.248.249	TCP	54	36880 → 443 [ACK]

When decrypted, yes...it's DoH

Source	Destination	Protocol	Length	Info
192.168.42.129	104.16.249.249	HTTP2	114	HEADERS[73]: POST /dns-query
192.168.42.129	104.16.249.249	DoH	155	Standard query 0x0000 A www.supermaliciousmalwaresite.com OPT
104.16.249.249	192.168.42.129	DoH	315	Standard query response 0x0000 No such name A www.supermaliciousmalwaresite.com



## Recognizing DNS Using DoH

This slide demonstrates the issue with trying to collect information on DNS queries when DoH is in use. The picture on the top half of the slide shows what Wireshark will see when observing DNS traffic using DoH *without* decryption being used. As you can see, there is no indication that you are looking at DoH or DNS traffic at all. The only clue you *might* be able to find (in this example, but not all) is the destination IP address.

104.16.248.249 happens to be a Cloudflare (the default Firefox DoH provider) IP address, so assuming (and this is an important assumption) the person using DoH in your network is using a well-known DoH provider, you could spot the attempt based on the IP address. But what if this was an advanced attacker that set up their own DoH server, resolving DNS from an IP address you couldn't find any information about? In that case, *you will not be able to distinguish DNS traffic from HTTPS traffic unless you are intercepting and decrypting the TLS connection to see that it is DNS inside*. In other words, without interception you would 1. Not know this was a DNS request, 2. Not have any way to see or log what domain name was being looked up.

Look now at the second picture on the slide above, if you *are* decrypting the traffic, you can see that this connection was indeed DNS. Wireshark identifies both HTTP2 and DoH protocol in use. The traffic was a POST request (because DoH does use normal HTTP remember) to Cloudflare's DoH server, querying for the website [www.supermaliciousmalwaresite.com](http://www.supermaliciousmalwaresite.com), which was completely hidden in the encrypted example above.

Yes, this may be diving deep for a management course, but understanding the ramifications of DoH are incredibly important for your future success as a SOC. To test for any "normal" uses of DoH on your network, search for port 443 traffic to well-known DNS server IP addresses like 1.1.1.1, 8.8.8.8, and 9.9.9.9, you may be surprised what you find. Windows, macOS, iOS, and more already support or have expressed their intent to soon support DoH for lookups (Android currently uses DoT – easier to block as TCP port 853), so the day where you become largely blinded to your DNS traffic is coming soon, make a plan now!

## HTTP2.0/3.0

HTTP content is the same, but the way it's sent is VERY different from before

- Nearly guaranteed usage of TLS for transport means interception is required
- Multiplexed data in one stream makes protocol analysis more difficult
- Server push means one request => many responses
- Headers are now binary and harder to view

Takeaways:

1. Interception is required to even view the protocol
2. Analysts will struggle to understand what's happening compared to HTTP/1
3. Available tools are not yet mature enough to perform all required tasks for analysis
4. Your tools might not even be capable of understanding the protocol yet

### HTTP/2 and HTTP/3

HTTP/2 and HTTP/3 are here and will further complicate data analysis for SOCs. While HTTP/2 and HTTP/3 carry files back and forth the same way the previous versions did, the way they represent that data has drastically changed (largely for performance improvement reasons). As of early 2021, tools have not yet caught up with the features they offer for analysis of the previous versions of HTTP (Wireshark cannot carve files out of HTTP/2 automatically for example and makes reading headers much more labor intense). Therefore, analysis of any malicious activity that occurs over these protocols may be 1. Impossible to see due to the nearly guaranteed usage of encryption, and 2. Slower or much more difficult for analysts due to the complex nature of the new protocols, or 3. Impossible if the SOCs current toolset does not support dissection of the protocol in a thorough way.

## Another Challenge: NSM in the Cloud

- Collecting network data on prem was easy
- Cloud collection options are new and less developed
- **They ARE still available, do NOT just give up**

Cloud network visibility feature names:

Data Type	Azure	AWS	GCP
<b>Flow Logs</b>	NSG Flow Logs	VPC Flow Logs	VPC Flow Logs (sampled only)
<b>Full Traffic Capture</b>	Virtual Network Taps	Traffic Mirroring	Packet Mirroring

## Another Challenge: NSM In the Cloud

While monitoring of traffic may have been difficult or impossible during the initial years of cloud deployed infrastructure, fortunately that is no longer the case. In all major platforms there are now at least partial solutions, if not solutions that are nearly identical to what you can do in your own network on-premise.

Most SOCs would consider their bases covered if they have at least flow log-level visibility for their cloud assets, and full traffic capture available for any specific instance or interface they are interested in. This level of visibility is available from Azure, AWS, and the Google Cloud Platform with varying levels of nuance. AWS seems to have the most feature complete offerings as of 2021, giving uses the ability to see all flow logs within any VPC, subnet, or interface, as well as mirror traffic from any interface to any other source (IDS, network metadata sensors, and more). Azure is nearly there as well, with only some incoming traffic mirroring capability (with potential restrictions on where traffic can be mirrored to), and GCP offers both mirror and flow logs, however flow logs are sampled at a roughly 1 in 10 rate. The moral of the story here is that you can likely get 99% of what you need, if not 100% from cloud-based network traffic, so do not give up your monitoring efforts, even for in-cloud assets!

## SOC Data Collection Summary (I)

- Data collection is not a yes/no – it is a multi-stage pipeline requiring careful planning
- Policies, costs, and detection goals determine what gets written and what gets collected
- Fundamental collection concepts:
  - Defining goals
  - What to collect
  - How to collect logs and data
  - Key data sources
  - Collection efficiency and cost and practical issues

### SOC Data Collection Summary (I)

You have probably heard the phrase, “garbage in/garbage out” to describe many computing concepts; this saying most certainly applies to SOC data collection. This requires clear goals, careful planning, and diligent ongoing management of data collection and processing. Your data pipeline begins with the right tools to capture and transmit various data sources in a way that is machine-readable, flexible, and easily enriched to maximize its value to your analyst team. We have talked about how we define what data is written and what is collected, which types of data to collect at a minimum, how to get that data into a system where it can be enriched, indexed, and viewed, and some of the practical challenges in collecting newer, privacy-oriented protocols. As SOC manager, you must be a good steward of your organization’s data and the investments made to collect and store it – this requires a solid understanding of the fundamentals we have covered in this section.

# Course Roadmap

- Book 1: SOC Design and Operational Planning
- *Book 2: SOC Telemetry and Analysis*
- Book 3: Attack Detection, Threat Hunting, and Triage
- Book 4: Incident Response
- Book 5: Metrics, Automation, and Continuous Improvement

## SECTION I

### Introduction

#### Mindset and Preparation

- Cyber Defense Theory and Mental Models
- SOC Data Collection
- **Other Monitoring Use Cases**

- *Exercise 2.1: Attack Path and Data Source Assessment*

#### Collection and Monitoring

- Using MITRE ATT&CK to Plan Collection
  - *Exercise 2.2: Prioritizing and Visualizing Attack Trees*
- Cyber Threat Intelligence
  - *Exercise 2.3: Writing Priority Intelligence Requirements*
- Practical Collection Concerns
- Prevention and the Future of Security
- Summary and Cyber42 Day 2



This page intentionally left blank.

## Other Monitoring Use Cases

- Network and endpoint data will take us far in the SOC
- But some use cases will require further planning
  - Additional telemetry, specialized tools, and collaboration with other teams to maintain adequate monitoring coverage
- Some unique use cases require a more deliberate, customized approach
  - **DevOps**
  - **Supply chain**
  - **Business e-mail compromise**
  - **Insider threat**

### Other Monitoring Use Cases

Attacks at the network and host layers – even in virtualized environments like the cloud – are far from the only detection use cases we must address in a modern enterprise. SOCs are increasingly responsible for a wide range of threat scenarios, including attacks targeting the application layer, supply chain, and trusted insiders. Many of these “other” use cases will leverage the same kinds of log data and analysis techniques we discussed in the last section, but we cannot simply bold on traditional security monitoring practices and assume they’re provide adequate threat coverage. In this section, we’ll discuss security monitoring in a DevOps environment, supply chain instrumentation, and insider threat as additional use cases where further planning and cooperation with other teams may be required.

## When You've Missed Something

All-too-common scenario:

- A new system shows up in your telemetry with alarming administrative activities, web traffic, interaction with other systems
- After tracking down the system owner, you discover a brand-new Internet-facing application
- More digging reveals the application runs a great deal of custom code written by your in-house dev team and vulnerable open-source software!



### When You've Missed Something

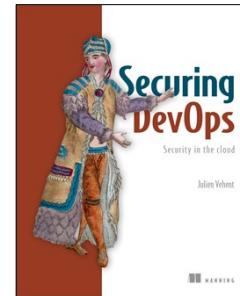
How many of you have found yourselves in this situation or one like it? It can be easy to ignore DevOps in our monitoring and response processes, but we do so at our peril. In 2017, Uber developers failed to properly secure credentials on a GitHub site, resulting in a compromise of an Amazon AWS instance where the personal details of 57 million customers and 600,000 drivers were stored. The company ultimately had to pay a \$100,000 ransom and untold incident response fees in order to recover from this breach – costs that may seem quaint by today's standards, but a significant breach, nonetheless. An effective SOC instruments the development process as one might instrument a network or an endpoint, identifying not only new systems and applications before they are deployed but also identifying common errors made throughout the process, like the insecure credentials that led to the Uber breach.

Some organizations use the term "DevSecOps" to describe a process in which DevOps incorporates various security functions at each phase of the development lifecycle. We won't split hairs about whether or not DevOps should include these functions natively (negating the need for a special "secure" flavor of it), nor will this course dive deeply into DevOps security best practices. But from a monitoring and response perspective, it is vitally important that we understand how DevOps can provide inputs to our monitoring and response processes and ways we can overcome organizational barriers to be active participants in the DevOps process.

The Uber breach is an interesting story not only for the DevOps/security context, but for how Uber handled incident response communications (which we'll cover later in this class). You can read more about that breach, and the fallout from it, here: <https://www.bloomberg.com/news/articles/2017-11-21/uber-concealed-cyberattack-that-exposed-57-million-people-s-data>

## What Is DevOps?

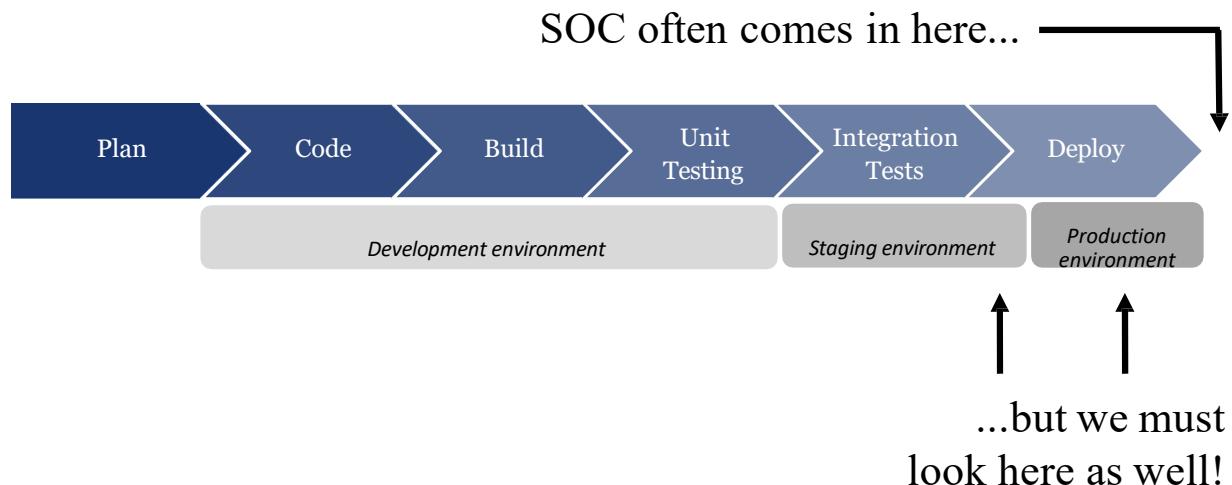
- Brings together two historically siloed teams: development and infrastructure
- Focus on continuous improvement through rapid releases, automation, and close collaboration
- Common components are:
  - Continuous Integration (CI)
  - Continuous Delivery (CD)
  - Infrastructure-as-a-Service (IaaS)
- Building security into this process can help us address issues earlier and faster (DevSecOps)



### What Is DevOps?

DevOps can be tricky to define as its implementation sometimes varies by organization. But at its core, it's an approach that brings together development and infrastructure support functions in a highly automated, continuous delivery model where applications and services can be delivered and updated rapidly. A secure DevOps approach is one that builds testing and continuous monitoring in at every step. Doing so not only reduces the likelihood of failures and downtime due to attack, but also helps the blue team maintain situational awareness. If you aren't familiar with DevOps or are looking for a crash course in securing a development pipeline, check out the excellent [Securing DevOps](#) by Julien Vehent, an Engineering Manager at Google and formerly the Firefox Operations Security Team Lead at Mozilla.

## Basic Deployment Pipeline

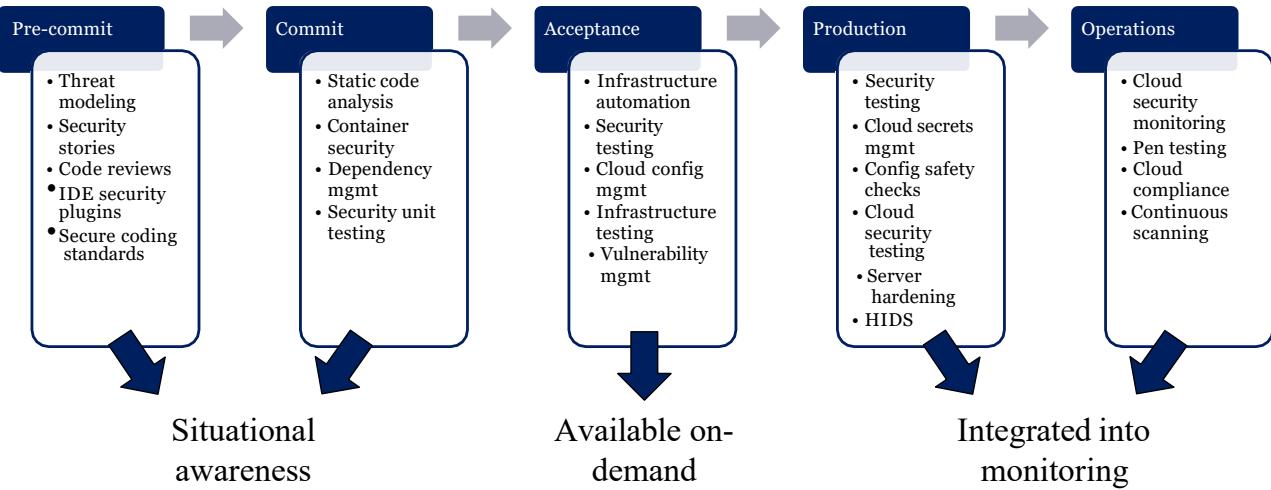


### Basic Deployment Pipeline

For those of you not familiar with development aligned to agile principles, this is what the process generally looks like: developers plan, write, compile, and test their code using a variety of different tools and varying levels of automation. In a DevOps shop, these activities normally move from a development environment to a test environment to pre-production (or staging) and then to production. Some of these activities may use fake data or simulated production systems to see how the application will behave after it has been deployed. The CI/CD pipeline and IaaS infrastructure underpin these environments and can be built from a variety of different tools, servers, and cloud infrastructure either on-premises or externally hosted.

In the SOC, we're often used to systems entering our area of responsibility only after they've been deployed to production. However, if we want to avoid the common scenario we discussed a few moments ago, we should be collecting telemetry from much earlier in the deployment pipeline. This telemetry can help us identify vulnerabilities (and, by extension, how much risk we're taking on with the deployment of a new system), assist in an investigation (understanding how a targeted application has been built and components it uses), or scope an incident (for example, visibility into trust relationships and communications within an application). Much like the network and host data we collect in production, we don't need all of it for alerting purposes; in the next slide, we'll look at the different kinds of data we can collect and how we might use it.

## Monitoring a Dev Environment and CI/CD Pipeline



### Monitoring DevOps

SANS' SEC540 class describes five phases in a secure DevOps toolchain that we can reference in implementing security controls and monitoring "further left" of deployment:

1. **Pre-commit:** these are the planning and development activities that occur before the code is checked into a version control system
2. **Commit:** these are checks performed during the build and continuous integration steps
3. **Acceptance:** functional testing and security scanning is normally performed at this stage
4. **Production:** in this phase, engineers conduct security checks before, during, and after code is deployed to production
5. **Operations:** ongoing security monitoring and auditing following deployment of the new code into production

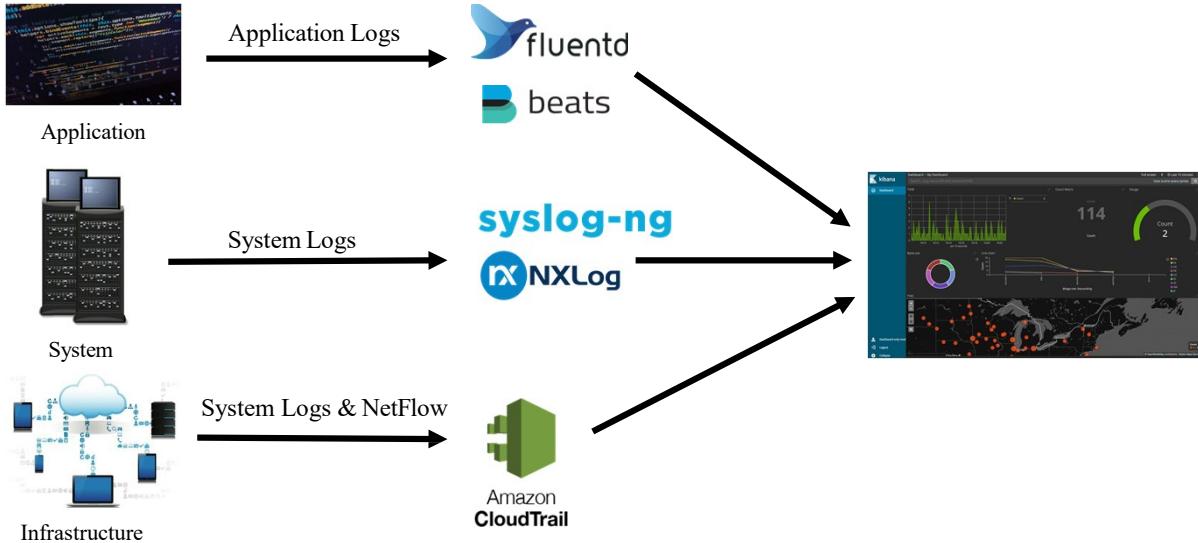
This graphic illustrates some of the controls and activities conducted at each phase to build security into the process. Obviously, our job in the SOC is not to manage all of these activities; rather, we must be aware of what's happening at each phase, security controls in place, and telemetry we might collect. Ideally, as shown in the graphic, the SOC monitors the outputs of security testing at the Production and Operations phases, has some access to testing results from the Acceptance phase, and is aware of controls in place at the Pre-commit and Commit phases for planning and investigation purposes.

All of these in more are described in the SANS Cloud Security and DevSecOps Best Practices poster, which can be downloaded for free from <https://www.sans.org/security-resources/posters/cloud-security-devsecops-practices>.

Of course, every deployment pipeline may look a little bit different. In the next slide, we'll focus on the kind of telemetry we're looking for regardless of what the deployment pipeline looks like, or the development process being followed in your organization.

## DevOps Telemetry

Test, Staging, & Production



SANS

MGT551 | Building and Leading Security Operations Centers

49

### DevOps Telemetry

If you've spent most of your career in the networking or systems world, a modern development process may sound like a foreign language. But at its heart, the deployment pipeline is really just a set of systems and software that can be instrumented like any other infrastructure. Just remember that your development environment is likely made up of a few different networks that include test and staging in addition to production, and each one hosts a subset of systems and applications used to write, test, and deploy code. All of these assets produce telemetry at the application, system, and infrastructure layers and can be monitored using many of the same tools you use for the rest of your environment. Applications write logs to their own custom channels or the system log locations where they can be collected using agents like Elastic's Beats or fluentd. Log collection utilities like syslog, syslog-*ng*, or NXLog in Windows environments can be used to instrument the underlying servers (systems) that run various development apps. Finally, logs and network data can be offloaded from cloud development infrastructures by custom-built services like Amazons CloudTrail or Google Cloud Logging and, depending on your provider, VPC flow logging as we discussed in the previous section. For third party applications like GitHub, you may have to utilize webhooks or APIs to collect audit logs.

## Choosing Application Events to Log

- Structured data is ideal
- Include key fields for investigations: timestamp, hostname, process ID, etc.
- What does bad look like in the application?
  - What content would indicate that?
- OWASP provides useful guidance for application logging



### Choosing Application Events to Log

Development teams are busy and deadline-oriented, so they won't have unlimited time to work with you on instrumenting the development pipeline. We'll talk about how you can build strong relationships with your development teams in a moment, but early on the most important thing is to provide reasonable, specific requirements for the data you need from them. This starts with standard contents and formats – think structured log formats like XML, JSON, or CSV that you can easily parse – and continues with key fields like timestamp, hostname, process ID, etc. Think about what bad might look like in the application telemetry you're going to get, and ensure that whatever data might indicate those activities is included in the logs. When it comes to exactly *what* should be logged, OWASP has some fantastic resources to guide discussions:

- The OWASP logging cheat sheet ([https://cheatsheetseries.owasp.org/cheatsheets/Logging\\_Cheat\\_Sheet.html](https://cheatsheetseries.owasp.org/cheatsheets/Logging_Cheat_Sheet.html)) is a list of events that should be recorded by an application
- OWASP's AppSensor project (<http://www.appspot.org/>) provides a useful framework that can be referenced for detection and analysis use cases in an application environment

## Getting (and Keeping) a Seat at the Table (I)

CALMS framework by Jez Humble:

- Culture (of "yes")
- Automation
- Lean
- Measurement
- Sharing



### Getting (and Keeping) a Seat at the Table (1)

Jez Humble, co-author of *The DevOps Handbook* and *Accelerate*, created the CALM framework to assess whether or not an organization is ready to accept DevOps processes and measure their progress following adoption. It is based on these five pillars: Culture, Automation, Lean, Measurement, and Sharing. In the SOC, we can use CALMS to build better working relationships with development teams in our own organization.

One of the fastest ways to break down organizational barriers that often isolate the security team from the rest of the organization is to build a "culture of yes," where security helps its constituents achieve their goals while minimizing risk. The security team must engage with development teams to understand their goals and effectively communicate security requirements without adding a lot of additional work or constraints to teams not designed to address them – keeping with the lean principles of the DevOps approach. Once these relationships have been established, the security team has to keep that communication going, looking for ways to improve and ask questions – especially when failures occur. This collaboration is bolstered by learning, in which the security team educates developers on relevant operational risks and seeks to better understand development tools and processes. Hard data and metrics collected in the course of the SOC's monitoring and response efforts can be of critical importance in demonstrating the need for ongoing collaboration between the teams.

Now you may be thinking, this all sounds great – but how do I make it happen?

## Getting (and Keeping) a Seat at the Table (2)

Injecting security into DevOps:

1. Provide training
2. Define requirements
3. Define metrics and compliance reporting
4. Use Software Composition Analysis (SCA) and governance
5. Perform threat modeling
6. Use tools and automation
7. Protect credentials
8. Use continuous learning and monitoring



### Getting (and Keeping) a Seat at the Table (2)

Microsoft has shared the following best practices<sup>1,2</sup> for injecting security into the DevOps process:

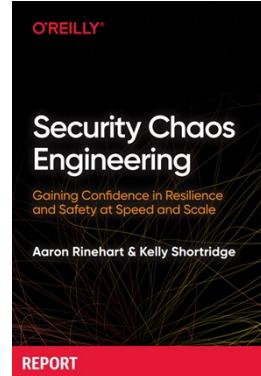
1. Provide training that covers common attacker TTPs and how they might target vulnerable applications and development infrastructure.
2. Establish a minimum- security baseline for controls and telemetry at each phase of the DevOps process, referencing a common framework such as the OWASP Top 10 or SANS Top 25 controls.
3. Define specific metrics that incentivize the right security behaviors and enable improvement over time.
4. Work to understand the impact of external, third party, and/or open-source components on the risk level of internal applications. Conduct security assessments (or consume the output of those assessments) of those components to incorporate them into your metrics.
5. Provide inputs to threat modeling exercises that align with the organizational threat model and known attacker TTPs.
6. Use or recommend tools that can be integrated into the continuous delivery pipeline, or offer access to security tools that create value for the development and infrastructure teams (such as vulnerability management or log management platforms)
7. Adhere to security best practices for protecting credentials as this is one of the most common attack vectors in the DevOps pipeline
8. Use the metrics you have established and telemetry you collect to continuously improve and refine your efforts, much as you would security monitoring and incident response for the enterprise!

1 <https://www.microsoft.com/en-us/securityengineering/devsecops#Training>

2 <https://www.microsoft.com/en-us/securityengineering/devsecops>

## Embracing Chaos

- Difficult to conceptualize modern enterprise applications
- Failure is as inevitable as compromise
- A resilient system is one that can absorb attacks, evolve along with threats
- Chaos engineering = testing to observe impacts of control, system, and response failures



### Embracing Chaos

This brief primer on DevOps and enterprise software development is not intended to make you an expert. Even if you already have some engineering knowledge, it can be tough to conceptualize all of the functions and components that make up modern application architectures.

*Security Chaos Engineering*, or SCE, is a set of guiding principles that can help us stay focused on high-level goals without getting bogged down in low-priority issues or technical minutiae. SCE essentially combines a focus on resiliency with a modern defensive mindset via two main ideas: (1) security controls will fail, and (2) we must be ready to quickly and effectively respond to incidents when they occur. From a systems perspective, this means that we must design security controls that fail gracefully without impacting uptime or users (or minimally so), and that enable us to identify and respond quickly to remediate the issue. For example, instead of relying on DDoS controls that will throttle system access in the event of an attack, we might implement a content delivery network (CDN) solution that will reduce end-user impact while providing a high degree of visibility and control.

SCE can also be a great reference for threat modeling and incident retrospectives, which in turn can produce valuable inputs to security planning. Let's talk more about planning and data sources now in more detail. If you want to learn more about SCE, check out the fantastic (and free) e-book [Security Chaos Engineering](#) by Aaron Rinehart and Kelly Shortridge.

## Supply Chain Risk and the SOC

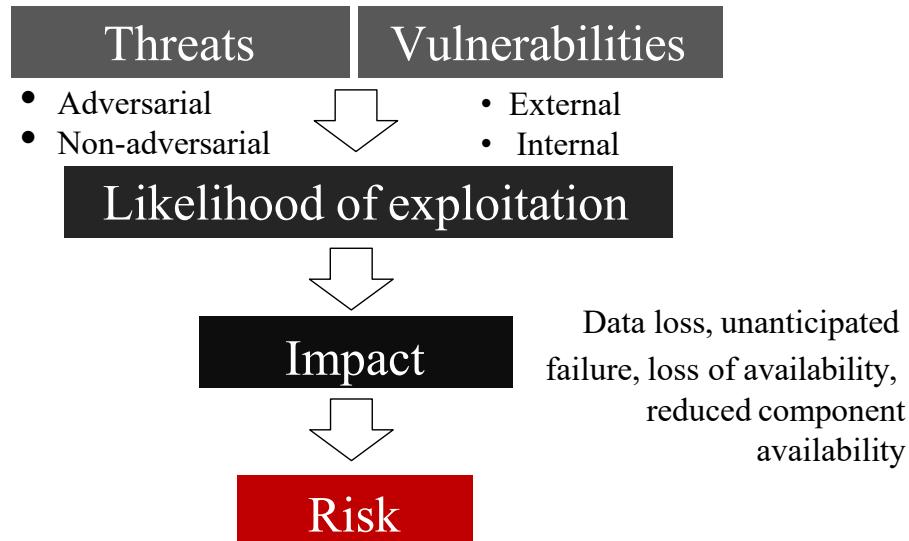
- Supply chain risk management is a huge challenge, much of which won't be addressed in the SOC; covers things like:
  - Security practices of third-party service providers or vendors
  - Compromised or vulnerable software or hardware purchased from suppliers
  - Third party data storage or data aggregators
- Likely represents a significant portion of your attack surface so we can't overlook it



### Supply Chain Risk and the SOC

Supply chain cyber risk can be an extremely large challenge to tackle, and it's a discipline most organizations struggle with if they are even addressing it at all. This area covers things like sourcing, vendor management, supply chain continuity and quality, transportation security, and many other functions that we rarely see, much less affect, in the SOC. That said, modern enterprises rely heavily on various kinds of third parties to deliver IT services, so we cannot overlook this large (and probably ever-growing) portion of our attack surface.

## SCRM Process

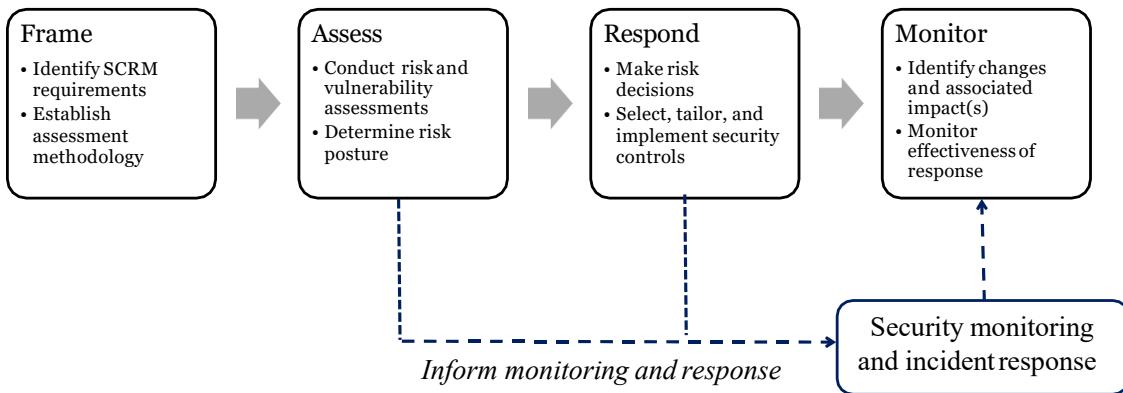


### SCRM Process

According to NIST Special Publication 800-161, risk in the supply chain comes from insertion of counterfeits, unauthorized production, tampering, theft, insertion of malicious software and hardware (e.g., GPS tracking devices, computer chips, etc.), and poor manufacturing and development practices. This graphic, based on the NIST publication, illustrates how risk is calculated based on the likelihood that various events and conditions (threats and vulnerabilities) will materialize in the environment. The SOC probably won't have sufficient awareness of all of the non-adversarial threats, or all of the weaknesses introduced into our environment by internal and external service providers or components, nor would we want to spend our time trying to develop that level of awareness. But we do need to focus on developing our telemetry to identify the impacts of those risks and anticipate impacts wherever possible. It's also important to understand how supply chain risk is introduced into the environment so we know where to prioritize our monitoring and response efforts.

You can download SP 800-161 from NIST  
here: <https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-161.pdf>

## SCRM Process (cont'd.)



### SCRM Process (cont'd.)

At a more tactical level, NIST breaks down the supply chain risk management process into the following steps:

1. Frame
2. Assess
3. Respond
4. Monitor

Unlike some of the other processes we have discussed, this is a risk management process- not a network defense one. In this context, "respond" means accepting, transferring, or reducing risk by deploying security controls or taking some other management action. Our job in the SOC is to support the monitoring step: watching for signs that software, hardware, data, or services provided by third parties have changed and ascertaining the impact of those changes. This may be relatively straightforward when it comes to things like Windows updates that include critical security patches, or new software deployed that has known vulnerabilities. But what about new server hardware deployed in the environment? New cloud infrastructure? Mobile devices? Maintaining awareness of these changes is a cross-functional effort that will require inputs from procurement, network infrastructure, application support teams, the service desk, and more. Even when we are aware of these changes, we may not always have the detail we'd like to confidently identify necessary changes in our monitoring and response procedures, or areas of higher risk that may require additional attention. As a security leader, it is your job to provide the SOC's inputs to the supply chain risk management process in the form of known risk (components we know are vulnerable and/or likely to be targeted), gaps in our visibility, and recommendations for improvement.

## The Role of the SOC in SCRM

- Study security policies, compliance requirements, and acquisition process
- Support thorough risk assessments
- Minimize and monitor vendor access
- Where possible, append vendor identifiable information to threat intelligence
- Suspect everything and everyone (kidding, sort of)

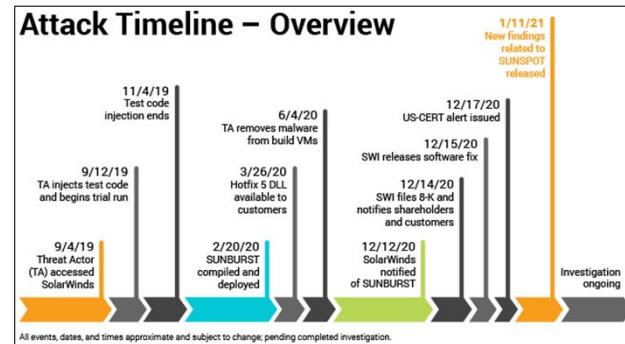


### The Role of the SOC in SCRM

Again, while we might not have much control over supply chain activities and the risk they introduce into our environment, the SOC should play a key role in supply chain risk management. This is a great opportunity for you to reach out to your acquisitions and legal team(s) to better understand security policies, compliance requirements, and other factors that influence how goods and services are brought into your environment (and any security-specific requirements your organization levies on vendors and partners). Advocate for thorough risk assessments of service and application providers and, at a minimum, vulnerability assessments for software and hardware. Of course, we're (hopefully) already aware of internal policies governing approved software, and a risk or vulnerability assessment should be a pre-requisite for any vendor looking to get onto that list. From an operational perspective, we want to minimize and monitor vendor access – many of the most high-profile and damaging breaches of the last decade+ have involved network access via external partner, so we must identify and minimize those trust relationships at the network and identity layers. Whether your SOC is producing or consuming threat intelligence (or both), look for ways to append vendor information relevant to your environment to the threat intelligence you have. Finally, while we often joke about being paranoid in our line of work, from a supply chain perspective it certainly pays not to trust anything or anyone in your environment. This is another area where zero trust principles – removing location as a measure of trustworthiness – can come in handy.

## Case Study: SolarWinds Breach<sup>1</sup>

- Threat actors modified a plugin in Orion product that was distributed to SW customers via software updates
- Trojanized component is digitally signed, disguises C&C as legitimate activity, includes multiple analysis evasion techniques
- Utilizes memory-based dropper to run customized Cobalt Strike Beacon
- Evidence demonstrates long-term planning, testing, and iteration and great opsec



### Case Study: SolarWinds Breach

Where were you on December 13<sup>th</sup>, 2020, and how was your day? Most of us probably wish we didn't know the answer to that question. The reality for many of us is that day may have been filled with chaos and confusion due to the announcement of the unprecedented-in-scale SolarWinds Supply chain compromise. This compromise, one of the most devastating supply chain attack compromises in history, showed the world the power of a focused and well-resourced adversary breaking into a single organization that has reach to thousands of other organizations worldwide. For many of us, it put supply chain attacks on the map as "that's real and could happen to us". While it may have been the first experience for some of us in a breach of this type, it certainly won't be the last. Given the tightening of security controls and prevalence of application control technologies, zero trust architectures and more, don't be surprised as we undoubtedly continue to see more and more supply chain attacks going forward. The question is, what can do we do about them?

[1] Timeline graphic from SolarWinds, reposted in Brian Krebs' article: <https://krebsonsecurity.com/2021/01/solarwinds-what-hit-us-could-hit-others/>

## Case Study: SolarWinds Breach

- Thousands of customers potentially impacted – how would we have detected this?
  - Reality for many orgs: we couldn't have, at least directly...
- **Attackers aren't magic**, they still must use traditional attack tactics after the supply chain compromise
  - Just because they snuck in, doesn't mean they're finished
  - Minimize and monitor access, even for software
  - Monitor internal host to host activity
  - Look for new and unusual activity by software/user account

### Case Study: SolarWinds Breach

While many of us were no doubt firmly in the camp of "cannot detect a supply chain attack", that doesn't mean we had no chance at detecting the rest of the TTPs that were used after the initial intrusion. Just because an attack was able to enter your organization through a trusted program and vendor, doesn't mean the rest of their operation is magically and automatically complete. They still need to perform traditional attacker tactics like lateral movement, command and control, execution, persistence and more, all of which can individually be detected by a SOC that is looking in the right places. If nothing else, this underscores the continued need for a strong and tested defense-in-depth security strategy, because some of those intrusion stages may be nearly impossible to detect.

## Protecting High-Visibility Users

- Line is increasingly blurred between personal Internet use and "official" Internet presence
- High-viz users like executives, communications professionals, advocates, journalists, negotiators may be targeted as a means to attack the organization
- While monitoring may not be within the SOC scope, education and some personal attention is a great idea



### Protecting High-Visibility Users

In the age of social media, some of your users may be high-visibility individuals who speak and act (either officially or unofficially) on behalf of the organization. These users tend to attract a lot of attention from adversaries; fake social media profiles created to spread false information or exploit trusted relationships and targeted social engineering attacks are just two examples of how these users are often targeted today. Many of these methods will be entirely outside of your control in the SOC – they're happening in other networks on systems you do not own and may have little recourse when they are used against you. Even so, we cannot afford to ignore these threats to our users, partners, and constituents. The next slide will cover some ways we can identify, respond, and advise on better online safety for high-profile users.

## Preparation for At-Risk Users

### Topics to address with your users:

- Personal online hygiene: introduction to password managers, VPNs, and safe use of technology
- Social media privacy and security
- Anti-doxing resources (reducing your online “footprint”)
- Spotting suspicious messages

### Common scenarios to prep for:

- You receive an e-mail notifying you that your bank account has been compromised along with a link to sign in and change your password.
- A colleague to whom you have delegated approval authority receives an unexpected request for a funds transfer.
- Someone is posting inflammatory messages from a social media account impersonating you.
- You are receiving scam/sales calls at your home and an unusually large amount of unsolicited text messages.



### Preparation for At-Risk Users

Protecting high-visibility users usually starts with the right training and awareness, since the user is often the one with the most awareness and control when it comes to their Internet presence. Be sure to address good online hygiene, security settings for their social media accounts, and how to spot suspicious communications with them. Also make sure that they are aware of anti-doxing resources and how to report harassment and attack attempts on platforms they may use (or offer to do that for them when they identify suspicious activity). Note these common scenarios and use them to guide this education effort.

One organization that works hard to protect a very high-profile user base is the New York Times, who has published a guide to protecting yourself from doxxing and other forms of online targeting:

<https://open.nytimes.com/how-to-dox-yourself-on-the-internet-d2892b4c5954>

This Wired article on using data broker services to opt out of personal data sharing is also informative:

<https://www.wired.com/story/opt-out-data-broker-sites-privacy/>

## Business Email Compromise / "CEO Fraud"

- Worth dedicating time to studying
  - A potentially *enormously expensive* attack
  - **\$1.8B** of losses across **19k+ reports** to IC3<sup>1</sup> in 2020!
  - Initial ID theft / non-BEC scams led to BEC with fraud accounts
- Not purely a tech problem, but the SOC can help
- Let's take a specific look at:
  - The most common tactics from a technical perspective
  - How the SOC can prevent/detect against those TTPs
  - How the SOC can assist other groups in stopping BEC
  - Business process that can back up technical controls



### Business Email Compromise / "CEO Fraud"

Business email compromise (BEC), an attack involving the social engineering of employees within organizations to transfer large sums of money or private information directly to criminals is a large and still growing segment of cybercrime. Due to the outsized payouts and lack of need for technical sophistication, it's one of the easiest ways for attackers to steal large sums of money, which explains its popularity and continued growth as a segment of cybercrime. The 2019 IC3 (Internet Crime Complaint Center) Report says that nearly 50% of monetary losses reported were due to BEC attacks! That's an incredible number given the breadth of scams on the internet. For this reason, it's worth dedicating some time specifically to studying them, as you can be almost sure that your organization will run into an attempt at BEC at some point.

BEC is not purely a technical problem and cannot purely be solved by technical means, however there are a lot of places where the SOC can help their organization prevent, identify, and halt BEC compromises. Therefore, over the next few slides we'll dive into some specifics on the attacks, and how the SOC can assist.

[1] [https://www.ic3.gov/Media/PDF/AnnualReport/2020\\_IC3Report.pdf](https://www.ic3.gov/Media/PDF/AnnualReport/2020_IC3Report.pdf)

## Case Study: Operation WireWire - 2018

- Summary: Energy Company in Texas...
  - Attacker "Colvis" uses the **company website for reconnaissance**
  - Attacker creates **lookalike domain** name
  - CEO's assistant receives an email on **Friday at 3pm** with person detail included "*I'm going to be offline this weekend because I'll be coaching my daughter's soccer game on Sunday, so don't try to reach me then, just make sure this is done today.*" (Info from CEO's Facebook account)
  - Message contained a **PDF** with a **fake due invoice** from large supplier
  - Employee pays the attacker **\$3.2M** via **wire transfer**
- FBI takes on case as "Operation WireWire<sup>1</sup>", results:
  - 74 arrests, \$2.4M seized, \$14M recovered
  - "Colvis" was Nigerian national living in Texas, got a \$300K cut of scam after payouts

### Case Study: Operation WireWire – 2018

To demonstrate the relative lack of technical sophistication required for one of these attacks compared to a traditional compromise, let's take a look at a case study from an unnamed Texas energy company that fell victim to a BEC compromise.

The attack started when a man who had identified himself as "Colvis" canvassed the company's website looking for a person who might be able to initiate a wire transfer and finds a potential victim. After finding the name of the CEO's assistant, the attacker purchased a lookalike domain name and prepped it for email sending. At 3pm on a Friday, the attacker sends an email to the CEO's assistant with a fake invoice saying a large money was owed to one of the company's large suppliers, including that it must be paid that day, and that they would be out of contact due to personal obligations (the attacker had done research on the CEO's whereabouts via Facebook). The employee didn't spot the fake email address and paid the invoice, resulting in a wire transfer to the attacker of \$3.2M! A successful attack for a large sum of money with almost no technical depth required, other than purchase a domain name with the ability to send email.

After the incident, the company had contacted the FBI which took on the case, working with partner agencies and law enforcement around the world to find the attacker and the assisting criminal groups that helped "Colvis" pull it off. In the end, this investigation, dubbed Operation WireWire led to 74 arrests worldwide, \$2.4M seized, and \$14M of victim money recovered. The perpetrator was a Nigerian national - Amechi Colvis Amuegbunam, living in Texas on a student visa<sup>2</sup>, that had pulled the scam on other organizations as well (forensic evidence of the attacker's name left in the PDF was one of the big tipoffs). As an interesting side note, as a result of the energy company scam, Colvis took home only \$300K of the \$3.2M stolen after paying out for help from "a transnational criminal organization that employs lawyers, linguists, hackers, and social engineers."<sup>3</sup>

While this story has a relatively happy ending, it also shows that these scams happen quickly and easily, without much sophistication. Consider how this attack might have played it out had it happened to your organization, would it have worked?

1 <https://www.fbi.gov/news/stories/international-bec-takedown-061118>

2<https://www.justice.gov/usao-ndtx/pr/nigerian-man-sentenced-role-business-email-compromise-scheme-caused-37-million-loss-us>

3 <https://insights.sei.cmu.edu/blog/business-email-compromise-operation-wire-wire-and-new-attack-vectors/>

## Common Attack Stories

Attackers take widely varying approaches to BEC:

1. Fake invoices from supplier to alternate (attacker) account
  - "Supplier: We changed banks, please wire funds to this new account..."
2. Executive/attorney-initiated transfer request
  - "Your CFO: Please urgent transfer funds to \_\_\_\_\_"
  - "We're in the middle of an acquisition, we need this confidential transaction to occur today. Don't tell anyone, stock prices could be affected."
  - "Our attorney will soon be in touch with you with procedures to pay today"
3. Outside of process tax info and employee PII request
  - "We're undergoing a corporate audit, please send me W2's for all employees..."

### Common Attack Stories

There are many different ways to approach a BEC, but they tend to fall into a few general categories:

- Fake invoices – A vendor (real account or spoofed) submits an invoice to be paid with the attacker's bank account or wire information attached. Being a good organization that pays their bills, your company dutifully and happily transfers money directly to the attacker, not realizing that it's not a real request. An attacker can either intercept and cut into the middle of a discussion for what might be a payment already in process or create a completely new invoice for something that never happened.
- Urgent requests from executives and attorneys – Stories vary widely here, but the theme is someone high up in the organization needs money urgently transferred for some reason, and there is a reason to not speak about it with coworkers due to the sensitivity of knowledge of the payment
- Direct requests for employee PII and tax info - After almost completely disappearing in 2019, in early 2020, W2 scams for US employee tax information made a huge comeback due to coronavirus related anarchy experienced by most businesses in the beginning of the pandemic.<sup>[1]</sup> In these attacks, attackers simply ask for a copy of all employee W2 forms with some story as to why it is needed and use the info to file fake tax returns with employee information.

[1] <https://www.agari.com/email-security-blog/business-email-compromise-bec-w2-scams-2020/>

## BEC Tactics and Techniques

Attack Stage	Tactics, Techniques and Procedures
Recon	<ul style="list-style-type: none"><li>Social media (LinkedIn and more)</li><li>News and press releases</li><li>Website and other OSINT</li></ul>
Delivery	<ul style="list-style-type: none"><li>"From" field spoofing</li><li>"Reply-To" address is different from sender</li><li>IDNs and homoglyph attacks (lookalike domains)</li><li>Compromise user email account via malware/phishing</li><li>Hidden email forwarding rules</li></ul>
Credential Theft	<ul style="list-style-type: none"><li>Linked to credential harvesting via cloned webpages and more</li><li>Malware and RATs</li><li>Links to malicious documents hosted in "safe" places</li></ul>

SANS

MGT551 | Building and Leading Security Operations Centers

65

### BEC Tactics and Techniques

What makes the approaches on the previous page possible? A set of well-honed attack tactics and techniques that enable attacks to act as employees of your organization via email spoofing, act as vendors, or create convincing lookalike phishing pages to convince your own employees to surrender their usernames and passwords. Beyond this, of course, is the normal attack route of using malware and more for direct harvesting of passwords through infected victim systems, although this isn't usually necessary.

For the Recon stage, attackers start to profile who is in your organization that might have the information they need. Through a combination of OSINT gained from news sources, press releases, social media, data from your website, and more, it is highly likely they will start to narrow in on the right person to target in the next stage of their attack. While your company likely needs to release much of this information, it's not a bad idea to periodically check for oversharing – assume the role of an attacker and see how long it takes someone to go from zero knowledge to a set of people they might target for a BEC attempt.

Delivery is the key stage for most BEC attacks. Through multiple different techniques, attackers are often able to land an email in the inbox of a victim that can be very difficult to spot as fraudulent. If anti-spoofing checks and detections aren't in place, it's often easy for attackers to produce email that appears to have come from an internal source, even though the true sender is somewhere else out on the internet. Even if these attempts are stopped, there's always the much more difficult to catch option of compromising a real user account and intercepting / sending email as the compromised user, which of course will have no technical indicators of being spoofed. This is where technical controls fail and business process and user awareness must take over.

Stopping credential theft for BEC is largely the same as stopping or detecting credential theft from any attacker. There are number of ways for attackers to proceed with this tactic – phishing, malware, and more. A comprehensive strategy that covers detection of phishing and cloned webpages, as well as email link rewriting, downloaded file inspection, and application control is needed to prevent compromise in the first place. Paired with a strong detection strategy for questionable logins and 2 factor authentication, even if passwords are stolen, your team will hopefully be in a position to spot the use of stolen credentials before attackers can use them.

## SOC-Focused Technical Controls

- Email-based Protection
  - **Disabling of email forwarding**
  - **Detection of suspicious mailbox rules**
  - **Prevention of spoofed email** (SPF, DKIM, DMARC)
  - Detection of **lookalike domain** creation and usage
  - Advanced email **scanning** and threat protection
- Credential Theft Prevention and Identity Protection
  - Check for anomalous logins – nature of login, IP geolocation, etc.
  - 2 factor authentication for accounts to disrupt password theft
  - Detection of phishing pages visited by employees
  - DMARC – Can receive reports of others spoofing email acting as *your* org.
  - Malware prevention that can lead to credential compromise



### SOC-Focused, Technical Controls

Considering the methods for accomplishing BEC often involve a certain subset of tactics, as a SOC, you can help out by focusing controls and detection techniques at those most used by BEC scammers. Remember the phrase "prevention is ideal, detection is a must" here. Our goal is to use all available controls to outright disrupt any attacks that can be positively identified and blocked. When that is not possible, ensure the SOCs attention is at least drawn for additional scrutiny to any cases that are too difficult to call with purely technical and automated means. We can break this into two main categories - keeping a very close watch on the details of emails being delivered, as well as keeping your own employee's credentials safe from theft.

On the email front, focusing on closing attack vectors commonly used by attackers can make a big difference.

- Can you prevent, or at least detect the presence of newly created and suspicious mailbox rules? (auto-forwarding to attacker)
- Will your environment deliver an email with a spoofed or non-aligned from address?
- What about if the from address of an email is "from" your company, but the reply-to address would lead a response to an outside domain?
- Have you ever tried making a lookalike domain for your organization and sending email from it to see if any of your technical controls will identify it?
- Do you have in-depth email scanning and threat protection from your email vendor?

There are plenty of avenues available here for prevention and detection that are low-hanging fruit, but often go un-tested by security teams.

Secondarily, how easy might it be for an attacker to steal credentials from your own employees and leverage them for an attack against other employees or another organization? If detecting spoofed emails is hard enough, imagine how hard it would be to stop a BEC compromise attempt launched from a real employee account sending an email to someone else in your organization! For these reasons, preventing credential theft, identifying successful credential theft, and making it difficult to use stolen credentials is another major piece to consider for your BEC prevention strategy. This includes things like monitoring for anomalous logins, 2 factor authentication for employee accounts on AT LEAST all publicly available services, technical controls to spot phishing pages and webpage cloning, and solid malware prevention to prevent credential compromise in the first place. There are also standards like SPF, DKIM, and DMARC, that can prevent spoofed email delivery *to* your org, as well as enable you to receive reports from other orgs on email they have received, claiming to be from your org, that may have been spoofed.<sup>1</sup>

[1] <https://seanthegeek.net/459/demystifying-dmrc/>

## Testing Against BEC Tactics

Testing yourself before attackers do!



- **Red team your email protections**

- Test *all* methods used by BEC scammers for spoofing
- Verify your ability to detect suspicious inbox rules
- Try sending various types of weaponized docs
- Phishing simulations utilizing common stories

- **Red team common phishing and credential theft tactics**

- Lookalike domain creation and traffic detection
- Cloned paged testing
- Train SOC analysts to assess email headers
- Test fake "stolen" credentials and see what telemetry can be used to find it

## Testing Against BEC Attacks

As will be discussed in more detail later in the course, you can never be sure that your controls and detection analytics will function until you have solid proof from a test, and even then, it's worth keeping a healthy skepticism. In the terminology we will eventually introduce later, testing against BEC would consist of designing a campaign of BEC-style tactics, and exhaustively rolling through each one in multiple forms, ensuring that each is either stopped, or at least generates an alert. For email, this would mean testing every method of spoofing, every commonly weaponized document format, and even running phishing simulations of the same nature you might expect to see in a BEC compromise. In addition, testing your ability to stop and detect credential theft techniques can back up the phishing tests by verifying users will be blocked or detected when visiting phishing pages and cloned websites. You can even test your SOC analysts with saved examples of BEC attempts from the past to ensure the evidence of spoofing is well understood and can be easily spotted in the future.

## What Else Can Be Done?

Non-SOC strategies to assist in BEC prevention:

Tactic	Response
Realistic stories and scenarios	User awareness training for typical stories, with examples of actual attempts and how to spot them
"Outside of process" requests for information	Controls on "outside of process" requests, expectations that this will not occur or must involve an in-person conversation
Requests for delivery of sensitive info via non-standard means	Data Loss Prevention and other policies about how personal data can be transferred, and standards for data protection
"We changed bank accounts"	Changed accounts must be verified <u>out-of-band</u>
Fake invoices / new accounts	Strict procedure for verifying first-time supplier payments



### What Else Can Be Done?

Considering the typical set of stories that are told in a BEC attempt email, the answer to interrupting what does make it through technical controls is a tight set of business policies and procedures. Creating a set of expectations of what can be asked for, how it can be provided, and how to verify the request makes BEC much less likely to occur. The slide above shows some examples on how to counter each of the techniques typically used for BEC. Largely, business process-based prevention revolves around ensuring unlikely stories are immediately spotted and processes for gathering sensitive PII information and sending it out cannot be simply worked around as a "special exception", even if there's a desperate need, out of band verification of payments must *always* take place.

## BEC Summary

### The bigger picture

- **People, business process, and technology** must all align to stop BEC
- The SOC can contribute:
  - Guidance on TTPs for employee awareness
  - Assessment and awareness of the risk
  - Real examples of attempts for employee awareness/training
  - Technical controls to auto-prevent as much as possible
  - Detection of potential issues and warnings to email recipients
  - Up-to-date intelligence on the new tactics
- Don't forget to have a procedure ready for if it *does* happen



### How the SOC Can Help Others

The big picture here is that given the security training and focus of those in the SOC, we are in one of the best possible decisions to drive the understanding of the actual risk, tactics, and prevention methods for BEC in our organization. Considering the enormity of the potential impact and the extremely common frequency with which attacks occur, the SOC can and should make it clear to upper management and other stakeholders that this attack is a very real and high-risk danger.

Prevention of BEC attempts will fall purely upon the SOC, but the best information on how these attacks work, and how a combination of technical controls, business processes, and user awareness can all be leveraged to provide the best coverage possible. Do not forget the "assume you will eventually, fail" mentality here. Having a prevention and detection strategy is great, but it needs to be backed up with a ready-made procedure of who to call if a problem is discovered. The Internet Crime Complain Center (IC3) has been successful in helping victims retrieve money that was sent (\$300M recovered in 2019 alone<sup>1</sup>) by working with the FBI Recovery Asset Team which has streamlined communication with financial institutions. Once money has been transferred, time is of the absolute essence, and stalling just a couple extra hours or even minutes might be the difference of losing millions of dollars or not! Have that plan ready!!

*Note: Here are the steps suggested by IC3 for BEC victims<sup>2</sup>*

- *Contact your financial institution immediately upon discovering the fraudulent transfer.*
- *Request that your financial institution contact the corresponding financial institution where the fraudulent transfer was sent.*
- *Contact your local Federal Bureau of Investigation (FBI) office if the wire is recent. The FBI, working with the United States Department of Treasury Financial Crimes Enforcement Network, might be able to help return or freeze the funds.*
- *File a complaint, regardless of dollar loss, with [www.ic3.gov](http://www.ic3.gov) or, for BEC/EAC victims, [bec.ic3.gov](http://bec.ic3.gov)*

1 <https://www.fbi.gov/news/stories/2019-internet-crime-report-released-021120>

2 <https://www.ic3.gov/Media/Y2017/PSA170504>

## Addressing Insider Threat (I)

- Many definitions of "insider threat"
- DHS: "...the threat that an employee or a contractor will use his or her authorized access, wittingly or unwittingly, to do harm..."
- Focus is often just on purposefully malicious actors, which can cause confusion
- Consider scenarios on the next slide; insider threat or not?



*Watch out for users hiding in the shadows, dressed like spies.*

### Addressing Insider Threat (1)

Some organizations treat insider threat as its own class of attacker, and your management may ask what your team is doing to defend against them. While any adversary could become an “insider threat” at a certain stage of the attack, it may be helpful to respond to this requirement with a common definition of what constitutes an insider threat, and how it might be treated differently than external, untrusted adversaries. For example, DHS defines *insider threat* as actions taken by an employee, contractor, or other user with authorized access. Does this align with your definition? Or should it also include accidental exposure caused by user mistake or poor security practice? These answers will guide your strategy for deploying additional controls or focused monitoring. SANS Senior Instructor Lance Spitzner has written an excellent blog post on this topic entitled “Decoding: ‘Insider Threat’”, which you can check out here: <https://www.sans.org/blog/decoding-insider-threat/>. On the next slide, we’ll look at some common examples of insider threat Lance covers in his post and discuss whether or not we think they should be included in our planning.

## **Addressing Insider Threat (2)**

- A disgruntled employee sells sensitive information to a competitor.
- A recently fired IT Admin remotely connects back into their old company using their (still valid) credentials and purposefully infects the network with a virus, destroying numerous systems including backups.
- An employee is romanced by a spy and ends up unwittingly sharing sensitive information.
- An intern's work account is taken over due to the fact they used the same password for their work account as they do for their personal accounts, one of which was recently hacked.
- A contractor loses their (unencrypted) laptop which had gigabytes of confidential information.
- An employee emails a sensitive document to the wrong person due to auto-complete in email.
- A Human Resources employee falls victim to an opportunistic phishing attack.
- An attacker hacked into the company's website due to a SQL injection flaw that a software developer accidentally left in the website's code.

### **Addressing Insider Threat (2)**

These scenarios were included in Lance's blog post as examples of how the concept of insider threat can be a bit messy. While all of these scenarios are potentially damaging to the organization, some are clearly user error or negligence; should we classify these users as "threats" at all? For our purposes as defenders, it's probably most appropriate to focus on malicious actors. Under that definition, we can consider the first two scenarios in this list as examples of insider threat. What about the employee who sends the sensitive document to the wrong person accidentally, or the one who falls victim to a phishing attack, or the developer whose code was vulnerable to attack? Would these trusted users be considered threats? In these types of cases, we would normally consider these individuals to be either negligent or victims themselves. The bottom line here is that victims and negligence are normally addressed by compensating controls and user education – not enhanced monitoring of internal users. If we are working to make our network more defensible, we should \*in theory\* be able to identify most of the activities perpetrated by a malicious insider. But the first step is identifying what that means to you and to your management.

## Behavioral Indicators of Malicious Insiders

- Remote access while on vacation, sick or at odd times
- Unusual network traffic spikes, volume of USB/mobile storage use, volume of off-hour printing activities, inappropriate use of encryption
- Works odd hours without clear justification or authorization
- Data or network access outside of the scope of a user's duties or role
- Signs of vulnerability\*

### Behavioral Indicators of Malicious Insiders

Here are some examples of malicious insider activity based on US-CERT guidance<sup>1</sup>. Put on yourself in the position of a detection engineer for a moment; do you notice anything useful in these scenarios that might lend itself to automated detection? There are clear markers here that we can use for insider threat detection – for example, traffic spikes at unusual times or from unusual locations. The only scenario here for which we might lack good telemetry is the last one: signs that a user may be vulnerable to coercion or manipulation. In those circumstances, we can only monitor the best we can and hope to catch the resulting action.

[1] [https://us-cert.cisa.gov/sites/default/files/publications/Combating%20the%20Insider%20Threat\\_0.pdf](https://us-cert.cisa.gov/sites/default/files/publications/Combating%20the%20Insider%20Threat_0.pdf)

## Detect and Deter

Capabilities for insider threat prevention /detection:

- Data/file encryption
- Endpoint logging
- DLP and other data-centric solutions
- UEBA
- Endpoint requires the least of amount of baselining, but won't catch legitimate activities done with malicious intent
  - *Example:* e-mailing a sensitive file attachment

### Detect and Deter

So how can we detect or deter insider threat? The best controls are usually focused on data at rest, endpoint visibility, and user behavior monitoring. Data loss prevention and UEBA tools tend to be very expensive to deploy in terms of training, configuration, and ongoing management, so be sure that the protections they offer will be worth the investment versus a monitoring-centric approach. Endpoint protection tools tend to require less baselining and ongoing management than those purpose-built solutions but may not help you identify legitimate activities (say, e-mailing or saving off a sensitive file) done with malicious intent.

## Insider Threat Best Practices

- Consult HR and Legal on your insider threat program; many insider threat controls aren't technical:
  - NDAs
  - User training
  - Acceptable use policies
  - Exit/offboarding processes
  - Segregation of duties
  - Job rotation
  - Least privilege
- Malicious insider doesn't absolve you from following IR process and regulatory compliance!

### Insider Threat Best Practices

At this point, your biggest takeaway from our discussion of insider threat may be, “it depends”. The best course of action if you are planning an insider threat program (or had it assigned to you for action) is to consult your human resources and legal teams. They can advise you on how best to approach monitoring specific user activity and provide non-technical controls that can help you manage the risk. And remember, although an authorized user may be exhibiting risky or malicious behavior, you and your SOC team are still bound by your corporate incident response policies and whatever regulatory/privacy requirements govern your access to user data.

## Additional Monitoring Use Cases Summary

- Some (DevOps) have their own threat model and telemetry
- Some (supply chain, BEC, and insider threat) may rely heavily on administrative controls)
- Incorporate them into your knowledge base and strategy, even if you can't include them in your telemetry yet
- Can't discount them simply because they are difficult – the SOC will ***still*** be held responsible if you're attacked via these vectors!

### Additional Monitoring Use Cases Summary

In this section, we tackled some of the more challenging monitoring use cases in a corporate environment. Some of these use cases, such as DevOps, have entirely different tools and telemetry your team will rely on for detections. Some, like supply chain and insider threat, may have extensive administrative or process controls that will not be apparent in your telemetry. Regardless of your monitoring constraints, these use cases represent significant attack vectors and can't be discounted simply because they are complex or lie outside of your monitoring purview. Even if your plans to include these use cases are aspirational at the moment, you must incorporate them into your enterprise defense and planning.

# Course Roadmap

- Book 1: SOC Design and Operational Planning
- *Book 2: SOC Telemetry and Analysis*
- Book 3: Attack Detection, Threat Hunting, and Triage
- Book 4: Incident Response
- Book 5: Metrics, Automation, and Continuous Improvement

## SECTION I

### Introduction

#### Mindset and Preparation

- Cyber Defense Theory and Mental Models
- SOC Data Collection
- Other Monitoring Use Cases
- ***Exercise 2.1: Attack Path and Data Source Assessment***

#### Collection and Monitoring

- Using MITRE ATT&CK to Plan Collection
  - *Exercise 2.2: Prioritizing and Visualizing Attack Trees*
- Cyber Threat Intelligence
  - *Exercise 2.3: Writing Priority Intelligence Requirements*
- Practical Collection Concerns
- Prevention and the Future of Security
- Summary and Cyber42 Day 2



This page intentionally left blank.

## EXERCISE 2.1

# Exercise 2.1: Attack Path and Data Source Assessment

### OBJECTIVES

- Consider visibility of high-impact attacks
- Convert attack models into more specific plans for data collection
- Group and categorize data collection and prioritize based on needs
- Continue developing our prioritized security logging and detection strategy
- Identify critical gaps in your security logging and visibility



### **Exercise 1.2: Developing and Implementing SOC Playbooks**

Please go to Exercise 1.2 in the MGT551 Workbook or virtual wiki.

# Course Roadmap

- Book 1: SOC Design and Operational Planning
- *Book 2: SOC Telemetry and Analysis*
- Book 3: Attack Detection, Threat Hunting, and Triage
- Book 4: Incident Response
- Book 5: Metrics, Automation, and Continuous Improvement

## SECTION I

Introduction

### Mindset and Preparation

- Cyber Defense Theory and Mental Models
- SOC Data Collection
- Other Monitoring Use Cases
  - Exercise 2.1: Attack Path and Data Source Assessment

### Collection and Monitoring

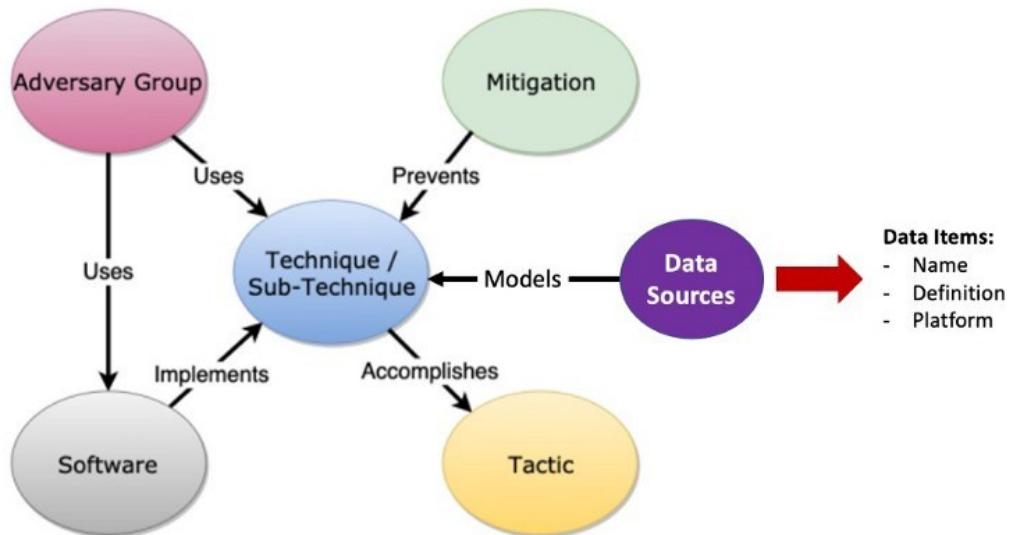
#### **• Using MITRE ATT&CK to Plan Collection**

- Exercise 2.2: Prioritizing and Visualizing Attack Trees
- Cyber Threat Intelligence
  - Exercise 2.3: Writing Priority Intelligence Requirements
- Practical Collection Concerns
- Prevention and the Future of Security
- Summary and Cyber42 Day 2



This page intentionally left blank.

## The Object Types in ATT&CK<sup>1</sup>



### ATT&CK Object Types

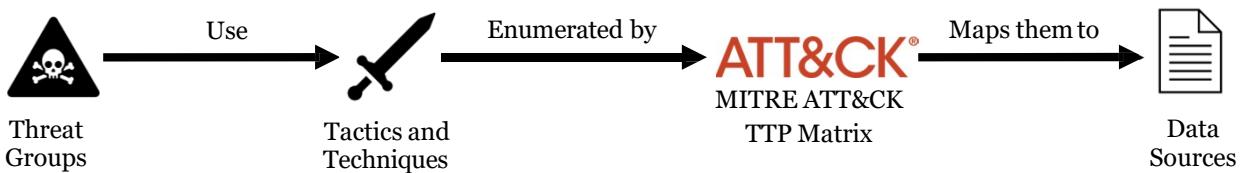
As noted by detection researcher Jose Rodriguez in his blog series “Defining ATT&CK Data Sources,” we often focus exclusively on tactics, techniques, procedures, detections, and mitigations when using the ATT&CK matrix as a reference model. This leaves out an important foundational element in monitoring strategy: data sources required to build detections. Since Jose’s series was published, MITRE has taken note of this omission and defined an initial methodology to define data sources behind each technique in the matrix.

- **Mitigations** – Mitigations are a list of methods to interrupt attacker attempts to perform that specific technique. Mitigations have a numeric identification scheme in the form of "M#####", similar to techniques.
- **Groups** – Group pages in the ATT&CK knowledge base list all techniques each known "group" has been reported to use in past attack campaigns as well as Software used by or attributed to them. Groups are sets of related attack campaigns attributed to a named actor within the threat intelligence community. Examples include the infamous APT1, Darkhotel, and Turla. Each group also has a list of associated groups and/or potential aliases that may be used by vendors for the same threat actor, as well as groups that may have partial overlap with the named group based on open-source intelligence reporting.
- **Software** – The software category enumerates the tools and open-source software used by attackers to conduct the behavior listed in the matrices. Like Groups, software also has an "associated software" property as certain tools may have partial overlap with others, leading to potential confusion depending on when and where that software's use was noted. Software also has a numeric identification scheme in the form of "S#####", similar to techniques and mitigations.
- **Data Sources** – Data sources list potential sources of information for detecting the usage of a given technique. While at the time of writing data sources are purely a list of standardized named sources listed under each technique, the ATT&CK 2020 roadmap notes that a more formal organization of Data Sources is coming soon. This additional organization will undoubtedly improve the ability to perform certain types of gap analysis as will be described later in this paper.

[1] <https://medium.com/mitre-attack/defining-attack-data-sources-part-i-4c39e581454f>

## MITRE ATT&CK For Data Source Planning

- Threat intel enables us to focus on specific threat groups
- Many threat groups are...
  - Already described using MITRE ATT&CK language
  - Can be further defined in ATT&CK based on your own intel
- ATT&CK tactics and techniques list **data sources**
- Therefore, a great starting approach is to utilize ATT&CK to assess data source priority, and collect top tactics/techniques



SANS

MGT551 | Building and Leading Security Operations Centers

80

### MITRE ATT&CK For Data Source Planning

Since we cannot "do it all" when it comes to defense, SOC teams must focus their effort on priority data collection items. There are two struggles involved in doing this, however. One is that we must know via threat intel which tactics and techniques are used by the different adversary groups that exist. The second is that we must know which data sources to collect in order to detect the use of those tactics and techniques in action.

Fortunately, the MITRE ATT&CK framework contains *both* of these pieces of information, and therefore makes a great starting point for designing a data source prioritization effort. If you can at least narrow down which groups are a threat to your organization, those groups can be translated into tactics and techniques, which then can be translated into the data required to spot the use of those techniques.

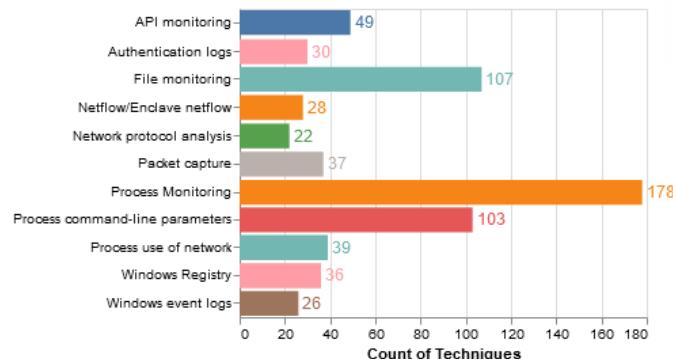
## Most Important Data Sources

What if you just want an easy place to start?

- Play the odds – collect the data sources providing the most coverage
  - MITRE team has calculated this using the tools from the previous slide

Which data source covers most ATT&CK techniques?

- Answer:



### Most Important Data Sources

One question often asked by both SOC analysts, managers, and architects alike is what is the *best* log source if you could only collect a very select source of data? While this question is difficult to answer since every organization is slightly different, there is one logical way to approach it. If we were able to look at all the attack techniques in MITRE's ATT&CK framework and the logs required to detect those techniques, we could see which log sources covers the most area.

Well, the good news is we can use the Python attackcti library<sup>1</sup> to query the matrix for this data and chart it out. While this should not be taken as gospel truth in every organization, it is a great starting point to see in general which log sources produce the most value. According to the analysis the top items are Process Monitoring followed by File Monitoring, Process Command-line parameters, API monitoring, Process Use of Network, Windows Registry, Packet Capture, Authentication Logs, and NetFlow. Not exactly a surprise, but it is useful to have a confirmation backed with real analysis of what catches attacks.

Since the ATT&CK Matrix is always changing this data may become stale over time but the code to reconstruct this based on the current version of the matrix is available at the link below<sup>2</sup>.

1 <https://github.com/OTRF/ATTACK-Python-Client>

2 [https://attackcti.com/playground/Export\\_All\\_Techniques.html](https://attackcti.com/playground/Export_All_Techniques.html)

## Personalized Data Source Assessment

### Building your collection strategy:

- Download all ATT&CK data in a structured format
- Maps techniques to data sources and specific event IDs
- Instrument in your audit/collection policy of those sources & events

### How: Four options to make this easy

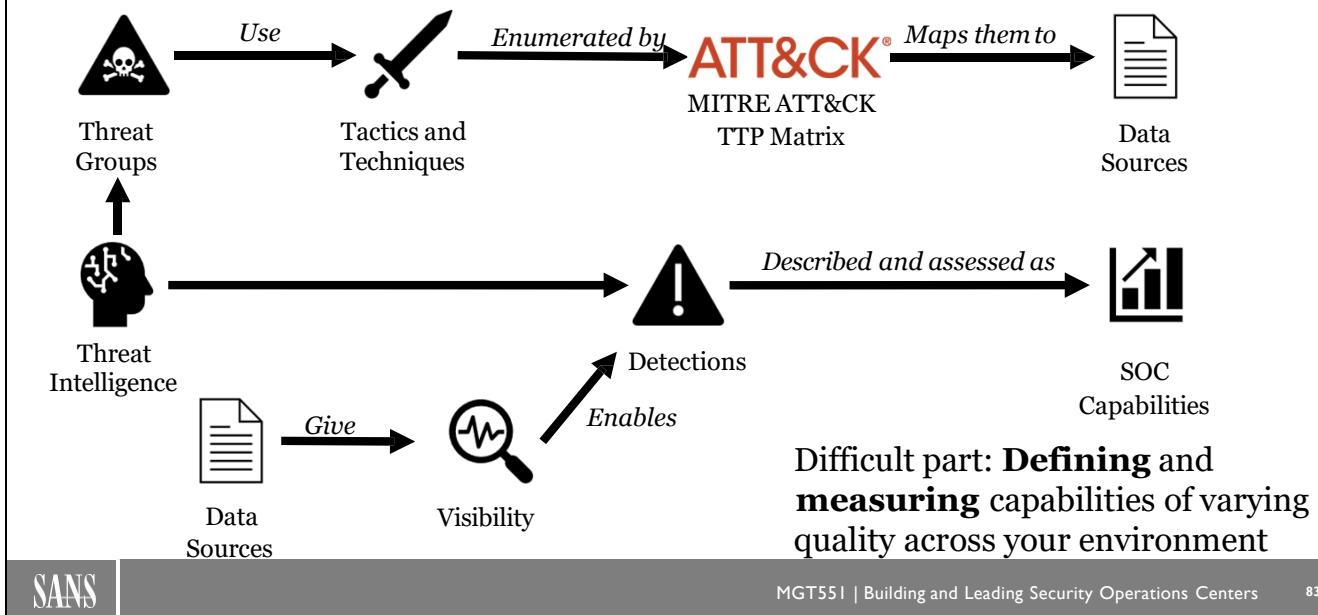
1. MITRE's **attack-scripts**<sup>1</sup>
2. Olaf Hartong's **ATTACKdatamap**<sup>2</sup>
3. Roberto Rodriguez **ATT&CK Python Client**<sup>3</sup>
4. Malware Archaeology **ATTACK logging cheat sheets**<sup>4</sup>

#### Personalized Data Source Assessment

How do you develop a specific log collection strategy tailored to your organization? The first step is getting an idea of who your enemy is and the tactics and techniques they're known for using. Once you have a list of the techniques that are most relevant to you, those techniques can be converted into the data source options for detecting them, as well as specific event IDs. This could be an enormous and laborious process to manually perform, fortunately, multiple groups and researchers have already made this mapping for us. Listed on the slide above are four references that provide code to download the most updated ATT&CK matrix data in a structured format, as well as pre-made maps (such as the Malware Archaeology reference) that go directly from a technique to the data source required to catch it. These herculean efforts of data organization can help you quickly convert the items you need into a specific audit and collection plan.

- 1 <https://github.com/mitre-attack/attack-scripts>
- 2 <https://github.com/olafhartong/ATTACKdatamap>
- 3 <https://github.com/OTRF/ATTACK-Python-Client>
- 4 <https://github.com/MalwareArchaeology/ATTACK>

## Capability Mapping Too!



SANS

MGT551 | Building and Leading Security Operations Centers

83

### Capability Mapping Too!

While looking at tactics and techniques lead us to the data sources we need, that doesn't fully describe the entire problem, as we know from the SOC functions model, collection is just the first step of the process. Beyond correct data collection, we need to know what specific analytics to apply to trigger an alert when a potentially malicious activity is observed and logged. Getting the right data sources just gives the visibility we need. The other piece of the puzzle is the necessary tactical level threat intelligence that can be applied to those logs through security appliance analytics. What results is a collection of data that is collected, and analytics that are applied based on threat intelligence that should, in theory, give us the capability to detect and act upon attempts to enact specific attacker tactics and techniques.

One problem though: Capabilities are not simple binary yes/no, "we can" or "we can't", there are many levels of nuance here. For example, maybe you have servers and desktops with different data collection and retention policies, meaning certain things are detectable in one population of assets and not the other – how do you note this? Ideally, we'd like to say about our detections "we can catch these attacks, in these environments, in these situations". So while capability mapping is great, it then introduces the struggle of trying to describe the quality with which you can perform each detection. This struggle has given birth to many different approach's and attempts to solve this problem in a standardized way as well.

## Capability Quality Ranking

- **Objective:** Quantitative or qualitatively describe confidence in detecting each attack tactic
- Factors to consider:
  - **Location** – Environments where the data is available (desktops, cloud, etc.)
  - **Completeness** – Is the data collected both complete, parsed, and usable?
  - **Visibility Level** – Extent of data availability (local collection only, centralized logs, SIEM collection, etc.)
  - **Count** – How many different analytics apply to this tactic?
  - **Confidence** – Is your analytic going to work in all instances of that tactic being used? (low, med, high)
  - **Timeliness** – How quickly can you move from event to activated alert based on that data?
- Result:
  - Metrics of quantitative or qualitative assessments for each ATT&CK Tactic detection capability

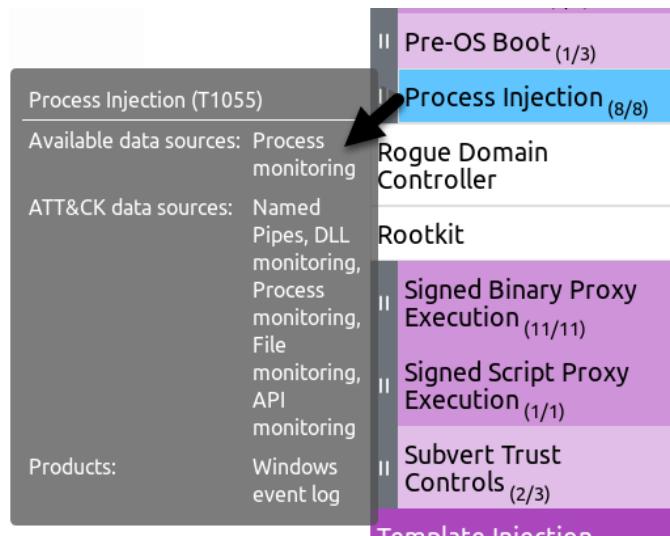
## Capability Quality Ranking

When setting out to track and describe detection capabilities, there are many variables you may consider tracking. Some of the most common are listed above. The goal in doing this should be to track meaningful metrics about what you can and cannot detect, and your limits and confidence in being able to do so.

The difficult part is finding the balance between going overboard with the details and not tracking enough. For each team, the balance will be slightly different, but aim to keep the 80/20 rule in mind here. Consider which specific questions you're trying to answer with this data and be sure to collect the items that will answer those questions, but don't be tempted to track anything and everything just because you can. Hit the priorities with minimum effort, then adjust as needed. In practice, this may mean tracking 2 to 3 of the above items such as location, confidence, and visibility level.

## DETT&CT

- A project to help you do capability ranking
- Python script and UI
- Creates YAML file
- Visualize with ATT&CK Navigator layer
  - Includes details on data sources and products



## DETT&CT

If you're looking for a great head start in labeling, organizing and visualizing of your capabilities, check out the DETT&CT project.<sup>1</sup> DETT&CT by Marcus Bakker (Twitter: @bakk3rm) and Ruben Bouman (Twitter: @rubenb\_2), which stands for Detect Tactics, Techniques & Combat Threats is a set of tools to help you label which data sources your team has available and how they apply to the MITRE ATT&CK tactics, and easily map and store that information into a YAML file which can be manipulated through a Python script or web application.<sup>2</sup>

The picture above shows the results of using the DETT&CT script to generate an ATT&CK Navigator layer based on data sources that visualizes and notates each tactic with both data sources that are available and possible for that item, as well as the products that it relates to.

- 1 <https://github.com/rabobank-cdc/DeTTECT>
- 2 <https://rabobank-cdc.github.io/detectt-editor/#/home>

## Additional Direction on Key Data Sources

- Microsoft guidance<sup>1</sup>
- MITRE ATT&CK<sup>2</sup> for Enterprise, ICS, Cloud, Mobile
- NSA "Spotting the Adversary" Guide<sup>3</sup>
- Malware Archaeology Logging Cheat Sheets<sup>4</sup>
- Florian Roth's Linux Auditd config<sup>5</sup>
- Blue Team Handbook Vol. 2<sup>6</sup>
- Ultimatewindowssecurity.com<sup>7</sup>
- Roberto Rodriguez's OSSEM Project<sup>8</sup>
- What2log.com



### Additional Direction on Key Data Sources

Additional views on what is important are always helpful since each may focus on catching a different target situation, so here are a few other popular sources of key log sources and data you should collect.

1<https://docs.microsoft.com/en-us/windows-server/identity/ad-ds/plan/security-best-practices/audit-policy-recommendations>

2 <https://attack.mitre.org/>

3<https://apps.nsa.gov/iaarchive/library/reports/spotting-the-adversary-with-windows-event-log-monitoring.cfm>

4 <https://www.malwarearchaeology.com/cheat-sheets>

5 <https://github.com/Neo23x0/auditd>

6 <http://www.blueteamhandbook.com/>

7 [Ultimatewindowssecurity.com](https://Ultimatewindowssecurity.com)

8 <https://github.com/OTRF/OSSEM>

# Course Roadmap

- Book 1: SOC Design and Operational Planning
- *Book 2: SOC Telemetry and Analysis*
- Book 3: Attack Detection, Threat Hunting, and Triage
- Book 4: Incident Response
- Book 5: Metrics, Automation, and Continuous Improvement

## SECTION I

### Introduction

#### Mindset and Preparation

- Cyber Defense Theory and Mental Models
- SOC Data Collection
- Other Monitoring Use Cases
  - *Exercise 2.1: Attack Path and Data Source Assessment*

#### Collection and Monitoring

- Using MITRE ATT&CK to Plan Collection
  - **Exercise 2.2: Prioritizing and Visualizing Attack Trees**
- Cyber Threat Intelligence
  - *Exercise 2.3: Writing Priority Intelligence Requirements*
- Practical Collection Concerns
- Prevention and the Future of Security
- Summary and Cyber42 Day 2



This page intentionally left blank.

## EXERCISE 2.2

# Exercise 2.2: **Prioritizing and Visualizing Attack Techniques and Security Controls**

### OBJECTIVES

- Learn to use MITRE's ATT&CK Navigator
- Use threat intelligence to prioritize threat group favorite techniques
- Use Navigator to assess technique coverage based on your security controls
- Align threat capabilities and mitigations to identify gaps in coverage
- Export Navigator layers for use in other security tools



MGT551 | Building and Leading Security Operations Centers

88

#### **Exercise 2.2: Prioritizing and Visualizing Attack Techniques and Security Controls**

Please go to Exercise 2.2 in the MGT551 Workbook or virtual wiki.

# Course Roadmap

- Book 1: SOC Design and Operational Planning
- *Book 2: SOC Telemetry and Analysis*
- Book 3: Attack Detection, Threat Hunting, and Triage
- Book 4: Incident Response
- Book 5: Metrics, Automation, and Continuous Improvement

## SECTION I

### Introduction

#### Mindset and Preparation

- Cyber Defense Theory and Mental Models
- SOC Data Collection
- Other Monitoring Use Cases
  - *Exercise 2.1: Attack Path and Data Source Assessment*

#### Collection and Monitoring

- Using MITRE ATT&CK to Plan Collection
  - *Exercise 2.2: Prioritizing and Visualizing Attack Trees*
- **Cyber Threat Intelligence**
  - *Exercise 2.3: Writing Priority Intelligence Requirements*
- Practical Collection Concerns
- Prevention and the Future of Security
- Summary and Cyber42 Day 2



This page intentionally left blank.

## Cyber Threat Intelligence Overview

- CTI informs every SOC function
- Can be done at all stages and maturity levels
- Best intel often internally sourced
- In this module:
  - Intel types and sources
  - Producers and consumers
  - Mental models
  - Operationalizing threat intelligence in the SOC



*Actual CTI team photo*

### Cyber Threat Intelligence Overview

There are few topics in cyber defense that get as much attention as cyber threat intelligence (CTI). Perhaps it's the overlap in terminology with traditional intelligence gathering, or the thought of facing off against a human adversary at the remote end of our collection capabilities. While its mystique may be debatable, the value of cyber threat intelligence in various security disciplines and capabilities cannot be overstated. Intelligence informs our threat model, our security monitoring priorities, our defensive controls, our alert triage approach, our incident response policies, and our strategic planning. Contrary to popular belief, CTI is neither an advanced capability (though it can get quite sophisticated) nor is it exclusively external-facing. In fact, some of the most useful and actionable intelligence comes from inside of your own environment!

CTI is also one of the most hotly debated topics in cyber defense today – whether it is the value of attribution or what exactly constitutes cyber threat intelligence, you'll be hard pressed to find a more passionate debate in infosec. In this section, we're going to establish a common set of terms and definitions for CTI, the types of CTI you'll work with in the SOC, the roles of intelligence producers and consumers, and ways in which CTI can be operationalized in the SOC.

## Cyber Threat Intelligence

- Collaboration is key
- Not all processes require the same level of automation
- Tools and data change as CTI capabilities evolve
- Requirements have become a staple of mature intelligence teams
- CTI is supported by a community: consumers and producers



### Cyber Threat Intelligence in the SOC

Whether CTI is a component of your SOC, or something provided by another team or third-party service provider, it's important to understand the role CTI should play in detection, enrichment, and response and what it takes to do so. Since CTI has evolved quite rapidly as a discipline in security operations, it's equally important to understand some of the pitfalls and misconceptions of applying and measuring CTI in a defensive environment.

The 2020 SANS Cyber Threat Intelligence (CTI) Survey consolidated insights from over 1,000 respondents across our industry, reflecting huge growth in threat intelligence capabilities over the past few years. Here are some of the key points about modern CTI capabilities based on the information gathered as part of the survey:

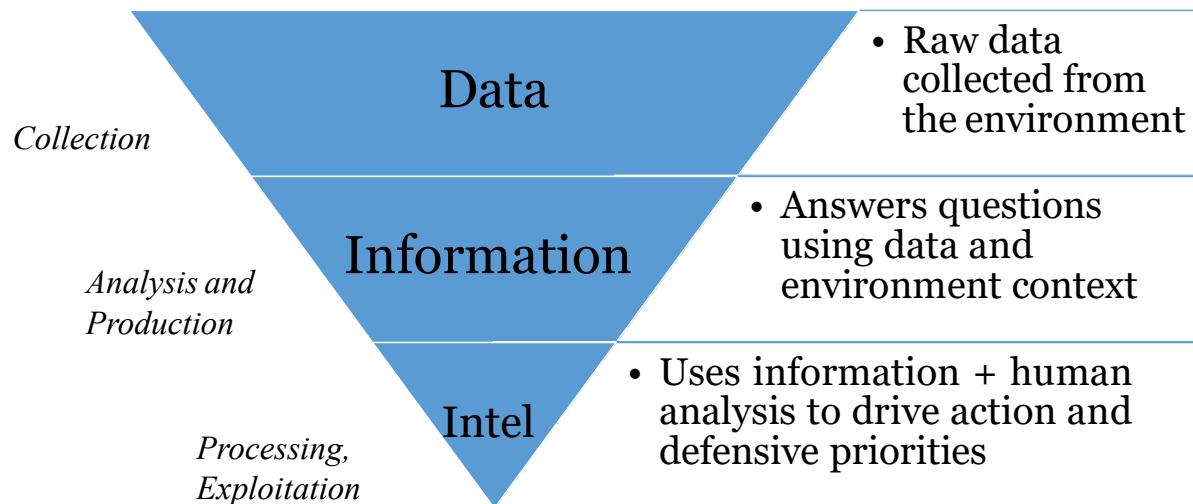
1. **Collaboration is key** – while many organizations have built (or are building) their own threat intelligence capabilities, partnering with vendors or external service providers is the norm.
2. **Not all processes require the same level of automation** – while many aspects of CTI work (such as data deduplication) can be repetitive, semi-automation is usually the best solution so as not to lose valuable insights (for example, the knowledge that a given field or attribute appears multiple times and thus requires deduplication). This is where Threat Intelligence Platforms (TIPs) come in handy!
3. **Tools and data change as CTI capabilities evolve** – as teams mature, they tend to derive less value from third party sources and data feeds and more value from internal data and visibility
4. **Requirements have become a staple of mature intelligence teams** – formal requirements for intelligence analysis play a key role in long-term success as they help teams focus their efforts
5. **CTI is supported by a community or consumers and producers** – while more organizations consume intelligence than produce it, a plurality do *both* which can be a great way to ensure that you're meeting all of your intelligence collection requirements

This last item is what we're going to focus on for the next few minutes, because where you fall on this spectrum will have a major impact on the costs and capabilities of your SOC.

You can read more highlights of the 2020 CTI Survey here: <https://www.sans.org/webcasts/2020-cyber-threat-intelligence-cti-survey-results-112005>.

And you can download the full results here: <https://www.sans.org/reading-room/whitepapers/analyst/membership/39395>

## Threat Data vs. Information vs. Intelligence



### Threat Data vs. Information vs. Intelligence

We covered technical data collection in the last module, but how can we supplement and prioritize this raw data using information we've gathered about the threat landscape? Turning raw data into intelligence is the best means of doing this and an important part of any SOC.

First, let's differentiate threat data vs. threat information vs. threat intelligence. This graphic is derived from a concept in the US Department of Defense Joint Publication 2.0 that illustrates how one becomes the next throughout the intelligence generation process.<sup>1</sup>

Threat **data** is the raw information and unarguable facts collected from the environment. In the case of network security monitoring, for example, this would be the logs and packets recorded by our sensors on the network. On their own, this does not represent intelligence or even information—merely the raw data that must be analyzed to create threat information or threat intelligence.

Threat **information** is an intermediate step that is all about using analysis to answer a question using the threat data as input. Threat information doesn't necessarily have the requirement to inform action; it is, however, the next step to producing threat intelligence. For example, is a system on the network compromised? We could answer that question using threat data collected from the environment, possibly by looking for signs of malware communication or execution.

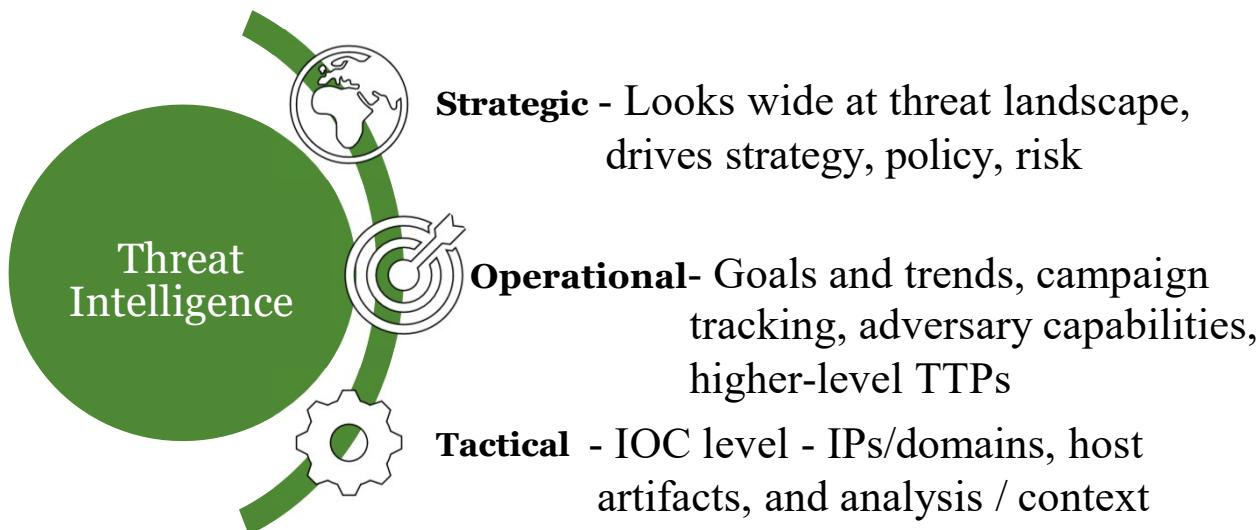
Threat **intelligence** requires gathering lots of threat information and aggregating it to make an assessment of some sort. This analysis of multiple bits of threat information drives an organization's security policy, spending, and defensive posture. It attempts to align defensive actions against what appears to be the TTPs of actors that are a threat to the organization or, in other words, "offense informs defense."<sup>2</sup>

For more information, threat intelligence vendor Recorded Future has a useful blog post that explains these concepts in more detail.

1 Joint Publication 2.0 – Joint Intelligence: [http://www.jcs.mil/Portals/36/Documents/Doctrine/pubs/jp2\\_0.pdf](http://www.jcs.mil/Portals/36/Documents/Doctrine/pubs/jp2_0.pdf)

2 Threat Intelligence, Information, and Data: What Is the Difference? <https://www.recordedfuture.com/threat-intelligence-data/>

## Threat Intel Levels



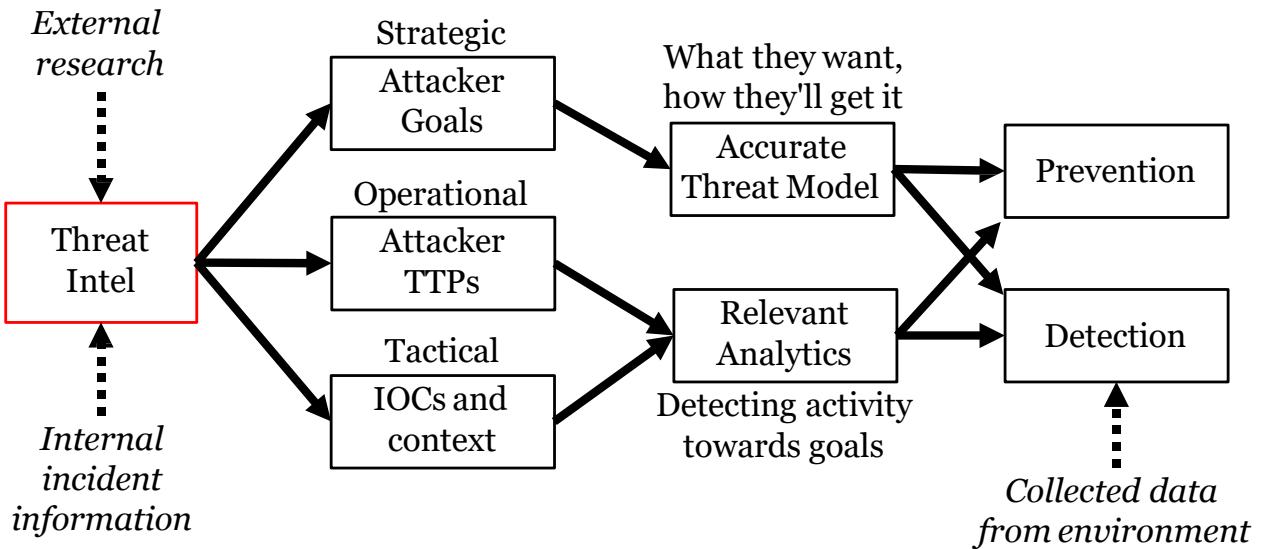
### Threat Intelligence Levels

Next, to discuss one of the basic mental models for types of threat intelligence. There is much confusion about what threat intel should and should not be throughout the infosec community. It wasn't long ago that purchasing a list of malicious domain names and IP addresses without any context passed as "threat intel", but fortunately those days are (or at least should be) over.

Whether you have a dedicated threat intelligence team or buy information from a vendor, what types of data are you specifically looking for? This can be broken down into three levels – strategic, operational, and tactical threat intelligence.

- Strategic threat intelligence is the highest level that is often dealt with at the CISO or board level and concerns who attackers are and what they want, but not the technical details about how they might get it. You may at this level know there are two groups APT 1, and APT 2, that primarily target your industry, and what types of things they have done to similar organizations in the past. This helps leaders and managers assess risk and prioritize resources as best possible.
- Operational level threat intelligence is a middle of the road type of threat intelligence often used at the security director or SOC manager level. It can be classified as attacker goals, and the tactics, techniques, and procedures (TTPs) they will use to achieve those goals. This may include broader campaign information, attacker toolsets, and specific attributions that can be made for tools or other methods the attackers use to get what they want.
- Tactical level threat intelligence is what analysts will deal with on a day-to-day basis. This is the level that most resembles the questionable value "list of IOCs" that was commonly purchased in the past. Yes, we will need a list of bad domain names, IPs, and the like, but if we ever do match them to activity in the environment, analysts need to know the *context* of why they were called bad as well. We'll discuss what makes quality threat intelligence in a moment.

## Threat Intelligence in the SOC



SANS

MGT551 | Building and Leading Security Operations Centers

94

### The Need for Threat Intelligence

While the previous slide should help in picturing where threat intel fits into the daily life in the SOC, here's a map of how those three levels of threat intelligence interface with tasks the SOC is responsible for. The threat intel team should be taking in both external and internal sources of data to understand your organization's likely and actual attackers. They should then exploit this information and produce the three levels of threat intelligence which in turn will feed what the SOC uses as a threat model. The threat model tells the SOC where to focus its resources and ensures it can put up the best preventative and detective measures possible. Simultaneously, the operational and tactical level threat intelligence can be passed to those who write the analytics for the SOC to confirm any attempts to make progress toward those assumed goals is immediately stopped or detected.

But who is responsible for the contents of that "Threat Intel" box? Many SOCs fight for this capability, and for a variety of reasons – it's a key element in prioritizing incidents, planning hunt, and improving detections. Also, let's face it, it just seems like a cool thing to do. But as we'll see in the next few slides, there are lots of cost and resource implications for CTI work depending on what CTI functions are performed within the SOC Team.

## Threat Intel Sources

- Incidents and investigations
  - Capture observables + context and TTPs during the incident response process, even if the attack is ultimately deemed unsuccessful
- Active research
  - Honeypots and sensors
- Forums, websites, feeds, and social media



### Threat Intelligence Sources

There are many different sources of intelligence your team can use to refine your threat model, enrich your data, and build custom analytics. Remember that *intelligence* is the result of an analytic process, not just raw data that we're going to jam back into our tools or processes. We can get quality data with context from the following sources:

- Incidents and investigations
- Active research (honeypots and other interactive collection infrastructure)
- Forums, websites, feeds, and social media

Incidents and investigations are one of the most useful, and most often overlooked, source of intelligence. Understanding how you have been targeted in the past, by what kinds of threat actors and their TTPs, is invaluable in prioritizing investigations and preparing for future incidents. Even if an investigation ultimately determines that an attack is unsuccessful or mitigated by your controls, it's important to capture techniques, infrastructure, and other artifacts that you have observed. Active research via honeypots and other interactive sensing infrastructure provides a low-risk way to observe attackers getting interactive with target infrastructure without putting your organization at risk.

Finally, many analysts have developed a list of favorite sources for intelligence information. These might be forums, RSS feeds, blogs, or social media accounts. Periodically poll your team(s) for these resources and ensure that they are shared, and the resulting information prioritized if it has analytic value. Social media in particular has recently emerged as a rich (and timely) source of threat intelligence, although the data set can be unruly and unverified.

## Intel Producers and Consumers

- There are threat intelligence **producers** and **consumers**
  - Many SOCs contain a threat intelligence group
  - Threat intel group produces intelligence, while analysts consume it
- Analyst jobs require using threat **data, information, and intelligence** to identify and protect against compromise
- **Threat intelligence platforms** help you accomplish this task
  - Knowledgebase of your threat data, information, and intelligence
  - Automates exchange and querying of data from other security tools
  - Do NOT produce intelligence for you!

### CTI Producers and Consumers

While many mature SOCs are both producers *and* consumers of CTI, this should be a conscious decision based on your SOC's scope and unique requirements. It also follows that you must be a relatively mature SOC to produce threat intelligence. As consumers, your analysts must use threat data, information, and intelligence to identify intrusions. As producers, they'll need to collect requirements and produce *actionable* information the team can use to do their jobs more effectively. Threat intelligence platforms can help automate this process regardless of where your team falls.

There are lots of different approaches and mental models that are useful for CTI production and consumption, which we'll talk about next.

## Consuming Intel

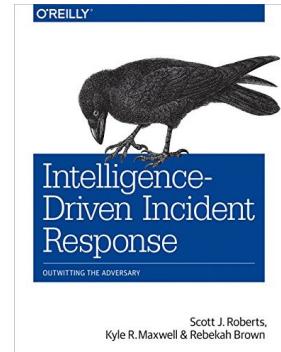
Quality threat intelligence should include<sup>1</sup>:

### 1. Collection source info

- Where it came from – Internal/external, automated collection?
- Relevant dates – First seen, last seen, collected

### 2. Analysis detail

- Context – Related activity, tools, attribution, etc.
- Potential biases – Any potentially relevant factors



Simplified phrasing from excellent Medium post <sup>2</sup> by Andy Piazza:

- "IOC = Observable + Context"
- "**PRO-TIP:** if your Threat Intelligence Platform (TIP) doesn't have a Description attribute for IOCs, you don't have a TIP. You have an Observable Aggregator."

### Threat Intelligence Quality

According to Threat Intelligence experts Scott Roberts and Rebekah Brown (authors of "Intelligence-Driven Incident Response", a must-read book on the topic), quality threat intelligence can be recognized by the inclusion of both collection sources and analysis information.

In the collection source category is both where the intelligence was collected from and associated dates. *Where* did the data come from, was it generated from internal incident data, or was the data collected by a public honeypot sitting on the internet? Internal data is the best and most relevant intelligence since it's guaranteed to be relevant, whereas public honeypot data collected in an automated way may be of less relevance. When looking up an IOC, analysts should ideally have a sense of *when* that piece of data was relevant. When was it first seen, last seen, when is it likely to be relevant? IOCs in information security are highly perishable, an adversary can change an IP, domain name, or hash in the blink of an eye, so knowing you were attacked by an IP that was malicious 3 years ago is much different than one that was identified as malicious today.

The secondary factors are analysis-based items, the context of the data as well as any potential biases that might be relevant. On the context, analysts must understand the situation in which the IOC was used and discovered. Who used it? Was it used for the delivery stage of an attack, or for interactive command and control? What type of malware was communicating with it? These are the questions an analyst who gets an alert for "threat intel match" will have to answer, and if it is not included from the vendor, they'll be left on their own to make the call by analyzing the data from scratch. The secondary factor is potential bias. If your collection system or threat intel vendor is focused on a particular type of data or threat actor, you should be aware of how that might color their analysis. Consider the things that information might *not* contain due to the source's data type, focus, or methods of collection.

1 <https://www.oreilly.com/library/view/intelligence-driven-incident-response/9781491935187/>

2 <https://klgrz.medium.com/cti-is-better-served-with-context-getting-better-value-from-iocs-496343741f80>  
(<http://mgt551.com/ioc>)

## Producing Intel (the EASY way)

### Chris Cochran's EASY Framework:

1. **Elicit requirements**
2. **Assess collection plan**
3. **Strive for impact**
4. **Yield to feedback**



*Requirements are fundamental and flexible*

In his SANS CTI Summit talk, Chris Cochran (former Threat Intelligence Operations Lead at Netflix) described an EASY model for improving threat operations:

1. **Elicit requirements** from your stakeholders to drive collection and analysis priorities
2. **Assess collection plan** - Identify internal and external information sources, and reassess your collection plan as your requirements change
3. **Strive for impact** – producing for the sake of production doesn't help anyone; frame intelligence to help the stakeholder make a decision or take action to improve security
4. **Yield to feedback** – make it easy for stakeholders to give feedback, be ready to communicate and change approach if needed

A key takeaway of the EASY model is that everything starts with requirements. If you don't want your CTI capability to be a "self-licking ice cream cone," listen to your internal and external stakeholders to understand how you can bring value to what they're doing. This might be strategic intelligence to your executive management for decision support, or operational intelligence to security and risk management, or tactical intelligence to support better detections in the SOC. Requirements may change over time as threats evolve or the business changes, so don't forget to revisit them periodically and be ready to adjust your collection plans if needed.

You can watch Chris' presentation on his EASY framework here in the video titled "**The Threat Intelligence EASY Button with Chris Cochran - SANS CTI Summit**": [https://www.youtube.com/watch?v=ecY5WW\\_qppc](https://www.youtube.com/watch?v=ecY5WW_qppc)

## Asking the Right Questions

### Strong requirements are:

- **Singular** – focusing on one question only
- **Atomic** – specific to a particular fact or event
- **Decision Centric** – should lead to making a decision or improvement
- **Timely** – within the timeframe for the intel to be useful

### Can be derived from:

- Past incidents\*
- Business model
- Geography
- Vertical
- Infrastructure
- One-offs

\* One of the **best** sources of intelligence!



### Asking the Right Questions

Quality intelligence starts with quality requirements. In his blog post *CTI Squad Goals - Setting Requirements*<sup>1</sup>, Scott Roberts describes the key questions that intelligence stakeholders – which include the SOC – might ask the intelligence function to answer. These questions, formalized in intelligence requirements, set the tone and priority of the entire intelligence analysis function: tempo, collection sources, processing required, and methods of dissemination. The requirements process for intelligence should produce requirements that are singular, atomic, decision centric, and timely. More specifically, requirements should be tied to a single question about a specific fact or event, and the intelligence required must lead to some action – a decision or an improvement. Finally, requirements must be time-oriented rather than open-ended.

Sources of intelligence can include internal sources like past incidents and infrastructure tracking data, business model, geography, sector, and infrastructure-specific research. Internal sources often produce the richest, most relevant intelligence – even when compared to paid feeds and other commercial sources. Tomorrow we'll talk about incident response and gathering those vital details during the course of an investigation that can inform your defenses and operational priorities later.

[1] <https://sroberts.medium.com/cti-squadgoals-setting-requirements-41bcb63db918>

## Asking the Right Questions (continued)

### Are these requirements strong or weak?

- Who might attack us?
- Is Actor X active in our area or sector?
- What detection rules should we use?
- What are current indicators of Actor X activity?
- What malware should we look for on our network?
- Will we get DDoS'd?
- Are attackers using Ryuk against my vertical?

#### Asking the Right Questions (continued)

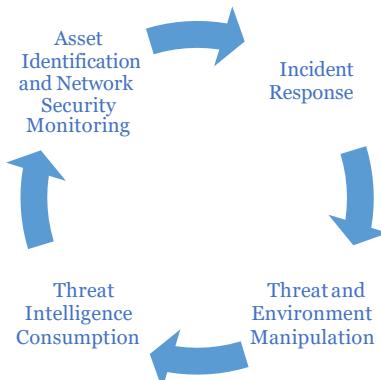
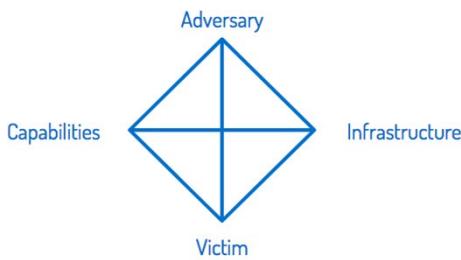
Look at these requirements and think about the characteristics we discussed from the previous slide. Based on those characteristics, are these requirements strong or weak?

- **Who might attack us?** This is a very open-ended requirement with no specificity on the “who” or the definition or the term “attack”. This one is pretty weak and will likely not result in specific or actionable intelligence.
- **Is Actor X active in our area or sector?** This one is better; it’s environment/sector specific and seeks to better understand a particular actor. Definitely a strong requirement.
- **What detection rules should we use?** This one is trickier. On first read it doesn’t sound very specific, but given some additional context – for example, specific kinds of rules we’re looking for – it could be a very strong requirement. Phrased another way, what detections cover relevant TTPs for threat actors and campaigns active against our sector/business/infrastructure?
- **What are current indicators of Actor X activity?** This one is strong, and maybe better expresses what we’re looking for in the above requirement. Note that this one is also time-oriented!
- **What malware should we look for on our network?** Malware is a fairly generic term, and this requirement is not bound by timeframe, nature of the threat, or target profile (specific infrastructure/applications/OS/etc.).
- **Will we get DDoS'd?** Another example of a broad, overly generic question. What sort of DDoS are we concerned about and why? Is there a specific campaign or observed attack that we’re concerned might impact us? If so that detail should be reflected here.
- **Are attackers using Ryuk against my vertical?** This is a good example of a requirement that is specific to a certain threat and our vertical and should produce a specific (and actionable) answer in the intelligence we receive.

## Mental Models for Threat Intelligence



The Cyber Kill Chain and Diamond Model  
(for producers)



The Active Cyber Defense Cycle  
(for consumers)

### Mental Models for Threat Intelligence

Here are some examples of mental models used to produce and consume cyber threat intelligence. You're probably already familiar with the **Cyber Kill Chain**, which is a model created by analysts at Lockheed Martin to describe sophisticated, multi-step attacks and campaigns; it is also a useful model for analyzing intrusions in order to generate intelligence. The **Diamond Model** is an approach to conducting intelligence on network intrusion *events*. The model gets its shape from the four core interconnected elements that comprise any event: adversary, infrastructure, capability, and victim. The diamond model is often used in conjunction with the kill chain to describe/categorize discrete events at each step of the kill chain. Identifying links between elements of each event can help us understand the relationship(s) between each event in the chain.

The **Active Cyber Defense Cycle** is a framework created by Dragos founder and SANS Instructor Robert M. Lee to break down intelligence analysis activities into four steps: incident response, threat and environment manipulation, threat intelligence consumption, and security monitoring. It is designed to bridge various security functions for the purpose of applying intelligence in a defensive context. It can start at any phase of the cycle, with the phases continually feeding into one another in order to create an ongoing process.

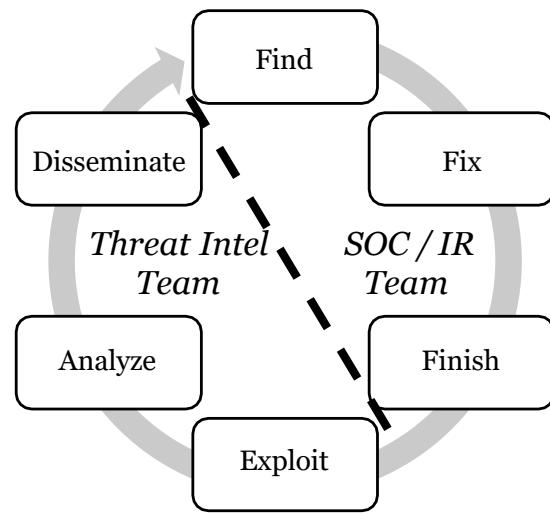
You can infer from the two models on the left that producing complete intelligence requires a great deal of visibility or telemetry. It isn't uncommon for software vendors like Microsoft to build complex and highly effective threat intelligence capabilities based on their vast telemetry. While this isn't necessarily a hard barrier to entry for producing your own cyber threat intelligence, it's a useful frame of reference in understanding the scope of what mature organizations are working with.

These are just a few of the hundreds of different models used for intelligence analysis. Deciding which models to use as the basis for your intelligence production or consumption processes is a function of (1) the time and resources you have for those processes and (2) the information to which you have access. Like all models, these are simply frameworks that we can use to add structure and repeatability to our efforts; find the one that is most useful and most effective, even if that means developing your own!

## Threat Intel / SOC Interfacing – The F3EAD Cycle

A threat intel source is no good if your team doesn't *use* the info

- The **F3EAD**<sup>1</sup> cycle is a great depiction of how it **should** work
- Shows the cyclic nature of
  - Threat intel team output going to SOC/IR
  - SOC/IR team output going back to TI
- Without this, Threat Intel becomes the dreaded "self-licking ice cream cone"



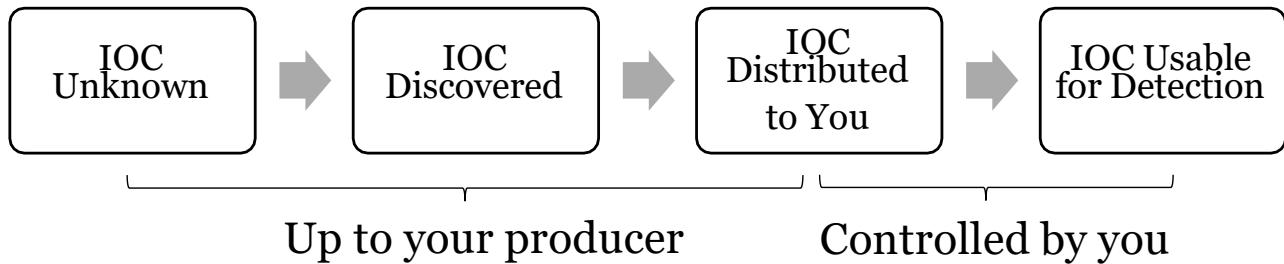
### Threat Intel / SOC Interfacing – The F3EAD Cycle

Another incredibly useful model from Scott Roberts' and Rebekah Browns' book is the F3EAD cycle as interpreted for cyber threat intelligence. The F3EAD acronym, which stands for Find, Fix, Finish, Exploit, Analyze, and Disseminate is a military-born model for combining both intelligence generation and operational usage of that intelligence. For the SOC, this serves as a perfect model of how the threat intelligence team should be an input to the SOC and IR team, and the SOC and IR team should feedback information as an input to threat intelligence. The stages of the F3EAD cycle are as follows:

- Find – Developing a thorough threat model and designing your network such that potential enemy activity in the environment can be successfully detected. This activity must be fed by the information gained from threat intelligence to know what attacker TTPs to prioritize, and what goals to assume attackers may be trying to accomplish. This is similar to the "prepare" stage in the incident response cycle we will discuss later.
- Fix – In this stage, you "get a fix" on the enemy, as in target where they have entered the environment through collected telemetry and establish what has been affected and the attacker's methods of operation. This correlates the closest with the Identify stage of the incident response cycle.
- Finish – This is the incident response stage where action is taken to remove the attacker from the environment and remediate any damage. It is this stage where most teams fail to complete the cycle. Every bit of information collected from actual incidents seen in the environment should be used by the threat intel team as a high-quality input about who your attackers are and how they operate. This stage correlates with the Contain, Eradicate, and Recover stages of the incident response cycle.
- Exploit – With the new incident information from the environment, in the exploit stage the threat intelligence team should gather all relevant data and IOCs and organize them in a way that can be used for the next stage.
- Analyze – This is the key stage for the threat intelligence team and is where the information about attacker goals and TTPs is synthesized based on all known information. In this stage, all data is

## Speed of Intelligence Use

Intel is another place we need a fast update pace:



- Ingested intel should be available for detections as quickly as possible
  - We want the smallest time possible for this whole process
  - But we only control the last step
- Ideal: **continuous** push of new indicators to security monitoring tools

### Speed of Intelligence Use

Threat intelligence ingestion is another process in which the speed of application can have large consequences. Any IOCs for known bad domains, IPs, etc., must go through a process of being discovered by someone, collected by a vendor, pushed to customers, and implemented by those customers for matching and detection. While you want the entire process to go as fast as possible, unfortunately you only have control over the final piece. To minimize the potential for missing malicious activity due to slow threat intelligence, consider how you can minimize the time from discovery of a bad IOC to your ability to match that IOC in any data feed in your environment.

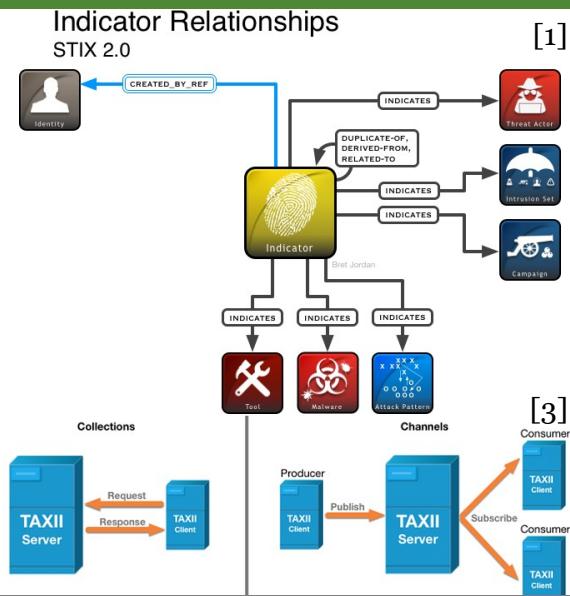
For evaluating threat intelligence vendors, it's worth questioning how their update process works and how quickly you can expect new, bad IOCs to be pushed out – once per hour, once per day, etc. Since malicious indicators can change at the minute scale, obviously the closer your vendor gets to *continuous* update publishing the better.

On your side, you should also consider how your ingest process works, and if it's also updating at the same speed. Your threat intelligence platform (or perhaps other tools that ingest threat intelligence) will likely be subscribing to many threat feeds, as well as have data added to them by analysts working through cases. Review the process and speed at which this intelligence goes from "entered into the threat intel platform or tool" to "active and able to match against traffic" and work to minimize it as much as possible.

## Threat Intelligence Transport: STIX and TAXII<sup>1</sup>

How is threat intelligence distributed?

- **STIX** = Structured Threat Information Expression
  - Machine readable key:value pairs, ingestible by your Threat Intelligence Platform
  - Goal: Make ingesting, publishing, and sharing CTI with context **fast** and easy
- **TAXII** = Trusted Automated eXchange of Intelligence Information
  - Application layer protocol for (STIX format) CTI exchange through an HTTPS API



### Threat Intelligence Transport: STIX and TAXII

One way to tighten the threat intelligence usability loop is to ensure you are ingesting IOCs from your threat intel vendor using one of the convenient standards for transacting CTI. The main standards in this space to be aware of are STIX and TAXII, created by the OASIS Cyber Threat Intelligence Technical Committee (CTI TC).<sup>1</sup> Vendors and tools that utilize STIX format indicators and share them via a TAXII server make ingestion of this type of data with a TAXII client fast and easy. Not only do STIX and TAXII take care of indicator expression and transport, but they do it all while keeping the data safe via HTTPS encryption, and support expressing the complex relationships between all the pieces of information (that all-important context that makes CTI worthwhile).

If you'd like to see an example of what STIX data looks like, see footnote 2 below.

- 1 <https://oasis-open.github.io/cti-documentation/faq.html>
- 2 <https://oasis-open.github.io/cti-documentation/stix/examples>
- 3 <https://oasis-open.github.io/cti-documentation/taxii/intro.html>

analyzed and new information and predictions about the attacker should be developed that can help prevent or detect future intrusions. This helps better ready the organization for the inevitable next wave of attacks.

- Disseminate – This stage completes the cycle as the newly synthesized intelligence information is passed back to the SOC, management, and any other stakeholders for the creation of new analytics and detection/protection methods.

[1] <https://www.oreilly.com/library/view/intelligence-driven-incident-response/9781491935187/>

## Applying CTI with a Threat Intelligence Platform

Much of your alerting is based on observables

- Known bad IPs, domains, hashes, etc.

Threat Intelligence Platforms (TIPs):

- Store context for observables
- Perform automated lookups via API
- Record context about stored items (NOT just atomic indicators)
- Identify associations between events
- Facilitate automated sharing with other organizations

### Applying CTI with a Threat Intelligence Platform

A threat intelligence platform's main purpose is to store the body of threat information, analysis, and indicators you have collected and then make it available in an easy-to-search way. The database will be manually searched as part of incidents and thus needs to have a user-friendly interface that makes the correlation of data across events easy. It also needs to have good integration capability so that tools such as your IMS and SIEM can send and pull information from it through an API of some sort. This is a crucial feature. Threat intelligence locked up in a proprietary system that can't be leveraged by your other security devices will be of minimal use, so communication across security tools is a highly important capability.

## Threat Intelligence Platform Types

Indicators or low-level configuration details? Intel collection or curation?

- Most TIPs handle observables with ease
  - IP Addresses, filenames, domains, hash values, URLs, etc.
  - Important feature: **easy bulk entry and integration**
- *Some* TIPs do a better job with context & additional features
  - Are you storing **malware configurations? Non-standard fields?**
  - How do you want to **correlate** across items stored?
  - Is **sharing** a required function?
  - What **volume** of indicators will you be storing?

### Threat Intelligence Platform Types

One of the items that may drive your decision is the depth of detail of intelligence that you need to store, as well as your interest in sharing your findings. John Bambanek of Bambanek Consulting, a prominent malware researcher, points out in his talk referenced below that many TIPs were not capable of storing the level of detail he needs for his work.<sup>1</sup>

Most threat intelligence platforms will be built for, and have no problem storing common indicators such as IP addresses, hashes, URLs, filenames and the like. But remember – IOCs = observables *plus* context. Is your platform equipped to capture and store context in a meaningful way? Are you looking to gather and store your own intelligence, or is collection from a third party (an intel data feed, for example) the basis for your CTI capability?

If this is your main need, it is likely almost any solution will provide what you need, and the choice can be driven based on integration with other tools. If you are doing an analysis that needs to include in-depth, arbitrary field names, however, some solutions are likely better suited than others. According to Bambanek, who has tried many of the solutions listed on the previous slide, MISP was the solution that provided him the required amount of field storage flexibility, as well as the ability to handle high volume indicators, sharing features, and correlation capability he was looking for.

[1] "Hack.lu 2017 How I've Broken Every Threat Intel Platform I've Ever Had (And Settled on MISP)" - <https://www.youtube.com/watch?v=6k-1QEZFgml>

## Operationalizing CTI with MISP

- A free, open-source analyst favorite
- Tracks "events" - reports, advisories, incidents, etc.
- Identifies commonalities in events and supports automated sharing
- Great web UI and REST API interface
- Classification and sharing functionality
- Flexible indicator storage
- Easy import/export



SANS

MGT551 | Building and Leading Security Operations Centers

108

### Applying CTI with MISP

MISP has lots of attractive features and is a full-featured threat intelligence management tool that competes with most of the commercial options out there. The team behind it has done an outstanding job of keeping it constantly updated with new features and taking input from the community on what should be added. This slide calls out some of the most useful capabilities, and here are a few more from the MISP site.<sup>1</sup>

- An **efficient IoC and indicators database** allowing to store technical and non-technical information about malware samples, incidents, attackers and intelligence.
- **Automatic correlation** finding relationships between attributes and indicators from malware, attacks campaigns or analysis. Correlation engine includes correlation between attributes and more advanced correlations like Fuzzy hashing correlation (e.g., ssdeep) or CIDR block matching. Correlation can also be enabled or event disabled per attribute.
- A **flexible data model** where complex objects can be expressed and linked together to express threat intelligence, incidents or connected elements.
- Built-in **sharing functionality** to ease data sharing using different models of distributions. MISP can synchronize automatically events and attributes among different MISP. Advanced filtering functionalities can be used to meet each organization sharing policy, including a **flexible sharing group** capacity and an attribute level distribution mechanisms.
- An intuitive user-interface for end-users to create, update and collaborate on events and attributes/ indicators. A graphical interface to navigate seamlessly between events and their correlations. Advanced filtering functionalities and warning list to help the analysts to contribute events and attributes.

## Operationalizing CTI with the Zeek Intelligence Framework

- Method of matching observables to network metadata in Zeek logs
- Observables are loaded via plain text tab-delimited files
- Built-in Zeek scripts extract atomic indicators from metadata to send to the framework
- Zeek fires on matches between metadata and observables it has been given



### Applying CTI with the Zeek Intelligence Framework

TIPs are not the only way to operationalize threat intelligence. Zeek is an example of a network security monitoring tool that supports direct application of observables via its Intelligence Framework. The framework is a relatively simple mechanism that allows a Zeek user to load a list of observables (e-mail addresses, IPs, domains, etc.,) and have Zeek alert on any matches in its metadata. The Intelligence Framework includes three functions that make this possible:

1. The intel load function, in which observables can be ingested via plain text, tab-delimited file
2. The "seen" function, where Zeek will extract atomic indicators from its metadata and send it to the framework to be checked, and
3. The "match" function, where Zeek will alert on indicators in metadata that match its list of known malicious observables

If you're running Zeek, the Intelligence Framework can be a streamlined way of ingesting atomic indicators and applying them directly to network data without the need for additional software.

## Threat Intelligence Summary

- Intel is an important input *and* an important output of the incident response process
- Collaborative process that includes producers and consumers
- Begins with the right questions -> solid requirements
- Good mental models exist for producers and consumers, find what works best for your team
- Continuous integration via agile processes and technology can help us maximize value

### Threat Intelligence Summary

Intelligence is an important input and output of the incident response lifecycle, informing everything from preparation to infrastructure hardening to automated detection and threat hunting. CTI collection, processing, and dissemination are collaborative processes that include producers and consumers. Your SOC will likely be more on the consumers side, though you may be doing both if the SOC includes a formal CTI function.

Wherever it's coming from, good intelligence begins with solid requirements that ask questions that are specific, event-centric, relevant, actionable, and time-oriented. There are plenty of mental models that can help us produce and consume actionable intelligence, including the Diamond Model, Cyber Kill Chain, and the Active Cyber Defense Cycle. Operationalization of intelligence in the SOC tends to be facilitated using standard data formats and threat intelligence platforms or IOC ingestion. Using these technologies to continuously refine and apply intelligence in near real time keeps management overhead low and ensures that we can maximize the value of the intelligence we receive.

# Course Roadmap

- Book 1: SOC Design and Operational Planning
- *Book 2: SOC Telemetry and Analysis*
- Book 3: Attack Detection, Threat Hunting, and Triage
- Book 4: Incident Response
- Book 5: Metrics, Automation, and Continuous Improvement

## SECTION I

### Introduction

#### Mindset and Preparation

- Cyber Defense Theory and Mental Models
- SOC Data Collection
- Other Monitoring Use Cases
  - *Exercise 2.1: Attack Path and Data Source Assessment*

#### Collection and Monitoring

- Using MITRE ATT&CK to Plan Collection
  - *Exercise 2.2: Prioritizing and Visualizing Attack Trees*
- Cyber Threat Intelligence
  - **Exercise 2.3: Writing Priority Intelligence Requirements**
- Practical Collection Concerns
- Prevention and the Future of Security
- Summary and Cyber42 Day 2



This page intentionally left blank.

## EXERCISE 2.3

# Exercise 2.3: Writing Priority Intelligence Requirements

### OBJECTIVES

- Capture key questions based on adversary research and defensive planning
- Consider leadership concerns in CEO's Intelligence Requirements
- Develop Priority Intelligence Requirements
- Validate, apply and provide feedback on the intelligence your team receives



### Exercise 2.3: Developing and Implementing SOC Playbooks

Please go to Exercise 2.3 in the MGT551 Workbook or virtual wiki.

# Course Roadmap

- Book 1: SOC Design and Operational Planning
- *Book 2: SOC Telemetry and Analysis*
- Book 3: Attack Detection, Threat Hunting, and Triage
- Book 4: Incident Response
- Book 5: Metrics, Automation, and Continuous Improvement

## SECTION I

### Introduction

#### Mindset and Preparation

- Cyber Defense Theory and Mental Models
- SOC Data Collection
- Other Monitoring Use Cases
  - *Exercise 2.1: Attack Path and Data Source Assessment*

#### Collection and Monitoring

- Using MITRE ATT&CK to Plan Collection
  - *Exercise 2.2: Prioritizing and Visualizing Attack Trees*
- Cyber Threat Intelligence
  - *Exercise 2.3: Writing Priority Intelligence Requirements*
- **Practical Collection Concerns**
- Prevention and the Future of Security
- Summary and Cyber42 Day 2



This page intentionally left blank.

## Practical Collection Issues Overview

- Defining collection goals
- Tactical collection
- Audit policy flexibility
- Multi-stage collection
- How to choose the highest-value sources
- Collection methods and filtering options
- Ensuring data quality



### Practical Collection Issues Overview

A collection system is highly complex and getting it all working the way SOC needs is a small feat itself. Now that we've covered the theory behind what we're trying to do we'll walk through a few of the practical needs of having an effective log and network data collection system. This starts with defining our specific collection goals which help us pick exactly the right number and type of logs to collect. Although this is a great start, audit policy flexibility is key since collection needs may change with short notice. Since as a SOC we must balance cost, efficiency, hardware resources, and most importantly, the ability to detect attacks, understanding what log sources are most important, collection methods, and what to look for to ensure your data is usable once collected must be covered as well.

## Collection Goals vs. Log Volume

- What do you **really** need to collect?
- Consider our **requirements** and **goals**:
  - IOC-based matching
  - Advanced attack detection / threat hunting
  - Audit
  - Compliance
- Collection policy drives **log volume**
  - SIEMs often charge based on rate / volume
  - Being **tactical** about collection == cost efficiency



### Collection Goals vs. Log Volume

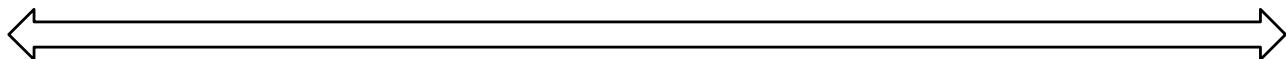
Part of the difficulty of creating a collection policy is making all of the "do we collect it or not" decisions. Developing a list of specific goals for collection can help clarify these questions when deciding whether a certain log source of event will need to be collected. In most cases, the goals you will be looking to meet will be derived from the following:

1. IOC-based matching – The ability to apply tactical level threat intelligence to the events occurring in the environment.
2. Advanced attack detection and threat hunting support – Since these attacks may use domains and IPs that are not known to be malicious, this requires you have a record of *all* external traffic events.
3. Audit – Do you need a record of the actions taken on a system by all users?
4. Compliance – Are you mandated by a compliance framework to collect certain logs from specific systems?

What you decide to collect will determine your log volume and your log volume will drive how much you spend and how fast you can search your SIEM. Therefore, when choosing what to collect, the answer should *not* be "collect it all" (unless you have unlimited licenses and hardware), you must choose what is most important. Even within compliance frameworks, there is often room to argue whether something really needs to be collected or not, do not simply accept that compliance means "full collection of everything possible".

## Tactical Log Collection

- What will your default collection strategy be?
  - Collect everything, just in case?
  - Only collect what we *know* we need?
  - Collect many things, assess for noise, adjust frequently



*Maybe not enough*



*Works for 99% of situations*



*Whoa too much!*

SANS

MGT551 | Building and Leading Security Operations Centers

116

## Tactical Log Collection

There are three common strategies applied for log centralization.

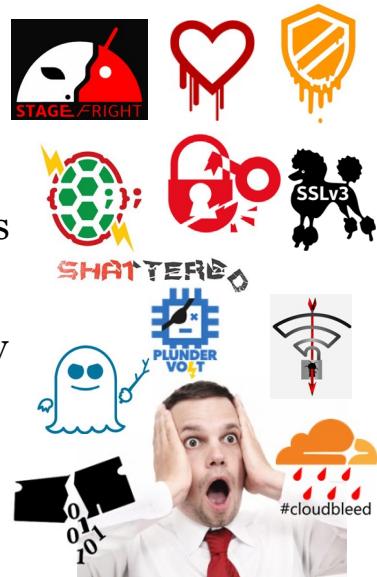
1. Input-driven – The collect it all, log hoarding, "I want everything, just in case" mindset. While this is great when you are trying to find something that you didn't know you needed, it's wildly inefficient cost-wise, and also, will bog down your SIEM with lots of unused and unnecessary data.
2. Output-driven – The use-case driven, "I only collect it if I know I need it" option. This is the most cost-effective and high performing way of collecting logs, but also takes a lot of thought up front to decide exactly what you need.
3. Hybrid collection – Start with an input-driven approach, but rapidly hunt down and weed out noise to a reasonable level. Items that are high volume *and* likely not useful should be turned off, low volume potentially useful logs may be left on.

We (the SANS Blue Team Operations curriculum) recommend the hybrid approach since it emphasizes tactical collection and balances performance, and cost. Note that this approach also means your auditing and collection strategy is constantly in a state of flux.

*Note: For those interested into diving deep into exactly what is and is not needed, Justin Henderson's SEC555 – SIEM with Tactical Analytics course is a great class for SIEM engineers and SOC analysts alike, and the hybrid approach is what is suggested in that course in order to stay nimble.*

## Audit Policy Flexibility

- Fact: Exploits are unpredictable
  - Commonly show up by surprise
  - Have potentially high impact
  - To detect, may require collecting of new events
- Keys to success:
  - Centrally managed log agents, collection policy
  - Fast approval process for changes
  - Speed of change deployment
- Ultimately: Speed up OODA loop



### Audit Policy Flexibility

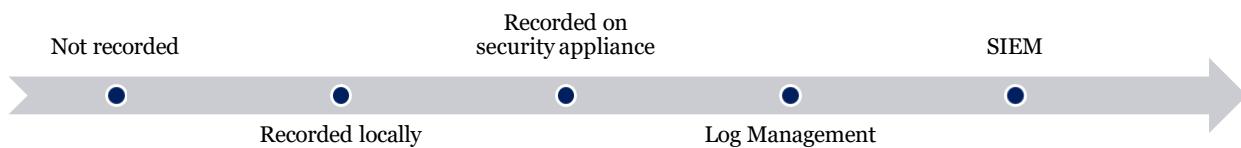
Why does your auditing policy need to be in a constant state of change? Do any of the logos on this slide look familiar? We all know exploits and attacker capability change at lightning pace and, considering the OODA loop model discussed earlier, we must be able to keep up to succeed. If our attackers are picking up and changing their capabilities within 24 hours, we need to be able to modify our collection policy within the same timeframe in order to keep up and defend. Therefore, your collection policy should be nimble and easily changed. This means:

- Having a deployment using some sort of centrally managed policy that, with proper permission can be modified en masse.
- Having a fast-track method of pushing changes in emergency situations
- Having those changes take effect as quickly as possible once pushed
- Having these items under control will ensure you continue to have an OODA loop iteration pace that at least matches with the attacker and gives you a chance to successfully observe and orient to their rapidly changing attacks.

Wondering what some of these logos are? Check out <https://io.netgarage.org/logo/> for a list of branded and named bugs that have been released.

## All or Nothing?

- **Auditing and collection is not "all or nothing"**
- Centralization is best, but we don't have room for *everything*
- There are a spectrum of other options available
  - Bring to log management solution
  - Record on a point product security appliance only
  - Record locally on the device, reference when needed



### All or Nothing?

So, if a tactical collection is the goal, are the only options "centralize it or don't?" Of course not. There are some log sources that, as much as we wish we could centralize them, they are just high volume and there's nothing we can easily do about that. One example of this is PowerShell logs. Attackers love PowerShell because it's not often recorded. Why? Because it can be very high volume. To compensate for this, FireEye recommends in their excellent article on how to log PowerShell<sup>1</sup> that organizations centrally log *some* specific events, while others stay audited, but logs stay local to the device. With this strategy, the centralized logs ideally give you the tipoff that something suspicious has happened and will summon a defender to the endpoint where they can gather additional, high-detail data. While this is just one example, remember there are other places you can store logs if you can't afford to bring them all to your SIEM. If you have a dedicated log management solution, space on your security point products, or your hosts, store additional data in these locations. At least the data will be there when you need it.

[1] [https://www.fireeye.com/blog/threat-research/2016/02/greater\\_visibility.html](https://www.fireeye.com/blog/threat-research/2016/02/greater_visibility.html)

## How to Collect?

### Options for Log Collection:

- **Easiest:** SIEM Agent
- **Most customizable:** SIEM Agent / Third-party agent
- **No agents allowed:** Built-in OS forwarding
  - Windows: Windows Event Forwarding<sup>1</sup>
  - Linux: Syslog Daemon (Rsyslog, Syslog-NG)
- **No extra processes running:** Agentless collection

### How to Collect?

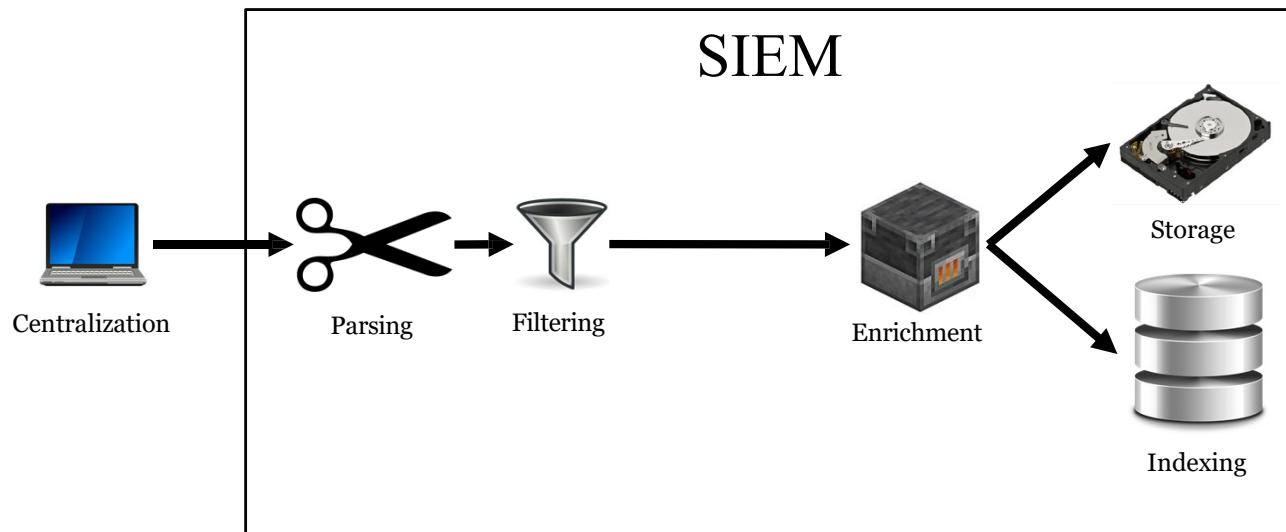
There are numerous ways you may choose to send your logs from the device they are generated on. Which one is the best? The answer depends on what you are looking to optimize for.

The fastest and easiest method is usually a SIEM agent. Most SIEMs come with their own agent that can be installed on each device and customized for collecting each required log source. This usually leads to a fast setup, deployment, and easy centralized configuration. The possible downsides to packaged agents are that they may not perform as well as you like, have advanced features, allow filtering or enrichment, or connect to 3<sup>rd</sup> party tools like log brokers. If you have specialized needs, you may want to go with a 3<sup>rd</sup> party log agent that provides them. There are both free and commercial third-party agents that can provide the additional features and performance you require. One suggestion that seems to have a great feature set and comes in a free and enterprise edition is NXLog ([nxlog.com](http://nxlog.com)).

Outside of specialized agents, you may also choose to use the method built into the operating system. For Linux/Unix this will be the syslog daemon, for Windows – Windows Event Forwarding<sup>1</sup>. Assuming the built-in method supports the log source you wish to send, this can be a great option when vendor, compliance, or organizational requirements disallow the installation of additional software on a system. Since these are a feature of the OS that simply must be turned on, they often offer the path of least resistance to centralized logging where technical requirements or politics get in the way. If even these are out of the question, you can always set up agentless pickup via a remote system that periodically logs in and grabs the required information.

[1] <https://docs.microsoft.com/en-us/windows/security/threat-protection/use-windows-event-forwarding-to-assist-in-intrusion-detection>

## After Centralization



### After Centralization

Now let's dive into what happens to your data once you've set up your logging pipeline to send the data to a centralized location (likely your SIEM). Inside the SIEM there are separable functions that need to occur as well, each determining how well the following can do its job. Although each SIEM might do this in a slightly different order, the concept stands that each of these steps must occur at some point.

- Parsing – Breaking the logs into their individual constituent fields that hold the data of interest
- Filtering – Determining whether or not that log is of interest and should be stored or thrown away
- Enrichment – Making the logs better by correlating with previously received logs or external data
- Indexing – Indexing the individual log entries by chosen fields for quick retrieval by analysts
- Storage – Storing the raw log as it was received

## Parsing

- Yes, you're *collecting* your logs...
- But can you *understand* and *use* their content?
  - **Data quality** will determine their usefulness
  - **Correct parsing** of all relevant data is required!
- **Unparsed data is unusable data!**
  - Can't be filtered based on field values
  - No alerting – fields can't be matched
  - No hunting – field values cannot be grouped / sorted
  - VERY slow SIEM searching – can't index fields

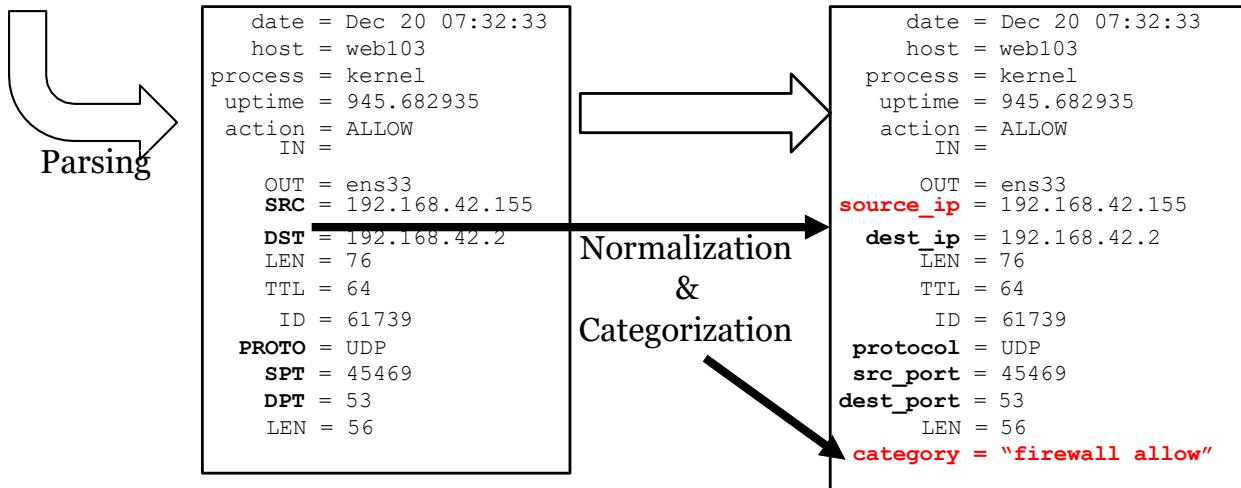


## Parsing

You've set up a great log collection pipeline, fantastic, now it's time to use that data. Now on to the next consideration – how usable is the data you've collected? A log can be stored in a SIEM as either a blob of uninterpretable data, or a beautifully parsed out specimen that has been categorized and with all fields parsed and normalized. Even the most perfect collection scheme in the world is useless if you can't do anything with the data, so ask your analysts and SIEM engineers how well the data they receive is parsed. If your data is not parsed correctly, your SIEM can't understand it (which means filtering at the SIEM won't work), analytics can't trigger on it, analysts can't use it for hunting, and the SIEM will be storing it as raw text (which means searches will be SLOW). Once you've set up collection, ensure your log parsing is in order, otherwise, all steps down the line will suffer.

## Categorization and Normalization

```
Dec 20 07:32:33 web103 kernel: [ 945.682935] [UFW ALLOW] IN= OUT=ens33 SRC=192.168.42.155  
DST=192.168.42.2 LEN=76 TTL=64 ID=61739 PROTO=UDP SPT=45469 DPT=53 LEN=56
```



### Categorization and Normalization

Once logs are parsed, another key operation most SIEMs perform is some type of categorization and normalization.

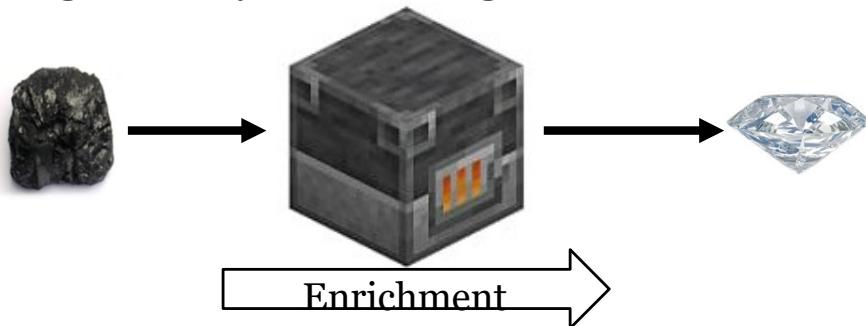
Categorization is important because it enables your team to identify events of the same conceptual type that may have very different formats and data sources. For example, without categorization, to search for logins across Linux and Windows for the same username you would likely need to put in search parameters for the way Linux login events look with an OR condition to also match how Windows events look. Categorization applies a “tag” of sorts to label both events as a “login” so you instead can simply search for “login” AND “username = bob” and find all instances of Bob logging in on any device. Especially with new analysts that may not understand how all common events look, categorization is important in ensuring all data analysts search for is found.

Normalization ensures field names that mean the same thing are searchable with a single term across all data sources. Your firewall might call the source IP field “src.ip” while your HIDS may call it “source\_ip”. The SIEM should understand both data sources and convert the source IP field from whatever it’s called in each source to a standard name. This allows analysts to search source IPs across all data sources without having to use the two different individual names originally used by the differing log sources. This is another crucial step to ensure analysts are finding all applicable data and should be applied, at a minimum, to all commonly used fields. Many times, this will automatically occur for you when using a SIEM’s built-in log agent and ingestion process.

## Data Enrichment

### Key Detection Concept: Log enrichment

- Provided by SIEM correlations / external data lookups
- Improves the quality of your logs
- Enables high-fidelity rule writing



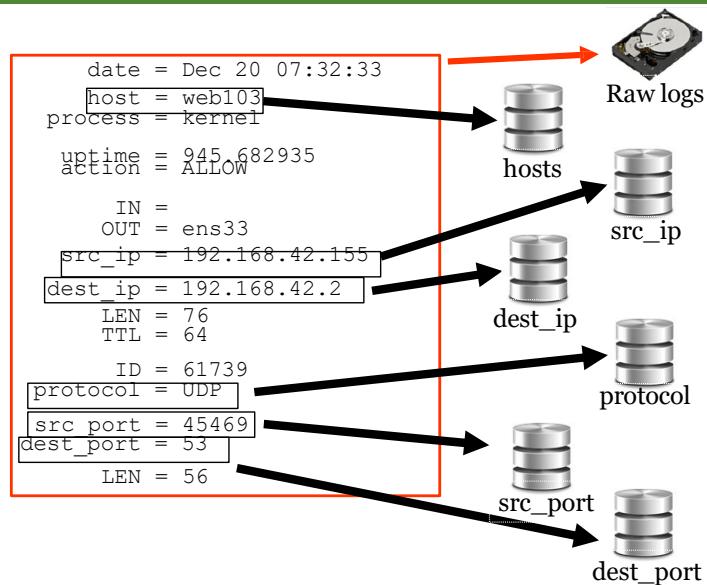
#### Data Enrichment

One key concept we must introduce at this point is the idea of data enrichment. Although we will leave the in-depth discussion for a later module, data enrichment is the step that takes logs that may be somewhat sparse or lacking in crucial detail and uses the SIEMs capabilities to make them better. This often involves external API calls, table lookups, or multiple data source correlations that take the information from the incoming log, and supplement it with additional context. Enrichment helps us take event logs that might not otherwise be actionable and turn them into a highly detailed threat hunting and analytic supporting data source.

## Storage and Indexing

Upon ingest:

- Raw log stored in full
- Fields put into database
  - Enables quick retrieval
  - Costs CPU/HDD/RAM
- Select only common search fields if low on resources
- Analysts know which fields are indexed for fast search



### Storage and Indexing

The final consideration from the collection phase diagram to consider is the storage and index phase.

Conceptually, each log that is ingested must have the data stored in several different ways. One way is the raw log, which will be stored to a hard disk, and can be displayed when analysts search and receive that log as a search hit. The second way is that some or all fields are indexed into some type of database or other quickly searchable data store. The reason is that when a SIEM user runs a search like “src\_ip = 10.0.0.1”, the database can be quickly queried for that IP, and each row will contain a pointer to the raw log which can then be brought up in full for display.

This functionality brings with it a decision that must be made – for each field that is indexed into a database additional resources are needed. While it would be ideal to have every field in a database, memory and hard drive space must be used to store and search the tables, and CPU resources must be used to create the tables on a continuous basis. Because of this, most SIEMs let the administrator of the system choose which fields are searched commonly enough to warrant the resource usage. Your analysts should know which fields are indexed as including them in searches whenever possible will ensure they are running the most efficient searches possible.

Note there are some SIEMs that do index all fields by default due to their underlying storage architecture – users of SIEMs based on Elasticsearch (the Elastic stack, LogRhythm, and Exabeam for example) will not need to make this choice since Elasticsearch always indexed all fields in a log by nature. This is something you may want to consider when deciding on a SIEM to purchase.

## SOC Data Collection Summary (2)

- Understand what drives collection volume
- Carefully plan key data sources
  - Strive for **tactical centralization** with added local logging
  - Consider both **network & host data sources required**
- Audit and collection policy should be nimble
- Consider collection agent vs. agentless
- Optimize **parsing, filtering, storage**
- **Enrichment** makes logs actionable
- **Categorize and normalize** for effective searching

### SOC Data Collection Summary (2)

In this module, we covered the key concepts of the collection piece of the core SOC activities. We discussed how collection can and should work, and the necessity to keep a flexible audit and collection policy that can change with the speed of attacks. We considered how to determine what your collection goals should be and some of the most high-value log sources from both the network and host side, and how to map those items to the MITRE ATT&CK framework. And finally, we reviewed what happens to collected logs after they are transported to the SIEM for storage. Although most times these items will be handled reasonably by default, some tweaking of parsing, filtering, and indexing is almost always necessary to optimize for your goals. Enrichment is the key item at this stage that will add value to your logs and improve the function of the SOC down the rest of the functions. While we will dive further into this later, realize that highly enriched and correlated logs make for high-fidelity detections, and that categorized and normalized logs ensure that analysts can find the logs they intend to locate with each search.

# Course Roadmap

- Book 1: SOC Design and Operational Planning
- *Book 2: SOC Telemetry and Analysis*
- Book 3: Attack Detection, Threat Hunting, and Triage
- Book 4: Incident Response
- Book 5: Metrics, Automation, and Continuous Improvement

## SECTION I

### Introduction

#### Mindset and Preparation

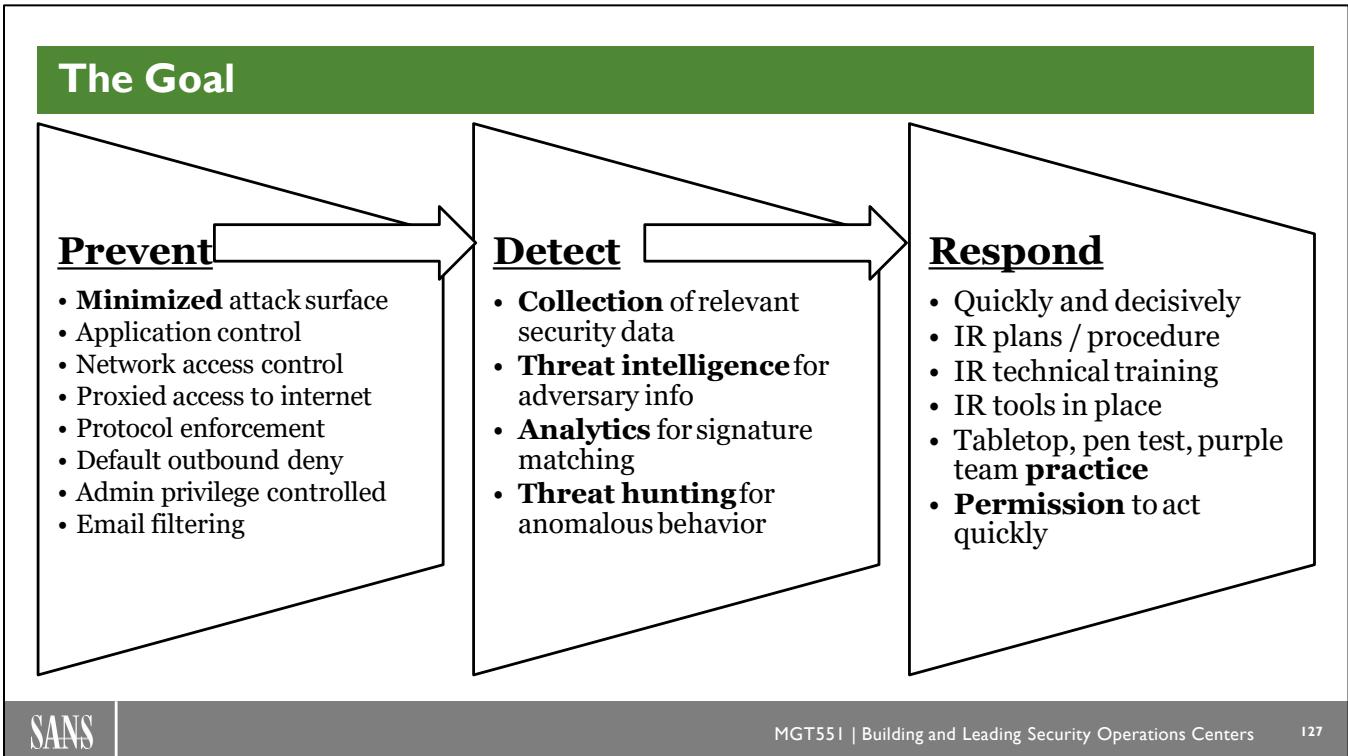
- Cyber Defense Theory and Mental Models
- SOC Data Collection
- Other Monitoring Use Cases
  - *Exercise 2.1: Attack Path and Data Source Assessment*

#### Collection and Monitoring

- Using MITRE ATT&CK to Plan Collection
  - *Exercise 2.2: Prioritizing and Visualizing Attack Trees*
- Cyber Threat Intelligence
  - *Exercise 2.3: Writing Priority Intelligence Requirements*
- Practical Collection Concerns
- **Prevention and the Future of Security**
- Summary and Cyber42 Day 2



This page intentionally left blank.



## The Goal

In security the phrase "Prevent, Detect, Respond" is one that should stick in everyone's head. The first step in any good cyber defense position is filtering out every possible attack before it even starts, that way there will be no mess to clean up. To this end, in this section, we'll be covering some of the facets of a good preventative stance. One that will enable your team to focus on the more important stuff – adversaries that are advanced and dedicated enough to bypass those preventions.

The slide above lists some of the most important parts of each piece of the prevent, detect, respond process. In the prevention area, we are focused on limiting attack surface at both the network and host level. This means not allowing any network traffic or protocols that aren't strictly necessary and monitoring closely and scrutinizing that which is still allowed. On the host level, this means controlling which programs are being run from the start with application control solutions, and monitoring what is left. Remember, a key component of the prevent stage is understanding and internalizing that it *will* fail, which is why it needs to be backed up with a solid detection strategy, and of course a proven ability to respond to what is detected.

## Richard Bejtlich on "Defensible Networks"

### Question: Do you have a fighting chance against attackers?

- Consider the list of what makes a "defensible" network
- How would you rate your maturity level in each of these?
  1. **Monitored** – Can you see what's happening?
  2. **Inventoried** – Do know what's plugged in?
  3. **Controlled** – Can you block / contain issues?
  4. **Claimed** – Do you know who own's it?
  5. **Minimized** – Is attack surface actively minimized?
  6. **Assessed** – Are you sure? Do you test monitoring/analytics?
  7. **Current** – Patching, how fast and complete are you?
  8. **Measured** – Do you have metrics to show effectiveness?

#### Richard Bejtlich on "Defensible Networks"

A classic blog post from Richard Bejtlich<sup>1</sup>, a Network Security Monitoring gives us a useful model to start thinking about the face we present to attackers. As Richard says, if you want to give yourself a chance at resisting intrusion, here are some of the most important variables that will influence your SOCs effectiveness to monitor for and respond to breaches. Consider where your organization is at if you were to do a maturity measurement for each of these items, are there any enormous gaps?

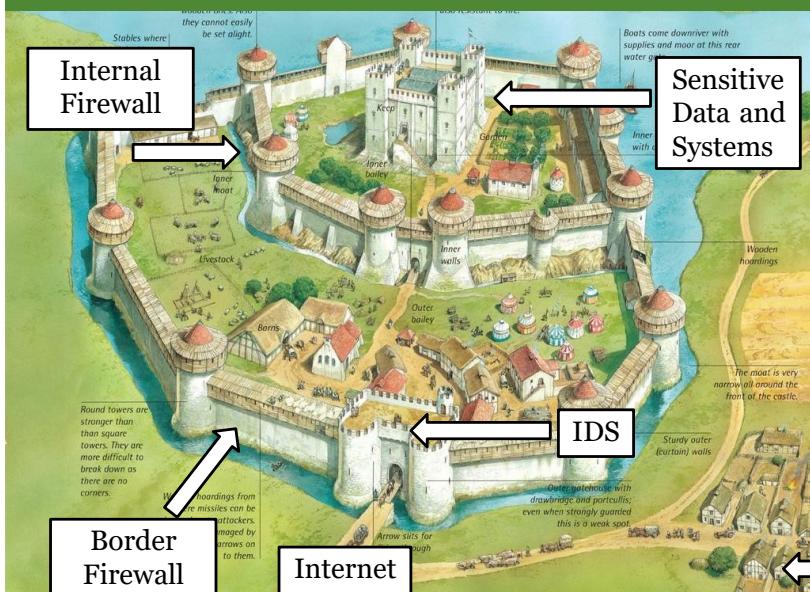
On the flipside, Richard also highlights in a separate post what he calls the "self-defeating network" which consists of a list of the opposite of many of these traits: things are unknown, unmonitored, uncontrolled, unstaffed (focus on products instead of people), etc. Which set of traits do you see more of in your network? The answer to this question may give you the baseline of results you might expect to achieve if nothing changes.

We'll talk about defensible network architecture a bit more on Day 4 when we discuss preparing for incident response!

1 <https://taosecurity.blogspot.com/2008/01/defensible-network-architecture-20.html>

2 <https://taosecurity.blogspot.com/2007/01/self-defeating-network.html>

## The Traditional Network Architecture and Its Failures



Works for medieval towns, not so well for modern IT

Key points:

- Items of importance are *inside* the walls
- Things inside are mostly *assumed* to be safe
- Remote access isn't necessary

How it fails:



Externalized services

### The Traditional Network Architecture and Its Failures

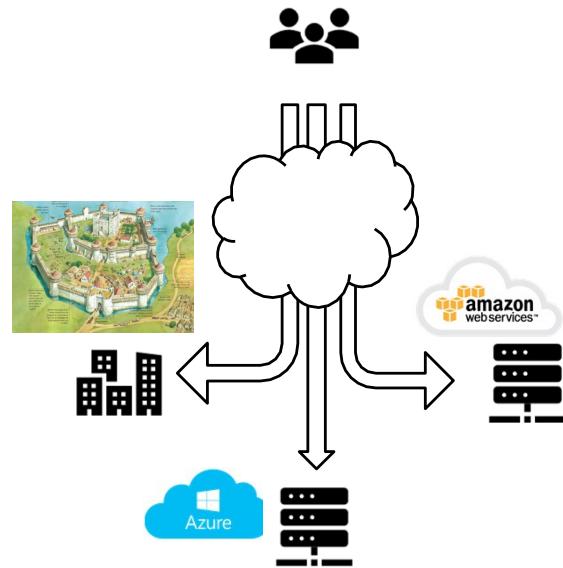
For years we've relied on a perimeter-based model of security setup much like a medieval town. We have firewalls that form the outside wall, keeping the rest of the evil-wishing world out of our personal space. We have externalized services available to the world outside the walls, tools like IDSs and firewalls that control and monitor traffic in and out of our network, and we probably even segment into additional layers inside the network with internal firewalls. In all honesty, it's not the *worst* model, it functions decently well against most types of attacks.

This security model's function rides on some key assumptions which were true of medieval towns, and early networks alike – everything important was inside the walls, things inside the walls are safe, and remote access wasn't necessary. Unfortunately, these things are no longer true of our networks – much of our data is outside the walls in cloud servers, phishing attacks are common and easy, things inside the network are not necessarily safe (because phishing can easily put an infected device "behind the walls"), and we must have remote access. If we look back at famous stories of medieval city security failing, we see the same attack tactics being used. While the independent city of Troy held strong against a 10-year siege attempt in the Trojan war, the Greeks were able to eventually get inside by hiding soldiers inside the trojan horse that got 30 highly skilled warriors behind the walls. The Trojan horse is the perfect analogy to a phishing attack – everything is great, until the attacker finds a way to get behind the walls and cause mayhem from within.

## Modern Network Access

A modern network must solve for security assuming:

- Data held outside the perimeter, on hardware you don't own
- Advanced attackers using phishing, lateral movement, and more
- Near-ubiquitous encryption of protocols, making visibility difficult
- Assumed constant remote access needed to int./ext. systems



### Modern Network Access

A modern network has many "new" challenges that don't equate well to the medieval town analogy. We now have a system in we should start to assume most users will be, or at least could be *outside* the perimeter, working from home, hotels, and airplanes (especially after 2020). While VPNs have enabled this in the past, these technologies were not meant to scale for everyone, and a rethinking of architecture must occur so that remote access is the rule, not the exception. Making this more difficult is the fact that our sensitive data is now held in multiple cloud platforms, not physically on site, and that access to that data (legitimate or done by an attacker) will almost always be encrypted, which make monitoring difficult. We also have legacy systems that were designed and architected for the old perimeter network-style protection that must continue to be accounted for. In other words - the modern network must enable securely working from anywhere and accessing data stored all over the world, while still enabling access to legacy systems and software. It also must still enable security teams for monitoring for anomalies, even in the face of new encryption and protocol standards. Hope you're up for a challenge!

## Closing the Doors of Opportunity – Network Level

- Part of prevention is restricting protocols and traffic
  - Goal: Only **necessary** communications are allowed
  - Everything else is carefully monitored / logged
- Quick check: How close is your network to "least privilege"?
  - Inbound: You've had this blocked for years, nothing new here
  - Outbound from inside: Goal - default *outbound deny*, unless through a proxy
  - Inside to inside:
    - Lateral movement prevented with host/network firewalls or switch VLANs
    - Next-gen firewall rules – are you blocking specific site *features*?
    - For what is allowed – is it tied to an IP addresses or identity?

### Closing the Doors of Opportunity – Network Level

The first step is locking down the network transactions that are allowed to occur to as close as possible to "least privilege". This is a defensive posture that's been best practice for years and forms a great base to add zero trust authentication and authorization on to. The goal is to make it so the only transactions that can even be attempted inside the network in the first place, are ones for which there is a legitimate business need. Of course, this a very hard thing to accomplish, but even partial progress shuts a lot of doors to attackers. To start off, ask yourself:

- Can you get to a default *outbound deny* posture, such that nothing can talk to the internet unless it goes through a proxy?
- Can you shut down lateral movement by denying things inside your perimeter from talking to each other, unless there is a clear business need for traffic to flow?
- Can you and are you shutting down specific features of websites that are high risk? Next-gen firewalls can do it. For example, if you want to turn off file attachments from Gmail, your firewall can facilitate that, shutting one door for exfiltration and insider threat.
- For what you *do* allow, is it tied to a person's identity such that only people who need to take certain actions can take them (not assigned by everyone in a given subnet or with a certain IP?)

Getting these types of preventative controls in place takes away *considerable* opportunities often utilized by attackers for multiple stages of an attack and makes attacker advancement much more likely to be detected as well.

## Closing the Doors of Opportunity – Host Level

- Aiming for least privilege on the host
  - Goal: Limit software to applications on pre-made allowed list
  - Apply this list to executables, scripts, PowerShell, etc.
  - Everything else is carefully monitored / logged
- How close is your network to "least privilege"?
  - Applications: AppLocker or other application control blocking or *at least auditing* what is running
  - Scripting controlled where possible – AppLocker, PowerShell JEA
  - HIDS/HIPS watching for questionable activity
  - Permissions carefully set and tested to prevent privilege escalation
  - Admin groups reduced to minimum
  - Privileged access workstations maintain separation of duties

### Closing the Doors of Opportunity – Host Level

Least privilege must occur at a host level as well, this means that you have followed through on all basic host hardening best practices including:

- Reducing which applications can be run with an application control solution
- Controlled running of scripts with application control and language specific features like PowerShell Just Enough Administration (JEA)
- Have a HIDS/HIPS solution watching for and blocking suspicious activity
- Carefully assigned operating system, file, and registry permissions and privileges to prevent easy privilege escalation by attackers
- Reduced administrative group membership to the minimum possible, and identify when administrative tasks are attempted by those who are not supposed to do them
- Following Microsoft best practice on privileged access and separation of machines for everyday work vs. high-risk system administration (more on this later)

## Zero Trust Principles

ZT addresses the traditional network issues by assuming:

- 1. The network is always hostile**
2. External and internal threats exist on the network at all times
- 3. Network locality is not sufficient** for deciding trust in a network
- 4. Every device, user, and network flow is authenticated and authorized**
5. Policies must be dynamic and calculated from as many sources of data as possible



### Zero Trust Principles

In practice, zero trust is not something many organizations have been able to accomplish yet, but that doesn't mean we aren't headed in the right direction. Like everything in cyber security, zero trust is not "100% on or off", there are stages and places it can be piecemeal applied for *significant* improvements in defensive posture. For most of us, the initial steps will look like implementing much stronger authentication tied to hardware and biometrics. Standards and technologies like 2 factor auth, Windows Hello for Business, FIDO2, and more will be the building blocks of the future secure network, and many of them we're already seeing widely implemented today. What we can say for sure, is that the perimeter model will not work going forward, or at least not *only* the perimeter model. Next generation secure network architecture and remote access solutions combined with new authentication methods tying access to people and their devices, instead of being inside or outside the network, will be one huge step in the right direction.

## Where We're Going

### In theory:

- Academic sense of zero trust, everything is always validated
- Not super practical given current technology

### In practice:

- STRONG authentication based on identity and hardware,
- Every device protects itself
- Throw away or augment your perimeter model
- A very different remote access architecture

### Example: Google's BeyondCorp

- Securely identifies managed devices based on certificates
- Securely identifies users with multi-factor auth and SSO tokens
- Applications accessed through internet-facing access proxy
- Proxy validates user, device, and authorization level for ALL requests
- No trust based on network location
  - No VPNs

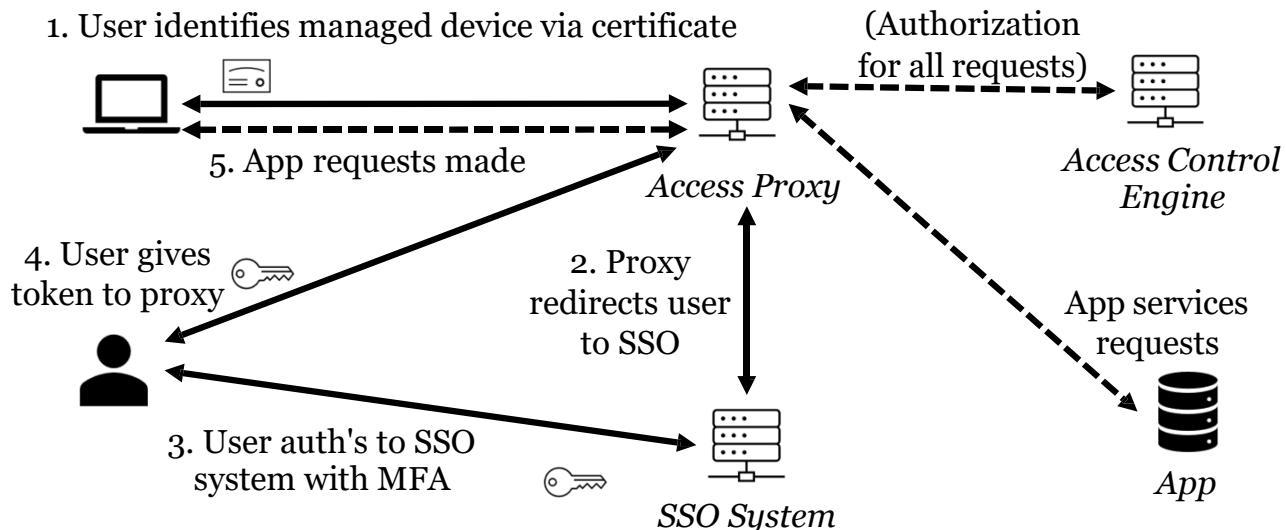


### Where We're Going

In practice, zero trust is not something many organizations have been able to accomplish yet, but that doesn't mean we aren't headed in the right direction. Like everything in cyber security, zero trust is not "100% on or off", there are stages and places it can be piecemeal applied for *significant* improvements in defensive posture. For most of us, the initial steps will look like implementing much stronger authentication tied to hardware and biometrics. Standards and technologies like 2 factor auth, Windows Hello for Business, FIDO2, and more will be the building blocks of the future secure network, and many of them we're already seeing widely implemented today. What we can say for sure, is that the perimeter model will not provide what we need going forward, or at least not *only* the perimeter model. Next generation secure network architecture and remote access solutions combined with new authentication methods that tie access to people and their devices, instead of being inside or outside the network, will be one huge step in the right direction.

Take Google's BeyondCorp setup for example, one of the closest implementations of true Zero Trust out there. Every device and user can be authenticated with strong authentication methods, regardless of the device's location. All access to applications is facilitated through an internet facing access control engine and access proxy which validates and authorizes every single request. This means that Google employees can work in the same way, without a VPN, no matter where they are – at work, hotel, home, or airplane.

## BeyondCorp Zero Trust Implementation Case Study



### BeyondCorp Zero Trust Implementation Case Study

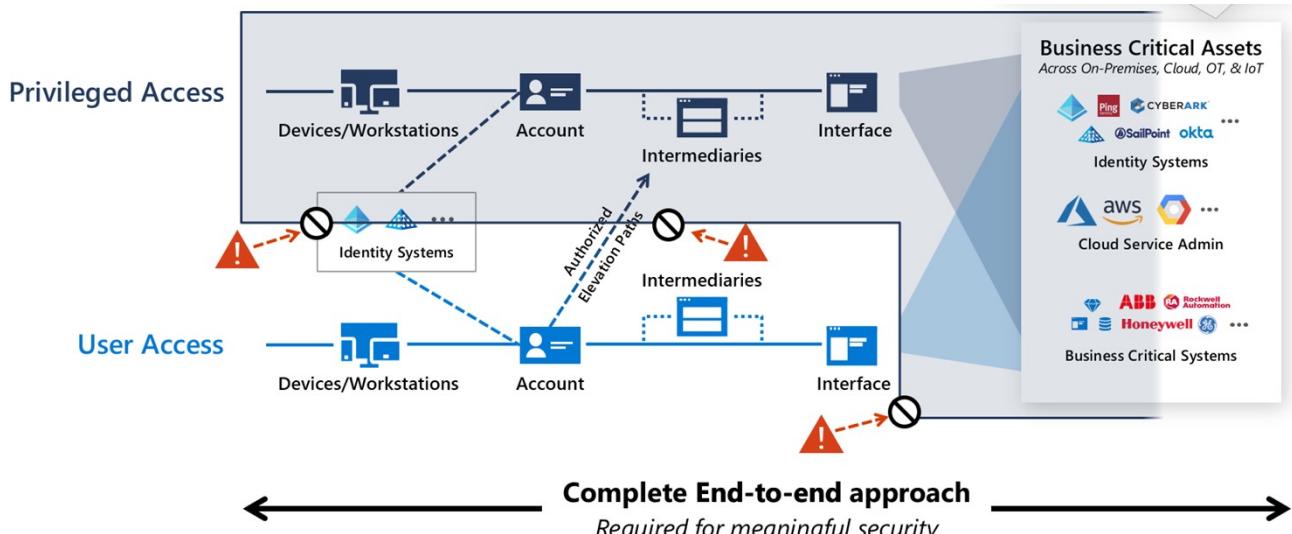
To make a real-life implementation of Zero Trust clearer, let's take a quick look at how this works in BeyondCorp. Understanding BeyondCorp will give you perspective on where security is likely going in the near future for most organizations. If you can apply the authentication and authorization workflow used here in even part of your organization, you will be moving in the right direction.

Assuming a person with an assigned Google managed device wants to access an employee only service, here's what happens, regardless of internal or external network location.<sup>1</sup>

1. The request is directed to an access proxy. The laptop provides its device certificate.
2. The access proxy does not recognize the user and redirects to the SSO system.
3. The user provides their primary and second-factor authentication credentials, is authenticated by the SSO system, is issued a token, and is redirected back to the access proxy.
4. The access proxy now has the device certificate, which identifies the device, and the SSO token, which identifies the user.
5. An Access Control Engine performs the specific authorization check configured for the application the user wishes to access. This authorization check is made on every request.
  - The user is confirmed to be in the appropriate group.
  - The user is confirmed to possess a sufficient trust level.
  - The device is confirmed to be a managed device in good standing.
  - The device is confirmed to possess a sufficient trust level.
  - If all these checks pass, the request is passed to an appropriate back end to be serviced.
  - If any of the above checks fails, the request is denied.

[1] <https://storage.googleapis.com/pub-tools-public-publication-data/pdf/43231.pdf>

## Microsoft Recommended Best Practice<sup>1</sup>



SANS

MGT551 | Building and Leading Security Operations Centers

136

### Microsoft Recommended Best Practice

Microsoft takes a similar approach with their recommended workflow for authenticating users and devices. The picture in the slide above is from their privileged access guidance and shows that whether you are authenticating basic user level access to an application, or high-risk privileged access, both the device and user account needs to be verified with a separate identity system, and an intermediary should be used before the request can be services by the interface for that application. In this case, it also shows the conceptual method for elevated privilege requests through a defined and authorized elevation path from user access to privileged access, facilitated through an intermediary. With both Google and Microsoft in alignment on these basic device and user-centric authentication and authorization principles, you can see the future of zero trust principle-based security is quickly arriving.

Microsoft makes it very clear on the page referenced below when they state:

*"Securing Privileged Access has two simple goals*

*1. Strictly limit the ability to perform privileged actions to a few authorized pathways*

*2. Protect and closely monitor those pathways"<sup>1</sup>*

[1] <https://docs.microsoft.com/en-us/security/compass/privileged-access-strategy>

## Leveraging the Cloud for Zero Trust

Cloud systems like Azure AD are now our main identity providers (IdP)

- A good thing – Microsoft is better at this than us
- Keeps organizations running, even during disruptive crisis
  - During large-scale ransomware attacks, cloud-based providers stay online while those with on-prem only IdP's commonly lose access
- Enables multi-org telemetry to be used for identity protection
- Enables user and device identity-based security
- Reduces reliance on network-based security controls
- Shifts the attack focus from passwords to identity systems themselves!
  - Example: Solarigate brings the world the **first recorded golden SAML attack**



### Leveraging the Cloud for Zero Trust

While this all sounds incredibly complex (and of course it is comparable to not doing any of it), the cloud platforms that so many of us rely on are already started to push these paradigms by default (like Azure AD). We now can easily leverage the cloud as "the one source of truth" for our identity and access management, or at least replicate what's on premise with a backup. With a single, cloud-based authoritative source for identities, security teams can make zero trust principles part of the everyday practice of application access and authorization.

Using the cloud-based PaaS systems for identity management also comes with some side benefits, one being that even if your entire organization is compromised with disk-wiping ransomware, it's likely users will still be able to use applications and systems that rely on cloud-based authentication as it will not be affected (a benefit Microsoft has observed over numerous incidents). It also enables Microsoft to leverage their aggregate telemetry for attacks and suspicious activity seen over multiple tenants to protect your employees and organization better than you could on your own. With a shift to identity-based access control as opposed to network location-based controls, we are ushering in our new zero-trust paradigm.

One final note on this, if network access controls are no longer the main barrier for attackers, then the focus of attacks will undoubtedly start to shift to compromising your identity systems. We have, in fact, already seen this with attacks like the SolarWinds supply chain compromise in late 2020 (aka Solarigate). For certain victims of this attack adversaries compromised the organization's identity provider (IdP) and gained access to SAML token signing private key<sup>1</sup> which allowed them to, in effect, have admin access to all systems leveraging SAML for SSO authentication, bypassing multi-factor authentication. Like a golden ticket attack for Kerberos, the "Golden SAML" attack may become more and more popular as organizations shift towards identity-based security and preventing these attacks will need to be a high-priority focus for security teams in the future.

[1] <https://www.cyberark.com/resources/threat-research-blog/golden-saml-revisited-the-solarigate-connection>

## Resisting Basic Identity Attacks

### 1. Password Spraying

- Attack: Trying a small set of likely passwords against *all* accounts
- Fix: Block weak passwords (Azure AD Password Protection), use multi-factor authentication, **block** legacy authentication methods that don't support MFA

### 2. Credential Stuffing

- Attack: Trying breached credentials from elsewhere on *your* infrastructure
- Fix: Multi-factor authentication, Azure AD risk-based sign-in protection

### 3. Phishing

- Attack: Cloned pages, lookalike domains, and more
- Fix: FIDO2 keys, Windows Hello for Business, Microsoft Authenticator app

### Resisting Basic Identity Attacks

Over the past decade and before, there have been numerous common identity-based attacks that attackers have used with a high degree of success. Things like password spraying, credential stuff, or stealing a password via phishing are some of the primary attacks that were, and still are used. While these may have been complex to fight in past years, new technology for preventing identity attacks has come a long way, but many organizations have no yet caught up. To stop and slow down attackers, and make their attempts much easier to spot, check into this list of technologies and solutions that can either totally prevent some of these issues, or at least make them significantly easier to detect.

- **Password spraying** – When trying a small set of common passwords against many accounts, the strength for the attacker lies in the fact that many people use weak passwords, and no accounts will be locked out if they only try to login a few times per account. How do we counter this in the SOC? Attack both pieces – block weak passwords in the first place. Microsoft Azure AD has a feature called Password Protection that stops users from ever setting a weak password in the first place. Additionally, backing up logins with required multi-factor auth means that any exposed passwords still might work for attackers. Enforcing multi-factor authentication wherever possible and ensuring to **block legacy authentication methods that do not support MFA** is a great step in the right direction.
- **Credential stuffing** – When attackers can source data breaches from other organizations and companies and try those passwords for your employee's accounts, if they used the same password, regardless of strength, this may expose those employees to a breach. How do we stop this? One piece of course is to ensure users can't login with a password alone, the second is detection and/or blocking of anomalous logins. Microsoft offers what they call Azure AD Identity Protection which applies various analytics to detect high-risk or anomalous logins, triggering additional verification or outright prevention for users with leaked credentials.<sup>1</sup>
- **Phishing** – When attackers can directly ask for a password via a lookalike page or domain where a common login page has been cloned, users may type in their password not realizing they are on a fraudulent website. How might we fix this? FIDO2 security keys are one incredible solution that ties authentication to a USB hardware token with a fingerprint reader that's used to activate it. That hardware token can communicate with a web browser and only authenticates if the user is on a site where credentials are saved. If they are on

- g00gle.com instead of google.com for example, the key will simply not respond as there are no credentials set up for that domain. The integration of the hardware with browser software allows this incredibly clever fool proofing to evade lookalike domains, cloned pages, and real-time phishing relaying two factor auth codes as well! How awesome is that?! In 2018, Google claimed their rollout of FIDO2 keys was so successful that they have had ZERO, (read that again, zero, none!) phishing incidents resulting in password breaches after their deployment.<sup>2</sup>

1 <https://docs.microsoft.com/en-us/azure/active-directory/authentication/tutorial-risk-based-sspr-mfa>

2 <https://fidoalliance.org/google-case-study/>

## Build a Strong Identity Authentication Foundation<sup>1</sup>

**Bad:** Password

123456  
qwerty  
password  
iloveyou  
Password1

**Good:** Password and...



**Better:** Password and...



**Best:** Passwordless



### Build a Strong Identity Authentication Foundation

If the future of security is zero trust, and zero trust relies on strong user verification, then put yourself in the best position by choosing the strongest identity verification methods! With some many different options and situations, which are the best? The slide above shows Microsoft's current guidance on the best authentication methods to use when possible. As you can see, single password is the worst, and passwordless authentication based on biometrics and hardware keys being the best because they replace something you know (passwords and that can be stolen), with something you have (hardware device), plus something you are (biometrics) or something you know (PIN tied to a hardware device). These methods dramatically reduce the likelihood and effectiveness of password theft and identity attacks in general, which in the bigger picture dramatically limits the number of attack vectors your adversaries have, and the chance that such an attack would occur undetected. Preventing and detecting attempted identity attacks is a core piece of setting your team up for success.

[1] <https://docs.microsoft.com/en-us/azure/active-directory/authentication/concept-authentication-methods>

## Preventing Trouble Before it Begins

You've heard it before: "*Move fast and break things*"

What developers hear:



What security team hears:



SANS

MGT551 | Building and Leading Security Operations Centers

141

### Preventing Trouble Before it Begins

We've all heard the phrase "move fast and break things", while the spirit of the saying can be a great motivator for productivity, taking it too far can, and many times has, cause significant security issues! In the modern cloud-driven organization, getting security right will revolve around making and sticking to a strategy for keeping tabs on all assets, accounts, and data that are being generated that process information for the organization. Over the next few slides, we'll discuss some of the most important aspects of getting your network, host, data, and account security right that can help place your SOC in a good position *before* problems begin.

## Don't Make it Too Easy

### Preemptively...

- Consider the paths your attackers use to steal sensitive info
- Search for publicized passwords and data using search engines
- Look at public network scan data (Shodan, Censys, etc.,) for your org.
- Perform OSINT recon on your org and sensitive / VIP employees
- Scan social media for oversharing by employees / job postings

### Micro-lab:

1. Open a browser
2. Go to <https://crt.sh> (certificate transparency log search)
3. Type in your org's domain
4. See anything you didn't expect?

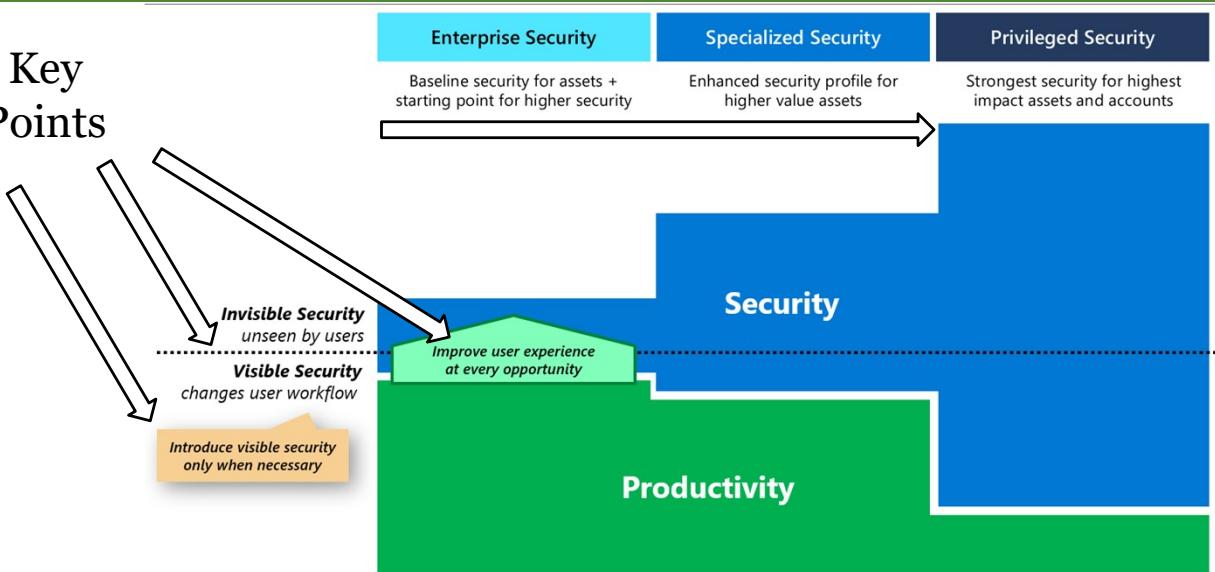
Lists like this make it easy for attackers to find systems you might not want publicized.

### Don't Make It Easy

One of the basic principles in setting up defense is thinking ahead of the attackers, and preemptively testing the same moves they would make before they do. Penetration testing is one of the common ways we do this in an in-depth multi-day scenario, but outside of that, it's very easy to have one of the analysts or other people in the SOC do some basic internet searching for low-hanging fruit. Sites such as Shodan and Censys, which scan the entire internet for open ports and grab banners from available services can quickly highlight mistakes you may have not known about. Data sources like the certificate transparency logs – data provided by certificate authorities on which certificates they issue and to who, can be used by attackers to get a list of potential hosts to attack, even though they may not have even been indexed or seen by a search engine, or mentioned anywhere on your website. While this isn't a class about technical tactics, it's important to have systems in place to stay aware of what your organization has available, and to constantly update and check that data for new items from employees that might have been a bit too ambitious about getting that new service running.

## Microsoft on Balancing Productivity and Security<sup>1</sup>

### Key Points



SANS

MGT551 | Building and Leading Security Operations Centers

143

### Microsoft on Balancing Productivity and Security

Part of the challenge of the SOC is helping the organization find that fine balancing between productivity and security. In an ideal world, we want developers, IT service, and anyone else who's trying to roll out a new capability be able to do so without any perceived roadblocks or interference from security. In the perfect situation, it would just happen in the background, "security baked in", but often this doesn't work as hoped. What we must therefore do, is look at the risk profile of different groups of employees and plan how we can more aggressively (and perhaps slightly more annoyingly) protect users that "hold the keys to the kingdom". This excellent Microsoft graphic from their securing privileged access success criteria<sup>1</sup> helps make these goals clear. It sums up this guidance on hitting that ideal, stating for normal baseline security and users "Introduce visible security only when necessary". As we move into the higher tiers of more at-risk users, security must be cranked up, while productivity inherently will suffer a bit. The interesting distinction they make is that not *all* of that security has to be user visible (the dashed line). As a SOC, the more security controls you can fit "above" the line, invisible to users at any stage, the better.

[1] <https://docs.microsoft.com/en-us/security/compass/privileged-access-success-criteria>

## Convergence of High Security and Convenience

Password + MFA  
where other options  
are not supported

Passwords + 2 Factor  
Authentication

Inconvenient

High Security

Best methods are  
becoming most  
convenient!

Passwordless authentication

Convenient

Passwords

Minimize systems that  
use password only

Low Security

SANS

MGT551 | Building and Leading Security Operations Centers

144

### Convergence of High Security and Convenience

The best of all situations is of course when the highest security solution is also the most convenient, and the good news about new authentication methods is, at least for some cases, this is becoming true! Take this illustration from Microsoft's guidance on recommended authentication methods<sup>1</sup> (with comments added by the authors). As we've seen, Microsoft recommends their passwordless authentication wherever possible as it is tied to a device and to biometrics, meaning it is a great way to verify both the person and device behind a login are what is expected. Tools like FIDO2 USB keys and Windows Hello for Business fall into these categories. These solutions *also* happen to be the most convenient, because for users, they simply look at their laptop camera or put their thumb print on a USB drive and they are immediately authenticated. The rare double win in this case, makes it easier to push organizations to adopt these new technologies as they will not put security in the way or productivity, it will actually enhance both! If this isn't a good reason to move towards new authentication technology, we don't know what is!

[1] <https://docs.microsoft.com/en-us/azure/active-directory/authentication/overview-authentication>

## Prevention and the Future of Security Summary

- Perimeter-based network design is showing its age
  - **Network locality is not a suitable base for trust**
- Cloud services are enabling "zero trust"-based security
  - The new **core of security is device + user identity** and access control
  - User identities are centrally managed, and proven with MFA
  - Managed devices can be tracked
  - Perimeters and VPNs will likely soon be a thing of the past
- Guidance:
  - Prevent attacks where possible, or make them as difficult as possible
  - **Move systems to align with zero trust** principles where possible
  - **Closely monitor legacy infrastructure** built with perimeter security assumptions

### Summary

While the perimeter-based models of security worked ok in many situations, attacks and attackers have evolved to the point where they no longer secure us. Combine new attack tooling with cloud services, and we have a whole new threat model to consider, and a new style of network architecture, assumptions and security to go with it. This section contained some forward-looking material to help familiarize yourself with what's coming up next, if you hadn't seen it before, and hopefully has shown you some of the possibilities when zero trust principles are implemented for a modern, cloud-based organization. The transition from on-premise systems to cloud may take away some of the importance of network-based security but monitoring of the traffic that still occurs will nonetheless remain and be an important part of day-to-day security. Our goal with this section is introduce some of the new mindset for security and give you a familiarity with what is possible so that you can start socializing these ideas within your organization if they haven't taken hold yet.

# Course Roadmap

- Book 1: SOC Design and Operational Planning
- *Book 2: SOC Telemetry and Analysis*
- Book 3: Attack Detection, Threat Hunting, and Triage
- Book 4: Incident Response
- Book 5: Metrics, Automation, and Continuous Improvement

## SECTION I

Introduction

Mindset and Preparation

- **Cyber Defense Theory and Mental Models**

- SOC Data Collection

- Other Monitoring Use Cases

- *Exercise 2.1: Attack Path and Data Source Assessment*

Collection and Monitoring

- Using MITRE ATT&CK to Plan Collection

- *Exercise 2.2: Prioritizing and Visualizing Attack Trees*

- Cyber Threat Intelligence

- *Exercise 2.3: Writing Priority Intelligence Requirements*

- Practical Collection Concerns

- Prevention and the Future of Security

- **Summary and Cyber42 Day 2**



This page intentionally left blank.

## Day 2 Summary

### In this book, we covered:

- Cyber defense theory and mental models
- SOC data collection
- “Specialty” monitoring use cases like insider threat, supply chain, software development, and business e-mail compromise
- Using MITRE ATT&CK to plan collection
- Planning, prioritizing, and visualizing attack paths
- How to write priority intelligence requirements
- Cyber threat intelligence
- Practical Collection Concerns
- The future of prevention and detection - zero-trust and cloud



### Day 2 Summary

In this section, we prepared our environment to be defended. In the SOC, this means assuming the right defensive mindset, planning our data collection, anticipating likely avenues of attack, and asking the right questions to gather intelligence about our environment and the threat landscape. To the extent that you can, this also means influencing your organization to adopt a better defensive posture – whether that means better policies or infrastructure changes to reduce attack surface or better approximate a zero-trust approach. From a SOC planning perspective, you did much of the hands-on work to build a solid foundation by planning and building attack paths, use MITRE ATT&CK as a reference framework for prioritizing your data collection, and write solid intelligence requirements. Whether you are managing a brand new SOC or taking control of an existing SOC, revisiting these elements can make a tremendous difference in the quality and reliability of your detection and response. In the next book, we will shift our focus to threat detection, which will rely almost entirely on the strong foundation we have built today.



# Cyber42 Simulation

## Day 2

### **Cyber 42**

Your instructor will now give you instructions on how to access the Cyber42 game. OnDemand students should refer to their supplemental documentation for instructions for access.

**551.3**

# Attack Detection, Threat Hunting, and Triage



SANS

THE MOST TRUSTED SOURCE FOR INFORMATION SECURITY TRAINING, CERTIFICATION, AND RESEARCH | [sans.org](http://sans.org)

SANS

# 551.3: Attack Detection, Threat Hunting, and Triage

© 2021 John Hubbard and Mark Orlando | All Rights Reserved | G02\_02

Welcome to book three of SANS MGT551: Building and Leading Security Operations Centers!

TABLE OF CONTENTS	PAGE
Introduction	of Class 1
Efficient Alert Triage	4
Detection and Analytic Design	22
Capacity Planning	39
Exercise 3.1 – Capacity Planning	61
Detection Engineering	63
Analytic Frameworks and Tools	82
Exercise 3.2 – Structuring, Documenting, and Organizing Use Cases	101
Threat Hunting	103
Exercise 3.3 – Planning a ThreatHunt	131
Off-Hours Alerting and On-Call	133
Active Defense	146
Summary and Cyber42 – Day 3	161



MGT551 | Building and Leading Security Operations Centers 2

This page intentionally left blank.

## Day 3 Overview

- **Introduction**
- **Creating and Processing Alerts**
  - Efficient Alert Triage
  - Detection and Analytic Design
  - Capacity Planning
  - Detection Engineering
- **Advanced Analysis**
  - Analytic Frameworks and Tools
  - Threat Hunting
  - Off-Hours Alerting and On-Call
  - Active Defense
- **Exercises:** Capacity Planning, Designing Use Cases, and Threat Hunting

### Day 3 Overview

Here is a list of topics we will be discussing throughout the third book of this course.

# Course Roadmap

- Book 1: SOC Design and Operational Planning
- Book 2: SOC Telemetry and Analysis
- *Book 3: Attack Detection, Threat Hunting, and Triage*
- Book 4: Incident Response
- Book 5: Metrics, Automation, and Continuous Improvement

## SECTION I

### Introduction

#### Creating and Processing Alerts

- **Efficient Alert Triage**
- Detection and AnalyticDesign
- Capacity Planning
- *Exercise 3.1 – Capacity Planning*
- Detection Engineering

#### Advanced Analysis

- Analytic Frameworks and Tools
- *Exercise 3.2 – Structuring, Documenting, and Organizing Use Cases*
- Threat Hunting
- *Exercise 3.3 – Planning a ThreatHunt*
- Off-Hours Alerting and On-Call
- Active Defense
- Summary and Cyber42 – Day 3



This page intentionally left blank.

## In This Module

- Alert triage model
- Triage process by SOC staffing model
- Practical considerations
  - Where to triage
  - Triage interface features for efficiency
- Making priorities clear in the triage queue



## In This Module

In this module, we will cover both the theoretical and practical considerations for alert triage including:

- How alert triage fits into the SOC functions model, inputs and outputs
- Different ways triage can be approached based on the SOC staffing model (tiered or tierless)
- Practical considerations for the triage stage such as where to collect alerts for triage and ideal features for the interface
- How to make the priority alerts clear to analysts, an important factor in fast, accurate triage

## The Next Step: Triage and Investigation

- Potential malicious activity items get queued for investigation
- Analysts choose the most important to examine first
- Determine:
  - False positive – Alert is dismissed
  - True positive – Data is validated, incident is created
- **Keys to success:** Speed, accuracy, context, automation

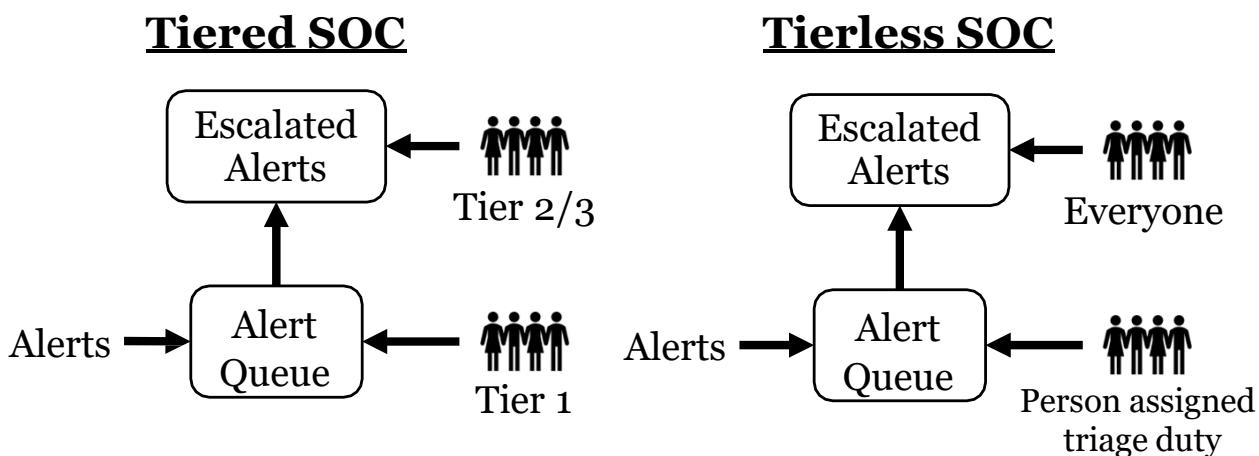


### The Next Step: Triage and Investigation

After your analytics, highlight any host or network traffic that is of concern—the next step down the line is adding the generated alert to the triage queue. In this step, analysts should be accurately able to determine which alert needs attention most urgently so that it can be given attention first. While many alerts may end up determined to be false positives during the initial investigation, validated alerts will continue on to become incidents that either the analyst or a dedicated incident response team might handle. The key to success in this phase is to orchestrate the workflow and help from automation tools such that items get triaged as quickly as possible once they hit the pile. Many SOCs, unfortunately, struggle with this stage, however, due to the high number of alerts coming in, which is why we addressed some of the fixes for that situation in the previous sections. Once an alert *is* generated, however, speed, accuracy, and context are the key items that will keep it moving through the process.

## Tiered vs. Tierless SOC Alert Triage

- Who does triage may depend on your SOC type



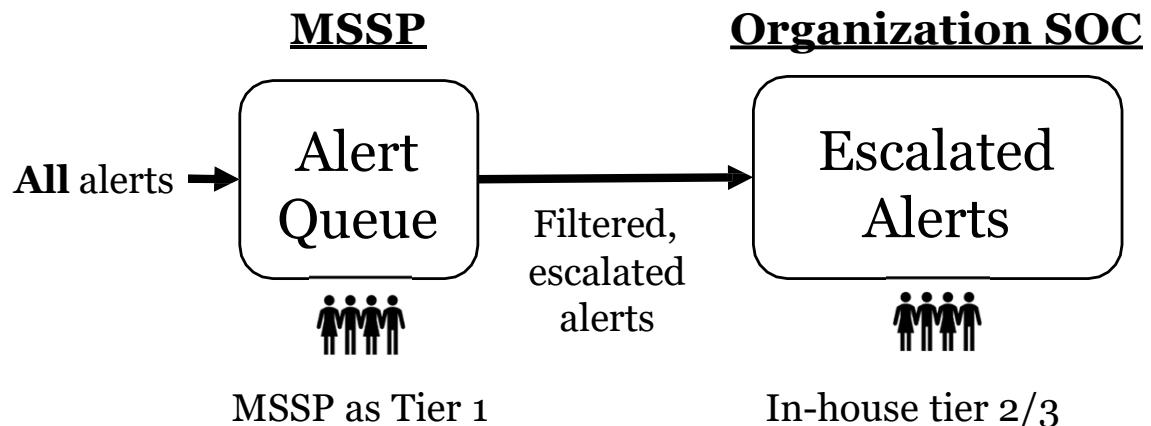
### Tiered vs. Tierless SOC Alert Triage

You're more than likely familiar with the way of doing alert triage in the traditional tiered SOC. In these types of SOCs, there are typically Tier 1 analysts that take the first look at all alerts coming inbound. In this situation, these analysts will review and qualify each alert for validity and toss it out if it's a false positive. If it is a true positive, Tier 1 analysts will then either take on the issue themselves or pass it up the chain for additional, more complex analysis by Tier 2 and above. A positive to this setup is that, in an immediate sense, it is very efficient and cost effective in that those who are capable of doing complex work only see the items that require it. One downside is that if there is an alert that takes complex skill to determine true or false positive, if Tier 1 doesn't escalate the alert, it may get improperly passed over.

In a "tierless" SOC, things are run a bit differently. While there is no single way that tierless SOCs operate, a common model is to select a rotating duty amongst *all* analysts for watching the alert queue and triaging the items that come in. Anything that is determined to be a true positive can either be assigned by this analyst, or the other analysts can select an already qualified alert off the pile. In a tierless SOC, there still may be Sr. positions, but this type of operation gives even those who are new the chance to deal with data they might not see or work with in a tiered SOC, which is great for growth. The downside is that each person must be more responsible and work together even more closely to ensure all the work gets done and that analysts don't start to only cherry pick their favorite type of alert to work.

## Triage via MSSP Hybrid Model

A strategy for only dealing with more complex alerts in-house



SANS

MGT551 | Building and Leading Security Operations Centers

8

### Triage via MSSP Hybrid Model

Another common method used for triage, which optimizes for having your in-house security team only receive high-priority and complex alerts is the MSSP hybrid model. In this model a MSSP is utilized as an outsourced tier 1, which takes on the job of filtering the important items out of *all* the alerts that are generated. This frees the in-house team up to focus on the most important items (assuming they are all identified correctly). Here are some of the biggest pros and cons to consider when going to this model:

Pros:

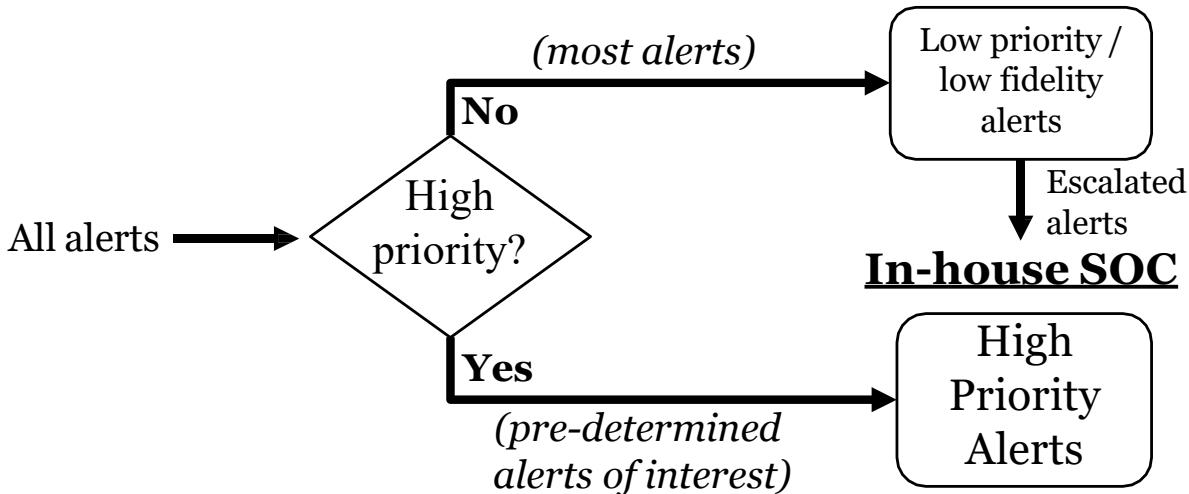
- Most exhausting portions of tiered SOC work are "someone else's problem" leaving your to only need to recruit a smaller group of highly experienced individuals for your team
- Analysts love not having to deal with "the small stuff"
- Potential cost efficiency or enabled of an at least partial in-house SOC for smaller organizations

Cons:

- Trusting your first line to someone else means those with the least context and knowledge of your company (unless you work *very closely* with them) are the gatekeepers to your SOC process
- Do you trust your MSSPs analysts to understand and accurately identify advanced attacks? If not they might be incorrectly marked as a false positive and dismissed allowing the attacker to continue
- Many MSSP analysts, and many organizations with this setup will tell you that the MSSP often doesn't have the same level of information as the organization itself, leading to unnecessary false positives being escalated or even false negatives, due to the lack of context

## Alternative MSSP / In-House Hybrid Model

Mitigating risk of incorrect labeling by MSSP



### Alternative MSSP / In-House Hybrid Model

To address some of the most potentially worrying issues with the "all alerts go to the MSSP" model, your SOC could also use the slight modification to the process on the previous slide as shown above. To address the possibility of high-importance alerts being incorrectly dismissed by the MSSP, each alert type can be pre-determined to be "high-priority" or not, and those alerts that fire from the high-importance category can be dealt with directly by the in-house SOC. Of course, doing this requires your team to do some work ahead of time to determine what is called high-priority (unless vendor defaults are accepted, which may or may not be relevant to your organization).

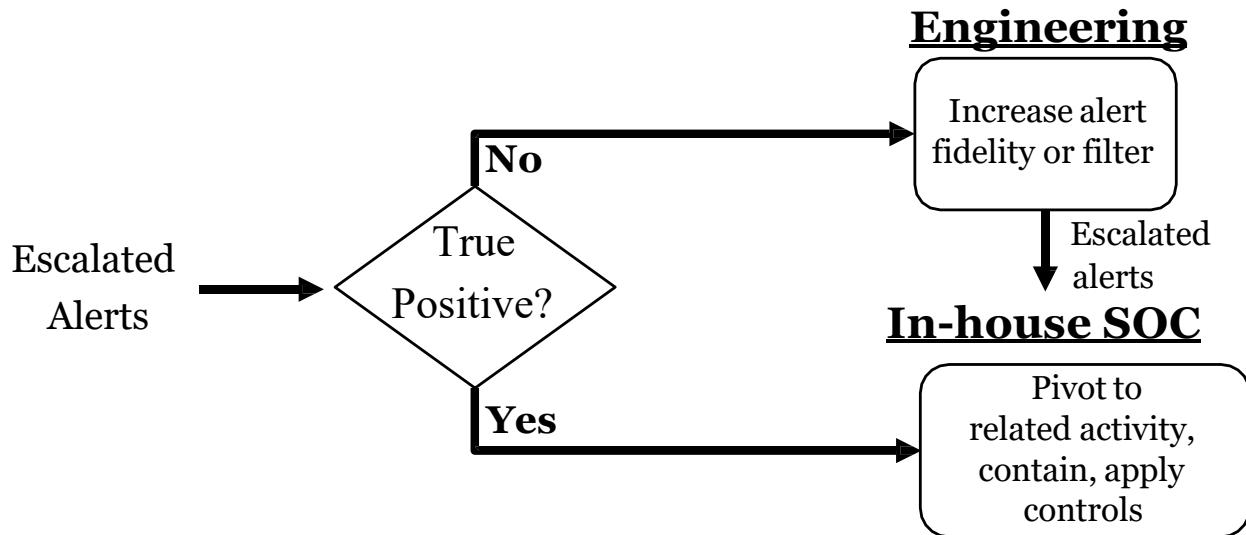
Pros:

- Mitigates risk of the MSSP incorrectly dismissing alerts that are pre-determined to be high-importance

Cons:

- Requires reviewing all alerts to determine a priority before implementation (frontloaded work)
- Analysts will see more false positives if high-priority alerts are not tuned very well
- Most of the cons from the previous slide – not all advanced attacks will set off high priority alerts, meaning you're still reliant on the MSSP to have the talent to identify anomalies that may be indicators of concerning situations

## MSSP Escalation as a Trigger



### MSSP Escalation as a Trigger

Regardless of what function a managed service provider is serving; you should treat any escalations or alerts from that service provider as you would the rest of your alerting pipeline. At the end of the contract term, you and your MSS should be able to trace a narrative of quality improvement in your alerting and reduction in false positives, not to mention actual incidents you've worked together to identify and contain. Consider tracking things like incidents by alert source, incidents by escalation source, false positive rate reduction, true positive rate increase, visibility improvements, and other alert-centric metrics as evidence that your MSS partner is helping to improve your visibility and detection over time.

## Requirements for Accurate Triage and Risk Assessment

- Understanding the alert
  - What it means
  - What attack stage / technique it may represent
  - What it *implies* (the ability to "read between the lines")
  - What data to use to investigate and validate the alert
  - How to run the searches required for investigation
- Asset and user context
  - Is the attack against a key server or administrator?
  - Is the target vulnerable?



### Requirements for Accurate Triage and Risk Assessment

In order to determine whether an alert represents a larger or smaller risk in the quickest way possible, analysts need to hone certain skills as well as have a wealth of data available at their fingertips. When first staring at a list of alerts, ideally your analysts will intuitively understand:

- What the alert means: Do they know the attack or issue referenced by the alert name?
- What attack stage or technique the alert could represent: Can they determine if this is an alert for attack delivery, command and control, exfiltration, lateral movement, etc.
- What it implies: This one is a bit trickier and requires some experience in attack models. Can an analyst read "between the lines" and answer "if this alert were true, how far has this attack progressed?"
- What data to use to investigate and validate the alert: Do they know what the next move is? How to validate the alert is a true positive?
- How to run the searches required for investigation: Even if they know what data they need, are they able to acquire it efficiently from the SOC tools?

Aside from these questions, analysts also need asset and user context—who and what is involved in this alert. Without this information, attacks on all assets and users might be looked at with the same level of priority, which of course is a non-optimal way to assess risk.

## Where To Triage Alerts

### Options for alert triage system:

- Consider queue count, enrichment, integration, workflow, UI



**Option 1:** Intrusion Detection Console

**Option 2:** SIEM

**Option 3:** Incident Management System

### Where To Triage Alerts

There are several ways you might capture alerts and disposition them. Each has their own benefits—consider which may work the best for you given your SOC workflow and available tools. Regardless of the option you select, be aware that tuning should be considered at every step of the alert flow.

One option is at a point product that has done the work to identify the potentially malicious event. The positives to triaging alerts in the product they came from is the interface is best designed to present the data from that specific alert type and the related data is obviously immediately available as well. The cons to doing it this way is even if you triage them in the point product, you will still have another queue to deal with for alerts generated by events flagged as suspicious at the SIEM, meaning you have two places that analysts must constantly watch.

Another option is to triage all alerts at the SIEM. Since all alerts from point products should be centralized here and the SIEM will make its own alerts, it makes sense as a "single pane of glass" for alert triage. The biggest positive of using the SIEM for alert triage, and this is a big one, is the ability to enrich the data from the alert with the additional context from the SIEM. As mentioned before, this can make the difference between an obvious false positive and wasted time. The downside to triaging alerts in the SIEM is that the data related to the alert is held in the remote system. If it linking to the alert can be easily done, this may not be a problem at all, but SIEMs aren't always the best way to display network packets and other binary data.

The third option is to gather all alerts from everywhere in the SIEM, perform enrichment, and then push those alerts on to the incident management system (this could be TheHive, ServiceNow, Archer, or some other such system). The benefit of this option is since ticketing systems are often specifically designed for this type of workflow, they may offer the best UI and user experience. This, of course, is highly dependent on the suite of tools your SOC uses. The downsides of this approach are similar to the SIEM in that you may have to reach back out to other systems for additional data.

## Alert Triage Software and Features

- Desired Features

- Display of metadata for log or packet source
- Signature or analytic that matched
- Easy pivot to PCAP / files
- Easy pivot to external information
- Group and sort
- Mass dismiss / accept

- Examples:

- Open Source IDS-based
  - Squil (SecurityOnion)
  - EVEbox
- Ticketing systems
  - TheHive
  - ServiceNow Security Operations
  - Archer
  - Resilient
  - RTIR
- SOAR tools



### Alert Triage Software and Features

To get the context required presented to the analyst, the software you use to triage your alerts must support the workflow and data display you're looking to achieve. The ideal experience is that well-formatted data enters whatever system you choose to triage your alerts in and is displayed to the analyst in a clear, field-separated manner. The analysts should be able to easily see all metadata, the original packet or log that triggered, and ideally the signature or analytic itself (to understand why the packet or log matched). From this view, the analyst should easily be able to enrich or pivot out to external data with a single click. When the alert is accepted as a true positive, the same fields should be transferred to the incident management system without an extraneous copy and paste or formatting changes required.

Examples of some point product-based alert triage systems that do a good job with these tasks are Squil<sup>1</sup> (which is the default for security onion) and Evebox<sup>2</sup>, which is similar to Squil, but made for triaging Suricata alerts. Both of which give you the immediate option to view packet metadata, full PCAP captures, signatures, enrichment information, and more. Of course, your SIEM interface or ticketing system may be able to provide you with the same experience. The key to success here is no manual process of moving data into different boxes or other non-value-added activity. If your security products do not supply data in the format your SIEM or incident management system wants, interject a step with SOAR or a script that does the work for you so that information flows seamlessly between systems.

1 <https://bammv.github.io/sguil/index.html>

2 <https://evebox.org/>

## Alerts Analysts Must Know to Prioritize (1)

Important alerts for immediate attention:

- Potentially **targeted attacks**
- Exploits against servers with **sensitive info**
- Exploits against servers with matching **vulnerabilities**
- Attacks against safety **critical infrastructure**
- Attacks against servers inside **highly protected subnets**
- Advanced attacker malware and **lateral movement** tools

### Alerts Analysts Must Know to Prioritize (1)

What are analysts looking for that should jump to the top of the priority in the alert queue? There are many items that would indicate an “out of the ordinary” attack or especially risky situation that should grab an analyst’s eye.

- Targeted attacks: If there is any indication of attack customization or specific targeting
- Exploits against servers with sensitive info: This implies attackers know where the important information is and may be getting close to it; it also likely signifies a targeted attack
- Exploits against servers with matching vulnerabilities: Being able to match exploit attempts with the likelihood that they succeeded will help prioritize response when multiple items are attacked at once
- Safety critical infrastructure: Obviously, anything that could cause injury either directly or in the long term via tampering should be at the top of the priority list if it can be identified
- Highly-protected subnets: Attacks that have proceeded far enough into the network that attackers have found the location of segmented, protected networks and are pursuing access imply a targeted attack that is already significantly underway. An exploit against a subnet that is air gapped or segmented should jump to the top of the list
- Lateral movement: Analysts should know that lateral movement detections mean post-exploitation tactics are already occurring and means there is already an incident in progress

## Alerts Analysts Must Know to Prioritize (2)

- Incidents nearing attacker's end goals
  - Exfiltration / theft
  - Data destruction
  - Denial of Service
- How?
  - Knowing your organization's specific **threat model**
  - Understanding **attack tactics** and capabilities
  - Using **data classification** to highlight threats to important data



### Alerts Analysts Must Know to Prioritize (2)

When faced with limited resources, analysts must look for the very worst items first. In this sense, “worst” often means attacks that are very near completion with completion being the achievement of the attackers’ goals. Focusing on targeted attacks here (since commodity malware is often simple and relatively inexpensive to remediate), these goals are often data theft, destruction, denial of service, or some other high cost and impact objective. If analysts see anything that implies one of these is happening or is about to happen, it’s time for quick action. The problem is that since attackers are very tricky, unless an alert says something like “10GB uploaded to a malicious server”, it can be hard to identify these activities straight away.

To give analysts the best chance at identifying these true high-priority items, they must first be familiar with your organization’s threat model and understand what an advanced attacker may want to achieve. Not just the vague idea though, analysts should know what users and systems relate to those assumed goals (which servers and subnets hold sensitive data for example) and have any alerts relating to those items called out in specific detail by the SIEM. If analysts have perspective on how attackers might operate and where they might use those techniques, it best positions us to make the call when an alert may show advanced attack stages. Getting analysts to this state is a combination of data classification, technology that assists in identification, experience within the organization, and possibly penetration test training or other Red Team activity exposure that can show them how a real attacker works.

## Prioritizing Accounts with Microsoft Defender for Office 365

- Prioritization must apply to email triage and more
- Microsoft now supports account tagging<sup>1</sup>
  - Enables *fast* identification of priority account attacks
  - Enables optimization of workflow for high-risk accounts
- **Tag your priority and high-risk accounts!**
  - VIP
  - Administrative users
  - Highly susceptible (based on position - example: Finance)

Priority Account



### Prioritizing Accounts with Microsoft Defender for Office 365

As of mid-2020, Microsoft has renamed their Office 365 Advanced Threat Protection to Microsoft Defender for Office 365, and within that suite, has added some new and *very* useful features. Considering the prevalence of phishing attacks (Microsoft states 90% of attacks originate over email<sup>1</sup>) and the potential enormous cost associated with "business email compromise" style breaches, fast and accurate triage of suspicious email has only become increasingly important over the years. Since so many orgs used Microsoft-based email, it's worth mentioning some new specific features they have added to address these problems.

To assist teams with the triage and investigation in high-risk phishing, Defender for Office 365 now includes Priority Account Protection in Defender for Office 365 Plan 2, including orgs that subscribe to Office 365 E5, Microsoft 365 E5, or Microsoft 365 E5 Security. This feature allows security teams to "tag" accounts they would like to raise the visibility of alerts pertaining to, and have those tags drive highly visible, differentiated workflows within the Microsoft suite of tools. Therefore, if you have this capability, your goal should first be to implement these tags on all user accounts worth differentiating (any more likely to be attacked or have a high-impact attack). This include VIP/C-Suite members, administrative users, help desk, and anyone highly susceptible to BEC style email compromise, such as those in account who can and do move large sums of money.

[1] <https://techcommunity.microsoft.com/t5/microsoft-defender-for-office/announcing-priority-account-protection-in-microsoft-defender-for/ba-p/1696385>

## Leveraging Prioritization in Defender for Office 365

### Optimizing alert **triage**

- Visual tag raises **visibility** of high-risk alerts for fast triage
- Enables **redirection** of high priority alerts to individuals/sub-teams
- Could trigger active **notification** pushed to SOC analysts

### Optimize **investigation** in Defender for Office 365

- Integrates with *submission explorer* to view VIP phishing reports
- *Quarantine* can be filtered by priority tags
- *Threat explorer* highlights alerts related to tagged accounts
- Integrates with *campaign views* to discover attack **trends**
- *Threat protection status report* gives granular **reports** on VIP attacks

### Leveraging Prioritization in Defender for Office 365

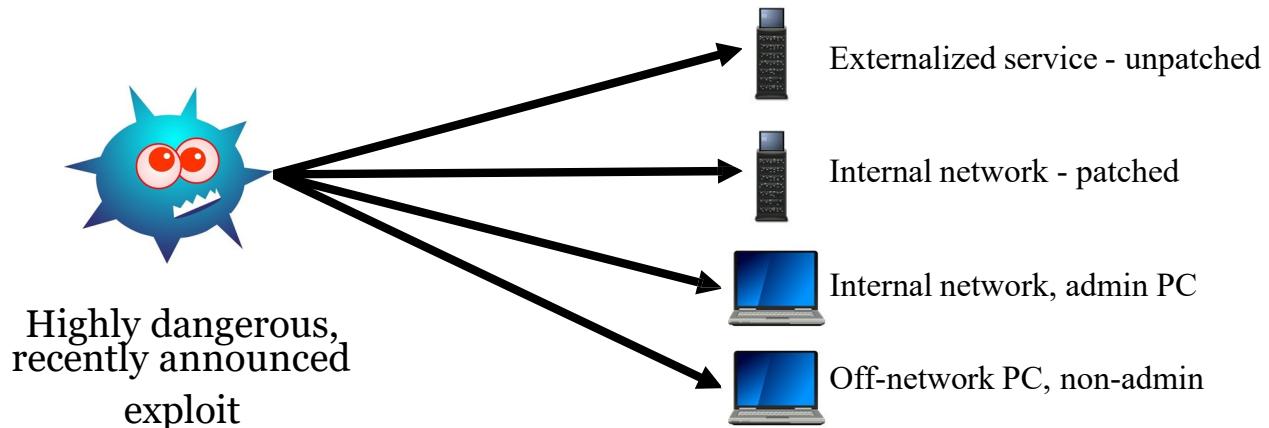
Leveraging these tagged accounts gives security teams numerous benefits in both the triage and investigation stage. During the triage stage, accounts with tags are clearly and visually labeled causing them to stick out in the pile of otherwise non-tagged alerts, drawing teams to them first. These accounts can then have custom workflows associated with them to increase the speed at which they are addressed. For example, a team could individually notify key SOC members that a priority account alert has been received via SMS or email, ensuring a fast response. They could also design a specialized workflow to reroute the working of priority alerts to special sub-teams dedicated to priority account support or to push the alert directly to higher-tier analysts for expert-level triage and investigation. Regardless of the route chosen, you should design the workflow such that alerts related to priority accounts get the "fast track" treatment as any successful attack can have outsized impact in a short period of time.

During the investigation stage, the interface for Defender for Office 365 supports this differentiated treatment through support of priority tags in the various submission queues, quarantine queues, threat explorer and campaign views, and even attack trends and reports<sup>1</sup>. This means any data generated by tagged accounts will have the priority label follow the issue all the way through the investigation lifecycle from email submission to attack trends and reporting. Being able to distinguish attack trends to priority accounts increases your ability to build relevant threat intelligence on those intent on causing high-impact damage to your organization and get in front of the problem before it happens.

[1] <https://techcommunity.microsoft.com/t5/microsoft-defender-for-office/announcing-priority-account-protection-in-microsoft-defender-for/ba-p/1696385>

## For Your Consideration

- What would your alert queue say in the below situation?
- Would it be *immediately* clear to an analyst looking at the alert?



### For Your Consideration

Let's say one day that 4 of your systems are attacker simultaneously – the ones shown on the slide above. If the exploit used in all 4 cases was exactly the same, how would these alerts appear on your triage queue to your analysts? Beyond the type of exploit listed in the alert, which would be no help in differentiating priority in this case, what other information would be either immediately or easily available for analysts? Would they automatically be presented with data that tells them which machine they needed to attend to first? Or would your security tools simply generate 4 alerts that look identical with no additional context, leaving analysts to manually identify which is the most important in a potentially haphazard and non-repeatable way? Hopefully your SOC is closer to the former.

## What It Takes to Get This Right

Alerts should come with immediately available context on:

- **User** – admin, VIP, non-admin?
- **Asset**
  - **Patching status**
  - Type (laptop / on-prem server / cloud / PaaS / SaaS / mobile / ICS)
  - Criticality and Compliance requirements,
  - Software - OS, applications, etc.
- **Location** – *Where* is the device located?
- **Data** involved – High importance?
- Potential **safety** implications (where applicable)

### What It Takes to Get This Right

How do we ensure the alerts on the previous page are identified in the correct priority from the start? By showing key information about the source and destination of the attack to analysts in the alert triage interface (ideally *without* requiring any extra work). If an analyst can clearly identify all the contextual information above about the destination of an attacked asset, they will be in a solid position to pick the most dangerous and important alert in the triage stage, giving you the best chance at stopping an attack early in its tracks.

## Operationalize Data Classification in Alerting

### Data classification is REQUIRED

- Analysts won't remember user / host names
- Your SIEM can, make lists of
  - Important users based on AD group
  - Important assets based on group
  - Label asset by type and location
  - USE THE LABELS in alerting logic - "if [attack] AND ([user] == high-risk OR [asset] == critical, then priority = highest)
- And keep that list up-to-date **automatically**

**TOP SECRET**  
**CLASSIFIED**  
**CONFIDENTIAL**

### Operationalizing Data Classification in Alerting

The SIEM is a key tool that can be utilized to get all the information from the previously slide operationalized and provided to your analysts during the triage stage. Most SIEMs have data labeling and vulnerability tracking capability built-in, but using it relies on telling the SIEM what is important and getting the data important. The main idea is to tell your SIEM which users and assets are on the high priority list and use that logic to increase the priority of any alerts that are generated so that they bubble to the top of the queue. Notice that this requires alerts to be triage at the SIEM or beyond (your IDS will not have user and asset priority lists built in, for example, unless you implement that into the signature themselves).

Therefore, you should strive to develop a process that will identify important users, assets, and information and keep it up-to-date in the SIEM over time. The key to success here is **automation of updates**. For example, having the SIEM monitor the "domain administrators" and "email administrators" groups through active directory so that these users can always be easily identified, even as the groups change. Loading the information at one point in time is better than not having it, but data, system, and user lists will quickly become out-of-date.

## Efficient Alert Triage Summary

- Fundamentals of alert triage and investigation
- Various strategies for alert triage depending on your staffing model
- Triage tools and approaches
- Alerts to prioritize
- Methods of adding context to alert data:
  - Asset status
  - High-profile user accounts in Microsoft O365
  - Data classification



### Efficient Alert Triage Summary

In this section, we talked about breaking down the massive workload that your SOC is likely faced with every day in the form of an alert queue. Whether the queue is in your SIEM, a ticketing system, or a combination of various interrupts, drive-bys, taskings, and indicators, only by effectively processing and triaging this data can you hope to convert the noise into meaningful signal. We discussed the fundamentals of alert triage, tools and approaches regardless of your specific toolset, and various strategies for managing alerts depending on your SOC staffing model. We also talked about the kinds of alerts your analysts need to know how to identify and prioritize in order to respond to late-stage or particularly damaging attacks first. Finally, we went over some specific examples and use cases for adding context to different kinds of alerts to separate them from the noise and ensure they get swift attention – for example, alerts involving high-profile users or sensitive data.

We have started with handling the alerts you have because this is the situation in which you are most likely to find yourselves with new teams, new tools, or environments that have suffered from a lack of alert “TLC”. Now that we have discussed what to do with the alerts you’re getting *right now*, we can move on to designing and creating new analytics that make for more meaningful alerts moving forward.

# Course Roadmap

- Book 1: SOC Design and Operational Planning
- Book 2: SOC Telemetry and Analysis
- *Book 3: Attack Detection, Threat Hunting, and Triage*
- Book 4: Incident Response
- Book 5: Metrics, Automation, and Continuous Improvement

## SECTION I

### Introduction

#### Creating and Processing Alerts

- Efficient Alert Triage
- **Detection and Analytic Design**
- Capacity Planning
- *Exercise 3.1 – Capacity Planning*
- Detection Engineering

#### Advanced Analysis

- Analytic Frameworks and Tools
- *Exercise 3.2 – Structuring, Documenting, and Organizing Use Cases*
- Threat Hunting
- *Exercise 3.3 – Planning a Threat Hunt*
- Off-Hours Alerting and On-Call
- Active Defense
- Summary and Cyber42 – Day 3



This page intentionally left blank.

## Upcoming Detection Topics

- Breaking down the detection function
  - Process diagram and variables to consider
- Analytic creation and outcomes
  - The relationship between false positives and negatives
  - Dealing with high-volume and low-priority alerts
- Analytic frameworks and analysis capture tools
- Threat hunting process and reporting
- Deception and active defense

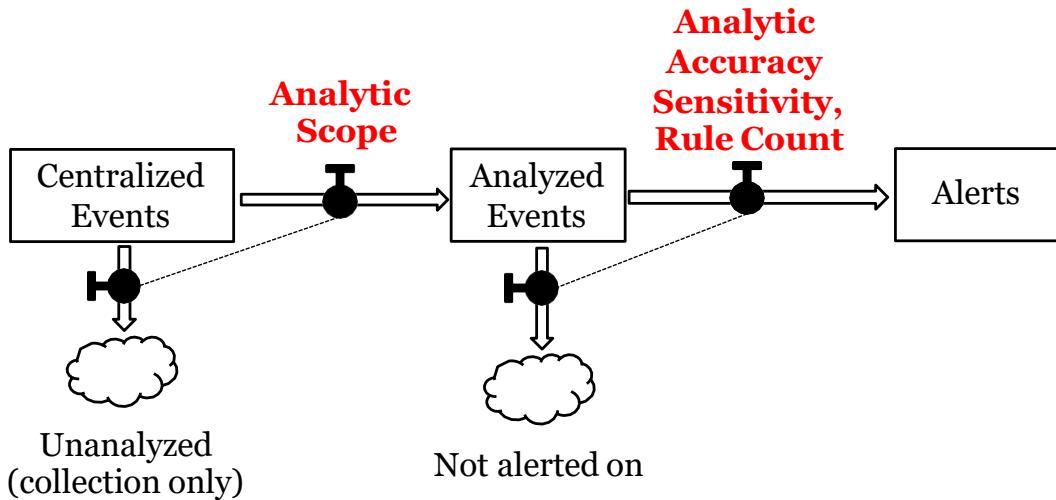


### Upcoming Detection Topics

In the next few sections, we'll cover the detection process in closer detail. The 2<sup>nd</sup> core SOC function is responsible for finding the malicious content in the events that were gathered by the collection system. We'll break down the detection system more granularly (as we did with collection) and look at the individual pieces that can be improved. Since alert tuning is a central struggle for nearly every SOC, we'll cover analytic creation and false positive reduction, digging into the inherent link between false positives and false negatives, as well as how to overcome that struggle as best as possible. We'll also cover use case development and storage for your detection analytics, and deception or "active defense". In addition, we have a dedicated section on threat hunting, both how to do it and how to show that it's having a positive effect.

## The Detection System

How detection works:



### The Detection System

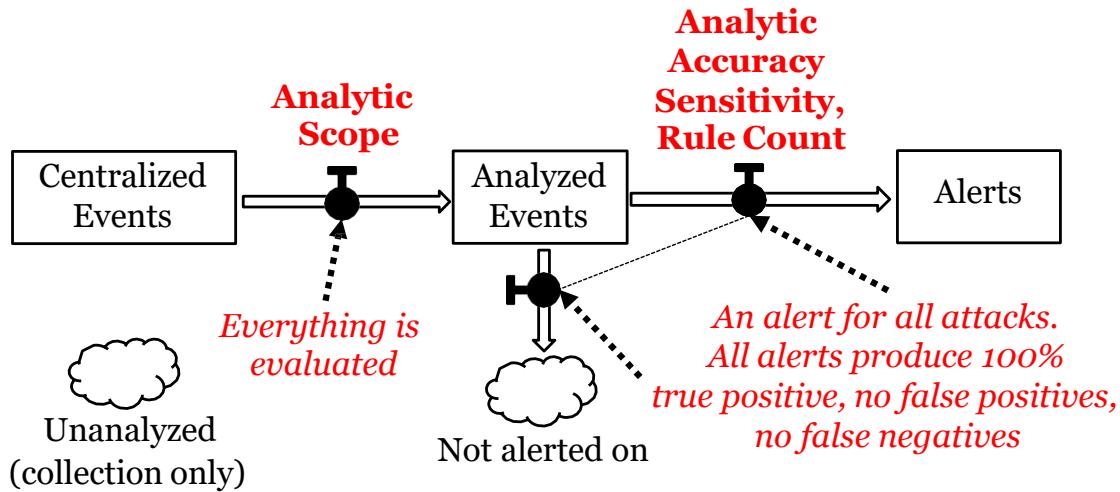
First up is to take a look at the detection system itself and ensure we are all working with the same mental model. This slide starts off with the "centralized events" box that was the final piece of the collection system. You can imagine that every event that was recorded and sent to a network sensor, security appliance, or the SIEM is now collected at this stage and ready for analytics to be applied.

Just because you have centrally collected a log or piece of traffic does not mean that all analytics are applied to it, so the first step to detection is actually submitting the captured event to analytic matching rules. In some cases, data may be only collected, but due to limited hardware resources or SIEM licensing, cannot be analyzed against the ruleset, which means the data will sit centralized waiting for retroactive manual analysis. Therefore, the next piece of confirming our SOC data flow works correctly is applying analytics to as much of our data as possible. If you must make a choice due to hardware or licensing restraints, be certain the logs that are most likely to identify an intrusion receive top priority for analysis.

All events that are analyzed are then subjected to the rules themselves and either produce a match or do not. This diagram shows the control that determines how many alerts will be made is a combination of the accuracy, sensitivity (how easy they are to trigger), and the count of how many analytics are active. Increasing rule count or making them more broadly applicable will increase the *volume* of alerts that your SOC sees – not that this doesn't mean they are true positives, just that more events will initially match the analytics. In the next few slides, we'll discuss the issue of fidelity.

## The Ideal Detection System

How detection works:



### The Ideal Detection System

If we were to design the ideal parameters for this part of the system, how would it look? In an ideal world, every bit of data would be analyzed and potentially capable of producing an alert if it contained signs of a potential attack. Of those analyzed events, we would then apply a perfect set of rules that could, with precise clarity, separate good events from bad. In this scenario, the majority of all recorded events would be benign and fall into the "not alerted on" category. Of course, this is the step where things start to diverge in a big way in any realistic system. Over the next few pages, we'll take a much closer look at the right half of this graph and examine how we can produce the best alerts possible.

## Designing for Reliable Detection

- Detection may or may not be based on a SIEM
- Must support:
  - Reliable and scalable collection and presentation
  - Health and welfare telemetry
  - A mature, repeatable approach to analytic development and tuning
  - Correlation and data enrichment with low admin overhead
  - Abstract (non-technical) requirements from stakeholders outside of the SOC

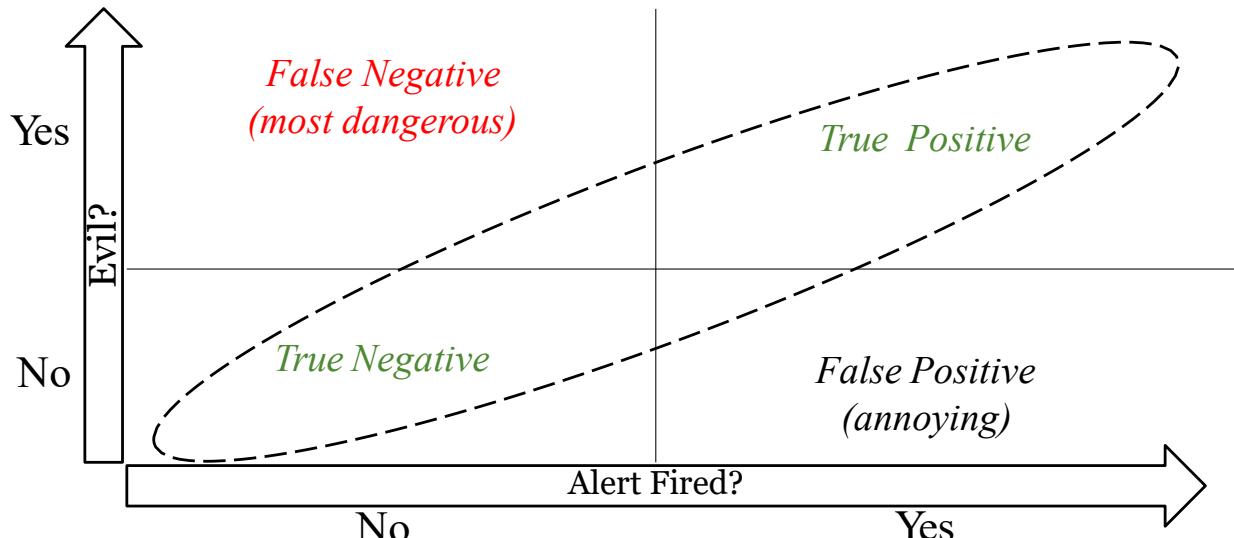
### Designing for Reliable Detection

Yesterday we spent a fair bit of time going over log aggregation, processing, and storage infrastructure. This SIEM function normally represents the “fuel” modern, detection-oriented SOCs run off of; but SIEMs can be extremely challenging to manage and the function they serve is increasingly outsourced or even abandoned for alternate approaches like Netflix’s event pipelines or big data analytics.

Regardless of the underlying infrastructure, there are some commonalities in effective detection solutions; the system must:

1. Be reliable and scalable to meet highly variable logging demands
2. Be healthy, with its own telemetry to indicate that the system is running as expected
3. Support a mature, repeatable approach to analytic development and tuning
4. Perform correlation and data enrichment with relatively low administrative overhead
5. Support analytic abstraction, where non-technical (or at least non-SOC) stakeholders can inject new detection requirements and see those detections implemented with as little friction as possible

## Analytic Outcomes



SANS

MGT551 | Building and Leading Security Operations Centers

27

### Analytic Outcomes

For all analyzed events being run through our analytic rules, there are four outcomes that may occur: true positives and true negatives for when things are correctly detected or not detected, and the bad options—false positives and false negatives. False positives are the bane of SOC analyst's existence and drive most of us crazy on a daily basis. The fourth outcome though—false negatives—is the worst since you now have an attack that you *weren't* alerted about. Obviously, if we had our choice, we would pick to have zero false negatives and minimal or zero false positives. Unfortunately, in a realistic detection system this cannot be done since we do not have perfect and complete information.

## The Tough Detection Decision

- You *will not* have 100% true positives and 0% false negatives
  - But you *can* optimize for zero false negatives OR zero false positives
  - You must decide on your preference, and everyone should understand it
  - Each direction has consequences...

Pros:

- Tolerable queue
- Happy analysts
- No alert fatigue

Cons:

- *High potential for false negatives*



Pros:

- Unlikely to miss attacks

Cons:

- *False positives*
- *Burnout, alert Fatigue*

### The Tough Detection Decision

The problem is false positives and false negatives are inextricably linked, when you dial one down, you inherently dial the other one up. The more sensitive your alert analytics are, the fewer positives you have, if you make analytics less sensitive, you may introduce false negatives.

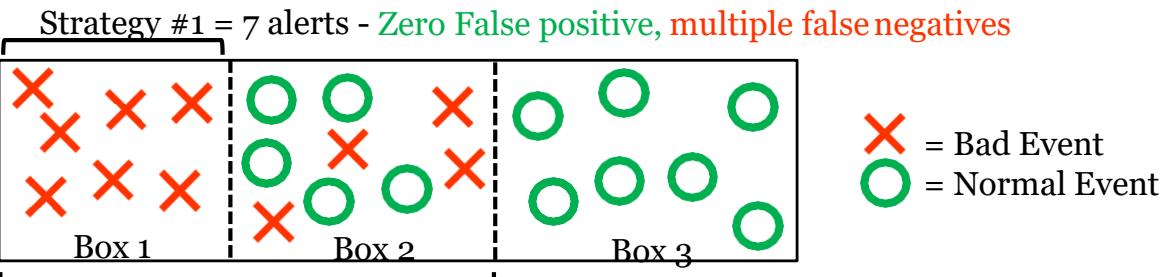
As a SOC manager, your job will be to decide which mix of the two you prefer. You must decide where to set the dial and ensure that your detection engineers and analysts all understand what decision was made and why. If analysts know the analytics were enabled with a strong disdain for false positives, they can enter the investigation of each alert knowing from the get-go that there's a very small chance of a false positive. If, however your alert sways the other way, analysts should know that each alert must be qualified to a higher degree, and that false positives will be more prevalent than true positives.

The extreme end of the choices has predictable consequences, and the decision is not an easy one. In the next few slides, we'll dive into this problem in more detail and explain what can be done to improve the situation as much as possible.

## The False Positive / False Negative Relationship

Unfortunately, most times you can't have both zero false positives and zero false negatives

- Imagine all events in your environment as shown on the bar below
- You must select a place to "set the bar" for alerting
- Imagine an **alert** being sent for every event **left of the arrows** below



Strategy #2 = 16 alerts - Zero False negatives, multiple false positives

### The False Positive / False Negative Relationship

Here's a simple way to visualize the relationship between false negative and false positives. This slide shows a population of events that happened in an organization and divides them into 3 conceptual "boxes":

- Box 1 – Events that are clearly bad and easily identified as such
- Box 2 – Events that are hard to identify as good or bad in a consistent way
- Box 3 – Events that are clearly not of concern, and do not need to be alerted on

Given these 3 boxes, how should we design our alerting strategy? Everyone would agree that we should create analytics that alert anytime a clearly bad event occurs (box 1) and should not alert any time something not concerning happens (box 3). It's the box 2 area that we have the big problem with, and this is the box where you determine the number of alerts you have and chances of false positives or false negatives. If you, as a SOC, decide you want to aggressively eliminate false positives, this would mean taking alert strategy 1, and only alerting on things you 100% know are bad. As a result, you won't receive false positives, but you might introduce false negatives. On the other hand, you may decide "we don't want to miss anything" (false negatives) and therefore you decide on alert strategy #2, which means alert count will be higher, you will alert on all bad things, but you'll also introduce false negatives. Which strategy is most appropriate for your organization? The choice is yours, but in general, most teams would rather have false positives than miss attacks, so strategy number 2 prevails, and thus, SOCs all over the world deal with false positive as part of everyday life.

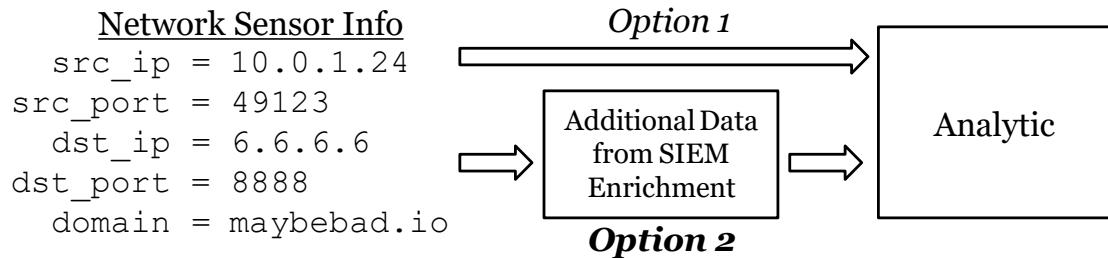
Of course, it's not exactly that simple, this scenario plays out in reality on a rule-by-rule basis, and your approach to any *single* alert can be of either strategy 1 or strategy 2. Your job, therefore, is to make the best analytics across the board that do identify all truly malicious items and *don't* alert on any false positives – cleanly separating good from bad where possible. Unfortunately, in many cases, writing a perfect rule is impossible. Why? Because we do not have complete information and context about every situation. In many situations, we simply will not be able to tell whether something is good or bad without additional human investigation.

## Writing the Best Possible Rules

**Key to success:** Where possible, *apply analytics downstream, post enrichment*

- The more information you have, the better you can separate good from bad
- The **SIEM** is the main tool for adding context and fidelity to events
- Enrichment "separates Xs and Os" on previous slide, making the choice easier

Which of these paths will make it easier to separate good from bad?



### Writing the Best Possible Rules

If writing high fidelity rules then comes down to how well you can cleanly separate good from bad events, how do we improve our ability to do this? In our line analogy, how do we better separate the Xs and Os such that the line we draw will be a cleaner division of the two populations? The way to do this is to use correlation and enrichment from other data sources to add information and context to the data you gather. It is the extra information that you can rely on to improve the accuracy of the analytic. In most SOCs, this will be the job of the SIEM. This slide shows an example.

One way of working alerting, for example, would be to take your Snort IDS sensor and have it send alerts for every single thing that matches one of the signatures and pass it on to the analysts to triage. Many SOCs operate this way and it works, but we can do better. Imagine, if instead of sending those IDS alerts straight to a triage queue, all the information Snort gathered was then forwarded to the SIEM for additional enrichment. Internal IPs in the alert could be cross-referenced against DHCP logs, hostnames, and vulnerability databases, and the information gained could be used to decide whether that alert still makes sense to address or not.

An example: Perhaps someone on the internet tries to exploit your externalized web server with an exploit for Apache. Snort doesn't know whether your webserver runs Apache or what patch level it's on, it merely can detect the attack. If you take the alert and forward it directly to a triage queue, an analyst is going to have to inspect it and the apparent victim system and manually decide whether it is of concern or not. But, what if, instead, you send the alert to your SIEM, and have your SIEM look up if you even run Apache on that system (maybe it's IIS or nginx) and whether that vulnerability has been patched or not? If Snort finds the attack was for CVE-2020-0001 and SIEM correlates that with your vulnerability data for that system and finds that attack is irrelevant to the system, the alert can be safely ignored without any human intervention. This is a simple example but hopefully the point is clear. Instead of alerting directly from the Snort rule that says, "if you see an exploit, alert me", wait until further downstream and change it so that it's implemented at your SIEM as "If Snort sends an alert for an exploit against a system AND that system runs the affected software and version, THEN alert". You've now saved your team a *lot* of work.

## Assessing False Positives and Analytic Accuracy

### Key action: Assessing accuracy of individual analytics

- Periodically run a report for each alert that fired
- How many times did that rule fire?
- How many times was it dispositioned as a true/false positive?

### How to do it:

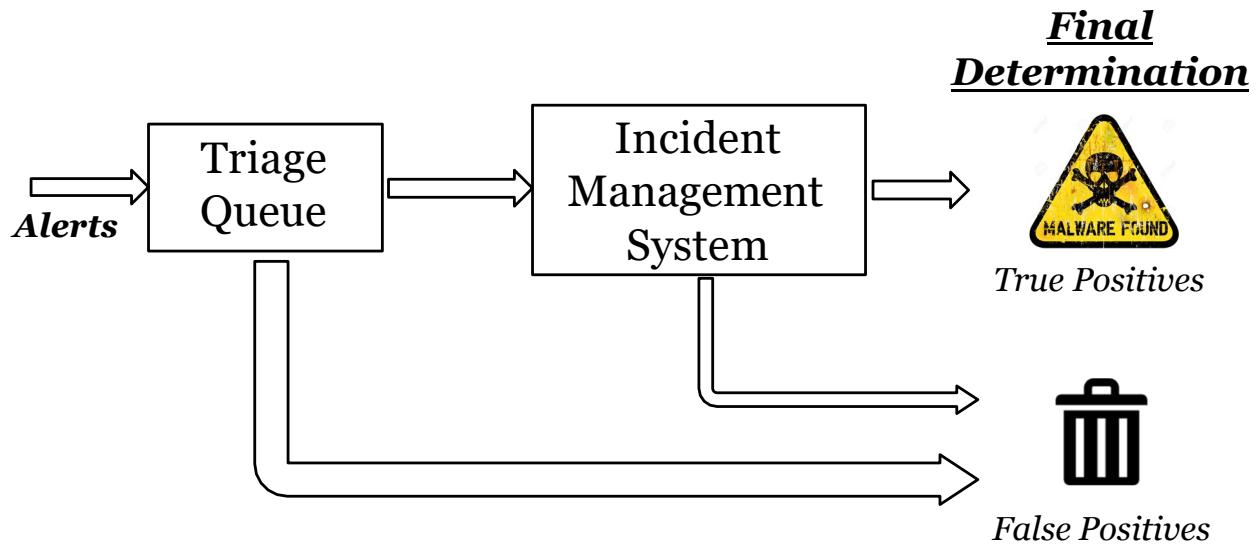
- All analytics have a **unique ID**
- Alert instances should include that unique ID and final disposition
- Once a week – pull all resolved alerts
  - Group by alert ID (gives a count of alerts that fired)
  - Sub-aggregate by disposition (gives percentage of true/false positive)

### Assessing False Positives and Analytic Accuracy

While it's hard to track false negatives, you can and should take metrics on the alerts you produce. Knowing the accuracy and count of each of your analytics is a key activity in keeping sanity in your SOC. To achieve this, each analytic you have should be identifiable with some sort of unique identifier (or at least a unique alert name.)

When your analytics do match, an alert is created with this unique ID and assessed by an analyst as being a true or false positive. After perhaps a week of this activity, you should be able to programmatically look back at all the fired alerts and create metrics on how many alerts per analytic were generated, and how many of them ended up being a true vs. false positive.

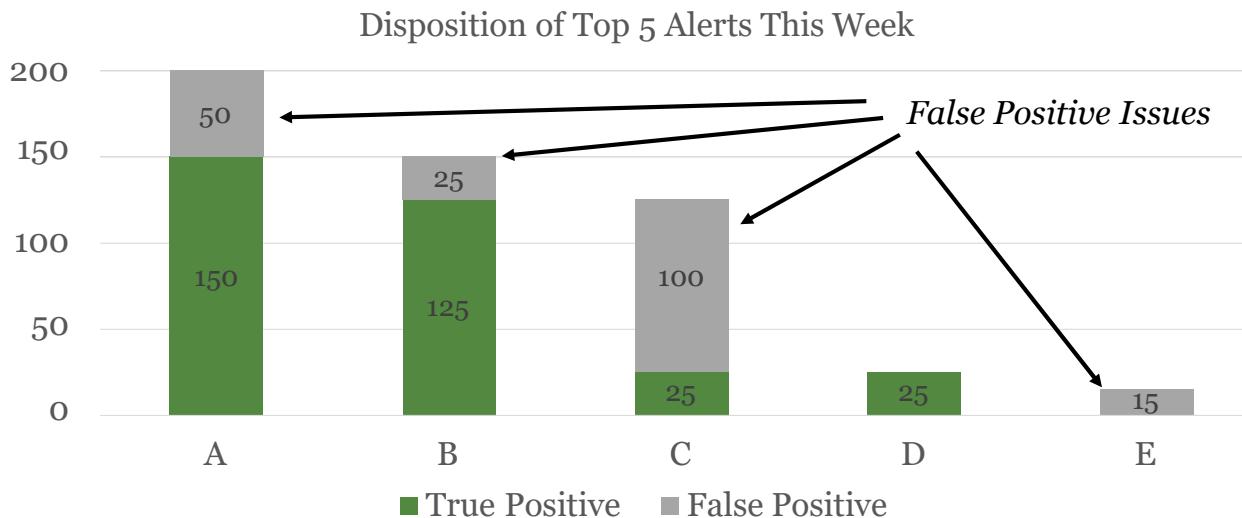
## Counting Final Disposition



### Counting Final Disposition

As alerts are processed, their final determination should be labeled in a way that can be easily counted for metrics. Most alerts determined to be false positives will be immediately dismissed from the alert queue, while others may at first appear to be a true positive, and then later added to the false positive pile. Both paths should lead to a label that can be counted as the ultimate "false positive count" vs how many alerts were true positive. To evaluate your best and worst analytics, you must be able to create a chart first grouping by the unique alert ID, and then within each of those groups, how many times it was called a true vs. false positive. Creating that chart will produce a visualization that lets you compare true and false positive counts and ratios per rule.

## Evaluating Alerts for False Positive Tuning



### Evaluating Alerts for Tuning

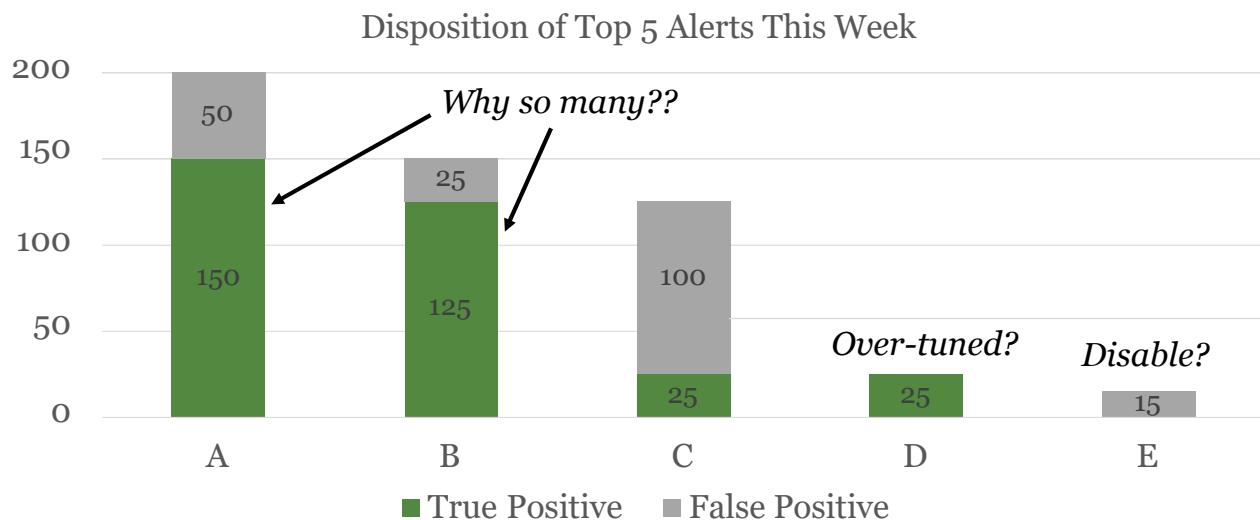
With the above chart showing the number of times each unique alert fired grouped by the unique identifier (called A,B,C,...etc., in this chart), we can easily find the most common alert the SOC sees. It's even more useful though to look at each broken down with their final disposition shown as well.

Here are the important points to draw from the chart above

- Alert A is our most common alert, and appears to be 75/25 in terms of accuracy creation 150 true positive and 50 false positives
- Alert B is our second most accurate alert at 125 true positives vs. 25 false positives, it is also the 2<sup>nd</sup> highest by volume
- Alert C is our least accurate alert by *percentage* with roughly 25 true positives and 100 false positives generated
- Alert D seems to be 100% accurate but lower volume
- Alert E has produced 100% false positives

Given this info, consider what, if anything you would do to tune each of these alerts. If our goal is to minimize time wasted on false positives (which will be the goal for most of us), then we should attack the source of the highest absolute *number* of positives first, which is alert C. Assuming that dealing with a false positive from all these alerts takes an equal time, then most time will be saved by fixing alert C. Which one comes next? The answer should be alert A since although alert E is 100% false positives, by count, alert A is still wasting more time by creating 50 false positives vs Alert E's 10 false positives.

## Other Conclusions Based on Alert Disposition

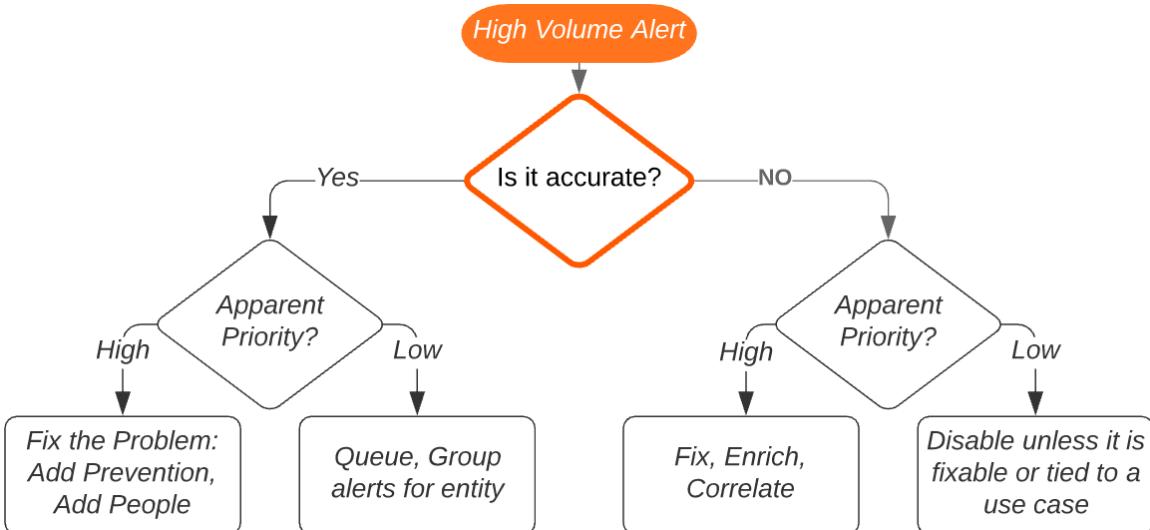


### Evaluating Alerts for Tuning

What else might we read into these numbers? Remember the idea is to identify all attacks, but also to drive true positive alerts as low as possible (meaning the minimum of bad things is happening). Given that mindset, here are some other options to consider when looking at these charts

- Alert A and B – Why are there so many alerts? Is there a lack of preventative control upstream that allows this many real issues to occur? Once false positives are tuned, this would be a great avenue to explore to bring the total true positive count down.
- Alert D – Is it *over* tuned? 100% is great, but only if you are sure there's no way something could slip by as a false negative. While bringing numbers down is great, you want to make sure you aren't missing anything.
- Alert E should likely be disabled if it continuously produces no value
- Bring in other metrics – For each alert, what is the average time it takes to work it, could better optimizations be made when considering additional factors? Is automation being used, or could it be used to bring down the time it takes to remedy whatever is happening in Alert A and B?

## Starting with High Volume Alerts



### Starting with High Volume Alerts

Another place to start with alert tuning is to ask, "which alert triggers the most?" regardless of accuracy and attack the problem based on volume. Given a high-volume alert, there are two variables we should separate the population with. The first is the analytics' accuracy, the second is the apparent priority (in other words, assuming it is correct, how severe of a condition is being identified?). This page shows a flow chart on how you can decide which action to take to move toward the ideal direction—low alert count with high accuracy.

- If you have high-volume, poor accuracy, low priority alerts (common for those who don't turn off default rule sets), these may either be disabled because they are just making noise and you don't care about their condition or fixed and dealt with as you would the high accuracy, low priority rules.
- If you have high-volume, poor accuracy, but high priority alerts (common for poorly tuned analytics), you likely want to keep these alerts, but need to fix them first. This is where enrichment, correlation, and rule logic tuning can come in. You don't want to get rid of highly valuable rules if you can keep them, so make your best attempt to fix them.
- If you have high-volume alerts that are accurate but low priority, the rule itself doesn't need improving, these alerts must just get in line for triage. Low priority items should be dealt with as you are capable, but to solve the high-volume problem, tactics, like alert grouping, may help you tackle the problem. The key to success here is increased efficiency in dealing with the issues.
- If you have high-volume alerts that are both highly accurate and high priority, you have a problem. This means your prevention capability is lacking in this area and the fix for you will be to attack the problem at its source—add prevention capability, additional controls, or anything else that can prevent it from happening in the first place. If you have done the best you can do, you may need more people if, after controlling the situation, you still can't deal with the number of true positives you're getting.

## Operations for Low-Priority / Non-Actionable Alerts

There are 1000s of analytics you *could* turn on, but...

- Attacks might not apply to you
  - Example: MSSQL exploit attempts if you don't run MSSQL
- They may identify internet noise
  - Recon scans, botnet exploits to perimeter, etc.
- They may be unclear to analysts
  - Protocol anomalies, server resets, out of order TCP, etc.
- Goal: Tie **all** analytics to a pre-determined use case

### Operations for Low Priority / Non-Actionable Alerts

There are many reasons your team may have many alerts in the "low accuracy, low priority" bucket. The primary reason is that most security appliances ship with 100s or 1000s of rules turned on by default, and security teams take the "tune-down" approach. The problem with this is that many rules fall under the following categories.

- Attacks that might not apply to you: Do you have alerts on that detect attacks for software you don't run? If so, why? Especially at the perimeter, looking for non-applicable alert signatures can cause a landslide of alerts that, while not wrong, don't really provide any value to the SOC other than knowing someone from that IP address is probably malicious.
- Attacks that identify internet noise: We know the internet is full of infected devices, hackers, and bots, and all of those things will be constantly scanning your perimeter. It's highly unlikely you will take any action since most attacks will be denied by a network or application-specific firewall, so why waste your time watching these? A significant amount of alert volume can be taken away by suppressing alerts you know will constantly occur due to internet noise.
- Alerts that don't imply a clear next action: We've all seen protocol anomaly alerts and alerts for things like server resets and other nearly "everyday" activities. The problem here for analysts is when they go off, what should happen next? Do you really want analysts chasing down why someone didn't send a TCP reset when shutting down a connection? For alerts like this, remember the aphorism of Hanlon's Razor which says, "Never attribute to malice that which can be adequately explained by stupidity." Sure, an application may have not used a protocol to the exact RFC specification, but is it more likely to be an attack or a lazy programmer? Alerts like these are often another large source of alerts that doesn't often bear fruit when the cause is chased down.

One step that can be taken to reduce unneeded and low-priority alerts is to actually look through all your available rules in your security appliances and deactivate all the ones that aren't tied to a specific use case. Yes, it's a big job, but it helps analysts understand that when an alert goes off, it *has* been pre-determined to be something of concern.

## Risk-Based Scoring and Alert Aggregation

**Problem:** Advanced attacks may trigger only low-priority alerts, get missed

**Solution:** Multiple low-priority alerts for one user or device should create a high-priority alert

**Method:** Risk-Based Scoring

1. Each alert gets a risk score based on pre-defined risk scoring rules
  - Based on user privs, vulns, intel, ATT&CK, kill chain, asset criticality, etc.
2. Collect all suspicious events into a holding area
3. Aggregate risk by user/host/asset/system
4. Draw an analyst's attention once a threshold is hit per user/device

### Risk-Based Scoring and Alert Aggregation

One issue you might worry about with disabling or otherwise passing over low priority alerts is that there is a chance they are alerting you to unknown advanced attacks. How can we leverage the fact that something anomalous but low priority is happening, and use that fact to discover the anomaly is actually a potential indication of compromise? We can do this by aggregating alerts over time for the same systems and users and escalating the situation if more than one suspicious activity occurs within a short time frame.

Risk-based scoring or alerting is what many SOCs use as a solution to this problem. Instead of chasing individual low-priority alerts or throwing them away, they walk a middle ground. In this system, a single low-priority alert may not be enough to inspire action, but their alerting systems collect all alerts that fire, scoring the apparently associated risk of each based on the user, system, and software involved. Once enough alerts occur for a single user or system, the situation is escalated to the point where an analyst will respond and take a look at it. The key assumption in this method is that advanced attacks may slip under the radar, only firing anomaly-style alerts that might not draw attention. If the attack continues, however, the attackers are likely to trigger more than one (low priority) alert for the same system, which can be used to identify actual attacks from true anomalies.

One specific method on how to do this was documented by Jim Apger (a staff architect at Splunk) at the 2019 SANS SIEM Summit. Jim's suggested method of "Risk-Based Attribution" reportedly brought dramatic changes to the true positive alert rate while bringing the alert count down at organizations where it was implemented. The slides on how this system works and is implemented can be found in the reference below,<sup>1</sup> as well as a recorded video from 2019 Splunk .conf where a case study on the system was presented.<sup>1</sup>

[1] <https://conf.splunk.com/watch/conf-online.html?search=%22Big%20Alert%22#/>

## Detection and Analytic Design Summary

- Alert count depends on scope, sensitivity, accuracy, and analytics enabled
  - Having more analytics is better ... if they produce true positives
- **False positives and false negatives are tied together**
  - Understanding tuning variables becomes *very important*
  - **False positives** are the worst part of many analyst's job
  - But **false negatives** are more dangerous—must "walk the line"
  - Testing (covered soon) can make all the difference
- High-volume alerts can be solved via **fixes or tuning**
  - **Risk-based scoring** can turn low-priority into high-fidelity alerting

### Detection and Analytic Design Summary

In this section, we discussed the detection system and the variables that affect it, as well as the theory and math behind tuning of analytics. One of the most important takeaways is the inseparable link in any detection system between false negatives and false positives. In a SOC, this isn't just an engineering decision but also a human factor one. Too many false positives can cause burnout and alert indifference, too few can stray into missing attacks. Each SOC must find its happy medium between these extremes.

The good news is there is a way to make the decision easier—data enrichment and constant analytic testing and tuning. With more information, true positive becomes increasingly separable from false positive, and accuracy improves without sacrifice. Given this, SOCs should consider it a top priority to understand their opportunities for enrichment with their SIEM and other security appliances and maximize their capabilities in this area. It quite literally ties back directly to alert fatigue and burnout prevention.

When fighting high-volume alerts, one approach does not fit all situations. We must consider whether the high-volume alert is pointing out valid concerns, or if it is a poor rule. Valid high-volume alerting indicates an issue in the environment that should be blocked before it becomes a problem, while bad rules can be remedied through tuning, enrichment, suppression, or risk-based scoring and aggregation. Throughout the next few sections, we'll discuss how to test, store, and document rules to help improve not only accuracy, but efficiency in triage as well.

# Course Roadmap

- Book 1: SOC Design and Operational Planning
- Book 2: SOC Telemetry and Analysis
- *Book 3: Attack Detection, Threat Hunting, and Triage*
- Book 4: Incident Response
- Book 5: Metrics, Automation, and Continuous Improvement

## SECTION I

### Introduction

#### Creating and Processing Alerts

- Efficient Alert Triage
- Detection and Analytic Design
- **Capacity Planning**
- *Exercise 3.1 – Capacity Planning*
- Detection Engineering

#### Advanced Analysis

- Analytic Frameworks and Tools
- *Exercise 3.2 – Structuring, Documenting, and Organizing Use Cases*
- Threat Hunting
- *Exercise 3.3 – Planning a Threat Hunt*
- Off-Hours Alerting and On-Call
- Active Defense
- Summary and Cyber42 – Day 3

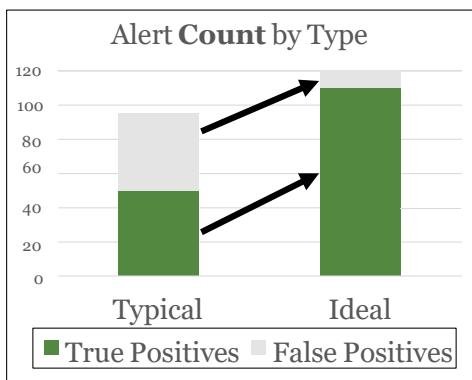


This page intentionally left blank.

## How Many Alerts Should You Create?

### Typical

- Missing *true* positives
- Too many *false* positives



### Ideal

- **All** true positives
- **Zero** false negatives
- **Minimum** false positives

Moving in the right direction:

- May increase true pos. *count*
- Decreases false pos. *percentage*
- May *not* change staff levels required

### How Many Alerts *Should* You Create?

Let's take a step back and think about how many alerts we *should* see in an ideal world. First, consider how many bad things are truly occurring in the environment. If we had perfect detection, all those things should be coming in as true positives, and seeing those alerts is a *good* thing. We also must add the inevitable small but non-zero amount of false positives, any more than that is a *bad* thing. As shown in the chart above, these two numbers added together represents the total alert *count* the SOC will see.

Notice though, to move in the right direction doesn't necessarily mean reducing the number of alerts you see. Seeing every attack in your environment means your alert count will now likely go up (because you are seeing more true positives). The good news is it may partially balance out with the reduction of false positive count, if nothing else, the percentage of alerts that are false positives will go down, which is always good.

## Factors Contributing to Alert Count

### Good causes for **more alerts**:

- New tools increase detection capabilities
- Additional (relevant) analytics enabled
- New sensors give more or better visibility
- Threat hunting

**Reasons:** Detecting more true positives that were previously missed

### Bad causes for **more alerts**:

- Poor tuning of enabled analytics
- More analytics active than necessary
- Lots of successful attacks

**Reasons:** Too many true positives, false positives

### Good causes for **less alerts**:

- Preventative controls stop attacks
- Additional data enrichment and correlation reduces false positives

**Reasons:** Blocking attacks before they become true positives, reducing false positives

### Bad causes for **less alerts**:

- Lack of visibility for attacks
- Disabling / ignoring valid alerts

**Reasons:** Missing or ignoring true positives



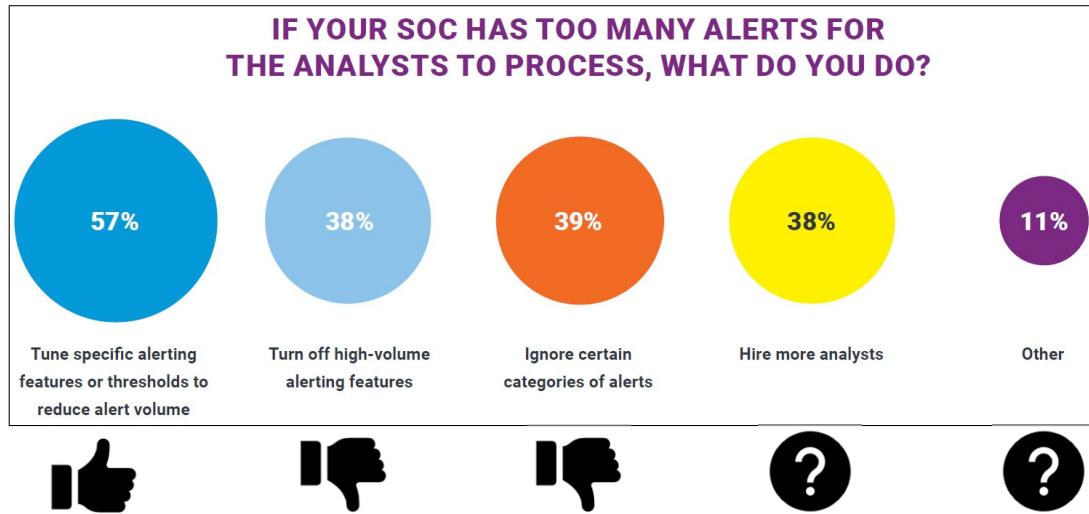
## Factors Within Your Control Contributing to Alert Count

Remember that while in general SOCs prefer less alerts as opposed to more, in a more specific sense, you could take an action which results in more alerts and have it be a good thing. From the previous slides, recall that we want to see *all* bad activity, and doing so might be detecting it with multiple different tools or appliances that all detect the same activity. If you are *not* seeing something that is indeed bad, and the action you take causes you to see an attack that was previously not visible, that creates more alerts, but *is* a good thing to have done.

This slide lists some of the factors that will take your alert volume up and down some of which are good, some of which are bad. Remember the goal is to drive down the number of bad things that do occur, while catching all the ones do, all while reducing false positives to the minimum.

## Discuss: How Do YOU Handle Too Many Alerts?

Compare reported actions to previous discussion



SANS

MGT551 | Building and Leading Security Operations Centers

42

## Discuss: How Do YOU Handle Too Many Alerts?

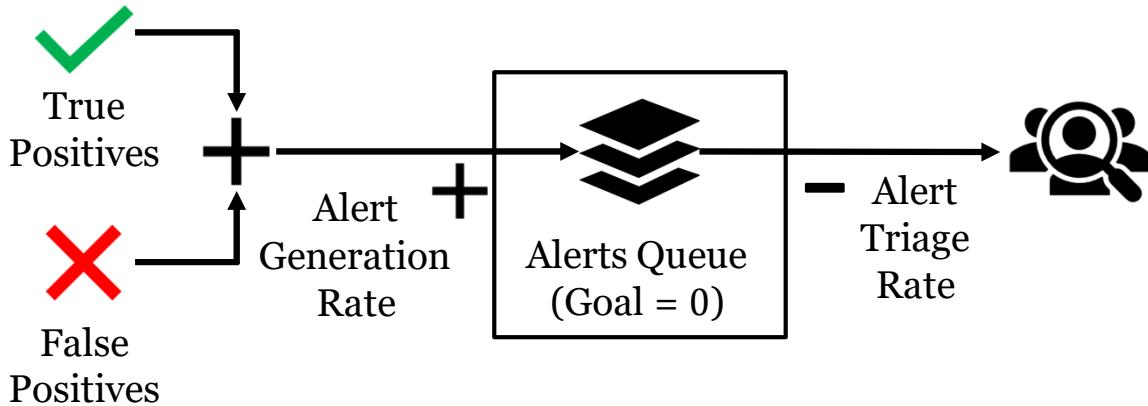
Considering the discussion on the previous slide, how does the typical SOC handle this issue? From the survey results, we see that survey respondents commonly take two approaches we have categorized as "bad" ways to get rid of alerts – turning off high-volume alerting features (which may reduce false positives, but also mask true positives), and ignoring certain categories of alerts. 38% of respondents said one of their tactics is hiring more analysts, this could go either way. Assuming alerts have been tuned, prevention has been maximized, and automation has been put in place, and the alerts are still overwhelming, this is the right approach, it should not be the first go-to method, however. The good news is 57% of SOCs report taking an approach from the "more alerts but good" box – tuning, which if done correctly should reduce false positives while keeping all the true positives.

Consider what your typical approach is when you have too many alerts, and if there's an action you can take that will reduce false positives (and therefore total alerts) without having to run the risk of obscuring potential attacks.

[1] [https://www.criticalstart.com/wp-content/uploads/CS\\_MDR\\_Survey\\_Report.pdf](https://www.criticalstart.com/wp-content/uploads/CS_MDR_Survey_Report.pdf)

## Alert Generation vs. Triage

- If generation rate > triage rate, queue > 0
- If generation rate =< triage rate, queue = 0



### Alert Generation vs. Triage

Once you have thought about the rate of alerts you are generating and why, consider whether that is in balance with the rate at which you can triage and investigate alerts. The alert generation rate is an independent variable when it comes to the SOC (there is a theoretically "correct" number as explained earlier). Whether or not you can keep up with that number is a different question.

We can think of this system as shown in the slide above. The inputs to the alert queue are the number of both true and false positives, the outputs from the alert queue is the rate at which you can deal with them. In a SOC, the goal is to keep the alert queue at an average size of 0, meaning you are always reviewing all alerts in a timely manner. If your alert generation rate is faster than the rate you can investigate them, you're going to run into a problem and will need to find the *correct* way to solve it. The correct way will depend on if the true positives or false positives are driving the overload of alerts.

## Capacity Planning

Predicting capacity requirements is *difficult!*

- Number, severity, and response time varies for every attack

We will discuss and provide guidance to help understand:

- How many alerts you should expect to create?
  - What is the min and max range?
  - How many analysts will be needed to cover them?
- How long should each alert take to triage?
  - What is the min and max range?
  - How does this vary by alert type?



### Capacity Planning

Over the following slides, we'll cover some of the methods that can be used for capacity planning, and the intricacies and difficulties of doing so. Naturally, anytime you are trying to predict the severity and arrival time of an unknown event (cyber attacks), as well as the effort required to remediate that event, there is a large room for error. Therefore, we will consider this problem in our approach, and seek to learn not just an average number for each of these factors but gain a sense of the range of possibilities so that we might plan with the best-and-worst-case scenario in mind.

## Basic Factors in Triage Capacity Planning Calculations

Some basic math identifies the key variables:

*(Scoped purely to alert triage activity)*

$$W = N * T$$

**W** = Alert workload in minutes

**N** = Average number of items<sup>1</sup> to manually address per day

**T** = Average time needed per item (minutes)

$$\text{Analysts required} = \frac{\text{Alert workload}}{\text{Available time per person}}$$

Assumes you want to get to alert count zero every shift

This however is *only an average*, far from the full story...

### Basic Factors in Triage Capacity Planning Calculations

Starting at the highest level, what are the basic factors that will drive the number of people required (at least for the triage role) in your SOC? While we will dive deep into each of these factors, at the top level we're looking at a simple calculation of the number of items that need addressing each day on average, and the length of time needed for each of them. Multiplying these two gives us an "alert workload" that can be looked at as the pool of work that must be done across some number of people. If you have 20 items to triage per day and it takes on average 15 minutes for each, you'll need  $20*15= 300$  min or 5 hours of work per day. This workload could conceivably be covered by a single person if their job allows them to focus that much of their time per day purely on triage. But will it?

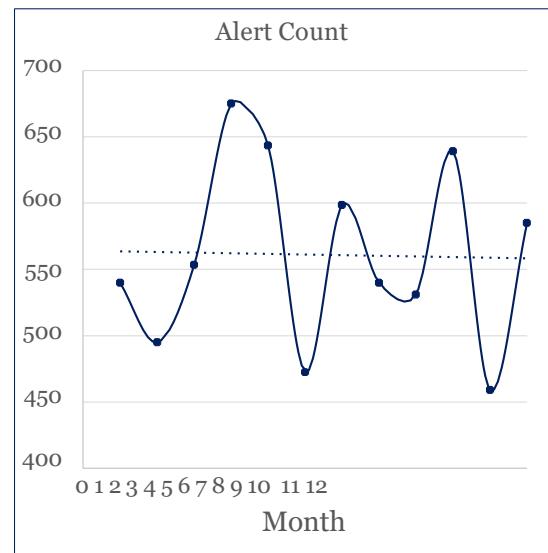
As you are probably thinking, however, doing math with such a reductionist approach is unsatisfying. We know that average time needed per ticket varies wildly, as will the time required to address each item. An average can of course still be calculated, but it isn't very representative or usable given the wide variance in the data day to day. Let's dive a bit deeper into this and tease out some of the complexities and how we might be able to produce a better and more meaningful calculation.

[1] I'm calling this "items" at this point because there will be an upcoming discussion on the complexities of this factor

## Approaches for Estimating Expected Alert Count & Workload

### Methods to estimate alert count

- 1. Historical metrics** (for established teams)
  - Best when you have months / years of data
  - Gives you both an average and sense of min / max
- 2. Surveys** from others (for new teams)
  - Average alerts worked per person
  - Total alerts by company size
- 3. Probability calculations** (for developing teams)
  - When you have *some* data, and want to get a sense of bounds
  - Relies on some key assumptions



## Approaches for Estimating Expected Alert Count & Workload

There are several ways to estimate the volume of alerts that you may expect to receive in a given period, each of which is appropriate for different situations.

- Historical metrics - Obviously, the best and most relevant information is going to come from your own SOCs alert count history. Looking into the past will give you a sense for not only the average number of alerts, but the variance within that average. This approach is best for SOCs that have months to years of data to clearly see trends, and trust that they will be at least decently representative of what you can expect. Do you normally have 100 alerts +/- 10 within a week? Or is it more like 100 alerts +/- 50? That's an important difference when it comes to capacity planning. To get an accurate estimation, you'll need to have some sense of the worst-case scenario, average, and lowest numbers to expect. What happens for the SOC that doesn't have years of data? Using the data you do have, combined with the other two approaches can help.
- Survey data - If you haven't started your SOC yet, you may need to leverage alert counts reported on surveys from others. As you might expect, this can be wildly inaccurate given the differences in organization size, security tooling, and alerting philosophy, but it's certainly better than nothing if you have nowhere else to start. The good news is this strategy will be rapidly replaced with historical averages once the data starts coming in. One way to bring some of the inaccuracy out of estimation is to try to work with an alerts per person number instead of a total.
- Probability calculations – When you don't have a long history of data (or even if you do) and want to get a sense of what your minimum and maximum numbers might be, probabilistic calculations can be surprisingly insightful to get a sense of boundaries and expectations. Under some key assumptions on the nature of how alerts are produced and the time they take to address, we can get a pretty decent sense of what to expect. You will see this approach in action in the upcoming exercise.

Over the next few slides, we'll discuss each of these approaches.

## Complicating Factors for Alert Count

What do we mean by "alert count"? Clearly definitions differ...

Cyberattackers are relentless and getting more sophisticated by the day.  
Businesses are under constant attack, with the average security operations team receiving over 11,000 security alerts daily.

6 SEP 2018 NEWS  
174,000 Alerts per Week Besiege Security Teams

How many security alerts does your security operations center (SOC) receive daily?



Considerations for alert count:

- What counts as an alert that needs to be addressed?
- What if one attack sets off 1000's of alerts?
- Are you counting false positives?

$\frac{1M \text{ alerts a day}}{100 \text{ analysts}} = 21 \frac{\text{alerts}}{\text{min per analyst}}$ ??

SANS

MGT551 | Building and Leading Security Operations Centers

47

### Complicating Factors for Alert Count

One of the biggest problems to deal with approaching this problem is defining what exactly an "alert" is *for purposes of time calculation*. While it may be clear to us that a security appliance sending a log that something bad may have occurred is an alert in the traditional sense, should we *really* factor in every one of those for time calculation? Probably not. It's not like we spend time evaluating each alert if a firewall tells us via 1000 alerts about a port scan from a single IP address – those types of things, and many other alerts, are aggregated into some type of issue to be investigated (or not). To further complicate things, we have headlines and images like the ones above<sup>1,2,3</sup> showing up in marketing hype and surveys touting tens of thousands to MILLIONS of alerts per day.

These enormous numbers have no bearing on reality for calculation of capacity planning. At 1M alerts per day, even with a team of 100 SOC analysts that would be 10,000 alerts per day per person. Assuming a perfect 8 hours of alert work that's 21 alerts per minute - clearly not a useful number. What else can we do, then?

1 <https://www.imperva.com/blog/27-percent-of-it-professionals-receive-more-than-1-million-security-alerts-daily/>

2 <https://www.infosecurity-magazine.com/news/174000-alerts-per-week-besiege/>

3 <https://blog.paloaltonetworks.com/2020/09/state-of-security-operations/>

## Complicating Factors for Alert

Which alerts "count" for capacity planning?

It's not so straightforward, consider:

- Duplicates
  - Firings of the *same alert*
  - Two different alerts fire for the same activity (seen by two security tools)
- Test alerts
- Simulations
- False positives
- Accidental triage queue explosions
- Automated actions taken for some alert types and not others

## Complicating Factors for Average Time Calculations

When trying to do capacity planning based on an absolute count of alerts you will likely find that things are unfortunately not as clean to calculate as you might hope. While it *is* possible to calculate a total average time per alert, you may find using this number is largely meaningless due to the makeup of that population of alerts. The largest factor is that many alerts are not "unique" or even malicious, meaning addressing them will be either very fast or unnecessary because they will be grouped in with others. There are a couple approaches to overcoming this issues that we will discuss over the next few slides.

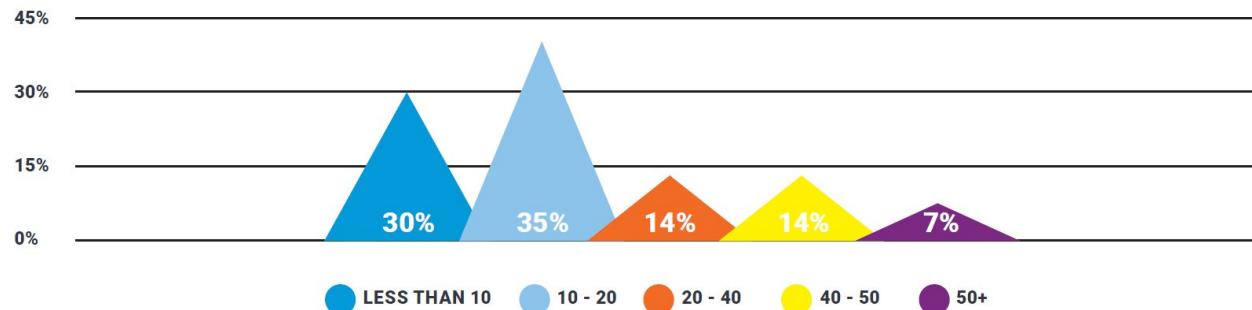
## Alert Count per Person Survey Data<sup>1</sup>

When you ask analysts:

- Why do these numbers not match up?
- Is this even realistic?

$$50 \frac{\text{alerts}}{\text{a day}} \cdot 6.25 \text{ alerts per hour}$$

### HOW MANY INCIDENTS/ALERTS DO YOU PERSONALLY INVESTIGATE PER DAY ON AVERAGE?



### Alert Count per Person Survey Data

If we check with survey data on alerts/day analyst claim to work, do these alert counts match up? Not at all – In this survey from Criticalstart<sup>1</sup> of "incidents/alerts" per person analysts self-report they deal with, the number is nowhere near 21/minute, 75% don't even seem to do 21/day. Why the discrepancy? The reason is that analysts don't often deal with alerts as a single item, they often deal with aggregated alerts that point to a single instance of a problem. Perhaps looking at it this way can bring us closer to an accurate understanding of workload.?

[1] [https://www.criticalstart.com/wp-content/uploads/CS\\_MDR\\_Survey\\_Report.pdf](https://www.criticalstart.com/wp-content/uploads/CS_MDR_Survey_Report.pdf)

## Narrowing Down the Count

What do we **really** want to know?

- How much time people spend triaging and investigating **individual potential incidents** each day (there's no standard term here)
  - Alerts are **not 1:1** with **individual potential incidents**
  - **Aggregated alerts** might be though

Sorting it out:

- Alert dashboards usually aggregate alerts related to same user/system
- Count these instead! Then consider "human interaction required"
  - How many "**unique issues**" were there to deal with?
  - Did you have to **manually investigate** them?
- **Suggestion: Measure aggregated issues requiring human investigation**



## Narrowing Down the Count

What we're looking to consider is not a total raw alert count, but how much time people spend on investigation individual potential issues each day. Raw alerts are *very much* not on a 1:1 basis with potential issues, a single attack may set of 10's to thousands of alerts, all for one compromised device. (Unfortunately, available terminology starts to break down here which is why we're using the "potential issues" phrasing – I'm trying to avoid using the word "events" since that is already defined. When I say "potential issues" on this slide, I'm trying to refer to instances of individual infections or attacks against a single system or user.) Therefore, to count in way that helps us predict time required, we propose you focus on counting "potential issues" investigated, and whatever term for this makes sense to your team. Most alert triage systems aggregate alerts for the same user, system, IP address, or other factor, and these items should align much closer as 1:1 with potential issues. Bottom line: If you can count the number of items (groups of aggregated alerts) analysts work through, regardless of the number of alerts that are aggregated into each, you will be much better at capacity planning.

## Estimating Count with Probability

A different approach that can **bound what to expect**

- A generic version of our problem:
  - *How many random events (cyber attacks) might occur in a unit of time, given an assumed long-term average rate?*
- How do we estimate this? Use the **Poisson distribution**
- A **Poisson process** is a process where...
  - **Events occur randomly** – approximately true of cyber attacks
  - **At a constant average rate** – which we have a sense of based on history
  - **And are independent of each other** – at least partially true (any threat actor can attack us at any time)
- Not a perfect model, but doesn't mean it won't lead to potentially useful and better conclusions than guessing and averages only

$$f(k; \lambda) = \Pr(X = k) = \frac{\lambda^k e^{-\lambda}}{k!}$$

### Estimating Count with Probability

Back to the discussion on how many issues we have to deal with, now that we're clearer on that definition. We mentioned the probabilistic way of approaching estimating how many issues you may need to deal with.

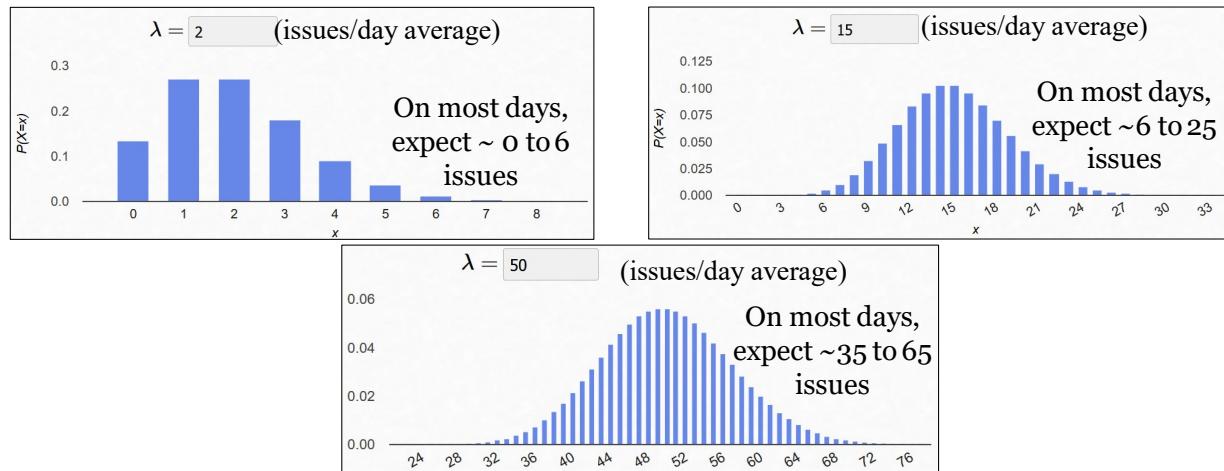
Consider in a more general sense the problem we're trying to solve here - what we have is a situation where events randomly occur (cyber attacks / intrusions), there is an underlying long-term average rate they occur at, and we'd like to know how many we might expect to see in any given unit of time. In other words, we want to first answer "what is the range of events might we see in a day, week, month, or year, given the average rate at which that random thing occurs?" If we can get this answer in more detail, we'll have a better sense of the "count x time per issue" calculation we first introduced.

Under these conditions, we can estimate cyber attacks as a "Poisson process"<sup>[1]</sup>. What the Poisson distribution gives us in this situation is a way to bound the number of attacks we might see, not just on average, but as a minimum and maximum as well, which is much more useful. To use this however, we must assume those attacks follow some rules (shown on the slide above). As you may be thinking, it's not a perfect match in the assumptions, but rarely do physical processes perfectly match a theoretical model. That mismatch, however, doesn't mean estimating using the Poisson equation can't lead us to better conclusions than we would have without it. It's another tool in the toolbox that's it's worth giving a try given the extreme simplicity.

[1] No need to dive too deep into the math here, but if you are a math lover, there are plenty of explanations available online and on YouTube about what a "Poisson process" is. In short, the problem is very similar to the "binomial distribution", which used for prediction in situations like coin flips. The Poisson distribution is just a specific case of the binomial distribution, where we assume there are large number of trials and the probability of any given trial being a success is very small. This situation converts well into a tool for estimating the expected rate of randomly occurring events.

## Applying Poisson to Issue Count per Day

If we assume cyber attacks are a Poisson process, we can fill in the average number of issues we see per day and predict bounds!



### Applying Poisson to the Issue Count per Day

To use the Poisson distribution to estimate how many issues/incidents you might expect to need to deal with in a given unit of time, all you need to know is the average rate at which you normally see attacks in that same period. On the slide above we have three different charts showing expected results for a SOC that saw 2, 15, or 50 alerts per day respectively. The bars over each number on the X access represent the percentage of days you would expect to see that number of attacks (adding the value of all blue bars would give you 1 or 100%). So, for a SOC that averages 2 issues per day, the upper left chart shows the following expected distribution.

- 0 alerts – 17% of days
- 1 alert – 27% of days
- 2 alerts – 27% of days
- 3 alerts – 18% of days
- 4 alerts – 9% of days
- 5 alerts – 4% of days
- 6 alerts – 1% of days
- 7 alerts – 0.3% of days
- 8+ - almost never

With this information one critical variable (issue count) of capacity planning can be estimated with the additional detail beyond an "average". With these numbers, the assumption that your worst days (roughly once a year at these percentages shown) you'll hit 7 issues a day. If your staff can handle that, you should have no problem handling the volume of activity expected. Keep in mind that there are some key assumptions built into this math that may not turn out 100% true, but this approach can at least help give you an idea of what numbers you might expect to be working with. Of course, this doesn't address the question of *how long* each of those issues takes to deal with on average but having a range for the count of issues variable is a great start.

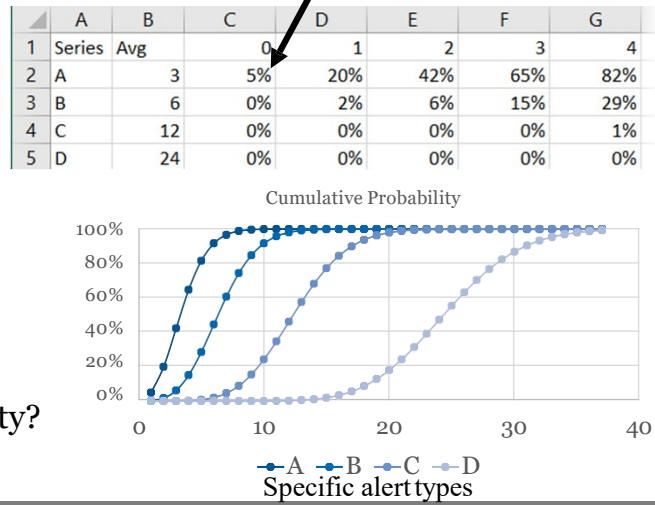
If you'd like to quickly make one of these charts – Google an interactive Poisson distribution graphing tool or go to wolframalpha.com and search something like "*poisson distribution mean 2*" where the number is the average rate of whatever timeframe you'd like to see the chart for. The charts on the slide above were made with the page at <https://homepage.divms.uiowa.edu/~mbognar/applets/pois.html>.

## Additional Questions We Can Answer with Probability

Useful SOC questions we can now roughly predict:

- What will *most days* look like?
  - "Most days" can be defined as any percent – 90%, 99%, etc.
- What *percentage* of days will we have over  $x$  number of issues to deal with?
  - In other words, how often will we be above capacity?
  - How often will we have extra capacity?
- For excel – see POISSON.DIST

=POISSON.DIST([x],[mean],[cumulative])  
=POISSON.DIST(C\$1,\$B2,TRUE)



### Additional Questions We Can Answer With Probability

Poisson process-based estimations are not just good for looking at bounds, using this model we can also get a grasp on some other useful questions that you might want to know as a manager such as how many alerts of a *specific type* will we get on 50% or 99% of days, weeks, etc.? (show as letters above – imagine A, B, C, and D are different alert types such as phishing, viruses, and more, with different average alert rates.) The same math that works for estimating all alerts counted together as a random process still works if you assume each individual *type* of alert is also a random process. For you Excel fans (aren't we all?) these models can be easily built with the POISSON.DIST excel function and equation as shown on the slide above. As opposed to the previous chart, the graph on this slide shows the *cumulative* probability.

Looking at the chart above - cell C2, for example, shows that a SOC with 3 alerts of type A on average would expect to see:

- 0 alerts of type A on 5% of days (cell C2)
- 0 or 1 alerts of type A on 20% of days (cell D2)
- 2 or less alerts of type on 42% of days (cell E2)
- 3 or less alerts of type A on 65% of days (cell F2)
- And so on.... Until we get to 100% (not pictured in the table)

As shown in the formula on the slide, the 3<sup>rd</sup> variable named "cumulative" in POISSON.DIST function (a TRUE/FALSE argument) determines whether your calculation shows the *cumulative* percentage of all numbers up to that point (as done in the previous bullet points) or just the probability of that *specific* number of alerts in a day. If graphing using FALSE as the cumulative variable, the graph would look like the curves on the previous slide. For you probability experts, this is showing the cumulative distribution function vs the probability distribution function.

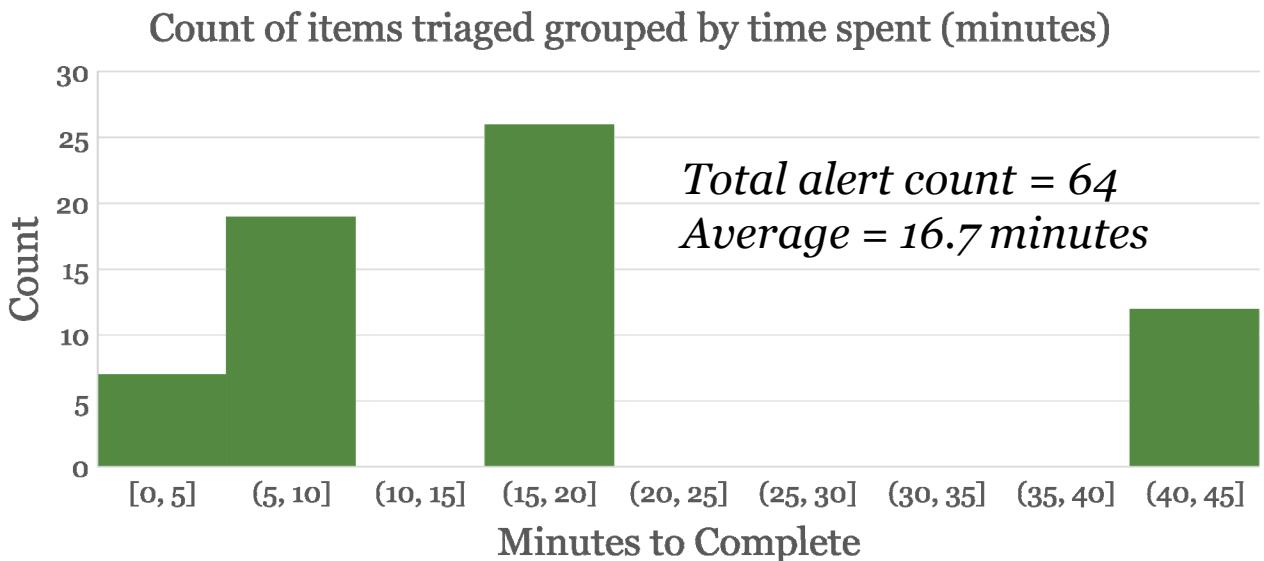
## What about Average Time per Investigation?

- We can estimate issue count with Poisson distributions
- What about estimating **time** to work each alert?
  - **If you have data – use it!** Group alerts by category, graph their investigation time and find the distribution you can use
  - **If you don't have this data** - back to probabilistic modeling and using surveys



This page intentionally left blank.

## Looking At the Population of Triaged Items

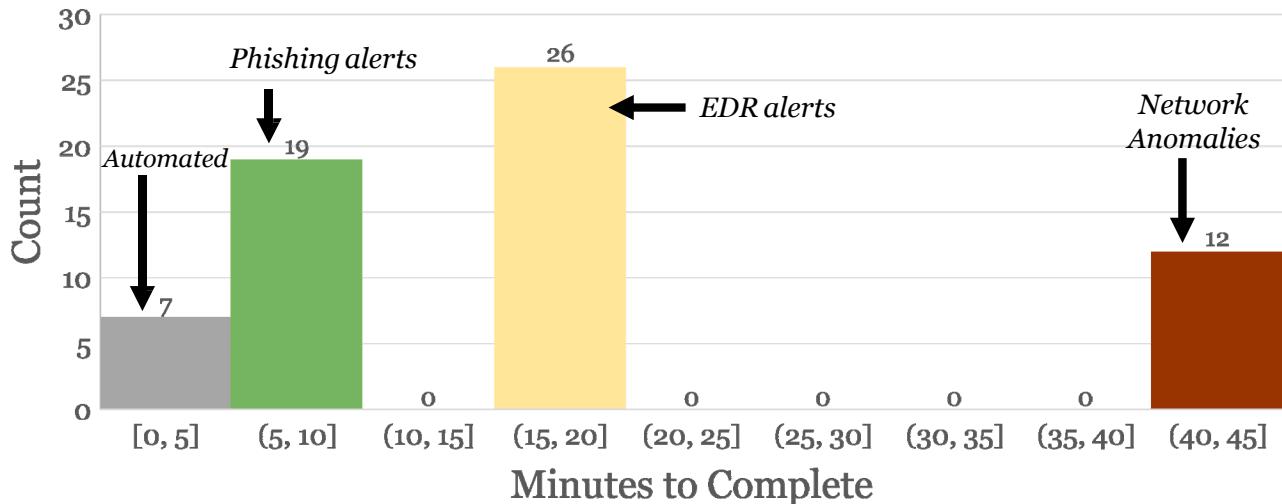


## Looking At the Population of Triaged Items

When we look at all the items that the team triage and group them by the time each took, what you will likely see is multiple clusters of items. Triage time (and time for analysis and incident response as well for that matter) is not a completely random variable. Items of one nature will tend to have an average time required that is independent of the time taken for a different type of item. While we can count the whole population and calculate the average, using this for prediction is likely to lead to large margins of errors. If we break this down into smaller groups and look at the distribution of each section however, we may be able to approach a much better understand of what is happening.

## Going Deeper...

Count of items triaged grouped by time spent (minutes)



## Going Deeper...

If we take the previous histogram and show the types of alerts that are underlying each, a more detailed and nuanced picture may emerge. When further grouped, the time taken to deal with alerts may show it aligns well with some category – alert type, priority, environment, or otherwise. While it's good to know what was shown on the previous slide (average alert count of 64 with an average time per alert of 17 minutes), it's much better to know an average count and variance for each *type* of alert, and the average and variance of the time taken for alerts of that nature. With that level of information, you could much more accurately predict workload. After collecting data for several months, you might be able to say something like:

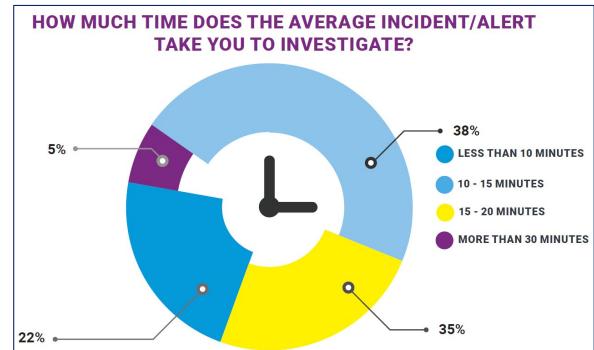
"Our workload typically consists of an average of

- 19 phishing alerts, each of which take an average of 8 minutes
- 7 automated alerts, each of which are almost immediately solved
- 10-20 network anomalies, which take 45 minutes on average to solve

You could then independently model and estimate the count and time associated with each type of alert you receive, and sum all of those numbers together for a estimate of the total workload. Getting too fancy may lead to diminishing returns, but it's worth giving a further breakdown of your data a try to see if it leads to additional insights.

## Using Probabilistic Models and Survey Estimates

- For estimating time, you have plenty of options for modeling
  - Surveys →
  - The normal distribution
  - The log-normal
  - Uniform distribution
  - Beta distribution
  - Historical data?
- Which is best?
  - Your data can tell you, if you have it
  - **The log-normal distribution** is a good starting place



### Using Probabilistic Models and Survey Estimates

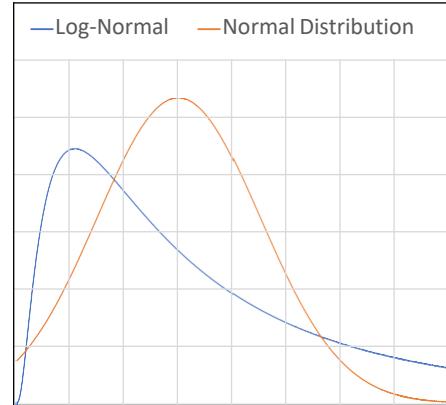
One approach to looking at expected times when you have no other estimates is turning to survey data. As shown in the slide from the CriticalStart survey in the footnote below<sup>1</sup>, we have some base data we can start with. While it's not ideal by any means "more than 30 minutes"...thanks. We can at least see that most analysts self-report that it takes them 20 minutes or less in most cases, and that the most common number is 10-15 minutes. Are these numbers reliable? Hard to say, but it's a MUCH better starting place than just guessing.

What if we want to try some probabilistic estimation though? Like our use of the Poisson distribution for modeling events *occurring*, there are also models that, if we choose a good option that matches reality, might also allow us to do the same for investigation timing. Which one is the *best* is a bit up in the air, and once you have some historical data, the right answer to use for your situation will become clear, but from the start, there is one that stands out – the log-normal distribution.

[1] [https://www.criticalstart.com/wp-content/uploads/CS\\_MDR\\_Survey\\_Report.pdf](https://www.criticalstart.com/wp-content/uploads/CS_MDR_Survey_Report.pdf)

## The Log-Normal Distribution

- Like a normal "bell curve", but skewed to the left
  - The normal bell curve has equal outliers on each side of the curve
- Log normal is different, best used when
  - ✓ Less than zero is not an option
  - ✓ Most things skew to one side (fast, well understood alert investigations)
  - ✓ Occasional outliers exist on one side (unique/difficult investigations)



In Excel:

`LOGNORM.DIST(x,[mean],[stdev],FALSE)`

### The Log-Normal Distribution

While your intuition may initially bring you to the normal bell curve style distribution for modeling alert times, in the authors opinion, this is likely not the best match. While the normal "bell curve" would model our investigation times mostly falling at or close to the average, there would be an equal number of outliers in both positive and negative directions. When it comes to investigating incident, the authors believe the situation is slightly different. During investigations, most of them will be "typical" and therefore have a known process for investigation that brings analysts to the answer rather quickly. There are, however, occasional outliers where an investigation is particularly difficult and has a much longer time associated with it. This means most investigations will skew shorter, with a few stick out longer investigation times. This is exactly the type of situation the "log-normal" distribution is optimized for modeling. Another benefit of the log-normal over the normal distribution is that log-normal does not support values less than 0 – the bell curve theoretically could produce negative investigation times. While those would be great for metrics ;), of course that would break our estimation capabilities unless we artificially stopped at 0. The log-normal distribution takes care of that from the get-go.

## Putting These Estimates Together

- We now have:
  - An average, low and high range for alert **count**
  - An estimation for average, low and high for **time**
  - Might even have data broken down by alert category/type - even better!
- **New problem:**
  - How to best simulate the results of multiplying two random variables?
  - We have random *counts* and random *times* per event
  - What total time required should we expect on low, avg, high days?
- **Answer:** Simulate with Monte Carlo analysis!
  - Coming up in your next lab!



### Putting the Estimates Together

We now have a much better approach to getting the answers for our two core variables for capacity planning:

- We can estimate the *count* of items we might see on a slow, average, and high-volume day using a known average and modeling alert arrival as a Poisson process
- We can estimate the *time* required for an easy, medium, and difficult alert by starting with the log-normal distribution and adjusting as our data begins to reveal more detail over time
- We can take either of these numbers and further subdivide them into alert types and categories as desired to produce even more detailed estimates

Even better, all of this can be done in Excel with built in functionality, and the only inputs we need are a reasonable guess for average counts and time ranges! With those items in place, we are now ready to run our numbers. The easiest way to do this is using Monte Carlo analysis. In effect, we take our simulated distributions for count and time and run hundreds of thousands of pretend "days", each of which contains a randomly selected number of alerts, each of which taking a randomly selected amount of time (both drawn from the appropriate models), and once all the data is in place, see what it tells us! Sure, we could do the math directly as well, but unless you're into higher-level probability math, simulation is usually the easier route. Our Monte Carlo analysis simply takes the two random variables and produces our desired goal – not just an average, but a *range* of how much total time we will need for dealing with manually investigated alerts on a good, bad, an average day! A much more detailed approach than simply multiply averages!

## Capacity Planning Summary

- Capacity planning can seem complex, and it is to some degree
- The *doesn't*, however, mean that we should:
  - Give up
  - Take a wild guess
  - Use incredibly basic averages only
- Our approach:
  - **Define** exactly what we are calculating before running calculations
  - Use **historical data** where possible, and as soon as possible
  - Lean on **models and simulation** to help gauge the rest
  - Look for **ranges**, not just averages, and staff for that!

### Capacity Planning Summary

In this section, we've taken a perhaps slightly different and more details approach to capacity planning than you might have used in the past. While hopefully the principles we've discussed here seem to make sense to you, in the exercise you are about to do, you will get a chance to get hands on with these techniques and see what type of numbers they produce for you. While we discussed how much time is *required* in this section, we didn't address the other part of capacity planning, looking at how much time you expect each person to have available. In the exercise, we'll make estimates for both of these numbers as rigorously as possible, and compare, and hopefully produce a best possible guess for you and your team's needs.

# Course Roadmap

- Book 1: SOC Design and Operational Planning
- Book 2: SOC Telemetry and Analysis
- *Book 3: Attack Detection, Threat Hunting, and Triage*
- Book 4: Incident Response
- Book 5: Metrics, Automation, and Continuous Improvement

## SECTION I

### Introduction

#### Creating and Processing Alerts

- Efficient Alert Triage
- Detection and Analytic Design
- Capacity Planning
- **Exercise 3.1 – Capacity Planning**
- Detection Engineering

#### Advanced Analysis

- Analytic Frameworks and Tools
- **Exercise 3.2 – Structuring, Documenting, and Organizing Use Cases**
- Threat Hunting
- **Exercise 3.3 – Planning a Threat Hunt**
- Off-Hours Alerting and On-Call
- Active Defense
- Summary and Cyber42 – Day 3



This page intentionally left blank.

## EXERCISE 3.1

# Exercise 3.1: Capacity Planning

### OBJECTIVES

- Learn how to use numerical methods for capacity requirement estimation
- Estimate how much free time for investigation and alert response you have
- Use the Poisson distribution to bound your expectations of alert count
- Use the log-normal distribution to understand expected investigate time
- Get a deeper insight on how to estimate analysts needed for coverage



### Exercise 3.1: Prioritizing and Visualizing Attack Techniques and Security Controls

Please go to Exercise 2.1 in the MGT551 Workbook or virtual wiki.

# Course Roadmap

- Book 1: SOC Design and Operational Planning
- Book 2: SOC Telemetry and Analysis
- *Book 3: Attack Detection, Threat Hunting, and Triage*
- Book 4: Incident Response
- Book 5: Metrics, Automation, and Continuous Improvement

## SECTION I

### Introduction

#### Creating and Processing Alerts

- Efficient Alert Triage
- Detection and Analytic Design
- Capacity Planning
- Exercise 3.1 – Capacity Planning
- **Detection Engineering**

#### Advanced Analysis

- Analytic Frameworks and Tools
- Exercise 3.2 – Structuring, Documenting, and Organizing Use Cases
- Threat Hunting
- Exercise 3.3 – Planning a Threat Hunt
- Off-Hours Alerting and On-Call
- Active Defense
- Summary and Cyber42 – Day 3



This page intentionally left blank.

## In This Module

- Detection engineering as a discipline
- Analytic types
  - How we can label them to assist analysts in investigation
  - Analytic lifecycle stages
- Use case development
  - Development, storage, and organization
  - Use case database workflow
  - Data to track, metrics
  - Testing and sample data
- Playbook creation

## In This Module

Before we get into the deep details of the SOC's detection function we must consider what we need to detect and why. On any given day analysts will likely run into a wide variety of alert conditions. Whether new or experienced, the sheer number of possible situations means that all analysts will need to have a reference for alert conditions of interest and how to react when a condition is potentially detected. This is where use case documentation and playbooks come in.

In this section, we're going to begin our discussion of detection by focusing on the different types of analytics, how their intended use should be recorded with documentation to support them, and how that documentation can lead to the creation of playbooks to guide analysts down the right analysis path. While this discussion primarily is about detection, don't forget that the use cases you decide on implementing will ultimately drive the data collection required of your SOC. Without the proper data coming in, you wouldn't be able to detect the selected conditions.

## Detection as Code

- Contribute to the “Githubification of InfoSec<sup>1</sup>”:
  - Ask a team member to research open models and tool-agnostic projects like Sigma, YARA, and Jupyter, and share them with the team
  - Ensure that detections shared internally and with peer companies align to open tools and reference models like ATT&CK
  - Send your team members to training on Python or interactive notebooks
  - Use your voice as a customer to encourage vendors to support ATT&CK, Sigma, and Jupyter
- Analytics that are:
  - Carefully curated
  - Contributor friendly
  - Extensible



### Detection as Code

In his seminal blog post “The Githubification of Infosec,” Microsoft’s John Lambert described a community-based, democratized approach to analysis that enables teams to share actionable, product neutral use cases and detections. In the next section, we’ll discuss Sigma, Jupyter, YARA, and Mitre’s ATT&CK matrix, which all support a common language for describing and implementing detections. Even if you can’t share analytics outside of your organization, implementing this approach inside your own SOC will make you less reliant on vendor solutions to define your threat detection capabilities, better prepared to consume intelligence products, and flexible enough to swap out parts of your defensive infrastructure with minimal coverage losses.

Your ability to identify and respond to incidents is only as good as your analytics. Strive to maintain a proven set of detections that are carefully curated based on your intelligence and your threat model, support a wide range of contributors and use cases, and extensible enough to be implemented in multiple platforms, regardless of vendor.

[1] <https://medium.com/@johnlatwc/the-githubification-of-infosec-afbdbfaad1d1>

## Detection Engineering Concepts

- Involves *analytics* driven by *use cases*
- Inputs:
  - Threat model
  - Intelligence
  - Visibility
- Process and rigor varies depending on SOC maturity and detection pipeline
- Involves data sources, event pipelines, correlation and enrichment, and detection mechanisms
- **Goal** is complete data sources, simple but robust pipeline, flexible detections, and as much correlation, enrichment, and automation as possible



### Detection Engineering Concepts

Detection engineering is quickly growing into its own discipline in the SOC. It's an area where engineering and analysis blend together to identify use cases and implement those use cases in detection tools. It's also a function that looks very different depending on how mature your SOC is and the makeup of your detection system. Inputs to the detection engineering process are your threat model – recall the threat assessment and attack trees we created on Day 1 – your threat intelligence, and the visibility you have within your environment. The process itself includes your data sources, event pipelines, correlation and enrichment provided within those pipelines or in your SIEM, and whatever mechanisms you can apply to your data to create detections.

As a manager, you'll need to understand the challenges and constraints inherent in this work, which is why we've spent some time discussing fundamentals of analytic design and the sensitivity/specificity tradeoff. That said, reliable detection and escalation is difficult to do well even if you understand these concepts. In this section, we're going to cover the basics of custom analytics, discuss the use cases that drive analytic development, and the larger detection engineering process. Our goal is to have a complete data set from which to work, a simple but robust data pipeline (normally part of your SIEM infrastructure, but not always), a variety of flexible detection mechanisms, and as much enrichment, correlation, and automation capabilities as possible to maximize the value of your data set.

## Guiding Principles

### ***Advanced teams<sup>1</sup>:***

- Don't solve detection problems with analysts
- Know where, and how, analysis work is being created
- Follow engineering best practices
- Arm analysts with as much information as possible
- Don't give bad alerts a pass
- Don't wait for an incident to test their detections



### **Guiding Principles**

In his excellent blog post [Lessons Learned in Detection Engineering](#), Ryan McGeehan summarizes the guiding principles he has identified in the highest-performing detection engineering teams:

- *Great teams don't solve detection problems with analysts*, meaning that we want to rely on automated detections, and the scaling that detection platforms afford us, to identify evil – not manual review by analysts.
- *We should know where, and how, analysis work is being created*. Ryan refers to this as the “law of the lever”: good detection engineers are aware of the work imbalance created by adding new detections (they create the rule once, but the monitoring/analysis team must deal with potentially much more “signal” as a result) and do their best to avoid the creation of wasteful, downstream work.
- *Engineering best practices* means that we build a certain amount of rigor into our process; in the next few slides, we'll talk about the analytic lifecycle and the importance of thorough testing before rolling out new detections.
- *We must arm analysts with as much information as possible* to reduce the amount of time they need to understand and act upon new detections. We'll also cover some concrete ways to address this in documenting new use cases.
- *Don't give bad alerts a pass* if they must be tuned or retired.
- *Don't wait for an incident to test detections*; we'll cover this at the end of the week when we walk through adversarial simulation and purple teaming.

Much of the content in this section and throughout this class as it relates to analytic development aligns to these principles, starting with analytic rule *type* and *trustworthiness*.

[1]: <https://medium.com/startng-up-security/lessons-learned-in-detection-engineering-304aec709856>

## Analytic Rule Type

Rules should clearly indicate their **type** and **trustworthiness**

- **Anomaly** – Detection of an unusual, but *not necessarily malicious* condition
- **Investigative** – An alert that, if correct, would be malicious behavior
- **High-Fidelity** – Trusted alerts that should immediately be considered a true positive detection of malicious behavior (based on high historical accuracy)

		<u>Accuracy</u>	
		Low	High
<u>Rule Detects</u>	<i>Malicious Activity</i>	<u>Investigative</u> <i>Action: Qualify detection</i>	<u>High-Fidelity</u> <i>Action: Activate incident response</i>
		<u>Anomaly</u> <i>Action: Qualify detection &amp; examine context</i>	<u>Anomaly</u> <i>Action: Examine context</i>

### Analytic Rule Type

Analysts should be aware that not all analytics have the same intention. Some will be incredibly good at being right at catching evil 100% of the time, others may simply point out an anomaly. If an analyst approaches these two alerts with the same mindset, at best they will be inefficient, at worse they may waste time on a false positive or false negative. A way to ensure this doesn't happen is to either use a naming convention that designates which type of activity a rule detects, or store that information in a use case database such that it is easy to reference when a rule is triggered. If we break down the most common types of alerts analysts will see, they can be broken into analytics that catch anomalies, and analytics that catch alerts, and we can further subdivide them into whether they do it well (99%+ of alerts are true positive) or whether they are lower accuracy and must be investigated.

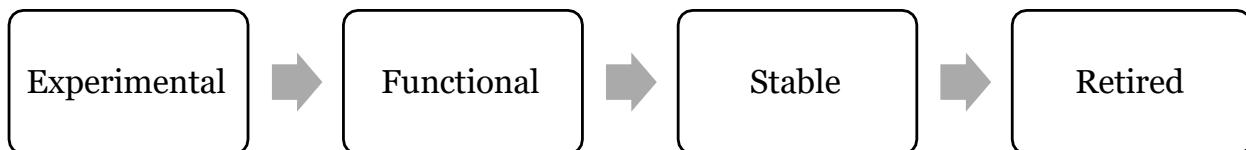
Within this combination, here is one naming convention that could be used to clearly signal to analysts what type of analytic they're dealing with.

- **High-fidelity** - Alerts that are tagged high-fidelity should be considered "always correct" and may even skip the triage process, immediately creating an incident ticket that incident response analysts can pick up. These are the best types of rules since they are nearly always correct.
- **Investigative** – Alerts tagged investigative *could* indicate badness, assuming they did indeed detect the thing they were attempting to detect. For example, maybe you wrote a snort rule for a specific piece of malware based on a URL pattern or User-Agent, but sometimes good applications may use the same data causing a false positive. That rule would be termed investigative, because if it is correct, malware is present, but there are known false positives that could and do occur.
- **Anomalies** – Anomaly alerts are the rules that detect things like "Word document with macro" or "EXE file in a zip file" or "JavaScript in a PDF". *Could* they be bad/weaponized documents? Sure, but even if the rule is right, it doesn't mean the file itself is necessarily bad. In the case of these alerts, instead of verifying the rule is correct (we're assuming it's accurate at detecting the condition of interest) they must analyze the context of the situation and decide whether that Word doc with a macro is actually a virus, or just a normal business document. If the rule is an anomaly rule, and is questionable at detecting the condition as well, analysts will need to both qualify the detection was accurate *and* examine the context of what happened. These rules should be evaluated if they are worth the effort.

## Analytic Rule Lifecycle

Consider a development lifecycle for tracking analytics

- Assists with metric creation and analytic tracking
- Gives analysts additional context
  - **Experimental** = Just made, brand new rule, testing feasibility
  - **Functional** = Somewhat vetted, still tuning, investigate with some suspicion
  - **Stable** = Thoroughly vetted, does not need to change unless situation changes
  - **Retired** = No longer needed/active analytics



### Analytic Rule Lifecycle

Another piece of information that should be tracked in your use case database related to your analytics is the stage of the lifecycle that analytic is in. While these may be correlated with the rule types described on the previous page, the idea isn't exactly the same.

The life of every analytic goes through similar phases. The analytic is first created and since it is new, it is still untrusted. In this stage, (which in my previous team we termed "experimental") rules are being tested for usefulness and feasibility. Alerts from this analytic may only alert the author when matched in order to not confuse the team at large. The author proves at this stage that the rule is viable and can either throw it out or decide to mature it after it is proven.

Once the analytic is relatively good it may be upgraded to what we called "functional". At this point, it still may be periodically tuned, but it should have proven itself useful and to generate a reasonable enough number of true positives to be worth sending the alerts to the main point of triage.

After an analytic has been thoroughly tested, it can end what might be called "stable" stage. At this stage the creator of the analytic is confident in its capability and has tuned it such that it should not need any further adjustment unless the situation it's detecting itself changes. Rules at this stage basically tend to stay static until they can be retired (if that day ever comes). Analysts that see "stable" rules firing should then know that they should have the fullest confidence that the rule is in its best possible state and can go into the triage and investigation with that assumption.

Aside from the additional context, tracking lifecycle stages has some practical benefits as well. As long as the lifecycle changes are date stamped, tracking these stages helps ensure all new rules don't sit in the testing stage forever and become forgotten. It also helps a SOC keep constant track of how many rules are active and in development which can show management the effort the SOC is putting in to improve detection.

## Use Cases

### Use cases:

- Define the conditions / attack techniques to detect
  - May be generic, high-level - "brute force login attempts"
  - May be more detailed, "External web application brute force login attempts"
- More than just the condition:
  - Why do you need to detect it
  - Where will you detect it
  - How will you detect it
  - Analysis steps when it is detected
- Ideally **documented** and **organized**, in a **structured** way
  - An example is coming in exercise 1.3...

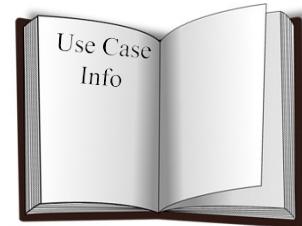


### Use Cases

Each condition you'd like to detect should be turned into a documented "use case". For each SOC, this may mean something slightly different, but the general commonality in what most SOCs call a use case is a well-documented condition they need to detect, as well as the reasoning and other details behind it. Putting this information into some sort of system in an organized fashion helps analysts reference it when needed and keeps management up to date on what is and is not covered in terms of analytics. We'll see some examples of documented use cases in upcoming slides and exercises.

## Use Case Databases

- New analytics need supporting info recorded
- Track them inside a designated *use case database*
- A dedicated place to answer:
  - What the analytic detects, why, how
  - What to do when it alerts
  - Known false positives
  - Metadata – data sources, framework alignment, author, lifecycle, change tracking, etc.
- A place to query for analytic-based metrics



SANS

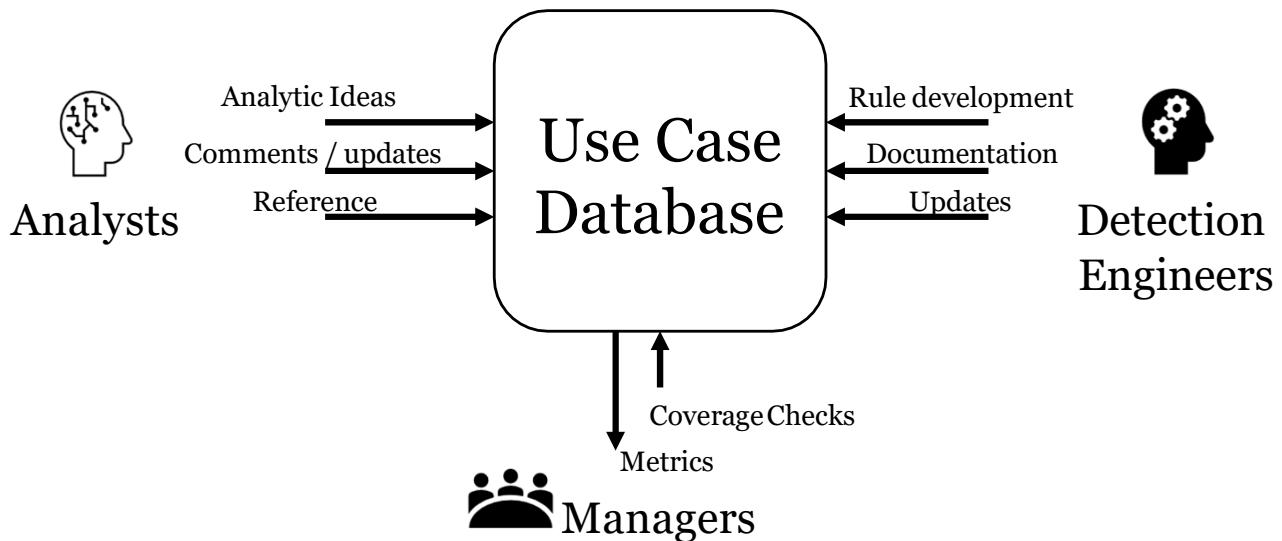
MGT551 | Building and Leading Security Operations Centers

71

### Use Case Databases

To document the new analytics your SOC should employ some type of *use case database*. This will be the designated location for analysts to document all the supporting details for how to interpret the analytic, why it exists, the data it relies on, its lifecycle stage, and more. Think of it as your own encyclopedia of detection information for the SOC. When analysts see an alert for the first time, they can refer to the use case information to understand how to interpret what they are seeing, known false positives, data sources, and other key information. In addition to serving to help analysts understand and interpret identified activity, the use case database should double as a way to generate metrics. As long as custom fields can be created to store each bit of information, and those fields can be queried via API it should be easy to produce metrics like the technology and data most commonly used for detection use cases, which ATT&CK techniques have detections implemented in stable status and more.

## Use Case Database Interactions



SANS

MGT551 | Building and Leading Security Operations Centers

72

### Use Case Database Interactions

A use case database is a system that will be used by multiple different people, each having their own type of interactions with it.

- Analysts – Analysts are the primary, and likely more important use of a use case database. They will regularly be interacting with the system to add new ideas, make comments on existing analytics, and use the system as a reference for additional information.
- Detection Engineers – Detection engineers (assuming you have them as a dedicated role), will be the ones architecting the analytics themselves, then documenting the details in the system for the analysts to see. Updates to use cases and other relevant information for any important changes will also be entered by whoever is doing this role. (Note that if your team does not include detection engineers, consider these interactions as done by whoever is writing and documenting analytics, possibly analysts)
- Managers – Managers typically have a very different interest in the use case database. Since managers are in charge of making sure the right things are being done and evaluating the state of the SOC, their interactions with the list of analytics will generally reflect this. Managers should be able to answer key questions like "do we have coverage for X?", and also should be able to pull automatic metrics from the system to show coverage against frameworks like the MITRE ATT&CK framework or any other information that is tagged on each use case.

## Suggested Fields to Track

- Name / UID
- Objective
- Lifecycle Stage
- Priority
- Author
- Analysis steps (playbook)
- Changes over time
- Data Sources
- Category/Type
- Technical Context
- Known false positives
- False negatives / blind spots
- Validation techniques
- Article references
- Analytic Logic
- Framework alignment  
(MITRE, VERIS, Kill chain)
- Compliance / audit support



### Suggested Fields to Track

Here are some of the fields that SOCs should consider tracking against their use cases. Remember each field tracked is another opportunity for important metrics that can show the effort your SOC is putting in outside of response tasks.

Although using all of these fields is not required, a minimum set should include the description/objective, category, author, change tracking, attack model alignment, and known false positives, false negatives, and blind spots.

## Use Case Database Platform Requirements

Requirements and desired features:

- Field customization capability
- Owners for each use case
- Nested parent/child items for organization
- Intel framework alignment capability
  - For tracking against Kill Chain, MITRE ATT&CK, etc.
- Lifecycle stage tracking
- Historical change tracking
- Export / API for metrics generation and backup
- More on this later...

### Use Case Database Platform Requirements

While there are many different tools you may want to use for use case tracking, each has its own strength and weakness. Again, the size and complexity of your SOC operation will also drive how formal you want to make the system as well. While smaller teams may keep it very simple with tools like Excel, larger SOCs may want to with custom solutions or tools like Jira or Redmine.

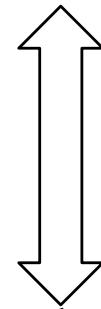
Regardless of the tools used, some of the most important features and requirements you should be looking for are listed on the page. Consider what you want to get out of your use case database and the types of interactions you want to have with it when picking a solution. The majority of us will want a tracking system that is simple to use, easy to customize, allows some sort of organization and change tracking, and ideally lets you generate metrics based on the fields in each use case. This capability allows you to answer questions such as "what tool is being used as the data source for the most use cases?" Knowing these answers then lets the use case database drive investments in new technology since you will be able to see where most of your detection capability is coming from.

## Use Case Database Storage Options

Potential use case database solutions:

- Excel
- Text files in Git repositories
- Markdown
- Wiki software
- Threat Intelligence Platforms
- Ticketing systems – Redmine, Jira, etc.

Simple,  
easy to learn



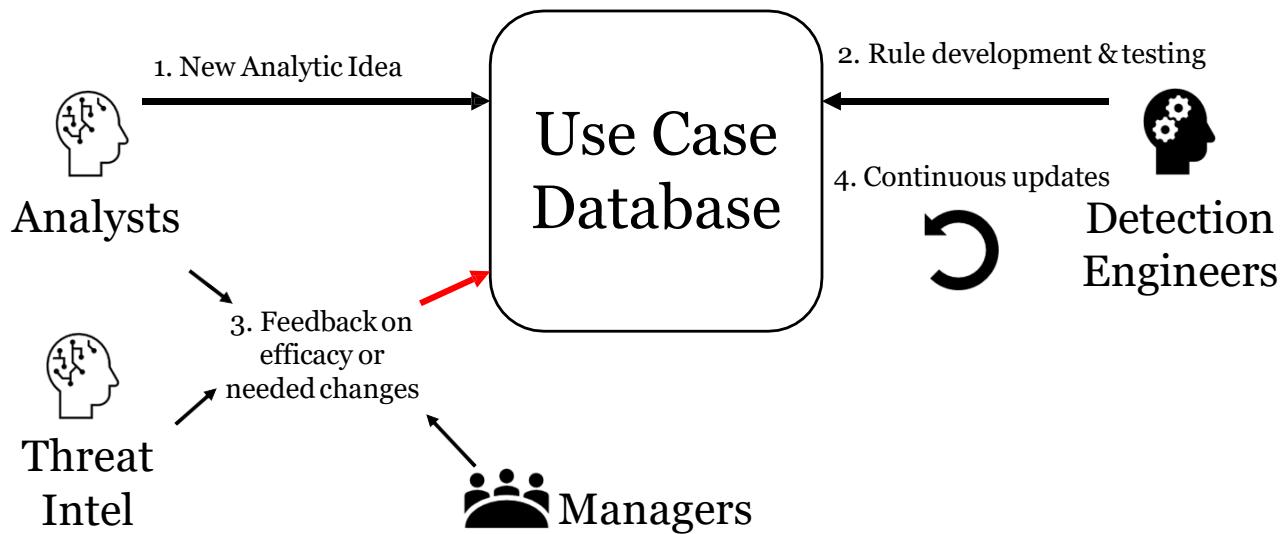
Complex, but  
highly flexible

### Use Case Database Storage Options

There are many potential solutions to storing use case database information and SOCs across the world have used methods as simple as Excel, Wikis, or markdown text files tracked via Git to complex systems using ticketing systems like Jira or Redmine. Your threat intelligence platform may even have a built-in tracking solution. What is highly important is that the solution matches your desired workflow and allows you to implement the custom fields required for your SOC, as well as enables you to generate metrics in a simple and automated way.

The arguably most important feature of the use case database, however, is that it is simple to reference and easy enough to modify such that the team will keep it up to date! You can implement the most complex and organized system in the world, but if no one wants to use the solution due to the pain involved, you will have achieved nothing. Look closely at the requirements you wish to track, but do not forget to keep usability as one of the prime requirements for the system.

## Typical Workflow



SANS

MGT551 | Building and Leading Security Operations Centers

76

### Typical Workflow

Typical workflow for a use case database is as follows:

- Analysts or threat intelligence produces a new idea or indicates the need for new analytic and enters it into the system with a status that indicates it needs to be created, as well as the priority.
- In parallel, all existing rules have a way to leave feedback and comments. Any rules that have failed, produced false positives, or acted in an unexpected way get marked as such, with the needed changes marked with a priority.
- The system shows detection engineers all use cases and their state of development or need for modification. As a daily activity, detection engineers or those responsible for new analytic creation check the queue, much like an analyst would check for new alerts, and find the most important analytic that needs development or modification.
- Working through in order of priority, detection engineers develop new analytics and modify existing ones, leaving comments and notes on their changes. Analysts and managers should be able to see what was done and the timeline of the change (it's important to time-stamp when analytic logic changes in case of a sudden change in false positive or false negative rates).

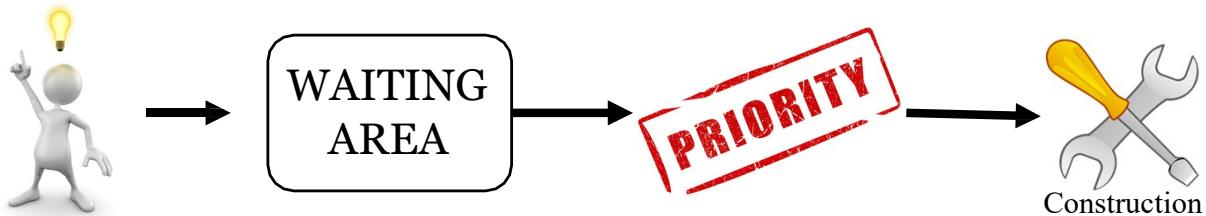
## Use Case Backlog

Analysts often have great ideas for new detection rules/changes

- But they may not know how to write it
- May not have time to write it

**Solution:** A backlog for new analytic ideas & improvements

- A great creative outlet and secondary activity for analysts



### Use Case Backlog

Since analysts are constantly seeing new attack techniques and learning new information about the environment they are defending, it's not uncommon to quickly develop lots of great ideas for new detection analytics. The problem, however, is often that analysts are either too busy in the moment, or otherwise unable to develop them, and these ideas can quickly be forgotten. To remedy the situation, the team should develop a process to capture these ideas in an analytic backlog of sorts. This way, all the ideas can sit in a queue and be prioritized for development. This queue can either be taken on by dedicated content engineers, if you have them, or use as a secondary task and creative outlet for analysts to pursue when alert queues are small. As the new analytic is proven out, the full documented use case should also be developed simultaneously.

## Use Case Database Example – Palantir ADS

Palantir described their alerting and detection strategy framework (ADS)<sup>1</sup> as follows:

1. Analyst has an idea for a new rule from news or threat intel source
2. Engineer starts development by emulating the technique and analyzing available telemetry sources
3. Engineer uses the findings to craft a new analytic, checks historical data for non-matching hits (**false positive testing**)
4. A new use case record is created in a GitHub repo with required info
5. **Peer review** of the new analytic, including **true positive testing**
6. Analytic is moved into production



### Use Case Database Example – Palantir ADS

If you'd like an example on how a mature SOC runs their analytic development, testing, and documentation, Palantir has posted their own in-house process in a Medium blog post entitled “Alerting and Detection Strategy Framework”<sup>1</sup> as well as a link to a GitHub repo with several filled out examples<sup>2</sup>.

In the post, Palantir describes some of the pitfalls and lessons they've learned and how they've developed a standardized process for new analytics to ensure they are durable, peer-reviewed, and documented before they go into production. That process is summarized on the slide above. As they express in the blog post, one of the key tenets of this system is documenting at the time of creation. This serves to ensure every alert has sufficient documentation, has been tested against false positives and false negatives, and is peer-reviewed to guarantee at least a minimum level of quality before the rule is placed into production.

1 <https://blog.palantir.com/alerting-and-detection-strategy-framework-52dc33722df2>

2 <https://github.com/palantir/alerting-detection-strategy-framework>

## Example Use Case – SSH Lateral Movement

**Goal:** Detect lateral movement attempts by monitoring for SSH connections from unexpected subnets.

**Categorization:** Lateral Movement / Remote Services

**Strategy Abstract:**

- Record connection attempts to destination port 22 via iptables
- Alert on any connections with a source IP outside the known admin subnet and scanner range

**Technical Context:** Attackers may use stolen credentials from unexpected locations to move laterally. This rule ensures that if they try, they will be caught.

**Blind spots and Assumptions:** We assume that iptables is up and running, and the logs are being centralized.

**False positives:** Vulnerability scanners and inventory scanners could trigger this if not suppressed in the SIEM

**Priority:** Medium

**Validation:** Test by SSHing to the server outside the expected IP range.

**Response:** Investigate the source IP, confirm the user or process that initiated the connection. Verify it is not malicious.



## Example Use Case – SSH Lateral Movement

Here's an example of the type of data and explanations you would want to track inside your use case database. This page shows how an analyst may document the reasoning behind the use case, why it exists, how it works, and other details (made brief to fit on the slide). This type of information is what analysts who are unfamiliar with the use case will need to know to understand the alert, how it works, and what it is intended to find.

## MITRE's CAR

- Cyber Analytics Repository
- Pre-defined, product agnostic analytics mapped to MITRE ATT&CK
- Includes data model for observables you can include in your analytics
- Specifies applicable platforms, open-source sensors (OSQuery, Sysmon, Autoruns) for observable collection



### MITRE's CAR

The MITRE Cyber Analytics Repository (CAR) is a set of analytics based on MITRE's ATT&CK matrix. CAR defines a product agnostic data model as well as targeted examples for specific tools (e.g., Splunk, EQL) in its analytics. The goal of the CAR project is to catalog a set of validated and well-explained analytics. Analytics stored in CAR contain the following information:

- A *hypothesis* which explains the idea behind the analytic
- The *information domain* or the primary domain the analytic is designed to operate within (e.g., host, network, process, external)
- References to ATT&CK Techniques and Tactics that the analytic detects
- References to MITRE's glossary of commonly-used terms in detection engineering
- A pseudocode description of how the analytic might be implemented
- A unit test which can be run to trigger the analytic

CAR also defines a data model for various observables (specific event sources) one might include in an analytic with references to specific tools one might use to collect the data. CAR can be a great starting point for building out a custom analytic repository, either as a reference for the kind of information your team might capture or as a starting point for a pre-defined set of detections aligned to the ATT&CK matrix.

You can learn more and fork the repository here: <https://github.com/mitre-attack/car>

## Use Case Development and Storage Summary

- Different alerts need different consideration
  - Anomalies vs. Investigative vs. High-Fidelity
  - New vs. vetted and stable
- Analysts should know what type and stage each is in
- This info and more can be kept in a use case database
  - Metadata and tracking info, response procedure, validation results etc. make for an encyclopedia of detection for the SOC
- “GitHubify” your detections
  - Use open standards and frameworks
  - Carry approach into interactions with peers and vendors

### Use Case Development and Storage Summary

In this section, we covered some of the useful mental models that analysts and others in the SOC can use to put themselves in the right mindset when triaging and investigating an alert. Knowing an alert is relatively new and identifies anomalies should almost always place it at a lower priority compared to a high-fidelity malicious activity detection. If analysts have no obvious method to tell which is which, however, the wrong call might be made leading to a delayed response of a critical issue. This is why you must track this information in a use case database and ideally have your alert triage system display it along with every fired alert. The more information and context analysts have, the more accurate they can be at assessing risk.

Use case databases can be stored in a number of ways, and there is no singular best option for everyone, but usability is one of the biggest concerns. These databases act not only as a reference, but also as a source of metrics showing the operational activity of the SOC—an important item to show for continued budget and support from the business. From these use cases, generalized playbooks can be made that guide analysts (especially new ones) in the right direction for alerts of a given type but should not be so restrictive that they become counterproductive. When breaking up alert investigation steps into playbooks, pay close attention for any automation opportunities. These are the atomic activities that often lend themselves to automation and quicker resolution of issues.

# Course Roadmap

- Book 1: SOC Design and Operational Planning
- Book 2: SOC Telemetry and Analysis
- *Book 3: Attack Detection, Threat Hunting, and Triage*
- Book 4: Incident Response
- Book 5: Metrics, Automation, and Continuous Improvement

## SECTION I

### Introduction

#### Creating and Processing Alerts

- Efficient Alert Triage
- Detection and AnalyticDesign
- Capacity Planning
- *Exercise 3.1 – Capacity Planning*
- Detection Engineering

#### Advanced Analysis

- **Analytic Frameworks and Tools**
- *Exercise 3.2 – Structuring, Documenting, and Organizing Use Cases*
- Threat Hunting
- *Exercise 3.3 – Planning a ThreatHunt*
- Off-Hours Alerting and On-Call
- Active Defense
- Summary and Cyber42 – Day 3



This page intentionally left blank.

## Blue Team Improvement: The Way It Used to Be

Every SOC is an island:

- Everyone gathers threat intel on their own
- Everyone writes their own analytics
- Everyone invents their own analysis techniques
- Clearly this is *super* inefficient 😞



### Blue Team Improvement: The Way It Used to Be

In the past, improvement on the Blue Team was tough, it was every team for themselves. In order to detect a threat, you would likely have to do your own research to produce your own threat intelligence. Then you would take that intelligence and develop it into your own analytic with logic you hope would work. Once an alert fired, you would also have to gather the data and hope you understood how to apply the required analysis methods to qualify it as a true or false positive. While advanced teams with experienced members wouldn't have problems with this after hitting a certain level of maturity, newer teams and newer analysts are left to learn on their own.

This state of being is obviously incredibly inefficient. Why should every team have to do this individually? An attack tactic that works at one organization, the analytic that finds it, and the analysis to verify it is likely highly similar across all organizations. The good news is things are now starting to change!

## Blue Team Improvement: The Way of the Future

Fortunately, the Blue Team community came to the rescue!

Standardization of attack descriptions	Standardization of file, packet & log analytics	Capturing complex analysis
<ul style="list-style-type: none"><li>Codified attack tactics, techniques, and groups that use them</li></ul> 	<ul style="list-style-type: none"><li>Generic detection languages for files, network traffic, and now logs!</li></ul>   	<ul style="list-style-type: none"><li>Repeatable complex analysis that's easy to share</li></ul>  

### Blue Team Improvement: The Way of the Future

In the last few years, we've seen an explosion of information sharing, attacker models, analytic tools, and sharable analysis methods that make these struggles a thing of the past. We share attacker tactic and techniques with attack models, the biggest and most popular of which right now is MITRE ATT&CK. The analytics we need to detect the usage of these techniques can be codified in Snort-style IDS signatures, YARA signatures for files, and now Sigma signatures for log content matching. Even analysis can be made repeatable with technologies like Jupyter notebooks and BinderHub, which allows complex analysis to be coded once and shared amongst anyone who can supply a fitting data set.

At long last being on the Blue Team has a bright future in which analysts from different organizations can efficiently share all types of defense knowledge with each other. Over the next few slides, we'll discuss the newest of these game changers, Sigma and Jupyter notebooks, in more specificity, as they look to be the next frontier in Blue Team advancement and standardization.

## MITRE ATT&CK Data Sources

- A standard for attack techniques and related metadata
- Each **Tactic** broken into **Techniques**
- **Techniques** are listed with **data sources**
- Therefore, we can map collected data sources to ATT&CK Technique coverage!

ID: T1197  
Tactic: Defense Evasion, Persistence  
Platform: Windows  
Permissions Required: User, Administrator, SYSTEM  
**Data Sources:** API monitoring, Packet capture, Windows event logs  
Defense Bypassed: Firewall, Host forensic analysis



### MITRE ATT&CK Data Sources

When looking to assess your threats and data collection, one method is to leverage the MITRE ATT&CK framework. This framework is broken up into tactics, which are further grouped into specific techniques (ways of accomplishing that tactic). These techniques are tied to groups that use them, attacker software that implements them, and even the mitigations that prevent them, making ATT&CK one of the most organized sets of data for getting a sense of your capabilities.

To assess your defensive position, a great activity is, therefore, to analyze your capabilities against the MITRE ATT&CK matrix of techniques. There are several ways of going about this activity, multiple of which we will do in exercises in this class. One specific method many SOCs use is to compare their collection capabilities against the listed "data sources" for each technique the SOC deems important. ATT&CK data sources list which logs or network information your SOC must collect in order to detect that technique in use.

## MITRE ATT&CK-Based Data Assessments

- **Input:** Security Data Sources, Threat Group OSINT
- **Output:** Prioritized ATT&CK Techniques
- **Tools:** ATT&CK Navigator<sup>1</sup>
- **Method:**
  1. Find which data sources are collected by the SOC
  2. Find which techniques your adversaries use
  3. Map data sources and attacker preferred techniques on ATT&CK Navigator
  4. Identify critical gaps and fix them!

### MITRE ATT&CK-Based Assessments

The assessment methodology, therefore, is straightforward. First, you gather a list of all types of data sources your SOC currently collects (items like proxy logs, DNS records, packet capture, etc.). Next, use this list to map those capabilities on the MITRE ATT&CK matrix, giving you a visual map of which techniques you should be able to catch in theory. Next, use your threat intel or the built-in information in the ATT&CK data set to see which techniques your most likely adversaries will use. If you identify any techniques for which you don't have any collection capability, you now know what to prioritize for changes!

## ATT&CK Navigator

- Run locally<sup>1</sup> or use online version<sup>2</sup>
- For visualizing gaps or technique coverage
- Tabular display of multiple “layers”
- Displays tactics/techniques with parameters
- Mark techniques using:
  - State (enabled/disabled)
  - Score (numeric)
  - Comment
  - Color (text / background)
- Use **layer math** to combine information!

Command And Control	Exfiltration
6 items	5 items
Commonly Used Port	Automated Exfiltration
Communication Through Removable Media	Data Compressed
Custom Command and Control Protocol	Data Encrypted
Data Encoding	Data Transfer Size Limits
Multi-hop Proxy	Exfiltration Over Alternative Protocol
Web Service	



### ATT&CK Navigator

To help facilitate the visualization of this and perform the activity we just described, MITRE has developed a tool called ATT&CK Navigator. ATT&CK Navigator can be used either online<sup>1</sup> or run locally<sup>2</sup> on your own system (Docker versions are available as well). Navigator is an interactive, web-based version of the ATT&CK matrix that lets users assign multiple parameters to each technique and has multiple layers of matrices with scores that can be combined together.

For your assessment, the first step might be to make a layer in ATT&CK navigator that represents the techniques for which you currently have the data sources. This is the starting point for which you can visualize all ATT&CK techniques you should be able to monitor for. If you want to add detail to the chart, multiple levels of "scoring" can be used to capture nuance. The picture in the slide above shows ATT&CK Navigator with numerically scored techniques using the coloring feature, allowing different colors to be assigned to each score. A score of zero can be represented as red for example, while a score of 10 could be shown as green. In addition to scoring, irrelevant techniques can be disabled and hidden from display, and comments can be attached to each technique.

One of the fantastic features of ATT&CK Navigator is that it enables doing math per technique on the scores from multiple instances or "layers" of matrices. Once detail on coverage by data source is scored, for example, you could create an additional layer with threat intel information scoring attacker favorite techniques. Using layer math, the overlap can now be easily produced with minimal effort!

1 <https://github.com/mitre/attack-navigator>

2 <https://mitre-attack.github.io/attack-navigator/>

## YARA<sup>1</sup>

- **"The pattern matching swiss knife for malware researchers"**
  - Like your own antivirus signature set
  - Defines strings, bytes, and other content to identify in files
  - Apply to files at rest or in transit
  - Widely supported by commercial and open source tools
- YARA resources:
  - <https://github.com/InQuest/awesome-yara>

```
rule silent_banker : banker
{
    meta:
        description = "Example signature"
        threat_level = 1
        in_the_wild = true

    strings:
        $a = {6A 40 68 00 30 00 00 6A 1}
        $b = {8D 4D B0 2B C1 83 C0 27 9}
        $c = "UVODFRYSIHLNWPEJXQZAKCBG"

    condition:
        $a or $b or $c
}
```



## YARA

One of the standards you may already be aware of that has become a large boon to the Blue Team is YARA.<sup>1</sup> YARA is a simple language for defining characteristics of files via byte patterns, strings, metadata, specific file format details, and more, designed to be applied anywhere files are collected or scanned. While in the past, scanning files may have been the lone job of our antivirus scanners, with YARA, the community can easily define and distribute signatures supported by a wide variety of both open-source and commercial tools.

Since it is highly likely you will want to produce your own file-based threat detections, your team should invest in learning the YARA language and storing any in-house developed signatures in a version control system. The simple text-based format makes change tracking and automated querying of your signature set easy, and having multiple tools draw on your YARA library means you write the signature once and it can be applied multiple places to spot a malicious file whether it is found on a hard drive or crossing the network. A suggestion for getting started with custom YARA detection would be implementing the excellent rule sets already created by Didier Stevens<sup>2</sup> and/or the set from Florian Roth.<sup>3</sup>

For an enormous number of YARA related tools, resources, and signatures, see the InQuest "awesome-yara" GitHub repo listed on the page above.

1 <https://yara.readthedocs.io/en/stable/>

2 <https://github.com/DidierStevens/DidierStevensSuite>

3 <https://github.com/Neo23x0/signature-base/tree/master/yara>

## Sigma

**"To logs, what Snort is to network traffic, and YARA is to files"**

- High-level **generic language for analytics**
- Written by **Florian Roth and Thomas Patzke**
- **Enables analytics reuse and sharing** across orgs
- MISP compatible: Share and store aligned with threat intel



### Sigma

Sigma, an open-source project on GitHub, led by researchers Florian Roth and Thomas Patzke, is emerging as the solution to one of the most frustrating problems in cyber defense: generic detection for log content.

Described as "to logs, what Snort is to network traffic, and YARA is to files", Sigma allows SOC teams to create, share, and distribute analytics that search for content in security logs the same way Snort can search for bytes and strings in network packets.

Up until now, every time you read a report about a new attack technique, every SOC on Earth would have to have a content engineer in the SOC come up with how to search for that specific condition with the SIEM or log management product they own, and the field names they use. With Sigma, one signature can be distributed to all security teams, and it is converted to the form needed to make it work in that specific environment. This project has the potential to revolutionize analytic sharing in cyber defense, and although not fully recognized by all vendors and tools, it has the most market share and continues to catch on. This, undoubtedly, is a project worth understanding and paying attention to.

[1] <https://github.com/SigmaHQ/sigma>

## Sigma Details / Benefits

- Pre-written analytics
  - Sigma default collection<sup>1</sup>
  - OSCD Community ATT&CK technique rules<sup>2</sup>
- Text-based YAML files...
  - Enable version control systems for analytics
  - Provide easy traceability to ATT&CK
  - Enable metrics and visualization on existing analytics via ATT&CK Navigator
- Eliminates vendor lock-in!
- Support growing among security tools

### **Supported Conversion Targets<sup>1</sup>**

- Splunk queries & dashboards
- Elasticsearch Query Strings / DSL / Watcher
- Kibana
- Logpoint
- Windows Defender ATP
- Azure Sentinel / Azure Log Analytics
- Sumologic
- ArcSight
- QRadar
- Qualys
- RSA NetWitness
- PowerShell
- LimaCharlie
- Grep with Perl-compatible regular expression support



### Sigma Details / Benefits

Aside from the obvious benefits of a standard way of writing generic signatures for logs, there are numerous, less obvious benefits as well. One plus of starting to use Sigma is the ecosystem of rules is already rather large. With the Sigma GitHub repo<sup>1</sup> coming with tons of rules and community efforts like OSCD<sup>2</sup> striving to get the Blue Team community together to cover the entire MITRE ATT&CK framework with Sigma rules, you'll be off to a fast start.

Another benefit is that Sigma rules are all text-based YAML format files. Each rule has its own text file that can be individually tracked and stored in a version control system which is great for ensuring rules can be tracked as they evolve or be rolled back easily when needed. In addition, each file has metadata sections that can refer to ATT&CK tactics and techniques the analytic covers. These data items can be automatically queried with the premade set of Sigma tools and placed on MITRE's ATT&CK navigator for coverage visualization.

Finally, moving to Sigma releases, your team from vendor lock in. It's not uncommon for a SOC to toss out one SIEM product as the industry evolves and an organization grows, but when you do, one of the biggest pains is the migration of all your analytics. If you implement Sigma, analytics only need to be run through the conversion process for the new product making migration a comparatively painless affair.

1 <https://github.com/SigmaHQ/sigma>

2 <https://oscd.community>

## Sigma Rule Format

Plaintext YAML files with:

### 1. Metadata

- Title, status, description, references, tags, etc.

### 2. Log Source

- What type, brand, and service is the log from?

### 3. Detection: List of Selectors

### 4. Condition: Logic for selector matching

```
title: DNS TXT Answer with execution
strings
status: experimental
description: Detects strings used in
command execution in DNS TXT Answer
tags:
- attack.t1071
author: Markus Neis
logsource:
  category: dns
detection:
  selection:
    answer:
      - '*IEX*'
      - '*Invoke-Expression*'
      - '*cmd.exe*'
  condition: selection
level: high
```

## Sigma Rule Format

This page shows an example of a Sigma rule on the right. As you can see, there is metadata on the top such as a title, status, description, author, and a tag to indicate the ATT&CK technique the rule relates to. Underneath the metadata is the log source the rule applies to, the detection logic, and the condition that must be met. Each of these fields is easily programmatically accessible due to the YAML formatting, which means auditing your rules for status, authors, and tags is incredibly easy. Knowing what you can and cannot catch, who's writing rules, where the data is coming from, and what each rule covers is a great way to perform gap analysis and ensure your SOC stays on top of the most important detections over time.

## Sigma Conversion Example

```
logsource:  
  product: windows  
  service: sysmon  
  
detection:  
  selection:  
    EventID: 1  
    ParentImage:  
      - '*\wscript.exe'  
      - '*\cscript.exe'  
    Image:  
      - '*\powershell.exe'  
  condition: selection  
fields:  
  - CommandLine  
  - ParentCommandLine  
level: medium
```



```
(source="WinEventLog:Microsoft-  
Windows-Sysmon/Operational"  
  (EventCode="1"  
  (ParentImage="*\\"wscript.exe" OR  
  ParentImage="*\\"cscript.exe")  
  (Image="*\\"powershell.exe")) | table  
  CommandLine,ParentCommandLine
```

```
SELECT 'CommandLine',  
'ParentCommandLine' from events where  
(LOGSOURCETYPENAME(devicetype)='Micros  
oft Windows Security Event Log' and  
("Event ID Code"='1' and (ParentImage  
ilike '%\wscript.exe' or ParentImage  
ilike '%\cscript.exe') and (Image  
ilike '%\powershell.exe'))
```

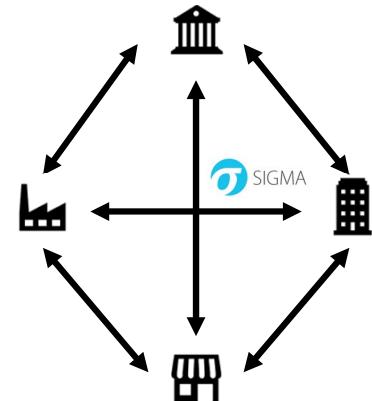
### Sigma Conversion Example

Here's an example of how the process looks. A generic Sigma format rule like what is shown on the left side of the page above is fed into the Sigma converter at each organization that wishes to use it. For example, let's say organization 1 uses Splunk while organization 2 uses IBM QRadar for their SIEM. Both organizations can feed the same generic rule into Sigma (which is configured for their product and fields) and organization 1 would receive the result in the upper-right box, a search query that can be dropped into Splunk in SPL language format. Organization 2 would receive the result in the lower-right box, a query that can be dropped into QRadar in AQL format. Instead of only storing the specific query in a use case database, the Sigma format rule can now be stored as well. This makes it easy to do analytic version control due to the simple YAML-based Sigma rule format, and also makes sharing the analytic with another organization simple.

## A World with Sigma

Imagine a world where intelligence reports come with Sigma rules!

- You now don't have to...
  - Write analytics
  - Test them (assuming they came from vetted source)
  - Don't even have to transcribe them
- Analytics are passed in Sigma YAML format
  - In vendor reports
  - On research sites and blogs
  - Via GitHub repos
  - Directly through your threat intel platform
- Converted and implemented immediately by your team!



### A World with Sigma

If we, as Blue Teamers, can get Sigma supported as a standard through the industry and our tools (or really any standard for generic log-based analytics) think about how incredible this would be. We could deploy analytics for new attack techniques as quickly and painlessly as deploying a new IDS signature. Vendors, blogs, GitHub repos, and our threat intel platforms can simply publish analytics in Sigma format, and our team can run them through the converter and apply them without having to worry about knowing specific field names, effectiveness testing (assuming we trust the source), or even the specific search syntax for our SIEM product! This would be a massive increase in ops tempo for defense teams as new detections can go live, worldwide, in all environments for log data as quickly as they could for IDS signatures!

## A Potential Sigma Alternative:YARA-L by Chronicle

### YARA-L<sup>1</sup> (the L is for logs)

- A new Sigma-like, log-based detection language by *Chronicle*
- Compatible and convertible to from Sigma, but takes it further
- Allows regex, and code-like
- Product specific for now, but claimed that open-sourcing is coming... keep an eye on this space
- White Paper for more info:  
[mgt551.com/yaral](http://mgt551.com/yaral)



```
profile susp_powershell_download_file {
meta:
  author = "Chronicle Security"
  description = "Rule to detect PowerShell one-liner to download a file"
  version = "0.01" created = "2019-12-16"
  reference =
    "https://github.com/swisskyrepo/PayloadsAllTheThings/blob/master/Methodology%20and%20Resources/Windows%20-%20Download%20and%20Execute.md"

condition:
  if re.regex(strings.lower.udm.process.command_line),
    ".*powershell.*net.webclient.*" then
    outcome.match()
  end
}
```

### A Potential Sigma Alternative: YARA-L by Chronicle

After many years of Sigma being the only effort in this space, Chronicle has now developed an alternative with a very similar aim named YARA-L, or YARA for logs. Created by many people involved in the original file-oriented YARA effort, YARA-L is designed for detection based on log content and has many of the features and benefits of the Sigma detection framework. There are some key differences between YARA-L and Sigma, however. One is that YARA-L allows more complex descriptions of what to look for in a log and enables code-like functions for defining what to do for data matching (some of this is due to specific Chronicle product features). The other big difference is that, for now, YARA-L is a proprietary language for Chronicle's product. The good news is that it has been implied through several tweets from those involved in YARA-L that the longer-term plan is to offer it as a standard for the entire community. Therefore, it is a language worth at least being aware of at this point so that its development and maturing can be monitored.

The right side of the slide shows an example of a YARA-L formatted rule. As you can see, the structure is quite similar to Sigma in many ways with metadata on top, and a condition for matching of logs based on the content of a field below. This example looks for the process command line field (udm.process.command\_line) to contain evidence of PowerShell creating a network connection as a condition for matching.

## Capturing Analysis with Jupyter Notebooks

- Jupyter notebooks allow us to capture **analysis techniques**
  - Ingest standard data format
  - Contain repeatable code blocks to step through
  - Provide visualization and live data ingestion capability
- In short: Jupyter notebooks capture complex analysis techniques for sharing!
- Even if...
  - Analysts don't know how to write Python code
  - Analysts don't understand how the analysis works
- Dramatically lowers the bar for applying data science



### Capturing Analysis with Jupyter Notebooks

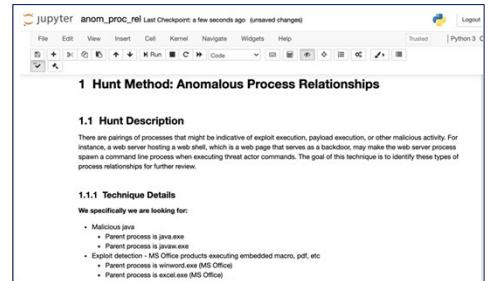
The sharing doesn't stop at analytics though. With the introduction of Jupyter notebooks, the information gathered for an investigation can now be run through standard data analysis procedures as well!

Jupyter enables anyone to write code (typically Python) and embed it in a shareable, web-oriented "notebook". With a copy of the notebook that contains the algorithm and your own data it needs for input, complex analysis techniques captured by analysts anywhere can be shared with organizations worldwide, regardless of whether the analyst knows how to write it from scratch or not. This innovation makes the threat hunting process and other complex analysis accessible to even the newest of teams, given they simply have their data and know how to set up Jupyter. Look for this to be a transformative technology for Blue Teams going forward.

## Jupyter Hunting Notebook Example

- YAML files store hunt configurations<sup>1</sup> ➔
- Analyst customizes config template with enrichments and data important parameters
- Notebook annotations and live data pulled in for analysis can be viewed in the notebook

```
1 # The notebook name is both the file name and the unique identifier for the hunt
2 notebook_name: anom_proc_rel
3 # The full name is used to reference the specific hunting technique
4 technique_name: Anomalous Process Relationships
5
6 # These are the fields that will need to be in every hunt to avoid key errors
7 column_list: ['record_id', 'timestamp', 'host', 'sensor_id', 'user', 'event_id', 'process'
8
9 # What is this hunt technique, what is it looking for? Reference the Threat Hunting Libra
10 hunt_description: |
11   ## Hunt Description
12   There are pairings of processes that might be indicative of exploit execution, payload
13
14 # This is the technical description of the hunt and what type of data is involved. Refere
15 technique_details: |
16   ## Technique Details
17   ===We're specifically we are looking for:===
18   * Malicious java
```



### Jupyter Hunting Notebook Example

Abstracting hunting analytics with Jupyter can be a tricky task if your SOC doesn't have much experience with data analytics. Managed services provider Expel has published a series of blog posts and presentations describing their approach to threat hunting using Jupyter. In this example, we see YAML configuration files that can be used to generate notebooks for various hunting use cases and data sets. Once the notebook has been created, the user can actually execute the code that is described in the notebook to hunt through the target data set.

Expel has also open sourced their configuration templates, which you can download here:  
[https://github.com/expel-io/notebook\\_builder](https://github.com/expel-io/notebook_builder).

1. <https://expel.io/blog/how-to-create-maintain-jupyter-threat-hunting-notebooks/>

## The Future: Binder and Jupyter Book Use Case Databases?

- *Binder* turns a Git repo into interactive notebooks
  - *BinderHub* uses Kubernetes to auto-scale multiple instances
- *JupyterHub* gives multi-user access to Binder built notebooks
- *Jupyter-book* turns notebooks into documentation
- Blue Team Usage:
  - Automated build and distribution of playbooks, analysis, documentation
  - Interactive use case database, visualizations with live data, dashboards, more?



SANS

MGT551 | Building and Leading Security Operations Centers

97

### The Future: Binder and Jupyter Books Use Case Databases?

Looking into the crystal ball, these technologies appear to be catching on quickly and it seems we might be able to predict where it will lead us. Given the supporting technologies around notebooks like Jupyter such as JupyterHub, Jupyter-book, and Binders / BinderHub, we may be headed toward a world where Jupyter is not just used to share analysis but could act as an interactive use case database and threat hunting documentation engine that could pull in live data. How cool would that be! With the way these technologies build on each other, it's easy to see how this could happen, and make the whole process highly automated as well. Binders (or binder-compatible repositories as they could be called) as Git repos containing instructions on how to automatically build the Python or other language environment required for analysis and put it into a Docker container. JupyterHub is the server technology that then allows multiple users to share and access the same notebooks, each with their own version and data. BinderHub is another technology that can be used with Kubernetes to automatically scale the whole system so that it can be used by an arbitrary number of users. An example of this already starting to happen is hunters-forge GitHub project and Threathunterplaybook.com<sup>1</sup>. These notebooks take example data set and show how analysis can be repeatedly applied to find ATT&CK techniques by anyone who loads the notebook.

Aside from this, Jupyter-book is another effort designed to produce documentation in a wiki-book style that can be converted into printable materials as well as accessed online and used interactively. It's not hard to imagine a world where use case databases migrate into Jupyter-book compatible formats. Needless to say, with so many new technologies that build on each other coming our way, it's an exciting time to be on the Blue Team! See the mordordatasets.com site for an example of a Jupyter-book.

[1] <https://github.com/OTRF/ThreatHunter-Playbook>

## Sample Data Sets and Tools for Analytic Testing

Pre-captured logs and network data for analytic testing:

- MORDOR<sup>1</sup>
- EVTX-ATTACK-SAMPLES<sup>2</sup>
- Splunk Boss of the SOC data<sup>3</sup>
- Secrepo.com
- NETRESEC PCAP file list<sup>4</sup>
- Malware-traffic-analysis.net



### Sample Data Sets and Tools for Analytic Testing

If you're looking for examples of attacks as captured by Windows logs, malware samples, or PCAPs with exploits in them for analytic testing, there are a number of high-quality resources available. The datasets listed here can be used if your analysts would like to test a detection hypothesis or validate a technique that they have developed against a "real" attack.

1 <https://github.com/OTRF/mordor>

2 <https://github.com/sbousseaden/EVTX-ATTACK-SAMPLES>

3[https://www.splunk.com/en\\_us/blog/security/boss-of-the-soc-2-0-dataset-questions-and-answers-open-sourced-and-ready-for-download.html](https://www.splunk.com/en_us/blog/security/boss-of-the-soc-2-0-dataset-questions-and-answers-open-sourced-and-ready-for-download.html)

4 <https://www.netresec.com/?page=PcapFiles>

## Testing with Full Lab Environments

You can even quickly spin up a full environment!

- **Detection Lab<sup>1</sup>** – Packer / Vagrant automated lab
  - Microsoft Advanced Threat Analytics
  - Windows Event Forwarding + Splunk
  - PowerShell Transcription logging
  - OSQuery
  - Sysmon
  - Autoruns sent to Windows events
- **Ypsilon<sup>2</sup>** – similar, uses Cuckoo and supports Sigma



DETECTIONLAB

SANS

MGT551 | Building and Leading Security Operations Centers

99

### Testing with Full Lab Environments

If you don't have a testing machine available, no problem. There are several open-source projects out there designed to painlessly spin you up an entire environment instrumented with host and network logging enabled to a SIEM!

One of those options is Detection Lab<sup>1</sup>. Detection Lab uses Packer and Vagrant scripts to immediately and automatically configure multiple hosts with best-practice logging enabled so you waste minimum time on lab setup and maximize time on high-caliber analytic testing. Included in the lab is Microsoft's Advanced Threat Analytics tool (ATA), Splunk forwarders pre-configured to ingest all data sources with indices already made, Palantir's Windows Event Forwarding setup for log collection, PowerShell transcript logging, OSQuery installed on each host, Sysmon installed and configured with SwiftOnSecurity's configuration, and all autoruns items logged to Windows events. Talk about a highly monitored setup! With this configuration, testing any new attack technique should very quickly reveal opportunities for detection.

Ypsilon<sup>2</sup> is another project with similar setup (Splunk, and automated VM setup with Ansible) but uses Cuckoo sandbox and also supports Sigma rules as well!

1 <https://github.com/clong/DetectionLab>

2 <https://github.com/P4T12ICK/ypsilone>

## Analytic and Analysis Framework Summary

As the saying goes - "*If you want to go fast, go alone. If you want to go far, go together.*"

- The future for the Blue Team is bright!
- The more we can organize, the more cyber defense improves
- Shared data now includes:
  - **Tactics and Techniques** with ATT&CK
  - **File detection analytics** with YARA
  - **Log detection analytics** with Sigma
  - **Analysis** and threat hunting with Jupyter notebooks
- If your team develops any of these, please consider contribution

## Analytic and Analysis Framework Summary

As the old saying goes "If you want to go fast, go alone. If you want to go far, go together." This is clearly true for the Blue Team as we've seen the uptake speed for new standards such as YARA and Sigma happen quickly throughout the industry. Blue teams yearned for a standardized common language for attack tactics and techniques, which brought the high-speed adoption of the ATT&CK framework. As these trends continue, Blue Teams, the world over, will become stronger and more capable as the discoveries of one team are increasingly easily shared with others, forcing the attackers to continuously invent new attacks.

Key recommendations from this section:

- If you aren't already, check out support for YARA and Sigma signatures within the tools you already have access to. If they don't support them yet, request it from your vendor.
- Consider storing your own analytics built with these formats. The simple text-based formats will help with version control and make the information you produce more sharable with the community.
- As you develop new and novel detections or analysis methods, consider sharing them at conferences, on GitHub, or anywhere else where the whole Blue Team community can benefit.

# Course Roadmap

- Book 1: SOC Design and Operational Planning
- Book 2: SOC Telemetry and Analysis
- *Book 3: Attack Detection, Threat Hunting, and Triage*
- Book 4: Incident Response
- Book 5: Metrics, Automation, and Continuous Improvement

## SECTION I

### Introduction

#### Creating and Processing Alerts

- Efficient Alert Triage
- Detection and AnalyticDesign
- Capacity Planning
- *Exercise 3.1 – Capacity Planning*
- Detection Engineering

#### Advanced Analysis

- Analytic Frameworks and Tools
- *Exercise 3.2 – Structuring, Documenting, and Organizing Use Cases*
- Threat Hunting
- *Exercise 3.3 – Planning a ThreatHunt*
- Off-Hours Alerting and On-Call
- Active Defense
- Summary and Cyber42 – Day 3

This page intentionally left blank.

## EXERCISE 3.2

# Exercise 3.2: **Structuring, Documenting, and Organizing Use Cases**

### OBJECTIVES

- Develop an example use case and categorize it with relevant details
- Enter the use cases into a tracking system
- Use the use case database system to look at metrics about your use cases
- Learn how to use the free open-source Redmine project management software for effective and free SOC data organization



### **Exercise 1.3: Structuring, Documenting, and Organizing Use Cases**

Please go to Exercise 1.3 in the MGT551 Workbook or virtual wiki.

# Course Roadmap

- Book 1: SOC Design and Operational Planning
- Book 2: SOC Telemetry and Analysis
- *Book 3: Attack Detection, Threat Hunting, and Triage*
- Book 4: Incident Response
- Book 5: Metrics, Automation, and Continuous Improvement

## SECTION I

### Introduction

#### Creating and Processing Alerts

- Efficient Alert Triage
- Detection and Analytic Design
- Capacity Planning
- Exercise 3.1 – Capacity Planning
- Detection Engineering

#### Advanced Analysis

- Analytic Frameworks and Tools
- Exercise 3.2 – Structuring, Documenting, and Organizing Use Cases
- Threat Hunting
- Exercise 3.3 – Planning a Threat Hunt
- Off-Hours Alerting and On-Call
- Active Defense
- Summary and Cyber42 – Day 3



This page intentionally left blank.

## In This Module

- What is Threat Hunting?
- Who should Threat Hunt?
- Threat Hunt scheduling
- Requirements and mindset of a hunter
- Threat Hunting methods and process
- Improving your hunt team maturity
- Demonstrating and documenting impact



### In This Module

One very important activity that SOCs focus on with varying level of success is threat hunting. While some organizations have very mature, defined-process threat hunting operations, others just seem to call day-to-day work "threat hunting" without it actually meaning anything specific. Considering the importance of true threat hunting, we want to manage a SOC of the former type. One where threat hunting is a well thought out, intentional process that supports the rest of the SOC activities and is part of the SOC's core competencies, and this module will cover how to start or continue down that path.

In this section, we're going to cover the consensus view about what hunting should be, why we should do it, who should do it, and the process to do so. We'll discuss the questions to ask, methods to form hypothesis, types of data to look for, as well as how to measure your threat hunting operation. Equally important, we'll also cover how to demonstrate to those outside the SOC that it is a valuable activity.

## What Is Threat Hunting?

### Assuming prevention has missed something

- Looking for **threats already in the environment**
- A **data-driven, proactive** search
- Follows from "presumption of compromise"

What is a **threat**? A threat must have

- **Intent**
- **Capability**
- **Opportunity** ... to do harm<sup>1</sup>



#### What Is Threat Hunting?

First of all, what is Threat Hunting? The short answer is threat hunting is looking for threats that have *already* bypassed our attack prevention measures and now have some control inside our environment. We assume this has occurred due to the previously mentioned "presumption of compromise" in our modern defense mindset. The key piece here is the focus on adversaries that are *already inside the environment*, which is what separates threat hunting activity from the normal day-to-day activity of protecting the environment.

A great intro resource on the topic of Threat Hunting, is the SANS Analyst Whitepaper paper written by SANS Instructors and Course Authors Rob Lee, well, Rob M. Lee<sup>1</sup> (yes, we have two of them.) In this paper, Rob M. Lee defines that a threat is anyone with the intent, capability, and opportunity to do harm. Without these three factors, that entity ceases to become an issue to you because they either don't want to, don't know how to, or can't attack you.

[1] <https://www.sans.org/reading-room/whitepapers/analyst/membership/36785>

## SANS Threat Hunting Survey Data

- Majority of organizations rated themselves as immature or in the process of maturing their threat hunting capabilities<sup>1</sup>
  - Only **37%** claimed they formally measure success and effectiveness
  - 78% of orgs base hunts on threat intel feeds
  - **Poor threat intelligence, lack of skilled hunters** listed as common challenges

### SANS Threat Hunting Survey

The SANS Threat Hunting Survey was designed to gain a better understanding of how organizations approach threat hunting, the barriers to success, and how those organizations measure their efforts. Analysis of the 2019 survey data offers various insights into our common understanding of threat hunting (spoiler alert: there seems to be little common understanding) and how organizations view the daily tasks of SOC analysts versus those of threat hunters.

Many respondents to the survey asserted that the demand for experienced threat hunters appears to outweigh the supply. The second challenge respondents face is the quality of threat intelligence feeding into the hunting process. Even though most respondents consume some type of threat intelligence for their hunting operations, only one of every three respondents said that they are highly satisfied with their sources.

The numbers tell a promising story for the future of threat hunting but also indicate that we have much work to do. 65% percent of respondents indicated that they already perform some form of threat hunting, and 29% plan to do so in the next 12 months. While 70% of respondents have dedicated in-house staff doing the threat hunting, only 29% believed that they are mature or very mature when it comes to performing the task. Digging into threat hunting approaches is a bit less reassuring. 78% of respondents indicated use of threat intel feeds compiled by either general security vendors or intelligence vendors, which is not by itself a bad thing. However, surprisingly, many organizations also indicated a preference for buying intelligence based on quantitative measures, such as the number of indicators (mostly due to the individuals leading the purchasing process).

Based on the 2019 survey, threat hunters mostly rely on tools such as SIEMs, IDS/IPS or EDR and generally have high confidence in these tools. To run threat hunting operations, hunters need well-curated and accurate threat intelligence that includes context for every indicator. Good intelligence and qualified staff emerged as clear trends in the challenges that organizations face to building and maturing their threat hunting efforts.

[1]<https://www.sans.org/reading-room/whitepapers/analyst/membership/39600>

## Why Should We Threat Hunt?

- Attackers are *very* good at what they do
- Well-funded attack teams:
  - Can launch attacks and cause damage in minutes
  - May persist in the environment for months/years
  - Use TTPs you may have never encountered before
  - Will likely not be caught by typical *reactive, alert-driven* process
- Threat Hunting's **proactive, data-driven approach** is needed to find anomalies and missed evidence
  - Seeks the "unknown unknowns"

### Why Should We Threat Hunt?

Is threat hunting a necessity? With a modern defense mindset and the presumption of compromise, the answer is a solid "Yes!" Though you may not think you will be the target of an advanced persistent threat-type actor, many organizations find themselves surprised. Even if you do not directly hold data that an attacker is interested in, you may work with vendors, produce software, or have connections to another organization that does. Therefore, there are still plenty of non-obvious reasons that a very highly skilled attacker may come after you, even if you don't know them, and because of this, threat hunting should be considered a necessary activity.

Threat hunting takes this assumption and runs with it. If you will be experiencing advance attacks, you should then also assume those attacks may very well be beyond your *initial* capability to detect them. Advanced attackers may use TTPs you've never heard of before, custom malware, and zero-days, and if they do, you're pre-existing alerting infrastructure may miss them. It is this assumed miss where threat hunting's data-drive, proactive threat-seeking approach steps in to balance out the odds. In short, why should you threat hunt? Because in all likelihood, you either currently are, or will one day miss something very important with your existing defenses, and threat hunting is your backup plan. It seeks the "unknown unknowns" the attack you didn't even know was possible.

## Hunt Team - Who

- Who?
  - While **any team can hunt**, maturity and team experience matters
  - Sr. Analysts and above may return the most value
- The ideal hunter has...
  - **Analysis skills and mindset**
  - **Environment familiarity**
  - **Understanding of the available data**
  - **Proficiency with all SOC tools**
  - Offense experience / knowledge
  - Knowledge of potential biases (confirmation, congruence, anchoring)

### Hunt Team – Who

How do we decide *who* should be performing threat hunting duties? While any team *can* hunt, your results will highly depend on a number of factors. The quality and availability of the telemetry data you have, as well as the analysis skill and experience of the person doing the hunting will all be major factors. The ideal hunter understands the data they have, how attacks look, where they would be recorded, and the environment they're looking for them in. Success can still be found without all of these factors, but each contributes a meaningful piece to finding the results you're looking for. Because of this, you may want to assign the most senior people on your analyst team for threat hunting. This does not mean it's impossible for newer analysts to generate insight and attack detections, it's just more likely that senior people will be able to consistently produce results you're looking for in the least amount of time.

## Threat Hunting Schedule

When should you be threat hunting?

- **Constantly!**

- Remember – "presumption of compromise"



Multiple options for scheduling—depending on team size:

- Defined portion of the day per person
  - On smaller teams, analysts may wear multiple hats: hunter/IR/etc.
- One rotating person per week dedicated to hunting
- Dedicated full-time "threat hunters"

### Threat Hunting Schedule

How often should you have analysts threat hunting? As often as possible! Remember the “presumption of compromise” from the modern defense mindset discussed earlier? If you assume the network always has some level of compromise present, your threat hunters should be heading out with the mindset of “we’re compromised until proven otherwise” instead of the converse.

There are multiple ways to divide the threat hunting load across the capable analysts and how you do so may depend on the size of your SOC. If you have a small team, you may direct certain people to spend a specific portion of their day each day dedicated to threat hunting. Alternatively, you can have one person dedicated via a “hunting rota” each week. This person will be excused from all normal alert triage and investigation requirements and allowed to explore data and new detection techniques throughout the entire week or day. If you have a large team, you may have enough team members to have dedicated full-time threat hunting staff!

## Data Quality

- For hunting, high-quality data is required
  - Low-data quality can seriously hamper your hunting
  - Leads to hunters wasting time cleaning and normalizing data
- Consider
  - Completeness (audit policy)
  - Field name normalization
  - Categorization
  - Parsing
  - Formatting
  - Accessibility



### Data Quality

One of the items that can *very* easily get in the way of threat hunting is low-quality data. One key difference that moves a group from a novice threat hunting capability and a more mature one is a set of clean reliable data. Since threat hunting requires searching through potentially gigabytes of data from thousands of devices, small data issues can balloon into huge time wasters.

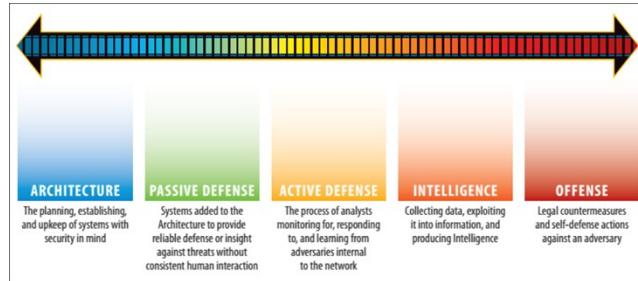
Ask your threat hunters:

- Are they spending time fighting understanding which field names match across different data sources?
- Are they understanding what common events like a login look like from different devices that record the logs in very different ways?
- Are the logs they want to use for threat hunting correctly and reliably parsed?
- Is the format conducive to quick searches and automated ingest? (Syslog for example is much more difficult to understand than well-formed JSON)
- Are the logs accessible or do they have to jump through hoops or wait days for data to extract before beginning their hunting sessions.

These are just some of the key questions you can ask threat hunters. It might pay to watch over the shoulder of someone on a hunting session and see where they spend time that isn't strictly looking for malicious activity. Small and simple improvements in audit or collection policy, or tweaks to log agents might have a large payoff in simplifying the data used for threat hunting.

## Preparing to Hunt

- Consider Robert M. Lee's *Sliding Scale of Security*<sup>1</sup>
- Mature threat hunting falls into Intelligence/Active Defense tiers
- Have a process – hunts can be unplanned, shouldn't be ad hoc
- Visibility is important but pre-collection is not required
- Hunts are at least partially manual



### Planning a Hunt

Before your team dives headlong into threat hunting, there are some basic controls and functions your SOC team should have. Refer to Robert M. Lee's *Sliding Scale of Maturity*, where organizations must focus on building the necessary foundational capabilities on the left side of the scale before pursuing more advanced capabilities on the right. A mature threat hunting capability falls into the Intelligence/Active Defense tiers, so if you haven't addressed secure architecture design or security monitoring yet then those are likely better places to start.

Threat hunting can be unplanned. Expect your leadership to hear something via "NEWSINT" and ask you to look through your telemetry to find evidence of \$fancynewattack. However, a hunt should not be ad hoc, lacking any structure or repeatable process guiding how it is executed. As with security monitoring, visibility is also a key ingredient in threat hunting. If you aren't collecting network and host data in a SIEM, NDR, or EDR, it may be challenging to find the data you need. But you shouldn't be afraid to look beyond the data you've pre-collected to dig into less obvious sources. Finally, it's important to understand that the nature of threat hunting is the search for non-obvious attacks; meaning, they are use cases that have *not* been pre-defined or configured in your automated alerting. Therefore, threat hunting is going to be at least partially manual and isn't something you can achieve through entirely automated tools.

Next, let's walk through some basic steps to prepare and execute a hunt.

[1] <https://www.sans.org/reading-room/whitepapers/ActiveDefense/sliding-scale-cyber-security-36240>

## Threat Hunting Process Overview

Formulate Hypothesis

What would *our* adversaries do?

Define Evidence

What evidence would this attack leave?

Data-Source Identification

Which data sources are available in our environment that would identify this evidence?

Gather and examine the data

Does it show compromise?

Respond to findings

Invoke incident response if needed

Analytic and protection improvement

Close the loop!



### Threat Hunting Process Overview

While the specifics of what you are hunting for will vary with each hunt, analysts should follow a structured process for coming up with what they are looking for while hunting to guide them. The process can be summarized as shown on the slide above.

1. Formulate a Hypothesis: What attack technique are we looking for, what would our adversaries do in our specific environment to try to steal our specific data? In this stage, analysts are thinking like an attacker. This is where having some experience in information security (especially on a Red Team), or doing pen testing, can come very much in handy.
2. Define Evidence: If the attack took the approach, we assume in step one, what evidence would that technique leave behind. Think both network and host-level artifacts.
3. Data Source Identification: Given the type of data you typically collect can you pull this information from your SIEM or PCAPs? If you have an EDR-type system that can query endpoints en masse, are there any forensic techniques you could use to acquire the data?
4. Gather and examine the data: Your analyst must know how to use the tools with proficiency to extract the data identified in step 3, ideally in an efficient and structured way. This is where having a SIEM that quickly returns searches can be of huge help as threat hunting often requires searching through large swaths of data from many machines.
5. Respond to findings: Once the data is collected, does it show evidence you predicted would exist if an attack were to have taken place? If so, time to spin up incident response!
6. Analytic and Protection Improvement: This is arguably the most crucial step. Once you have hunting in a successful way, you shouldn't have to do that hunt again, it should just become "the way". Analysts that find new successful techniques should automate them where possible and write any new analytics that can help identify that attacker or those techniques in the future.

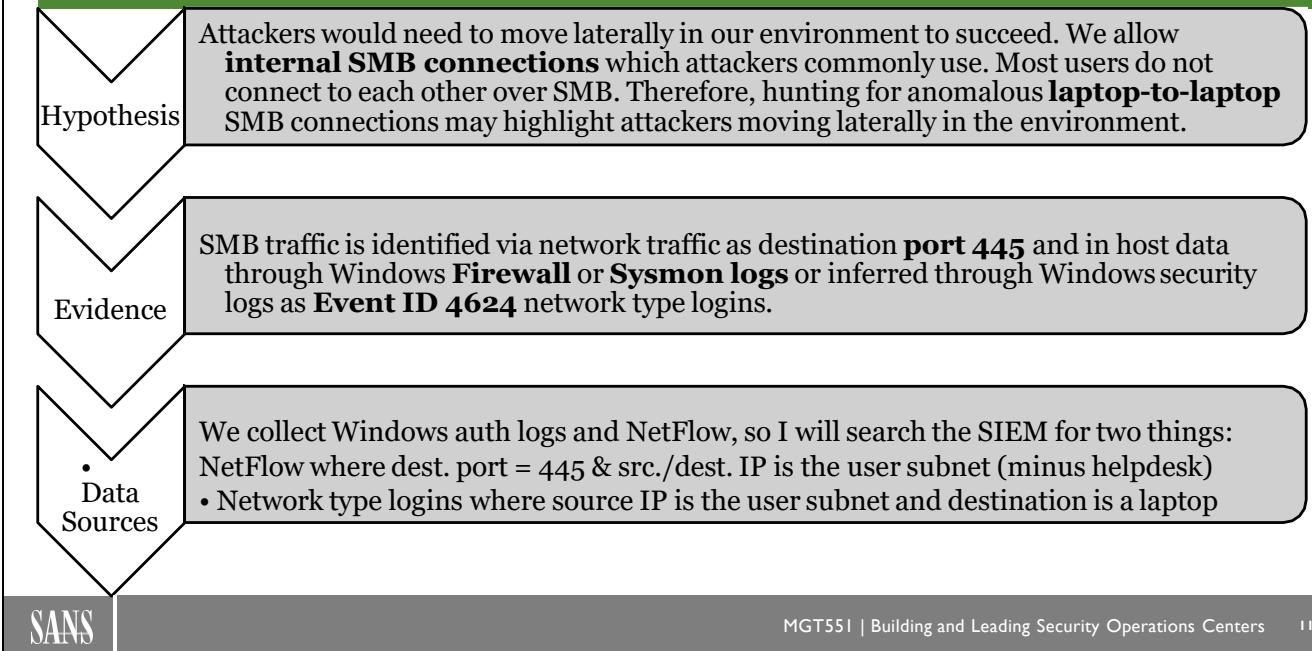
## Executing a Hunt

- Start with a standard framework or reference model to avoid bias, purely reactive hunting (via NEWSINT, for example)
- Develop a hypothesis statement
- Refine via questions and conditions:
  - “Would attacker do x?”
  - “Users will normally do x but not y”
- Gather evidence
- Pivot until your theory is disproven or you discover an incident, gap, or useful new detection

### Executing a Hunt

Executing a threat hunt doesn't have to be mired in process, structure, and documentation (though we *do* like documentation, right?). In many cases, hunts are unexpected activities that must be undertaken when your management asks a question to which you don't have a ready answer, you learn about a critical new vulnerability that is actively being exploited, or other unanticipated events. Especially in these cases, it is important for your hunt to be quick, agile, and iterative. Start with this basic workflow of question > hypothesis > refinement > evidence gathering > pivot until you have exhausted your data sources or have the answer you seek. In the next slide, we'll look at an example of a threat hunt as we move through some of these actions.

## A Simple Threat Hunting Example



SANS

MGT551 | Building and Leading Security Operations Centers

114

### A Simple Threat Hunting Example

Following the previous slide process, here is what an analyst embarking on a quick threat hunt may formulate in their head.

**Hypothesis:** I want to find lateral movement in the environment, and I know one common way is for attackers to leverage SMB. SMB uses destination port 445, and while this is very common inside Windows networks, it is *not* common for individual users to use it directly between each other. If a list of SMB connections that were made directly from one laptop to another can be produced, the associated context will likely highlight whether it is legitimate (perhaps from an IT laptop to a user for a fix), or illegitimate (a person in Engineering connecting to an HR laptop).

**Evidence:** When an SMB connection is made, it could be observed in multiple ways:

- Network data: Network-based data that could highlight SMB connections would be NetFlow, Layer 7 metadata, or full PCAP data that shows a destination port 445 connection from an IP address in user subnets (minus IT and the helpdesk) to another user subnet device.
- Host data: Windows records network connections locally using Windows firewall logs and in tools like Sysmon that locally record network connections. It's also recorded as an authentication attempt in the Security event channel in event ID 4624 as a type 3 network login which records the source IP and workstation name of the connection source, the username/domain used, and the hostname of the device recording the event would be known as well.

**Data:** Our organization currently collects NetFlow data and the Windows Security event channel, and we know the IP addresses assigned to servers and IT/helpdesk laptops. Given these logs and the IP info, both of these are viable options for highlighting potential anomalous SMB connections. To find them I will search the SIEM for:

- NetFlow records with a destination port of 445 and a source and destination IP in the user subnets where we wouldn't expect an SMB connection to originate

- Windows logins of the network type sourced from IP addresses we wouldn't expect an SMB connection to originate and going to a user subnet as well. Using hostnames known as user devices with these events can also work instead of IP addresses since the laptop assigned to each user is known.

Performing these searches would highlight all anomalous SMB traffic in the environment. If any items were found that were suspicious, incident response could be spun up to handle it. If nothing suspicious is found, you can still use the logic you produced for the hunt to make an analytic that will run going forward which would immediately alert you to user-to-user SMB traffic in unexpected locations.

## Hunting Options

Hunting based on	Difficulty	Specificity	Intel Quality Required
IOCs	Easy	High	Low
Attack Technique	Medium	Medium	Medium
Attack Tactic / Model	Medium	Low	Medium
Specific Threat Actor	Hard	High	High
Anomalies	Medium/Hard	Low	None/Low

### Hunting Options

When hunting, there are various methods and data types an analyst might pursue to seek attackers that are already in your environment. The simplest method is to go directly for IOCs—scanning for hashes, domains, or IPs that were not checked automatically and perhaps are new from an external report. They could focus on a specific attack technique from a model such as MITRE ATT&CK, looking, for example, for the use of a specific mechanism of persistence. Going more broadly, they could focus on one whole tactic/step of an attack model like "Command and Control". This casts a much wider net and may take longer but is potentially more thorough. If your threat intel quality is high, advanced hunters may look for all TTPs known to be associated with a specific threat actor, searching for the presence of a particular group in your infrastructure. Finally, when looking for something out of the box, there's also anomaly hunting. Taking your whole data set and using it to cut away "known good", identify oddities, and see if what sticks is malicious. Each of these methods has their pros and cons, and slightly different requirements for analyst skill and intel level as well.

This slide shows a table of the difficulty, specificity, and intel quality requirements for popular threat hunting methods. If you have highly capable analysts and great threat intel, you may be able to jump right to looking for specific TTPs you know your attackers would be using in your environment. If you have a brand-new team with members who aren't very familiar with the organization, you might want to set them after IOCs, specific attack techniques in MITRE ATT&CK, or on a single step of the kill chain.

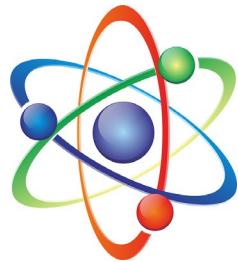
## IOC-Focused Hunting

### IOC-based hunting

- Easy, low false positive rate, highly specific
- Focuses on atomic indicators

### IOC-based hunting methods:

1. Matching to known bad
  - One-off search of new info
  - Retro-active searching of historical data
2. IOC-based anomalies
  - Domain length, randomness, TLDs, IDNs, age, etc.



### IOC-Focused Hunting

IOC-based hunting is perhaps the easiest place to start with threat hunting. One of the most basic methods is extracting indicators from what was observed in the environment – hashes, IPs, domains, etc. and matching them against your tactical threat intelligence list. Note that this is something that *should* be done automatically for most events, but there are some cases where it may need to be a separate activity included in a threat hunt. The main situation is the common occurrence of coming to work and seeing a new report that threat actor group X has been using domains Y and Z for targeted compromise on your industry. Since these indicators have not been known or actively checked against your historical traffic, analysts will need to go through old data and retroactively search for activity to that IP, hash, or domain.

Beyond retroactive searches is a slightly different IOC-based method—looking for IOC-based anomalies. Take domains visited for example. An analyst might start off a threat hunt by assuming that attackers might try to use domain generation algorithms for malware that might be in your environment. To detect DGAs, they then could collect all DNS request data or proxy log info and extract the entire list of domains visited by your organization in the last week. Running this list through tools that can check for long, random values could highlight anomalies in the list which could then be individually investigated.

## Attack Technique-Focused Hunting

### 1. What attack/technique are you trying to find?

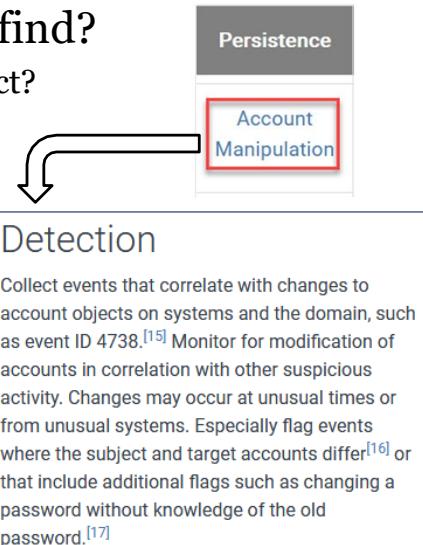
- Is it a specific implementation you're trying to detect?
- Even better, can you make a generalized detection?

### 2. What data do you need to identify it?

- What time frame can you find it in?
- Is the data available real-time only, ephemeral?

### 3. Do you have the data?

- Do you know how to search/extract it?
- Is it high-quality, parsed correctly, enriched?
- Is it available from all hosts on the network?



## Attack Technique-Focused Hunting

One of the main methodologies in threat hunting is taking a *technique* that you do not have current coverage for (either because it was just announced, or you haven't addressed it yet) and searching for evidence of it on your network. Techniques, in this case, are used in the sense that MITRE ATT&CK<sup>1</sup> uses them—as specific methods of accomplishing a larger tactic. Methods for the tactic of persistence, for example, are Registry Run Keys, BITS Jobs, or New Services. Analysts looking for evidence of a specific unfamiliar attack technique can break the process down into a few core questions to be answered.

1. What technique and implantation are you trying to find? They should be specific here. Do they know how the attack looks in one *specific* implementation, or can the detection be generalized such that it would catch it in *all* implementations?
2. What data do you need to identify it? Do you need host logs, network PCAP files, or must you live query hosts on the network in order to find this attack?
3. Do you currently collect that data or have access to query those areas? Do you know how to search that log source and format the query? If it requires logs to find the evidence, do those logs contain the information necessary, are they parsed correctly, and is that data enriched in any way that might make it easier to find?
4. Finally, collect the data as specified previously and search it for evidence of the attack in question.

The MITRE ATT&CK website is an amazing resource for this kind of hunt as each technique is listed with a detection suggestion that hunters who are unfamiliar with the technique can lean on. Stepping through these items gives structure to the hunt and makes sure they aren't trying to bite off too much without planning what is truly needed to answer the question they're focusing on.

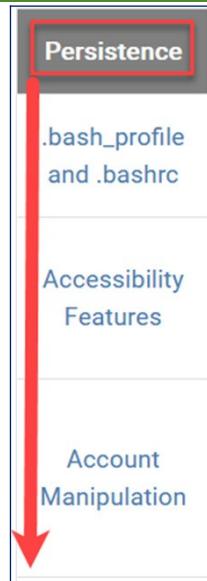
## Attack Tactic-Focused Hunting

Pick an attack model, pick a tactic/step

- Cyber Kill Chain, Mandiant Attack Cycle
- MITRE ATT&CK Tactics

**Define priority** techniques of interest within that step or tactic

- Prioritize based on threat intel
- For ATT&CK – leverage threat actor data
- For other models, consult threat intel reports, news, blogs, GitHub, Twitter, etc.



### Attack Tactic-Focused Hunting

Zooming out a bit, an analyst may choose to focus on an entire MITRE ATT&CK Tactic category for hunting or perhaps an equivalent step in the Kill Chain or Mandiant Attack Cycle. A hunt like this would be a much larger task because you're essentially searching for multiple techniques for accomplishing the tactic. A hunter may set out to "find evidence of any new persistence mechanisms as-yet undetected in the environment", the hypothesis being, of course, that while you may cover some of them, you likely missed at least one instance of persistence of some type.

Does this mean the analyst has to check the entire list of all possible persistence techniques? Not necessarily. Although, it would be great to have full check of all known persistence techniques, it is highly likely that all of them do not apply to your organization. Hunters casting the net this wide should first define and prioritize the techniques they wish to focus on based on the visibility available and what threat intel says is the most likely. Again, leveraging the MITRE ATT&CK website helps ensure newer analysts are aware of the wealth of options that are available for each "tactic" used in common attacks. For information and steps beyond ATT&CK (such as if you were trying to find all "Delivery" stage events based on the Kill Chain), additional research from threat hunting blogs, vendors, Twitter, and intel reports should be leveraged to make a list before starting.

## Threat Actor-Focused Hunting

If intel is available, hunters can focus on one threat actor!

- Get started with ATT&CK's<sup>1</sup> technique tracking by group

### Techniques Used

ID	Name	Use
T1087	Account Discovery	APT1 used the commands <code>net localgroup</code> , <code>net user</code> , and <code>net group</code> to find accounts on the system. <sup>[1]</sup>
T1119	Automated Collection	APT1 used a batch script to perform a series of discovery techniques and saves it to a text file. <sup>[1]</sup>
T1002	Data Compressed	APT1 has used RAR to compress files before moving them outside of the victim network. <sup>[1]</sup>



### Threat Actor-Focused Hunting

Even better than a single technique or tactic hunt is a hunt focused on the totality of what you know about a threat actor. If you have the threat intelligence (or can gather it from something like the groups section in the MITRE ATT&CK Matrix, as shown in the page above), you can hunt with laser-like focus on your most important adversary. To do this, your threat intelligence team will need to have information on the threat actor from higher on the pyramid of pain—TTPs, and your threat hunting team will need the data and the understanding of how to hunt for those TTPs. If you can manage to hunt at this level, it may be one of the most high-value versions of threat hunting you can do. It's focused on groups you know are after you, using techniques you know they use. Of course, just because one threat actor uses a technique doesn't mean others don't as well, so even though this type of hunt is focused on a threat actor, it has the bonus benefit of finding any other group that uses the same tactics as well.

## Hunting via Anomaly Investigation

When you don't have a specific attack in mind:

- Select a large data set with multiple attributes
  - Example: All logged authentication events
- Investigate the distribution of the attributes
  - Example: RDP vs. interactive, NTLM vs. Kerberos, domain vs. local
- Subtract *assumed good* until only anomalies remain
- Look at the remains—do they seem legitimate?
- Can you extract any new knowledge about your environment for future analytics?
  - Example: All legitimate local logins are in the 10.0.1.0/24 subnet

### Hunting via Anomaly Investigation

An alternative method for threat hunting is going “IOC-less” and instead focusing on anomalies in the network. This method has some positives and negatives compared to looking for a known attack technique. The cons are that you’re more likely to chase false positives—things that are anomalous but are *not* malicious. The upside is that it can catch unknown attack techniques that no one is aware of yet.

Anomaly detection can be done in a number of ways, one of the most common is “stack ranking”, or “long-tail analysis”—taking a list of events and sorting them by the frequency of values in one of the fields. The theory is that the values that show up the most are likely the least interesting, at the bottom of the list will be things that happen infrequently (anomalies) and might lead you in the direction of an attack. Grouping data by frequency of field values and other exploratory data grouping methods can yield anomalies in unexpected ways.

This approach can be followed up by elimination of the most frequent items, carving away what is assumed to be good until only anomalies remain. Once this is done, you can look at the logs in multiple different visualization types, using the multiple angles on the data to perhaps produce new insights on what is and is not normal. Anomalies can be investigated and even if you don’t find anything, the process will result in new knowledge about what is “normal” in your network, and that information can be used to produce new analytics.

## Suggested Hunting Sessions

- Where is the best place to start?
- Focus on the riskiest stage – **post exploitation:**
  - Internal recon / lateral movement
  - Unexpected privileged account / service account usage
  - Local accounts used over the network
  - Unexpected protocols outbound, protocol non-compliance
  - Persistence mechanisms
  - Exfil via DNS, SSH, other



### Suggested Hunting Sessions

If you're wondering where the best place is to start in threat hunting, here are some tried and true methods to potentially immediately return some findings that can show the value of hunting.

- Lateral Movement: Focus on identifying traffic by port in your environment where both the source and destination is a person's personal computer. If you find SMB, PowerShell, or other traffic between two points that don't make sense, investigate! Remember this data can be sources from an incredible number of places—host or network firewalls, NetFlow, Sysmon logs, PCAP capture devices, anything that records that a network transaction occurred and the IP/port that was used can be helpful.
- Privileged Account Usage: Attackers love to steal and use privileged accounts and service accounts and often use them in anomalous ways. Audit both where and how these types of accounts are being used. Where are they logging in *from*? If the answers don't make sense, one of them may be compromised. *How* are the logins occurring? Should anyone be using your service accounts to login via RDP? (hint: no)
- Local accounts used over the network: If you utilize local administrative accounts, audit where and when they are being used. Although it is against best practice, many organizations, unfortunately, have the same local admin account password used in multiple places. If the adversaries can find it, they may use it as a backdoor to login from other compromised assets. The good news is Windows event logging will pick this up in login events and you can easily identify it.
- Unexpected protocols outbound: Do you have users using SSH, FTP, DNS or any protocol directly outbound to destinations that don't seem clearly good? It's worth double checking what's going on. All of these protocols and more have been used for data exfiltration in many attacks through the years. A twist on this is to look for protocol non-compliance. Many network sensors and next-gen firewalls should alert on this automatically. If yours don't however, look, for example, for usage of port 80 that isn't actually carrying HTTP traffic.
- Persistence mechanisms: If you have the visibility, compare startup items across your enterprise and look for the least common ones. By definition, infected machines will have unique startup items, all you need to do is query for them and group the data!

## Common Mistakes (1)

- Hunting before basic visibility and reliable detection has been established
- Prioritizing hunts exclusively based on individual preference
- Not documenting hunts
- No measurable output or metrics

### Common Mistakes (1)

We've already referenced some common mistakes organizations make in threat hunting, but it's a topic that tends to trigger much debate so it's worth revisiting those mistakes before we move on. Remember the sliding scale of security – while threat hunting is less of an "advanced" capability than it used to be, and now has a much more important role in detection and response, it can't take precedence over basic visibility and reliable detections. Address those fundamental elements of defense before you worry about planning and executing proactive hunts. Another common pitfall of threat hunting is leaving it to individual analysts to prioritize TTPs and hunt objectives based on their own individual preference. For example, you may have an analyst who loves tracking down ransomware and is quite skilled at doing so. While ransomware may be a highly concerning threat in your organization, applying some structure around how hunts are chosen and prioritized based on threat intelligence and your organizational threat model can prevent your analyst's ransomware hunts from taking cycles away from other high-priority threats. The process should be documented, structured, and repeatable – regardless of who is on a hunting rotation or filling a hunting role.

Lack of documentation is a common problem in the SOC across all functions. It's of particular concern in threat hunting, where the approach and results can sometimes be subjective. If we want to manage out threat hunting efforts, then we must be able to monitor them; that's very challenging if the hunter does not take notes and document their efforts. Finally, lack of tangible results is the hallmark of a poorly-designed hunting program. As we've discussed, even those hunts that do not identify a new threat or incident should still uncover visibility gaps or improve automated detections. At the very least, they can validate the absence of a known threat at a specific point in time. Tracking these measurable outputs and a larger set of metrics around threat hunts will help you understand your return on investment and adjustments you may need to make.

## Reference Guides

- Generating hunting hypothesis:
  - *Generating Hypothesis for Successful Threat Hunting*, by Robert M. Lee and David Bianco<sup>1</sup>
- Suggested hunts:
  - Threat Hunter Playbook<sup>2</sup>
  - Threat Hunt Library<sup>3</sup>
- Testing hunts/analytics:
  - Atomic Red Team<sup>4</sup>



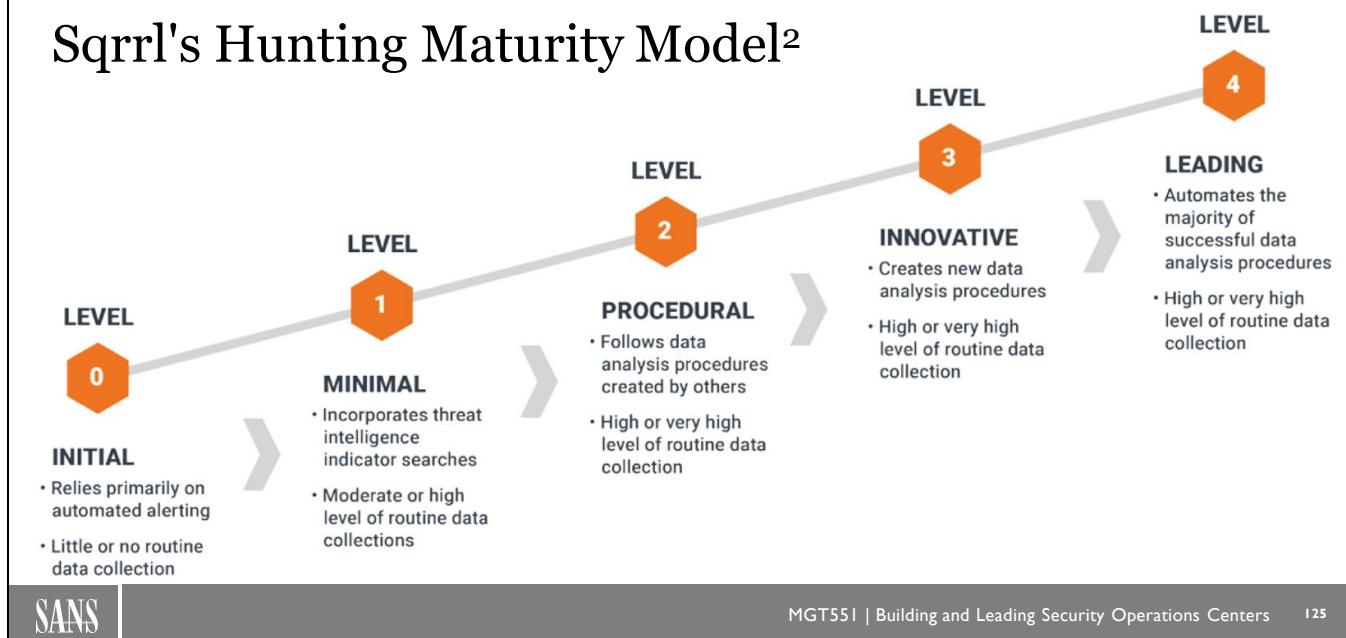
### Reference Guides

There are many great references on threat hunting, some of which we've already mentioned in this course. If you're just getting started in threat hunting and need to develop hunting hypothesis, check out *Generating Hypothesis for Successful Threat Hunting* by Robert M. Lee and David Bianco. If you're interested in a pre-defined library of hunt playbooks and analytics, there's the Threat Hunter Playbook and the Threat Hunt Library projects. Finally, you can "unit test" your detections to prove or disprove hunt hypothesis using Atomic Red Team. Links for these projects are below.

1. <https://www.sans.org/reading-room/whitepapers/analyst/membership/37172>
2. <https://github.com/OTRF/ThreatHunter-Playbook>
3. <https://github.com/svch0stz/TheThreatHuntLibrary>
4. <https://github.com/redcanaryco/atomic-red-team>

## Hunting Maturity Model

### Sqrrl's Hunting Maturity Model<sup>2</sup>



SANS

MGT551 | Building and Leading Security Operations Centers

125

### Hunting Maturity Models

Those with a current hunt team that are looking to improve can start by benchmarking themselves against the Hunting Maturity Model originally developed by David Bianco<sup>1</sup> and included in the excellent Sqrrl (now acquired by Amazon) Hunting Guide available at the link below.<sup>2</sup> This model shows that you are not simple "hunting or not hunting", but that there is a progression of hunting process and maturity that even the best teams can improve on.

The HMM and related literature below has a great set of questions to help assess which level you're at and suggestions on how to level up. Suggestions on getting started, for example, include focusing on acquiring automated data collection systems for centralized logging, establishing an incident response team and integrating external intelligence feeds for automated detection. To step into level 0, the guide suggests establishing dependable security data collection from host and network sources (sounds familiar), and that analysts should routinely use that data for basic searching of indicators. Suggestions for teams at level 3 to go to level 4 are automation, sharing of successful hunting techniques with the community, machine learning, and more.

1 <http://detect-respond.blogspot.com/2015/10/a-simple-hunting-maturity-model.html>

2 <https://www.threathunting.net/files/hunt-evil-practical-guide-threat-hunting.pdf>

## Simulating TTPs Using Atomic Red Team

- Select a test relevant to your environment
- Execute the test based on the commands or instructions given
- Collect evidence: where in the process did your detection(s) fire?
- Develop and/or refine detections
- Measure progress, continuously improve

### Simulating TTPs Using Atomic Red Team

The process of simulating TTPs with Atomic Red Team begins with selecting a test that is relevant to your environment. This should reflect your threat model, a new detection priority, or a gap you've identified. After executing the test, it's time to look for evidence – did any of your detections fire as a result? If so, continue executing the test until you are satisfied that your detections will work consistently and reliably everywhere you might see that same TTPs. If you missed it, revisit and refine your detections until they fire based on the test. Atomic Red Team is a great way to keep this iterative process going to prove out your detections using a scientific, evidence-based approach. We'll see an example of Atomic Red Team testing on the next slide.

# Atomic Red Team Example

## ø T1003.004 - LSA Secrets

### Description from ATT&CK

Adversaries with SYSTEM access to a host may attempt to access Local Security Authority (LSA) secrets, which are used to store sensitive information such as password hashes and other credential materials, such as credentials for service accounts.(Citation: Passcape LSA Secrets)(Citation: Mimikatz LSA Secrets) LSA secrets are stored in the registry at `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Lsa\Secrets`. LSA secrets can also be dumped from memory.(Citation: ired Dumping LSA Secrets)

`Reg` can be used to extract from the Registry. `Mimikatz` can be used to extract secrets from memory.(Citation: Mimikatz LSA Secrets)

### Atomic Tests

- Atomic Test #1 - Dumping LSA Secrets

Attack Commands: Run with `command_prompt` ! Elevation Required (e.g. root or admin)

```
#psexec.exe -accepteula -s reg save HKLM\security\policy\secrets %temp%\secrets
```

Cleanup Commands:

```
del %temp%\secrets >nul 2> nul
```

A handle to an object was requested.

Subject: Security ID: S-1-5-21-3845758792-3269411503-3879198917-1000  
Account Name: raveydalegravvy  
Account Domain: RAVEYDAVEYGRAVY  
Logon ID: 0004D628

### Object:

Object Server:	Security
Object Type:	Key
Object Name:	\REGISTRY\MACHINE\SECURITY
Handle ID:	00000000000000000000000000000000

### Process Information:

Process ID:	00001154
Process Name:	C:\Windows\System32\reg.exe

### Access Request Information:

Transaction ID:	{00000000-0000-0000-0000-000000000000}
Accesses:	READ_CONTROL



## Atomic Red Team Example

In this example, we can see technique T1003.004, LSA Secrets, from the MITRE ATT&CK matrix as listed in the Atomic Red Team project. Below the title and description, the entry has specific commands we can run to simulate this technique as well as cleanup commands to remove artifacts our testing might leave on the target system. The goal here is to see if this test generates a detection, and what information is available within that detection that we might use for an analytic. You can see in the red box on the right that, thanks to Windows Object Access Auditing, our test has been flagged in the Windows Event ID 4656, *A handle to an object was requested.*

## Showing the Value of Hunting

- Tag hunting results in the incident management system
  - **Create metrics** based on incidents tagged "hunting"
  - Capture all positive outputs
- Hunting reports at the end of week/rota
  - Save in a "threat hunts" database for future reference
- Presentations to new team members
  - Technique focused on why it's important;
  - How you looked for it;
  - Scanning results and conclusions
- Ultimate goal: improve detections!



### Showing the Value of Hunting

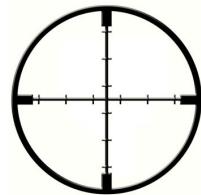
One problem you can run into when tasking analysts with threat hunting instead of reactive ticket-based work is that upper management may doubt they will produce anything useful. That means until your threat hunting program is mature and trusted, you will need to be extra diligent to show the fruits of your labor. Here are some common ways that I've seen used in the past to prove the value of threat hunting:

- While threat hunting, have analysts mark every alert and incident generated from their activity with a "hunting" tag or something similar in your incident management system. As long as each item is labeled, it's easy to then query the system over the long term to see how many and which incidents were the results of a threat-hunting exercise. Cross referencing these alerts with their severity and closing disposition can help show that threat hunting is worth the time invested.
- Once an analyst has gone on a successful hunt, the idea is to turn that technique into a new analytic so that specific threat hunt doesn't have to be repeated again. At the end of the week the person on hunting duty should write up what they did, what they found, and the results into the SOC knowledgebase. These reports are useful in two ways. Management can reference these reports and see the effort that went in (as well as metrics on new analytics created), and future threat hunters can see the detection techniques that have already been attempted, how well they worked, and how it was implemented. Just because someone didn't find something on one week doesn't mean it won't be found the second time and starting where the first analyst left off can save time on the second round.
- A final tip for getting value out of threat hunting regardless of whether something was found or not is to have the Sr. Analysts that were hunting do a short presentation to the newer analysts at the end of the week. This should include the technique they focused on, a bit of explanation about it and why they chose it as a potential threat to your environment. Then, they should explain the tactics they used to look for that activity in your environment as well as the results of what they found and the conclusions that were drawn. Using this method, if nothing else, an unsuccessful hunt at least turns into a training exercise that gets new analysts more familiar with the environment and give them the mindset of a Sr. Analyst.

- In addition to these items, there are many other metrics that can be collected as well such as vulnerabilities identified, bad practice found, and detection gaps discovered and remedied. Any time there is a positive result of threat hunting, it is a wise move to document that outcome so the win can be acknowledged and championed.

## Threat Hunting Summary

- Hunting is for **all organizations**
- But it requires:
  - A presumption of compromise
  - Network and host visibility
  - High-quality data
  - A defined process and procedure
  - Analysts with interest and capability
  - A good source of information on new attack techniques
- **Document** all hunting-related wins!



### Threat Hunting Summary

In this section, we covered the what, who, why, and how of threat hunting and many of the considerations for making or improving your own hunt team. In short, threat hunting's key tenet is that *you missed something, and it's already in your network causing damage*. This is the difference between hunting and typical day-to-day network defense. The problem is that they are an "*unknown unknown*" -something you are unaware you don't know, which means finding them will be difficult. To do so, we ideally dedicate some of our best people to leverage the best threat intel we have and form a hypothesis about things we may have missed, as well as attack techniques attackers may use, and set them on their way.

Although not all hunts will result in success (which is *good* assuming the hunters were thorough), that doesn't mean they aren't producing wins for your team. Aside from finding advanced attackers, the secondary output needs to be a well-documented list of improvements, visibility gaps, vulnerabilities, and anything else that is found as a result of the activity. Like so many activities in the SOC, communication is the name of the game. When done properly, threat hunting can give the SOC an incredible reputation and build trust within the organization, but only if they know the results, they're getting for the time put in.

# Course Roadmap

- Book 1: SOC Design and Operational Planning
- Book 2: SOC Telemetry and Analysis
- *Book 3: Attack Detection, Threat Hunting, and Triage*
- Book 4: Incident Response
- Book 5: Metrics, Automation, and Continuous Improvement

## SECTION I

### Introduction

#### Creating and Processing Alerts

- Efficient Alert Triage
- Detection and AnalyticDesign
- Capacity Planning
- *Exercise 3.1 – Capacity Planning*
- Detection Engineering

#### Advanced Analysis

- Analytic Frameworks and Tools
- *Exercise 3.2 – Structuring, Documenting, and Organizing Use Cases*
- Threat Hunting
- ***Exercise 3.3 – Planning a ThreatHunt***
- Off-Hours Alerting and On-Call
- Active Defense
- Summary and Cyber42 – Day 3



This page intentionally left blank.

## EXERCISE 3.3

# Exercise 3.3: Planning a Threat Hunt

### OBJECTIVES

- Understand various hunting triggers
- Create an investigation abstract and enrich it using MITRE ATT&CK, threat intelligence, and other metadata
- Determine data sources and analysis techniques required to prove or disprove your hypothesis
- Document and share hunt findings



### Exercise 2.1: Prioritizing and Visualizing Attack Techniques and Security Controls

Please go to Exercise 2.1 in the MGT551 Workbook or virtual wiki.

# Course Roadmap

- Book 1: SOC Design and Operational Planning
- Book 2: SOC Telemetry and Analysis
- *Book 3: Attack Detection, Threat Hunting, and Triage*
- Book 4: Incident Response
- Book 5: Metrics, Automation, and Continuous Improvement

## SECTION I

### Introduction

#### Creating and Processing Alerts

- Efficient Alert Triage
- Detection and Analytic Design
- Capacity Planning
- *Exercise 3.1 – Capacity Planning*
- Detection Engineering

#### Advanced Analysis

- Analytic Frameworks and Tools
- *Exercise 3.2 – Structuring, Documenting, and Organizing Use Cases*
- Threat Hunting
- *Exercise 3.3 – Planning a Threat Hunt*
- **Off-Hours Alerting and On-Call**
- Active Defense
- Summary and Cyber42 – Day 3



This page intentionally left blank.

## Off-hours Alerting

- Recall our Day 1 discussion about 24/7 coverage
- Many options for on-call rotations, off-hours alerting
  - Splunk On-Call
  - PagerDuty
  - ZenDuty
- Whatever technical solution you use, should integrate with collaboration and security tools to provide ready context and escalation



### Off-Hours Alerting

This is a topic that will be *highly* specific to your environment, SOC charter, the composition of your team, and other factors. But our adversaries don't punch a clock, so we need to make sure we have some way of ensuring coverage even if we do not have analysts on duty 24/7. Fortunately, there are many options for off-hours escalation and coverage, some of which we'll cover in this section.

## On-Call Best Practices

- Questions to answer:
  - What events require immediate escalation to the on-call analyst?
  - Who will participate in the on-call rotation, and how will participants be selected?
  - How long will a rotation last?
  - What happens when escalations go unanswered?
  - How far should the on-call analyst take investigation and response activities?
  - How will the team handle unscheduled rotations and other unforeseen events?
  - Who are the on-call stakeholders?
- Collect metrics on the number, frequency, and type of incidents that trigger off-hours escalation

### On-Call Best Practices

On-call rotations are going to be managed differently from organization to organization, but there are some best practices we can keep in mind when planning this kind of coverage approach. Think through your standard day from the perspective of alerts, service requests, and other work streams. Which of these will require immediate handling regardless of the time of day? Which have outcomes that are unlikely to change if they wait until business hours? Think about contingencies like what happens when escalations go unanswered, and how far should the on-call analyst take each investigation or escalation? Consider these and other basic questions and then collect detailed metrics on your on-call workloads to understand if you need to adjust or revisit your strategy or answers to these questions.

## Using Amazon SNS as a Free Notification Service

Send alerts to text messages / email for free!

- Amazon SNS service offers a simple API
- Free tier allowances per month
  - **100 SMS** messages
    - Per msg rate after (in USD): **US**=\$0.0065, **CA** = \$0.0023-0.0068, **UK**= \$0.039, **NL**=\$0.11 , **SG**=\$0.052-0.068, **JP**=\$0.075
    - **1000 emails** - \$2/100k emails after
    - **100k HTTP/S** requests - \$0.60/1M after
  - Supports SMS, email, HTTP/S, Lambda, and more

### Using Amazon SNS as a Free Notification Service

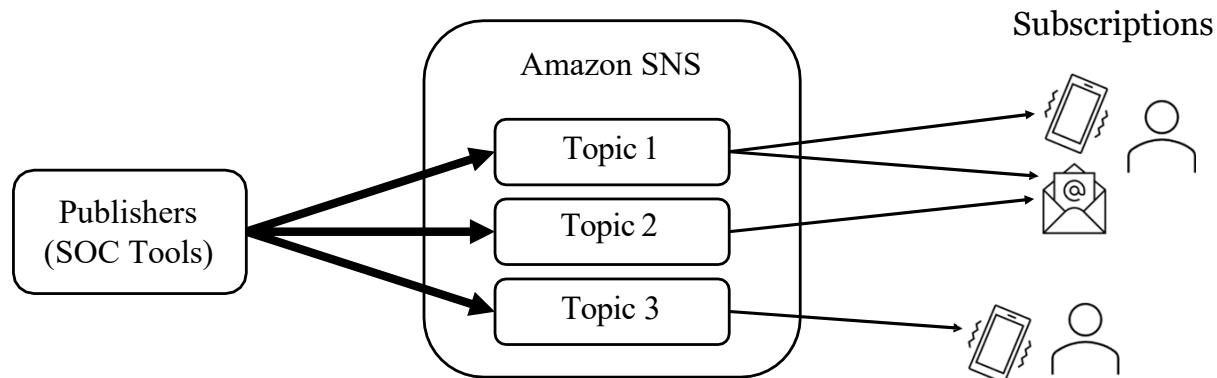
Every SOC needs a way of sending automated email and text messages to analysts, infrastructure group members, and more out of office hours. A system to do this can be set up in under 10 minutes using Amazon SNS (Simple Notification Service). Assuming you have access to an AWS account where you can log in and set up Amazon SNS, configuration is incredibly simple and fast. The best part is, if you send less than 100 text messages or 1000 emails a month, the service is completely free! After hitting these limits, the cost is still incredibly low and should fit within any SOCs budget.

If you are interested in sending more than just simple email and text message-based notifications, SNS also supports many other message types such as HTTP/S requests, AWS Lambda function calls (serverless function-as-a-service code), mobile push notifications (if you have an app, you can push them to) and much more.

## How Amazon SNS Works

Amazon SNS simplified for SOC alerting:

- You make "topics" and list subscribers (SMS numbers/email addresses) for each
- "Publishers" push "messages" to a topic, which is sent to subscribers



## How Amazon SNS Works

While usage of Amazon SNS can become quite complex for those looking to use all its features, making a simple text and email message notification system is incredibly simple. Here's all you need to know:

- In the terminology of Amazon SNS *publishers* send messages to *topics* – in effect, your infrastructure and tools are the publishers which may send notifications of different types to the different topics such as malicious activity or infrastructure alerts
- *Topics* relay messages to *subscribers* of that topic – SMS phone numbers and email addresses are listed as subscribers, which will receive a notification of the messages sent to any topic they subscribe to

## Creating a Topic

In Amazon SNS, select Topics, then "Create Topic"

### Create topic

Type [Info](#)  
Topic type cannot be modified after topic is created

FIFO (first-in, first-out)

Standard

Name

SOC\_Alerts

Maximum 256 characters. Can include alphanumeric characters, hyphens (-) and underscores (\_).

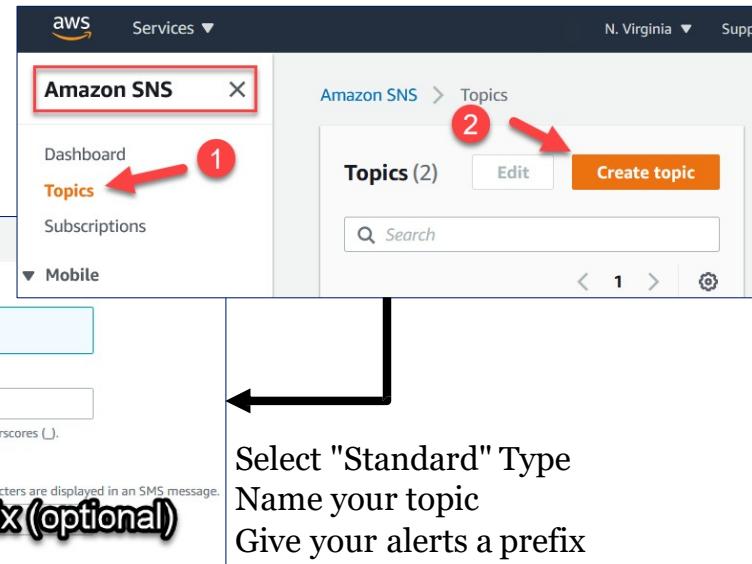
Display name - optional

To use this topic with SMS subscriptions, enter a display name. Only the first 10 characters are displayed in an SMS message.

SOC\_Alerts

**SMS Prefix (optional)**

Maximum 100 characters, including hyphens (-) and underscores (\_).



Select "Standard" Type  
Name your topic  
Give your alerts a prefix

## Creating a Topic

The first thing you need to do to use Amazon SNS is to create a "Topic" for people to subscribe to. Doing so is as simple as selecting "Topics" in the sidebar of the Amazon SNS page and selecting to "Create topic" to bring you to the new topic creation page. On this page, selecting a "Standard" type topic, give your topic a "Name" (what appears in Amazon SNS interface to label this topic), and a "Display name" – what your messages will use as a prefix when delivered via SMS or as an email display name when emailed.

For this step, it is likely you will want to create several topics for different types of events, depending on the organizational structure of your SOC. You should consider "High Priority Detection" alerts, "Infrastructure" alerts, and anything else the members of your team may need to be pushed information about in an urgent manner.

**Adding Subscriptions – Email and SMS**

The screenshot shows the AWS SNS interface. On the left, there's a sidebar with 'Amazon SNS' at the top, followed by 'Dashboard', 'Topics', and 'Subscriptions' (which is highlighted with a red arrow labeled '1'). Below that is a 'Mobile' section. In the main area, there's a 'Subscriptions' page showing 5 subscriptions, with a 'Create subscription' button highlighted with a red arrow labeled '2'. A large callout box covers the 'Create subscription' dialog. Inside this box, the 'Protocol' dropdown is set to 'Email' (red arrow labeled '3'). The 'Endpoint' field contains 'john@l...com' (red arrow labeled '4'). At the bottom right of the dialog is a 'Create subscription' button with a red arrow labeled '5'.

## Adding Subscriptions – Email and SMS

Subscribing people in your SOC to topic alerts is as easy as listing their email address and phone numbers as a "Subscription".

To perform this step

1. In the Amazon SNS sidebar click "Subscriptions" then click "Create subscription"
2. In the Create subscription page, select the "Topic ARN" for the Topic you just created. (The ARN – Amazon Resource Name, is a labeling system that uniquely identifies all objects in AWS using a colon delimited value, the topic name you just created should be at the end of the ARN).
3. In the Protocol box, type select either Email or SMS.
4. In the Endpoint box, type the email address for email protocol subscriptions, or a phone number for SMS subscriptions, then hit Create subscription.

That's it, you're done!

## AWS CloudFormation – SNS Alerting Infrastructure As Code

### Additional option:

- Deploy using **Amazon CloudFormation**
  - "Infrastructure as code"
- Utilizes YAML template to describe alerting system
- Automatically deploys topics and example SMS/email subscription
- File included in link below



```
Resources:  
SOCAlertTopic:  
  Type: 'AWS::SNS::Topic'  
  Properties:  
    DisplayName: SOC_Alerts  
    TopicName: SOC_Alerts  
SOCAlertEmailSubscription1:  
  Type: 'AWS::SNS::Subscription'  
  Properties:  
    Endpoint: email@example.com  
    Protocol: email  
    TopicArn: !Ref SOCAlertTopic  
    DependsOn:  
      - SOCAlertTopic  
SOCAlertSMSSubscription1:  
  Type: 'AWS::SNS::Subscription'  
  Properties:  
    Endpoint: 155555555555  
    Protocol: sms  
    TopicArn: !Ref SOCAlertTopic  
    DependsOn:  
      - SOCAlertTopic
```

### AWS CloudFormation – SNS Alerting Infrastructure As Code

For those who want to dive all the way into the cloud workflow (or at least understand a common feature), this setup can be easily deployed all at once with topics and subscriptions all in place via Amazon CloudFormation. Amazon CloudFormation is a way to describe and configure AWS assets you would like to create via a simple YAML or JSON text template file. Those templates can be loaded as a "stack" in the CloudFormation section of AWS and deployed to create all required items in one shot. (While the simplicity of our Amazon SNS setup doesn't necessitate doing it this way, for those interested, this *is* a nice way to introduce yourself to CloudFormation with a simple setup that is highly unlikely to result in high-cost errors.)

- **Topics** – Duplicate the Topic section once for each topic you would like to create. The bolded top-level item "SOCAlertTopic", which is just a name for the resource in the template, *must be a unique name* for each topic. If you want 3 topics, make 3 copies of this block, each with a unique name, and modify the subitems for DisplayName and TopicName for each block as desired.
- **Subscriptions** – To create subscriptions, just duplication the subscription block as many times as necessary for each subscription you'd like to create. Each subscription block needs four things modified
  - Name – Show at the top of the block above, "SOCAlertEmailSubscription1" for example
  - Endpoint – The email address or phone number to send the notification to
  - Protocol – Either "sms" or "email"
  - TopicArn – This block is how you signify which topic this subscription is subscribing to, modify ONLY the bolded section with the name of a topic block you used elsewhere in your template ("SOCAlertTopic" is selected as the TopicArn in the slide above, meaning both the sms and email subscriptions on the slide would be subscribed to the single "SOC\_Alerts" named topic above)

Once your template is complete, head over to the CloudFormation section of AWS. Select "Create Stack" with new resources and choose to upload your template file. You will need to choose a name for the stack, other options can be left blank/default or modified as needed, click next until you deploy the stack. You should now see it listed under the name you gave it in your CloudFormation stacks list. If anything goes wrong and the stack can't deploy you will see the error listed in the Events tab of CloudFormation under the stack name, you can just delete the stack and re-create it as needed. If it does successfully deploy but isn't what you wanted, you can delete and redeploy or fix the created items individually in the Amazon SNS page.

## Sending Alerts



Easy options for sending alert text to SNS for notifications:

1. Logstash (free, simple, and flexible)
  - Configure arbitrary format log input from any device, modify as needed before sending
  - See notes for config to receive message on a listening port, forward it to a SNS notification
2. AWS CLI (scripted)
  - `$ aws sns publish --topic-arn "arn:aws:sns:[region]:[number]:[topic]" --message "hello world!"`
3. Pre-made generic-webhook-to-sns topic app<sup>1</sup>
4. Custom Amazon API Gateway integration

### Sending Alerts

To send alerts to your new notification service, you have a few options. Depending on how fancy you want to get and how much flexibility you need, you can rely on pre-created applications that easily integrate incoming data with the AWS API or create your own.

(Technical security note: Regardless of the method below you chose, to send alerts safely to AWS, you should create a dedicated alerting user in AWS IAM given the least privilege necessary to get the job done. This means creating a Policy in AWS IAM with "sns:Publish" permission, attaching that policy to a new SNS alert sending Role, and giving your notification user that new SNS notification Role. Once the user is created, you create a IAM user "access key id" and "secret key" that will need to be used for any of the below methods.)

1. **Logstash** - If you have a SIEM or other security tool set up to send text-based alert logs for conditions of interest, Logstash<sup>1</sup> is a great option for simple integration. Logstash is a free tool which you can define inputs and outputs of different types and connect them together to take action for you. For this setup, you simply set up Logstash with a configuration file to listen for incoming logs in any of a variety of methods and output any input to the "sns" output plugin<sup>2</sup> (which only requires configuration of the access key ID, secret key, and ARN.) See the included logstash configuration at the bottom of this page for the configuration required, tested with Logstash version 7.10.
2. **AWS CLI** – Amazon provides the AWS CLI application for use on any system that can directly make API calls to AWS services. If you have a system that can take the notification you'd like to send and include it as a command line parameter to the aws cli application, then generating a notification is as simple as making the call shown on the slide above (after you provide aws cli the access key id and secret key for your SNS alerting AWS user).

**3.Generic-webhook-to-sns** – If you'd like to go the pre-made code route utilizing AWS API gateways and Lambda functions, the "generic-webhook-to-sns" project is available on GitHub <https://github.com/vacationtracker/generic-webhook-to-sns>. This project will quickly deploy an Amazon AWS API Gateway that will receive messages over HTTP and shuttle them to a Lambda function, which will then push them to your SNS topic.

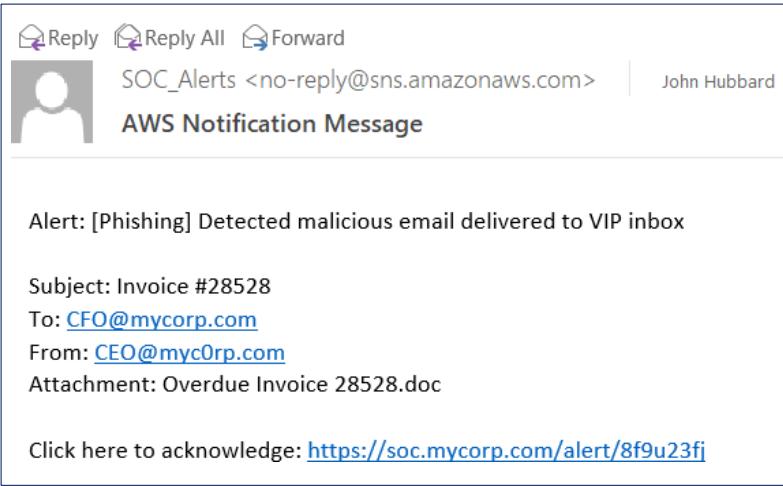
**4.Custom code** – If you'd like to create a highly customized experience for alert sending, creating your own AWS API Gateways and/or Lambda functions is not as complicated as it sounds, especially for those familiar with this process. A custom Amazon AWS API Gateway can ingest HTTP requests sent from any tool in any form and convert them into the form needed to be sent as a notification. This approach is similar to using Logstash, but instead does the work in the AWS cloud. Costs for AWS API Gateway are negligible at the volume that would be required for this use.

- 1 <https://www.elastic.co/logstash>
- 2 <https://www.elastic.co/guide/en/logstash/current/plugins-outputs-sns.html>

Complete Logstash config for forwarding any text sent to Logstash on port 1514 to a chosen SNS topic notification:

```
input {  
    tcp {  
        port => 1514  
    }  
}  
  
filter {  
    mutate {  
        rename => { "message" => "sns_message" }  
    }  
}  
  
output {  
    sns {  
        access_key_id => "[fill me in]"  
        secret_access_key => "[fill me in]"  
        region => "[fill me in]"  
        arn => "[fill me in]"  
    }  
}
```

## Notification Sent!



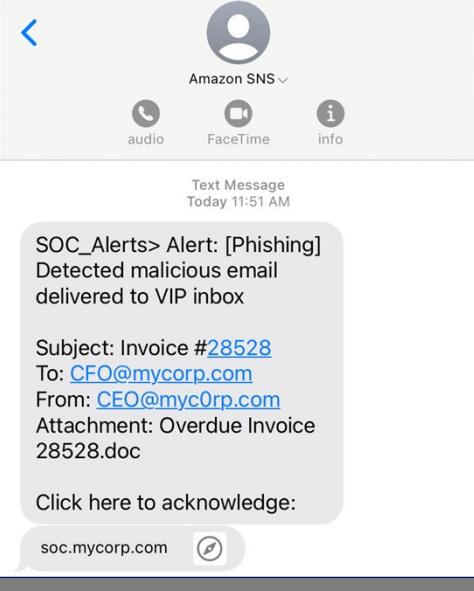
**SOC\_Alerts <no-reply@sns.amazonaws.com>**

**AWS Notification Message**

Alert: [Phishing] Detected malicious email delivered to VIP inbox

Subject: Invoice #28528  
 To: [CFO@mycorp.com](mailto:CFO@mycorp.com)  
 From: [CEO@mycorp.com](mailto:CEO@mycorp.com)  
 Attachment: Overdue Invoice 28528.doc

Click here to acknowledge: <https://soc.mycorp.com/alert/8f9u23fj>



**Amazon SNS**

Text Message  
Today 11:51 AM

SOC\_Alerts> Alert: [Phishing]  
Detected malicious email  
delivered to VIP inbox

Subject: Invoice #[28528](#)  
 To: [CFO@mycorp.com](mailto:CFO@mycorp.com)  
 From: [CEO@mycorp.com](mailto:CEO@mycorp.com)  
 Attachment: Overdue Invoice 28528.doc

Click here to acknowledge:

soc.mycorp.com 

SANS
MGT551 | Building and Leading Security Operations Centers
143

### Notification Sent!

Once you have the topic set up, added subscribers to each topic and chosen an alert sending method, you should be good to go! You can test the system by using the "Publish Message" feature directly from topic page on the Amazon SNS website to send a test message.

Here is the info that will be received in email and SMS form by simply filling out the SNS "message" field, (and *not* filling out an SNS "subject" such as would be done with the Logstash example). If the SNS subject field *is* filled out, that text will appear as the email subject line, but will not appear anywhere in the SMS. For this reason, it is advised to keep all data in the message area for notifications.

As a final point, while the URL to acknowledge in this screenshot was made for demonstration purposes, AWS also offers an "API Gateway" service that could easily be made to craft this functionality on your own. As a manager, knowing the message was seen by one or multiple intended recipients is a requirement for this type of notification. The link could either refer analysts directly to cloud-based security tools, pass a secure acknowledgement through AWS API Gateway infrastructure connected to your security tools, or simply trigger another message to be sent to the SOC manager saying the alert was acknowledged.

## Asynchronous Investigations

- Communicate, communicate, communicate!
- Whether on-call or 24/7 coverage, make sure the requirements for triggering incident response are clear and well-defined
- Identify functions that require immediate execution (i.e., taking a compromised machine offline) and those that can wait
- Include off-hours investigations in your tabletops

### Asynchronous Investigations

Investigations begun off hours live and die by the quality of communication. Whether you have 24/7 coverage or rely on on-call coverage to handle critical events, ensuring that you have a clearly-defined process for when and how to escalate/involve other teams, document initial findings, and cadence of updates and reporting until regular hours resume will save you a lot of headaches. Walk through your incident response plan and ensure that any staff working on-call or accessing your environment in a way that is different from business hours access can perform their duties just as they would during the day. Understand which stakeholders and IR participants are available off hours and which are unlikely to respond to requests until business hours. Finally, ensure that there are “warm” (i.e., real-time communications) hand-offs from one shift to the next or from on-call to on-duty that include updates for all incidents and investigations in progress, with documentation to match.

We will talk about tabletops and incident response planning in Book 4, but for now just keep in mind that any common scenario, including off-hours investigation and escalation, should be well-practiced within your team.

## Off-Hours Alerting Summary

- Fundamental questions for on-call coverage
- Example notification using Amazon's SNS
- Best practices for asynchronous (off-hours) investigations
- Common themes are focusing on what absolutely can't wait and *communication*

### Off-Hours Alerting Summary

In this section, we talked about a common approach to off-hours coverage: maintaining an on-call rotation. We discussed some fundamental questions to guide your on-call strategy, worked through an example escalation using Amazon's Simple Notification Service (SNS), and covered some investigative best practices for cases worked off-hours. Remember that just like every other function in the SOC, this is an area that should be measured, improved, and properly resourced to produce value to your network defense. On-call rotations are an easy way to burn out your team and magnify problems in alerting or investigative process, so keep a close eye on utilization and outcomes and be ready to adjust if needed.

# Course Roadmap

- Book 1: SOC Design and Operational Planning
- Book 2: SOC Telemetry and Analysis
- *Book 3: Attack Detection, Threat Hunting, and Triage*
- Book 4: Incident Response
- Book 5: Metrics, Automation, and Continuous Improvement

## SECTION I

### Introduction

#### Creating and Processing Alerts

- Efficient Alert Triage
- Detection and Analytic Design
- Capacity Planning
- *Exercise 3.1 – Capacity Planning*
- Detection Engineering

#### Advanced Analysis

- Analytic Frameworks and Tools
- *Exercise 3.2 – Structuring, Documenting, and Organizing Use Cases*
- Threat Hunting
- *Exercise 3.3 – Planning a Threat Hunt*
- Off-Hours Alerting and On-Call
- **Active Defense**
- Summary and Cyber42 – Day 3



This page intentionally left blank.

## What is "Active Defense"?

US DoD definition<sup>1</sup>:

- *"The employment of limited offensive action and counterattacks to deny a contested area or position to the enemy."*

Terms commonly associated with active defense:

- Active Defense
- (Breach) Canaries
- Deception
- Honeypots
- Tripwires

### What is "Active Defense"?

"Active defense" is another strategy the SOC can deploy to assist the detection function. The goal is to create an environment where it is extremely difficult for the attacker to operate and advanced without exposing their position. This is done through deception, confusion, and other methods of creating complication for the attacker in a way that will encourage them to make a mistake, and give you, the defender, a high-fidelity and early-stage signal. Throughout this section, we'll dive into some of the different methods, tools, and tactics that can be employed as part of an "active defense" strategy.

[1] <https://www.jcs.mil/Portals/36/Documents/Doctrine/pubs/dictionary.pdf?ver=2020-06-18-073638-727>

## Can I Do That?

- Isn't that illegal?
  - Your organization will instinctively say *yes*
  - Truth: It depends\*... some techniques are likely allowed
  - **Annoyance/Confusion/Attribution/Breach Detection** = Probably ok
  - **Active attacking** or "hacking back" = Almost certainly illegal
- **Goal:** Create an environment with a **better chance at detection**
  - **Confuse** the attacker
  - **Detect** a breach in action
  - **Attribute** their activity
  - **Ultimately - Slow down and complicate** the attacker's job

\*I am not a lawyer, check the laws in your area



### Can I Do That?

Once your team has met the basics of detection by collecting necessary events and security appliance data it's time to consider taking your defense to the next step—actively trying to deceive attackers that may get into the environment. Suggesting active defense often causes lawyers and upper-level management to quickly go on the defensive with complaints of, "You can't hack back!! That's illegal!", but not so fast ...

In a literal sense, they may be correct, but not all active defense is "hacking back". A great reference on this topic is the "Offensive Countermeasures"<sup>[1]</sup> book by John Strand. In this book, John breaks down offensive countermeasures into 3 types: **annoyance**, **attribution**, and **attacking**. While the attacking part matches the likely illegal "hacking back" scenario, the annoyance and attribution types of active defense can quite easily fit within the law. The goal of active defense is to, in a *legal way*, slow down the attacker, frustrate them, cause them to expose their location within our environment, and even perhaps help attribute the activity. Implemented properly, active defenses open some amazing detection opportunities for the Blue Team!

[1] <https://www.blackhillsinfosec.com/projects/books/>

## Active Defense: Confusion and Annoyance

- Poisons the attacker's OODA loop
  - Observations cannot be trusted
  - Orient becomes difficult / impossible
  - Decision based on false information
  - Actions cause you to identify them
- Slows attacker's OODA loop
- Improves your OODA loop through fast, accurate detection



### Active Defense: Confusion and Annoyance

One goal of active defense is to confuse and annoy attackers early-on in the attack. Even simple active defense measures can bring great value to the Blue Team if they provide a significant annoyance factor to an attacker. Why? Because once the enemy has to consider whether you are actively trying to deceive them, they must reconsider all their actions and question if what they are seeing is real. Attacking becomes much harder because they cannot easily tell what a real system is and what is a trap. This "poisons" their OODA loop because the observations they make can no longer be trusted. If they can't trust them, they can't correctly orient to the situation, make a solid decision on how to move forward, and are likely to act in a way that makes them reveal themselves. This is the true goal of active defense—forcing errors and obvious actions to be taken such that you can locate the attacker and kick them out.

## Active Defense: Breach Detection Tactics

**Goal:** Detect data that was, or is about to be stolen  
**Tactics:**

- Bugged Word documents and PDFs
- Monitored Windows folder access
- Honey tables and databases
- Fake cloud API keys
- Fake communication app API keys



### Active Defense: Breach Detection

A later attack stage implementation of active defense is "breach canaries" used to detect when items either have been stolen or are in imminent threat of being stolen. This can take the form of folders, services, API keys, and more. The idea is to use features built into these services and formats, to let you know when someone has accessed a resource, folder, or file, and set out fake tempting looking resources that would attract an attacker. As an example - Word and PDF documents can reach out to the internet upon opening through auto-loading remote pictures. An attacker that steals a tempting looking document or PDF and opens it on their own system (assuming it's connected to the internet) will then automatically cause a DNS or HTTP query to be made in a unique way to a server of your choice, which can then send an email notifying the security team that a specific document that was set out as a trap has indeed been stolen or accessed. While you may think at this point it's "too late", remember that in a breach not all data is stolen at once. Knowing that data has started leaking is a great way to ensure it doesn't continue for months like it might have otherwise taken you to realize it.

Some options for the annoyance type of active defense that are commonly deployed:

- Honeypot systems that look like and are named like the real production systems, but any contact with them sets off an alert to the SOC. This method works for tables in databases, users, or even just listening ports on systems that may be scanned by attackers.
- Unusable tripwire accounts left in privileged groups: These accounts sit and wait for discovery and attempted use by an attacker. Any activity with them at all is an immediate, high-fidelity detection and login is disabled so that they do not pose any additional risk.
- Deceptive DNS records: You can use DNS records for systems that do not exist, knowing that any traffic that attempts to access them is malicious.
- Frustration inducing webservers: I left this one vague because there are so many different variations on this. Blocking user-agents, web labyrinths, fake folder names, and more can make a webserver look normal, but under a scan from an attacker cause them to become frustrated, and also give up their intentions.

## Active Defense: Attribution Tactics

- Goals:
  - Expose the actor behind the attack
  - Improve/gather threat intelligence
- Common methods:
  - Honey documents with web bugs
  - Web-bugged emails
  - Location logging features



### Active Defense: Attribution

Taking active defense a step further can bring potential to gain additional information about the actor behind the attack. The goal of attribution-focused active defense is to figure out *who* is behind the attack by either identifying their physical or network location, or more about their intentions (which may indirectly help you understand who they are).

While there are numerous gray-area methods and tools that can be deployed for this type of activity, the page above lists some of the most-likely to be legal, and safe to use methods.

- Honey documents – Utilizing the previously mentioned bugged word document and PDF tactic can also lead to attribution. If the attacker has an active internet connection that isn't cloaked, you now have their IP address to either use for intel, attribution, or to give to authorities in the case of extortion or blackmail attempts.
- Web-bugged emails – This remotely loading resources technique from documents also works for tracking an attackers IP when they open an email (assuming they load pictures in the email). Legal approval of this technique should not be hard to gain. There are many documents out there with these features in use already and nearly every marketing email you've ever received uses the web bug technique to know if you've opened it or not..
- Location logging features - This is a broad catchall description for any method of attempting to geolocate either the user of a program, or who accesses a specific sensitive part of your website. The idea is to get a true geo-location of an attacker if they slip up and allow the lookup to occur. One method is to put location identification code into the administrative pages of your web services. Browsers are capable of asking the user for location, so if the attacker has this feature turned on, or accidentally accepts it, you will receive a rough geolocation of where they are located. The second form of this is to embed code in sensitive applications that does the same thing. If the code is ever taken and run somewhere unexpected, you will have a record of that. Again, these techniques are common in many websites and software already deployed and, therefore, legality of this should be easy to argue, but always consult with your legal team to be sure.

## Active Defense Frameworks: MITRE Shield<sup>1</sup>

- A newer MITRE project focusing on **active defense** and **adversary engagement**
  - First released in August 2020
- Contains:
  - **Matrix** of tactics and techniques, like ATT&CK
  - **High-level** CISO-ready considerations
  - **Low-level** practitioner-ready **Tactics, Techniques, Procedures**
  - Structured and unstructured data

### Active Defense Frameworks: MITRE Shield

A new project called related to active defense was released in August of 2020 by MITRE called "Shield". The Shield project, according to MITRE, "Shield is an active defense knowledge base MITRE is developing to capture and organize what we are learning about active defense and adversary engagement. Derived from over 10 years of adversary engagement experience, it spans the range from high level, CISO ready considerations of opportunities and objectives, to practitioner friendly discussions of the TTPs available to defenders."<sup>1</sup>

While the framework is still an early work in progress, the creators have already done a great amount of work to standardize and organize active defense measures in the same style as the popular MITRE ATT&CK matrix. This info includes both high-level information such as CISO-ready considerations as well as low-level, practitioner-focused tactics, techniques, and procedures.

[1] <https://shield.mitre.org/>

## MITRE Shield Matrix

- Tactics for what the **defender** is trying to accomplish
- Techniques for **how** to achieve the tactic
- Each technique also has **Opportunities, Use Cases, Procedures, and ATT&CK Mapping**
  - Each is technique and associated info is uniquely numbered

Channel	Collect	Contain	Detect	Disrupt	Facilitate	Legitimize	Test
Admin Access	API Monitoring	Admin Access	API Monitoring	Admin Access	Admin Access	Application Diversity	Admin Access
API Monitoring	Application Diversity	Baseline	Application Diversity	Application Diversity	Application Diversity	Burn-In	API Monitoring
Application Diversity	Backup and Recovery	Decoy Account	Behavioral Analytics	Backup and Recovery	Behavioral Analytics	Decoy Account	Application Diversity

## MITRE Shield Matrix

As with the ATT&CK Matrix<sup>1</sup>, the data in Shield is organized with the active defense *tactics* horizontally across the top row, and techniques for achieving that specific tactics underneath in each column. What's different from ATT&CK, is that tactics represent what the *defender* is trying to accomplish in Shield, as opposed to the attacker (like in ATT&CK).

Each technique in the Shield matrix is described in terms of

- **Opportunities** – high-level active defense possibilities for a technique
- **Use Cases** – high-level descriptions for how a defender can take advantage of an opportunity the attacker's action presents
- **Procedures** – specific implementations of a technique
- **ATT&CK Mapping** – Traceability to MITRE ATT&CK matrix techniques that relate to the technique

[1] <https://shield.mitre.org/>

## MITRE Shield Technique Example – Decoy Account<sup>1</sup>

Description: "Create an account that is used for active defense purposes. A decoy account is one that is created specifically for defensive or deceptive purposes. It can be in the form of user accounts, service accounts, software accounts, etc. The decoy account can be used to make a system, service, or software look more realistic or to entice an action."

### Use Cases

ID	Description
DUC0004	A defender can create decoy user accounts which are used to make a decoy system or network look more realistic.
DUC0044	A defender can use decoy accounts and monitor them for any activity that might reveal adversary manipulation.
DUC0187	During an adversary engagement operation, a defender can utilize decoy accounts to provide content to an adversary and encourage additional activity.

### Procedures

ID	Description
DPR0020	Create a user account with a specified job function. Populate the user account's groups, description, logon hours, etc., with decoy data that looks normal in the environment.
DPR0021	Create a user that has a valid email account. Use this account in such a way that the email address could be harvested by the adversary. This can be monitored to see if it is used in future attacks.

## MITRE Shield Technique Example – Decoy Account<sup>1</sup>

Let's look at an example technique, Decoy Accounts, an item that appears from multiple tactic categories (techniques can belong to more than one tactic). The description for this active defense tactic describes the idea – creating a fake account that is not used by any human or service, but purely for tactical detection purposes. As listed, Use Cases for this technique are numerous – making a decoy system look more realistic, monitoring for activity, and for tempting an attacker into additional activity during an incident response engagement.

Procedures listed for this technique give additional guidance on carrying out the account creation – adding additional realistic looking details to the account, as well as creating an email address for it and leaving it in places you might expect attackers to look for building phishing lists and more.

In addition to these Use Cases and Procedures, the Decoy Account technique also lists the follow Opportunities:

- DOS0001 There is an opportunity to study the adversary and collect first-hand observations about them and their tools.
- DOS0004 There is an opportunity to introduce user accounts that are used to make a system look more realistic.
- DOS0187 In an adversary engagement operation, there is an opportunity to present decoy accounts to the adversary during the enumeration process.

Through the listed Use Cases, Procedures, and Opportunities, the Shield framework gives you the information needed to send you down the path of implementing this technique for active defense purposes, as well as references the ATT&CK Techniques that it can be used to detect (Valid Accounts, Account Discovery, and Account Manipulation). For additional information on getting started with Shield, see the "Getting Started" page and referenced documents within in the footnote below.<sup>2</sup>

1 <https://shield.mitre.org/techniques/DTE0010/>

2 <https://shield.mitre.org/resources/getting-started>

## Deception and Active Defense Linux Distros

- For all things active defense, check out *ADHD*<sup>1</sup>
  - "Active Defense Harbinger Distribution" by Black Hills Infosec
  - <https://www.blackhillsinfosec.com/projects/adhd/>
  - A Linux distribution with active defense tools ready to run
- Tools
  - Cowrie – Interactive SSH Honeypot
  - Docz.py – Honeydoc maker
  - Weblabyrinth – Site scanning tarpit
  - HoneyPorts – Auto IP block after connection



### Deception and Active Defense Linux Distros

The easiest way to step into trying out active defense techniques is through Black Hills Infosec's Active Defense Harbinger Distribution (ADHD)<sup>1</sup>. This pre-configured Linux distribution comes with a whole list of useful tools ready to go and example instructions on how to use them.

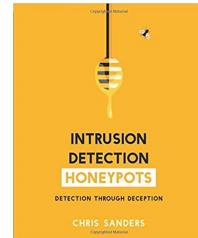
For details on the tools included and examples see reference [2].

- 1 <https://www.blackhillsinfosec.com/projects/adhd/>
- 2 <https://adhdproject.github.io>

## Active Defense References

### Books:

- Intrusion Detection Honeypots<sup>1</sup> – Chris Sanders
- Offensive Countermeasures: The Art of Active Defense<sup>2</sup>
  - John Strand, Paul Asadourian, Benjamin Donnelly, Bryce Galbraith, and Ethan Robish



### Tools and Links:

- Thinkst Canary - [canarytokens.org/generate](http://canarytokens.org/generate)
- T-Pot<sup>3</sup>
- Anomali Modern Honey Network<sup>4</sup>
- GitHub awesome-honeypots repository<sup>5</sup>

### Active Defense References

For those looking for in-detail tactics and tools to build up their active defense strategies, this slide lists some resources to get you started. There are multiple books available on the subject as well numerous open-source software packages that can be used for nearly any commonly attacked service and protocol. See the footnotes below for details.

- 1 <https://chrissanders.org/2020/09/idh-release/>
- 2 <https://www.blackhillsinfosec.com/projects/books/>
- 3 <https://github.com/telekom-security/tpotce>
- 4 <https://github.com/pwnlandia/mhn>
- 5 <https://github.com/paralax/awesome-honeypots>

## Example - Website Clone Detection<sup>1</sup>

**Problem:** Cloning your org's login pages is *very* easy, makes phishing simple

**Solution:** Identify your "site" being hosted anywhere else on the internet

**Active Defense Approach:** Place JavaScript on your site to load a remote resource if the page is *not* hosted on the expected domain

```
if (document.domain != "https://yourorhere.com") {  
    var l = location.href;  
    var r = document.referrer;  
    var m = new Image();  
    m.src = http://canarytokens.com/ +  
"5xuldrjqqmct8s3bq0wsmt6dd.jpg?l=" + encodeURIComponent(l) +  
&r=" + encodeURIComponent(r);  
}
```

### Example – Website Clone Detection

Here is an example of one of my favorite types of active defense tactics as provided by canarytokens.org<sup>1</sup> – website clone detection. This is a *fantastic* way to get ahead of your enemies by preemptively embedding active defense in your externally facing login pages.

Here's how it works: Adversaries want to phish your employees, so they will use tools like the Social Engineering Toolkit, which for example, is included in Kali, and clone your organizations internet facing portals that employees use to login. Once cloned, they will have a lookalike site they can phish people with to steal your credentials. How can you programmatically detect these sites? By embedding code within your own that will check if the site is hosted in the expected location and make a unique URL callout if the code from your page is cloned and hosted anywhere else (the JavaScript on the page above). In essence, the code says if it isn't hosted at yourorhere.com, load a URL at canarytokens.com with a unique identifier and include the location of the cloned page in the callback. As long as you have visibility to the callback site, you immediately are notified if anyone loads a cloned version of your webpage. How cool is that??

You may be saying "well can't they just detect this code and remove it in the clone?" – technically yes, but this is only one method to accomplish this, you can also obfuscate the JavaScript and embed it within other code, making it harder to identify for attackers. Do not get stuck in the "all or nothing" thinking trap, tactics like this are worth their weight in gold as they will often detect attacks early in the kill chain, *before* attackers make any meaningful progress.

[1] <https://canarytokens.org/generate#>

## When to Move to Active Defense

- Active defense *can* be used by anyone!
  - Are you ready?
- Success factors:
  - ❑ Legal / business approval
  - ❑ Specified use case / threat model driven strategy
  - ❑ Knowledge of deployment environment
  - ❑ Control or knowledge of false positive generating traffic
  - ❑ Understanding of safe implementation
  - ❑ Management and response plan



### When to Move to Active Defense

Active defense is an amazing capability due to its highly effective nature and low false positive rates, but you may be wondering whether your organization is ready for it? We opened this section saying you should focus on getting the basics right first before moving in this direction, and there is some truth to that, you should "walk before you run" as they say. This doesn't mean, however, that you can't get immediate value out of some simple active defense and deception techniques with very little effort, even without a perfect environment.

Consider the criteria on the page above—if you meet these requirements, even if it's for a small section of your network or subset of users, you may be able to easily deploy active defense or deception techniques with great success. Given you have the items listed below, you may be ready to test out some of the tools discussed in this section!

- The legal and business approval
- Understanding of your attacker and the techniques you're trying to identify
- Understanding of your environment and how false positives might be produced / how you can filter them
- How to safely deploy the techniques
- A management and response plan for your tools

## Common Mistakes (2)

- Using insights to actively target attacker infrastructure
- Hacking back!
- Blazing a trail without plan or strategy
  - Document the where and why of your deception tech
  - Capture in your alert documentation as you would any other analytic
- Snaring users and administrators in the net

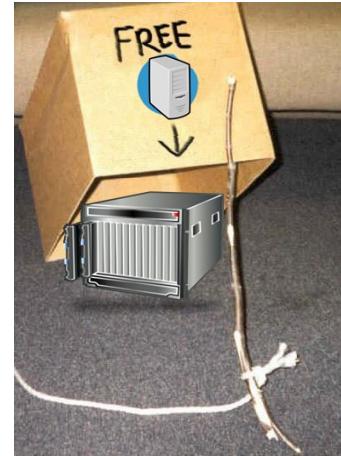
### Common Mistakes (2)

Based on some of the advice we've given you in this section, you may be able to surmise some of the mistakes one could make when building out an active defense capability. However, given the risk of collateral damage to your team, some of these probably bear repeating. The main goal is to break the intruder's OODA loop, cause them to doubt their observations, and drive up their cost to the point where it is no longer advantageous to target your infrastructure. It's entirely possible to carry this line of thinking into active disruption of the attacker, and potentially targeting infrastructure the attacker is using. Doing so can not only compromise your operational security but can also land you in very hot water with your legal division and even law enforcement. Remember, active defense is not hacking back!

Of less concern but perhaps even more common is deploying deception measures without proper planning or strategy. If you're using fake accounts, files, or other tokens as a detection mechanism, remember to track where those mechanisms are, how they are deployed and monitoring, and what you should do if the detection is triggered – just as you would any alert source. Finally, while we want to make these tokens and decoys as attractive as possible to attackers, we don't want our administrators and users tripping these defenses. Avoid leaving token files, directories, and other artifacts with attributes that would entice any user in places where users are likely to come across them – for example, leaving a token called "private salary information" in a user's home directory.

## Active Defense Summary

- Usually legal to do in *many* useful ways
- Ruins the attacker's OODA loop
- Gives a low false-positive detection
- Low maintenance once set up
- Can be used for
  - Annoyance – Slowing the attacker down, making them reveal their presence
  - Attribution – Figuring out who they are
- Lots of fun too! 😊



### Active Defense Summary

Active defense is one of the tools in the defender's toolbox that's incredibly fun to deploy, very sneaky, and can lead to high-value detections. The largest hurdle is typically just getting it approved. As long as you are not aggressively pursuing attackers or setting up a situation that could be considered entrapment, there are usually plenty of methods of active defense and deception that can be legally deployed. Since these techniques work beautifully to disrupt, confuse, annoy, and potentially even help identify the attacker, they are absolutely worth pursuing once your team has the time and approval to do so.

# Course Roadmap

- Book 1: SOC Design and Operational Planning
- Book 2: SOC Telemetry and Analysis
- *Book 3: Attack Detection, Threat Hunting, and Triage*
- Book 4: Incident Response
- Book 5: Metrics, Automation, and Continuous Improvement

## SECTION I

### Introduction

#### Creating and Processing Alerts

- Efficient Alert Triage
- Detection and AnalyticDesign
- Capacity Planning
- *Exercise 3.1 – Capacity Planning*
- Detection Engineering

#### Advanced Analysis

- Analytic Frameworks and Tools
- *Exercise 3.2 – Structuring, Documenting, and Organizing Use Cases*
- Threat Hunting
- *Exercise 3.3 – Planning a ThreatHunt*
- Off-Hours Alerting and On-Call
- Active Defense
- **Summary and Cyber42 – Day 3**



This page intentionally left blank.

## Day 3 Summary

In this book, we covered:

- Alert triage and capacity planning
- Analytic design
- Use case design and management
- Detection engineering as a formal SOC discipline
- Analytic frameworks and tools
- Threat hunting
- Off-hours coverage and escalation
- Active defense



### Day 3 Summary

In this book, we covered the basics of detection: creating the alerts, triaging them, and planning capacity to make sure we won't become over-subscribed. We also talked about detection engineering as a formal discipline, and some of the structure and rigor you'll need to formalize it as its own discipline within your team. Then we moved into more advanced analytic topics like structured analysis frameworks and tools, proactive threat hunting, handling alerts outside of your normal coverage window. Finally, we covered principles of active defense – what many groups now refer to as “deception technology”. By combining these repeatable processes and looking ahead towards more advanced analytic techniques, your SOC can continue to evolve its detection capabilities in a way that is innovative *and* measurable.

## Cyber42



# Cyber42 Simulation

## Day 3

SANS

MGT551 | Building and Leading Security Operations Centers 163

### Cyber 42

Your instructor will now give you instructions on how to access the Cyber42 game. OnDemand students should refer to their supplemental documentation for instructions for access.