

55 I.4

Incident Response

The SANS logo consists of the word "SANS" in a bold, white, serif font.

THE MOST TRUSTED SOURCE FOR INFORMATION SECURITY TRAINING, CERTIFICATION, AND RESEARCH | sans.org



551.4: Incident Response

© 2021 John Hubbard and Mark Orlando | All Rights Reserved | G02_02

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

TABLE OF CONTENTS	PAGE
Introduction	of Class 1
Incident Response Planning and Preparation	4
Investigation	37
Exercise 4.1 – Investigation Quality Review	58
Identification, Containment, and Eradication	60
Incident Response in the Cloud	92
IR Tools	102
Exercise 4.2 – Planning Responses with RE&CT	124
Crisis Management and Continuous Improvement	126
Exercise 4.3 – Designing Tabletop Exercises	145
Recovery and Post-Incident	147
Conclusion	156



MGT551 | Building and Leading Security Operations Centers 2

This page intentionally left blank.

Day 4 Overview

- **Introduction**
- **Preparation, Planning and Review**
 - Incident Response Planning and Preparation
 - Investigation
- **IR Execution**
 - Identification, Containment, and Eradication
 - Incident Response in the Cloud
 - IR Tools
 - Crisis Management and Continuous Improvement
 - Recovery and Post-Incident
- **Exercises:** Investigation quality review, Incident response with RE&CT, and Designing tabletop exercises, planning



Day 4 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

Planning, Preparation, and Review

- **Incident Response Planning and Preparation**

- Investigation

- *Exercise 4.1 – Investigation Quality Review*

IR Execution

- Identification, Containment, and Eradication

- Incident Response in the Cloud

- IR Tools

- *Exercise 4.2 – Planning Responses with RE&CT*

- Crisis Management and Continuous Improvement

- *Exercise 4.3 – Designing Tabletop Exercises*

- Recovery and Post-Incident

- Summary and Cyber42 – Day 4

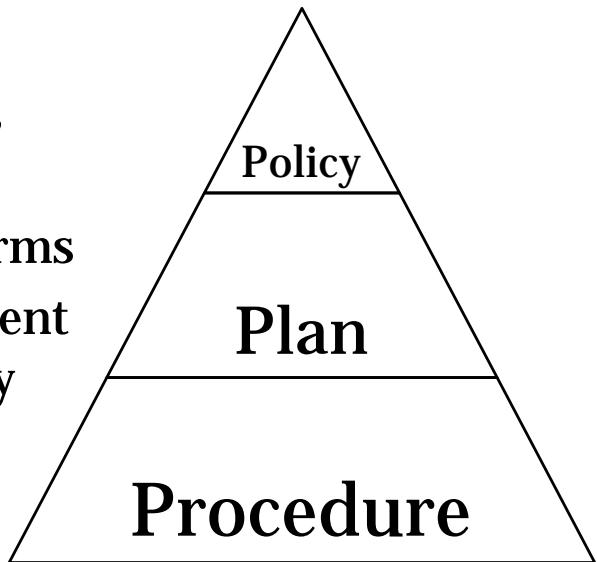


This page intentionally left blank.

Preparing for Incident Response

Prepare for IR by creating an incident response policy, plan, and procedure:

- **Policy** sets the rules and norms
- **Plan** defines how to implement the capability based on policy
- **Procedure** is the steps to follow during an incident



Preparing for Incident Response

Now it's time to shift the conversation to the incident response function. In this module, we'll discuss incident response as if it were a potentially separate group within the SOC since, in many cases, this is true. Incident response can be thought of logically as another function, or in smaller teams, just something the analysts also take care of once an alert is validated. Either way, the planning and considerations here apply, the documentation may just be merged with the rest of your SOC policies and procedures if you do not have a dedicated IR team.

To start off, we'll quickly discuss the three foundational documents your incident response capability will be based on—the policy, plan, and procedure. This page describes what each of these documents is designed to do: the policy is a high-level direction setting document, the plan is how that policy will be brought to life, and the procedure will be the low-level tactics the IR team uses.

Incident Response Scenarios to Consider

Which systems might you need IR capabilities for?

- User systems – local, remote, virtual, cloud, BYOD
- Servers – physical, virtual, local, remote, cloud, containers
- Cloud Services – PaaS, SaaS, FaaS
- Mobile Devices – Android, iOS, ...
- Internet of Things
- Operational Technology & ICS Equipment

Options for each:

- Insource or outsource the capability
- Determine data retention periods and impact on IR

Incident Response Scenarios to Consider

Incident response scenarios can be highly variable depending on the location and system type where the attack occurs. For example, work on a cloud-based server breach is highly different compared to working physically with an infected user device. Each scenario has a very different experience, with different set of tools and data sources. Therefore, a main consideration for planning is which scenarios your SOC is willing, interested, and capable of doing incident response for. It is likely your SOC charter specifies this in a high-level way as your scope of responsibility, but it is better to consider each device type (desktop vs. server vs. IoT vs. mobile) and location (on-premise, remote site, cloud, etc.) and specifically consider whether you are ready for a situation of each type. You may be ready with traditional tools if one of your on-premise servers are compromised, but what about a cloud attack - does your SOC have the data necessary from the cloud platform itself to spot a breach? Will a short data retention period make it difficult after a short period of time? Will you be outsourcing help from others? Knowing where you will turn beforehand prepares your team so that you don't waste time evaluating incident response vendors and getting contracts put in place under the pressure of an active incident.

Incident Response Policy

IR policy document should include:

- Statement of management commitment
- Definition of an incident
- Purpose and scope
- Org. structure, roles, responsibilities
- Guide to incident prioritization
- Performance measures

Goal: High-level, short, to the point



Incident Response Policy

The incident response policy is the "top of the pyramid" document which should be a relatively brief description of the things the IR team is hoping to accomplish, documented backing by management, the purpose and scope of the team, prioritization guidance, and other such items. This is the first document you will likely make as it defines the type of team you are bringing to life and what their mandate is. This should not be so detailed that you need to consult it or modify it any time a new procedure needs to be enacted or toolset changes.

Incident Response Plan

IR plan: How you intend to achieve the policy

- ❑ Mission
- ❑ Strategies and goals
- ❑ Organizational incident response approach
- ❑ Internal and external communication methods
- ❑ Roadmap for maturing IR capability
- ❑ Fit into the overall security and IT organization
- *See notes for example from Virginia Tech IT Department¹*



Incident Response Plan

The incident response plan is the next item you should look to create. The IR plan will define exactly *how* you are going to operate, the mission, strategies, and goals of the team, as well as your high-level approach to building the team, communication standards, and your roadmap for continuing to mature into the future. You also may want to document how the IR team fits specifically into the SOC or security organization at large. A good example of an IR document that has these details plus some of the policy items can be found on Virginia Tech's website at the reference below.¹

[1] https://security.vt.edu/content/dam/security_vt_edu/downloads/incident_response.pdf

IR Team Formation

Common IR team structures:

1. Centralized Team

- A single IR team for whole organization
- Good for small and/or single geography organizations



2. Distributed Team

- Multiple IR teams, split authority
- Good for highly segmented, geographically distributed orgs

3. Coordinating Team

- Providing help to other team that has higher authority
- Less common, good for federated or hierarchical organization

IR Team Formation

Since your IR plan will require considering how to construct your team, now is the time to think about the IR specific requirements (if they aren't already part of your SOC at large). Some of the common incident response team formats are listed on the page above.

If you work in a large organization with mostly in-house capabilities, a larger budget, and a full SOC, you'll probably go for the Distributed team. These types of teams will be available in a physical sense in most places where an incident might occur and can give the fastest response times. In a smaller and more geographically bounded organization, you might choose the single centralized SOC model, which works great if it can cover all the ground necessary. In some situations, especially government scenarios where there are multiple levels of authority reporting up through one another, you may also find the coordinating SOC, but these are less common. Additional considerations for choosing your team type are on the following page.

Staffing Model for the IR Function

Common staffing models: Factors in selection:

- **In-house:** Typically found in large organizations with big budget
 - **Partial outsource:** Small to medium orgs, may outsource only some functions (forensics, malware)
 - **Fully out-sourced** – For small or newer organizations that need immediate capability
- Expected workload
 - Team size
 - Full-time vs. part-time team
 - Budget
 - Expertise available in-house
 - Availability requirements (24x7 or 9x5?)

Staffing Model for the IR Function

Here are some options to consider within each of the previously mentioned staffing models. When thinking through the options, consider the requirements and restraints on your team. Do you know how many cases you expect to have to work, the team size you will have, if people will be dedicated to the function or only part-time responders, etc.? These, plus the available in-house talent, can guide you to the best answer for your situation.

The simplest model for staffing is a full in-house set of employees dedicated to incident response. If you can afford a team such as this and it is necessary due to the number of incidents you may deal with, it's likely the most cost-efficient option. Large and more mature organizations will usually have the full suite of capabilities—malware analysts, forensics, and more, available fully in-house.

If you work at an organization with a much smaller budget or workforce, or you just have great preventative controls, it may be rare that you have such a large incident that a dedicated team is necessary. In this case, you may choose to partially outsource the capability, letting your analysts cover the basic cleanup and simple cases, and calling in specialists if/when something particularly large happens once in a blue moon. You may also choose to only outsource very small parts of incident response for those subspecialties you rarely need, such as manual, deep malware analysis. This can help alleviate the costs of keeping dedicated IR practitioners on staff that won't be needed most of the time.

Fully outsourcing the capabilities is often done for the smaller businesses that do not have the staff or experience in-house, and where it does not make sense to move in that direction over the longer term. The other option, for a new team, is to start out fully outsourced but use a hybrid partial outsourcing model to help stand up your capability until you can work toward incident response independence.

Communication Guidelines and Methods

Communication considerations:

- Who will you share information with?
 - Internal: System owners, business units, board, etc.
 - External: ISPs, vendors, partners, law, media
- When you will share it?
- How will you securely communicate with stakeholders?
 - (Hint: **Out-of-band!** Attackers will compromise email if they can)
- How often and who will give updates?
 - Designate an "Incident lead" position for communication
 - Ensure analysts are not bogged down giving updates



Communication Guidelines and Methods

In the heat of battle, it can be easy to pass up on letting outside the SOC stakeholders know what's going on. Analysts and responders will be deep into details and you won't want to break their concentration to make them write status emails, and for good reason—time is of the essence! For this reason, it is best to pre-plan how communications will work, both in terms of who will be read in, the frequency, roles, and also practical matters such as what tools you will use to communicate.

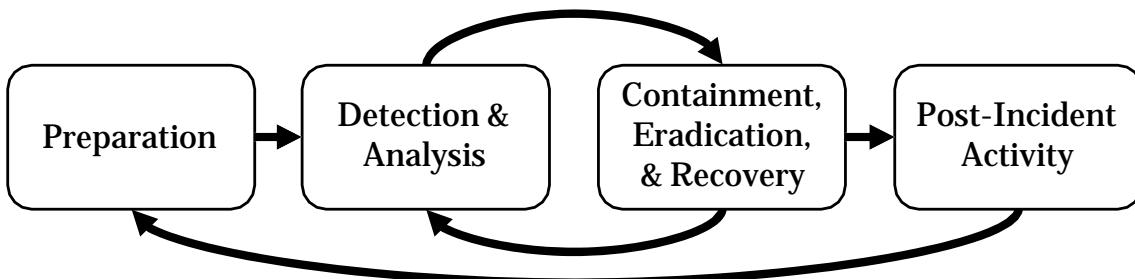
One item to write into your procedure is who you will be communicating with and when it is permissible and advisable to notify them. This includes both internal and external parties. For external parties especially, it is crucial to be crystal clear about when and how communications will go out as letting the outside world know your company has been breached can have large ramifications. You also need to know at what point something has progressed far enough to contact the asset owners and let them know what's happening before taking action on their system.

Also consider *how* the communication will be done. For typical incident response that doesn't involve anyone's user account, email, Skype, and Slack may be fine. For advanced attackers that have compromised a considerable portion of the environment, however, a backup plan for *out-of-band* communication should be set up in order to avoid adversary eavesdropping on your cleanup attempts. Suggested methods for this include Signal, Telegram, iMessage or any other secure private messaging app, non-company email, and plain phone calls to personal phones.

The final key consideration is who will give updates and how often they will be given. Since incident responders and analysts will have their hands full, dedicating a single point of contact as an "incident lead" can ensure the distractions to others stay at a minimum, while the incident lead takes charge of gathering and summing information for outward communication from the SOC.

Incident Response Procedure Overview

- IR procedure:
 - Based on the policy and plan
 - The specific techniques and processes to use in an incident
- Organized into the following phases:

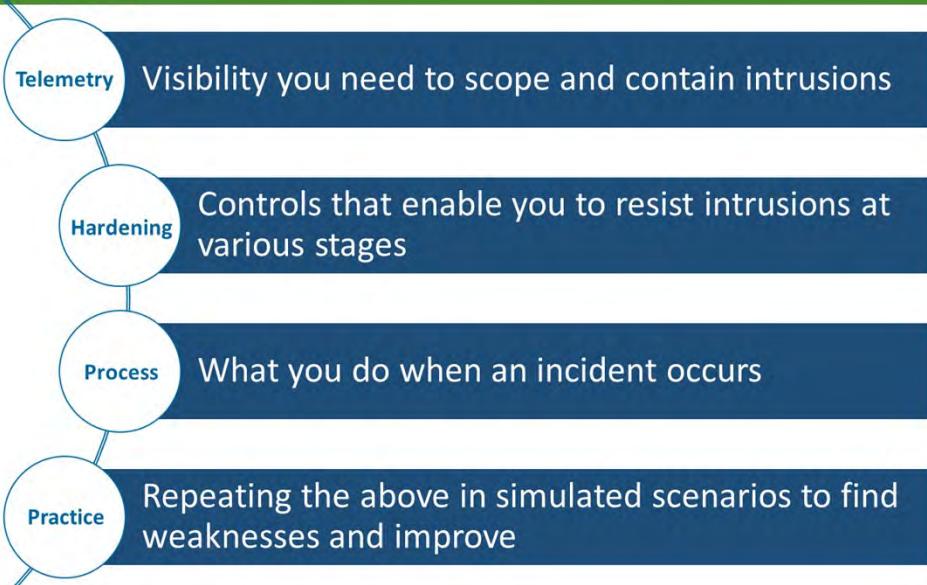


Incident Response Procedure Overview

Now that there is an established IR plan and policy, it's time to look at the procedures that will be put into place. Looking again at the guidance from NIST 800-61r2, the phases laid out for incident response are "Preparation", "Detection and Analysis", "Containment, Eradication and Recovery", and "Post-Incident Activity". When crafting the incident response procedure, these items help us decompose and keep track of what needs to be done, and in what order.

In this course, we've already covered planning from a whole SOC perspective and the detection / analysis phase as well. Therefore, in the rest of this module, we will continue to discuss *incident response specific* planning as well as the containment, eradication, recovery, and post-incident activity portions of this diagram.

Preparation (1)



Preparation (1)

In *Intelligence-Driven Incident Response*, Rebekah Brown and Scott Walker outline four key elements of incident response preparation: *telemetry, hardening, process, and practice*. We've already talked about preparing your network for reliable monitoring and detection, but it's important to reiterate that regardless of the role your SOC plays in IR, visibility will be foundational for everything you do in an investigation or response activity.

But having the raw data isn't enough – devising analytics and identifying key assets, users, and other contextual information will help your team prioritize response efforts (and, depending on the affected users or systems, may change your response entirely). Likewise, you should be working to make your network more resistant to attack by deploying or improving security controls and addressing gaps wherever you find them. Procedures aren't something you want to figure out as you go, so having a solid set of process documentation with tools configured to enforce it is a must-do as well. Finally, practice is perhaps the most often overlooked part of incident response; it's vitally important to give your team a chance to walk through procedures in a safe environment before a real crisis occurs.

Prevention IS Ideal

- Long term IR preparation: preparing your infrastructure to resist intrusion
 - Best practices like CIS security controls
 - Reference models like Defensible Network Architecture, least privilege, zero trust
- Focusing on short-term prep without these strategic initiatives will prevent you from making meaningful progress



Prevention is Ideal

While a modern defensive mindset is detection-oriented, we obviously want to prevent impacts from as many threats as we can. Before we talk about short-term preparations for incident response, we want to make sure we're working to prepare our environment to resist intrusion and support detection when prevention eventually fails. This means deploying controls, collecting telemetry, and striving for an ideal security posture using reference models like zero trust and least privilege. The next several slides will cover some of these references and best practices in more detail.

Defensible Network Architecture: The Return

- ...From Day 2! In incident response, DNA means:

- *Monitored*: Security team can view host, network, and application logs
- *Inventoried*: Security team can identify asset location, purpose, data classification, criticality, and owner info
- *Controlled*: Access control enforced at host, network, application levels
- *Claimed*: Asset owner is tracked in an inventory system
- *Minimized*: Access to asset ("surface area") provides only what is necessary to support business function
- *Assessed*: Asset security posture is routinely evaluated
- *Current*: Patch and configuration levels are kept up to date
- *Measured*: **Security team measures progress against previous status**



Defensible Network Architecture

Recall that we talked about Defensible Network Architecture as a reference model for giving your network a fighting chance against attackers. A defensible network is one that is monitored, inventoried, controlled, claimed, minimized, assessed, current, and measured. We've covered these items at a high level, but here we see the implications of each from the SOC's perspective. The last item, added based on feedback to Richard Beitlich's initial post, is perhaps the most critical of all of these, especially in environments where some of these (or all of these) may not be in place: tracking progress against previous status. Tracking and reporting the defensibility of your network is a great way to identify gaps and gain a better understanding of where your ability to detect and respond to intrusions may need additional work.

CIS Benchmarks

- Best practices for secure configuration
 - Windows Server
 - Amazon Web Services
 - Google Cloud Computing Platform
 - Microsoft Azure
 - Many, many more
- Depending upon your org, you may also be subject to PCI-DSS, ISO 27001, SOX, HIPAA hardening requirements



SANS

MGT551 | Building and Leading Security Operations Centers

16

CIS Benchmarks

Hardened infrastructure starts with good configurations, but knowing what settings and changes constitute "good" from a security perspective across a wide variety of different operating systems and applications can be tough. Enter CIS benchmarks[1]! The freely-available benchmarks are a set of best practices for secure configurations on everything from operating systems like Windows Server to cloud infrastructures like AWS, GCP, and Azure to applications like Zoom. Each set of recommendations references specific CIS controls (which we'll get to next) designed to improve an organization's resistance to cyber attack. The benchmarks provide two levels of security settings:

1. **Level 1**, which are basic security requirements that should have little impact on functionality, and
2. **Level 2**, which are changes for greater security that could limit or reduce system functionality

CIS also offers hardened images[2], which are securely configured virtual machine images based on CIS Benchmarks hardened to either a Level 1 or Level 2 CIS benchmark profile.

1. <https://www.cisecurity.org/cis-benchmarks/>
2. <https://www.cisecurity.org/blog/cis-hardened-images-now-in-microsoft-azure-marketplace/>

Other System Hardening Resources

- **NIST SP 800-123: Guide to General Server Security**
 - US regulatory standards like HIPAA, HITRUST, CMMC are based in part on these measures
- **NIST National Checklist Program Repository**
- **DISA Security Technical Implementation Guide (STIG)**

Other Resources

CIS may be the most comprehensive, community-driven initiative to provide system hardening guidance, but it is by no means the only resource out there. The National Institute for Standards and Technology's Special Publication (SP) 800-123 [1] provides high-level, general guidance for hardening server infrastructure. NIST also hosts the National Checklist Program Repository, which provides a searchable index of hardening guides for all kinds of systems and applications. Many of the resources you'll find in the NCPP are documents created by the US Defense Information Systems Agency, or DISA. These Security Technical Implementation Guides, or STIGs, are in-depth step-by-step checklists for locking down systems. As anyone who has worked in the Department of Defense implementing STIGs will tell you, your mileage may vary in terms of how restrictive these checklists can be – they are normally meant to be tailored and applied piece-by-piece to avoid breaking application or server functionality. But, when used correctly, the STIGs can be some of the most useful system hardening guides out there. And they're all free for download!

- 1 <https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-123.pdf>
- 2 <https://nvd.nist.gov/ncp/repository>
- 3 <https://public.cyber.mil/stigs/>

CIS Controls (1)

- Prioritized set of actions for defense in depth sourced from community
- Basic tenets:
 - Offense informs defense
 - Prioritization
 - Measurements and metrics
 - Continuous diagnostics and mitigation
 - Automation
- Basic > Foundational > Organizational
- Classified by implementation group



SANS

MGT551 | Building and Leading Security Operations Centers

18

CIS Controls (1)

The CIS Controls™ are a prioritized set of actions, sourced from various community experts, which form a defense-in-depth set of best practices to mitigate common attacks. These controls are intended not only to support detection, but to minimize attack surface and improve resistance to attack at all phases of the kill chain. These actions reflect the following five basic tenets:

- **Offense informs defense:** the CIS Controls are informed by real-world attacks
- **Prioritization:** the controls are organized in such a way that investing in them in order will provide the greatest risk reduction against the most damaging threats first
- **Measurements and metrics:** the CIS Controls are intended to drive common metrics for communicating an organization's risk management posture as a result of the controls it has implemented
- **Continuous diagnostics and mitigation:** assumes continuous measurement of the implemented controls to gauge their effectiveness and drive prioritization of additional controls
- **Automation:** whenever possible, defenses should be automated in order to help organizations scale and manage costs

In addition to being presented in priority order, the CIS Controls are grouped into *basic, foundational, and organizational* categories to communicate scope and scale of implementation. Each control is also classified by implementation group based on the organization's size, sophistication, and security maturity.

CIS Controls (2)

CIS Top 5:

1. Hardware asset inventory
2. Software asset inventory
3. Continuous vulnerability management
4. Admin privilege control
5. Secure configurations



SANS

MGT551 | Building and Leading Security Operations Centers

19

CIS Controls (2)

The CIS reference contains 20 controls, and each one can be a massive undertaking in a large enterprise. Implementing the top 5 controls alone will have a major impact on your organization's ability to resist intrusion, not to make your monitoring and response operations MUCH easier – so it's a great starting point. In any new environment (or in your current environment if you aren't doing these things well), understand what you're defending is fundamental in identifying and responding to incidents. The first two controls involve maintaining a current and accurate view of IT assets, which is an ongoing and dynamic process. This can be achieved via regular scanning and passive identification, the output of which can be used to build more advanced defensive capabilities like application control and unauthorized device detection. Continuous vulnerability scanning can help your team enrich alert data and prioritize activity based on attacks that are most likely to be successful. Administrative privileges should be cataloged and monitored to ensure your users aren't abusing their privileges or, even worse, that those privileges haven't been compromised. Finally, maintaining secure configurations can help us filter out benign activity and other noise and more effectively investigate anomalies for malicious activity.

System Inventory

- Preventing exploitation step 1: **Know your infrastructure!**
 - **We cannot protect what we don't know we have**
 - Network scanning checks the network for software running on each endpoint
 - Nmap is a good example scanner
 - Scans for open ports
 - Grabs banners for listening services
 - Runs scripts
 - Checks operating system versions



Software Inventory

Step 1 of preventing exploitation is having a true inventory of what is running on your endpoints. Without a list of which services and programs are running, you have no hope of protecting them from exploitation. There are two general ways this information is collected, via inventory systems and via network scanning. Inventory systems are like vulnerability scanners (discussed next) in that they can periodically log in to each machine and create a list of installed software and save it into a centralized database that can be polled. This is a great first step but is a potentially incomplete view of what is truly present. The inventory systems are only as good as the methods they use to enumerate installed software, and depending on how applications were installed, they may or may not get logged correctly.

A way to round out this view is to scan each machine over the network to see what listening services are presented to the network. Network scanning tools such as Nmap are a good representation of what these tools are capable of. Nmap can be given a list of IP addresses and attempt to connect to any open ports, enumerating what services if any are available at each one. Scans can be done at varying levels of depth and are not just a simple TCP connection. Once a successful connection is made, even without logging in, scanners can grab and interpret banners returned from each server, use heuristics to predict the operating system version, and even run scripts to probe listening services in ways that service provides. For example, a script may be used for any FTP servers found to test for anonymous logins and enumerate any files available. An SMB script may make a connection to pull the remote machines hostname, operating system and more. Combining information from network scans and software inventory systems gives us a great start at enumerating our network's attack surface so that we can protect and reduce it.

Continuous Vulnerability Scanning

- **Cyber threats change extremely quickly**
- A server that is safe today may become exploitable overnight!
- **Vuln. scanning** keeps track of versions of software on each host
 - Helps prioritize patch rollout
 - Categorizes hosts by criticality, highest risk vulns, OS type, compliance
 - Periodically logs in to each device, **not real-time**
- 2 types of scans:
 1. **Unauthenticated:** Scanning by probing over the network, limited
 2. **Authenticated:** Logging in and doing true enumeration

Continuous Vulnerability Management

Given the ever-changing landscape of cyber threats, what is completely safe one day may become your company's biggest vulnerability the next. You never know when the next surprise zero-day exploit will hit, so once you have identified software and where it is located with the inventory and network scans, you must be able to detect when it is out of date.

Vulnerability scanners are another common security product category that is offered by many vendors. The goal of these appliances is to track all deployed software throughout the enterprise, put it in a database that can be queried, and assist with prioritizing patch deployment. Vulnerability scanners collect vulnerability information from each individual host with either *authenticated* or *unauthenticated* scans. Unauthenticated scans are like network scanning—they attempt to gather as much information as possible through interacting with the host over the network, but do not have the ability to log in. These scans are extremely limited as many services will never provide the version number of the software being run. Authenticated scans allow the appliance to log in to the host being checked so that its version numbers can be truly enumerated, and the scanner can know for certain what is installed.

Administrative Privilege Control

- Administrative privileges are a **common target**
- Normally an admin user doing something unsafe while logged in with elevated privileges, or credential theft/reuse
- Many built-in capabilities and approaches to minimize this risk:
 - Privileged Access Workstation (PAW)
 - Multi-factor authentication
 - Event logs for login activity, group membership, impersonation
 - Unique passwords
 - Admin account inventory
 - PowerShell's Just Enough Administration (JEA)

Administrative Privilege Control

The misuse of administrative privileges is an extremely popular method for attackers to move around inside a target enterprise. This is normally achieved one of two ways: a user who is logged in with administrative privileges executes malicious code, or credentials of an administrator account are compromised and reused. Fortunately, most modern operating systems have built-in features to minimize these risks on individual systems and in a domain – some of those capabilities are listed on this slide. One control that you may not be familiar with here is PowerShell Just Enough Administration (JEA)[1], which enables delegated administration for anything managed by PowerShell. With JEA, you can:

- **Reduce the number of administrators on your machines** using virtual accounts or group-managed service accounts for privileged actions
- **Limit what users can do** by specifying which cmdlets, functions, and external commands they can run
- **Better understand what your users are doing** with transcripts and logs that show you exactly which commands a user executed during their session

The ability to identify abuse or misuse of privileged access also means we can detect various attack activity, from initial exploitation to privilege escalation to lateral movement.

[1] <https://docs.microsoft.com/en-us/powershell/scripting/learn/remoting/jea/overview?view=powershell-7.1>

Secure Configurations

- Can be challenging to implement
- May require multiple baseline configurations based on user group, system function, location, etc.
- Requires:
 - Secure images kept under version control
 - Configuration management tools
 - Configuration monitoring – in the SOC, we can use FIM, continuous scanning, application control

Secure Configurations

Developing configuration settings with good security properties is a complex task beyond the ability of individual users, requiring analysis of potentially hundreds or thousands of options in order to make good choices (the Procedures and Tools section below provides resources for secure configurations). Even if a strong initial configuration is developed and installed, it must be continually managed to avoid security “decay” as software is updated or patched, new security vulnerabilities are reported, and configurations are “tweaked” to allow the installation of new software or support new operational requirements. If not, attackers will find opportunities to exploit both network accessible services and client software.

For a complex enterprise, the establishment of a single security baseline configuration (for example, a single installation image for all workstations across the entire enterprise) is sometimes not practical or deemed unacceptable. It is likely that you will need to support different standardized images, based on the proper hardening to address risks and needed functionality of the intended deployment. For example, a web server in the demilitarized zone (DMZ) versus an email or other application server in the internal network. The number of variations should be kept to a minimum in order to better understand and manage the security properties of each, but organizations then must be prepared to manage multiple baselines.

High-Value Assets

- Identify critically important
 - Servers
 - Applications
 - Users / Accounts
 - Data
- And for those items, consider incident response tasks
 - Identification - Data availability and analysis requirements
 - Containment– Data isolation, asset, application, or user containment capabilities
 - Recovery - Are there backups? Wipe and rebuild difficulty?

High-Value Asset IR Planning

Let's revisit our previously created list of high-value assets and expand it a bit to include the applications on those system, the users that access those applications, and the data involved. Have you considered all the items that would be on this larger list, and what you would need to do to contain and remediate an incident involving them? Some questions:

- Is data collected, available, and usable for these critical systems that will help incident responders positively identify intrusions in a timely manner?
- For each of the items on this list, you should pre-consider what you will do if that system or application must go offline, if data access must be severed, or if a critical user or service account must be isolated, what will be done in the meantime? Is there a backup system that can provide coverage while the primary is down?
- Are there manual fallbacks for any business-critical processes that may be stopped as a result of an incident?
- If a system does become infected and cannot be trusted, are there working, *tested*, and current backups that can be utilized for a quick recovery?

Again, the idea is mapping out each business-critical application and considering the most high-risk and likely cyber incident scenarios and thinking through them before anything happens to ensure that a plan is in place that can be swiftly enacted in the case of emergency.

Incident Response Procedures

Items to include in your procedure:

- Technical processes
- Checklists
- Forms & Templates
- Roles and responsibility assignment
- Flow charts / key decision guidance
- Communication directions



Incident Response Procedures

Each section of your incident response procedure will be significantly different, but the types of data inside them is the same as any procedure. Be sure to include any specific technical procedures to follow, checklists used for guidance, forms and templates that should be filled out, and any other standard actions. The procedure should also explain at what point, and how, roles and responsibilities will be delegated to the incident responders when the time comes to declare a major incident (minor incidents usually do not need specific roles). Flow charts for actions (when to invoke disaster recovery or BCP, etc.) as well as guidance for key decisions may be included as well to standardize the SOC's reaction at key points throughout the incident (when to involve law enforcement, etc.).

Cloud Incident Response Preparation

Are you ready for cloud incident response?

Data to collect for CSPs, services and instances:

- Cloud management plane logs
- Cloud network traffic (Flow logs, traffic mirroring)
- Host logs (IaaS)
- Application logs (SaaS, PaaS, IaaS)



Goals for cloud IR data collection

- Audit activity on the cloud service itself
- Audit network, host, and service activity of cloud assets

Cloud Incident Response Preparation

Prepping and skilling up for cloud-based incident response is a very different game than physical, in-person IR. Since you won't be dealing with any physical assets, cloud-centric IR is more about ensuring that all IR-relevant data is being recorded *before* an incident occurs, and making sure your team know what that data is, where it is, and how to read it.

Cloud-based incident response can be conceptually broken down into a couple of different areas:

- IR for the **cloud platform** itself – what happens if an attacker gains access to one of the accounts in AWS and has access to consoles, GUIs, or other management plane features?
- Cloud **network** traffic – Will you know which cloud assets talked to what?
- Cloud **host** activity – When in control of the whole host (IaaS), do you have the operating system level logs to see what occurred (in other words, are you monitoring your cloud instances like you monitor your on-prem servers and services?) Don't forget this also includes edge cases that don't fall neatly into these categories, like containers.
- Cloud **application** activity – For the platforms that are higher-level (PaaS, SaaS) are the logs being collected, and quickly accessible to the team?

Software Incident Response Preparation

Participate in requirements engineering

Add detail to attack trees/threat modeling

Advise on secure network and system configurations

Provide threat awareness "training"

- ❑ Keep development teams aware of evolving threats

Close the feedback loop post-deployment

- ❑ Provide feedback to development teams as part of recovery when applications are involved

Software Incident Response Preparation

The incident response team (whether or not it is contained within the SOC, or the SOC is simply a participant/contributor) has a unique understanding of attackers, motives, targets, and techniques actually observed within the environment. This insight is invaluable to the requirements engineering phase of the secure development lifecycle. If your organization adheres to any kind of SDLC, the security team should be involved in security requirements elicitation to provide different views and probabilities for what type of attacks might be realistically executed. This interaction should be bi-directional, where the security team can get more detailed insight on application components and architecture to add to their own attack trees/threat models. While the SOC may not be able to provide much input on secure development best practices, it can absolutely advise on secure network and system configurations for the development and production environments and provide some level of training on general security best practices. Finally, incident response preparation where software development is concerned doesn't stop at deployment. We'll talk about the recovery and lessons learned stages on IR in a few minutes, but for now keep in mind that giving feedback to application owners and development teams is an important part of the incident response process where those applications may have been targeted or exploited. This feedback can serve as requirements in future iterations of those applications.

Writing Playbooks

- Guidance for responding to various scenarios in accordance with the *Incident Response Plan*.
- Goal is to be concise, relevant, and flexible
- A well-written playbook can reduce panic during the early stages of a response
- Playbooks need not be technical
- Refer to incident history to drive playbook requirements



Writing Playbooks

Playbooks are step-by-step guidelines for responding to various incident scenarios in accordance with our *Incident Response Plan*. In the preparation stage, it can be tempting to write lengthy, detailed process descriptions and workflows and playbooks to communicate readiness and maturity. But don't over-engineer your incident response process. The goal in writing playbooks (sometimes referred to as "runbooks") is to be as concise, relevant, and flexible as possible to be of use in a real response scenario. Remember that process often goes out the window when we find ourselves in a crisis and focus can be difficult; strive to answer key questions like "who do I call first?" and "what information do I need to gather?" and "who can help?" as initial steps.

Playbooks do not always need to be technical. Conducting a risk analysis, managing a potentially volatile termination, and dealing with account hijacking are all possible scenarios that may require a playbook but don't necessarily involve technical tasks. If your playbooks are technical in nature, write them as simply as possible and include references and screenshots. Most importantly, review your playbooks periodically for relevance and timeliness. If you're unsure of which incident types require playbooks, refer to historical incident reporting to see which threats are the most prevalent and which processes the team is likely to have to execute again in the future.

In the next few slides, we'll look at reference models you can use for writing technical response playbooks.

RE&CT Framework

RE&CT framework:

- Like MITRE ATT&CK for defensive tactics
- Broken into "response actions" (similar to tactics) with a unique ID

The diagram illustrates the RE&CT framework. On the left, a response action card is shown with the following details:

Title	Block external URL
ID	RA3105
Description	Block an external URL from being accessed by corporate assets
Author	@atc_project
Creation Date	2019/01/31
Category	Network
Stage	RS0003: Containment

A solid arrow points from the 'ID' field to the 'RA3105' entry in the matrix below. A dashed arrow points from the 'Title' field to the first row of the matrix.

The matrix on the right maps response actions to incident response stages:

Containment	Eradication	Recovery	Lessons Learned
Patch vulnerability*	Report incident to external companies	Reinstall host from golden image*	Develop incident report
Block external IP address	Remove rogue network device*	Restore data from backup*	Conduct lessons learned exercise
Block internal IP address	Delete email message	Unblock blocked IP	
Block external domain	Remove file*	Unblock blocked domain	
Block internal domain	Remove registry key*	Unblock blocked URL	
Block external URL	Remove service*	Unblock blocked port*	

SANS

MGT551 | Building and Leading Security Operations Centers

29

RE&CT for Playbook Creation

When it comes to creating use cases and playbooks, there's a new project out there called RE&CT by Timur Zinniatullin Daniil Svetlov, Andreas Hunkeler, and Patrick Abraham that you should know about. RE&CT was created in the model of MITRE ATT&CK as a similar set of techniques and tactics, but instead of being focused on attacker action, it is a list of defender-focused incident response tactics called "Response Actions".

Response actions are listed across the columns by incident response cycle ("PICERL") stage and contain actions for the stage they're under like MITRE ATT&CK techniques each have tactics underneath them. Each individual response action has a unique ID, such as RA3105 on the slide above. In RA3105, the first and second digit have special meaning. The first digit classifies the response action as part of a specific incident response stage (3 means Containment), the 2nd digit classifies a category it belongs to (1 means "Network" category in the case of RA3105). Each response action can be clicked on for additional detail and plotted on the Navigator web application, just like the ATT&CK framework. While RE&CT seems to be a newer effort, it's already in a very useful state for idea generation at a minimum, but for those who would like to use it further, it also comes with Case Templates for TheHive and a premade Confluence knowledge base.

[1] <https://atc-project.github.io/atc-react/>

RE&CT-Based Playbook Creation

RE&CT is amazing for playbook creation!

- Not sure where to start, don't want to forget any good options? RE&CT has you covered!
- Instructions:
 1. Consider which situations you need a playbook for
 2. Gather all applicable potential response actions for each playbook
 3. Filter to the best and most useful actions (analyst input needed)
 4. Consider marking each step as required/optional
 5. Enter those items into your IMS for testing and usage!

SANS

MGT551 | Building and Leading Security Operations Centers

30

RE&CT-Based Playbook Idea Generation

If your team is looking for inspiration for steps that can be taken as part of your playbooks, the ATC RE&CT project is an outstanding place to start.¹ Let's say you want to make a playbook of steps for positively identified command and control traffic emanating from an infected host to the internet. Look at the Containment category, since you're already past the preparation and identification stage, there a multitude of potential options. From a brief look at the list, here are some response actions you might consider including in your Containment playbook steps.

- Block external domain
- Block external IP address
- Block port external communication
- Block external URL
- Block data transferring by content pattern
- Block process by executable content pattern
- ... (and many more)

As you can see, although the ideas themselves are fairly standard the value in RE&CT is that it is an exhaustive, pre-thought-out of list of potential actions. One that can be easily leveraged for creation of playbook steps to ensure analysts don't overlook anything in the heat of the moment. If potentially appropriate responses are considered ahead of time for a standard type of situation, then you can be reasonably sure that any response to that situation will be largely thorough and complete, regardless of who responds to it.

[1] <https://atc-project.github.io/react-navigator/>

RE&CT-Based Assessments of Capabilities

- Assess your SOC on Navigator web app for RE&CT!

Preparation	Identification	Containment	Eradication	Recovery
101 items	61 items	26 items	8 items	14 items
Access external DNS logs	Analyse email address	Block domain on email	Delete email message	Restore data from backup
Access external HTTP logs	Analyse file hash	Block external domain	Remove file	Restore quarantined email message
Access external packet capture data	Analyse filename	Block external IP address	Remove registry key	Restore quarantined file
Access internal HTTP logs	Analyse IP	Block external URL	Remove rogue network device	Enable disabled service
Access internal network flow logs	Analyse jar	Block internal domain	Remove service	Reinstall host from golden image
Access DHCP logs	Analyse macos macho	Block internal IP address	Remove user account	Unblock blocked domain
Access external network flow logs	Analyse MS office file	Block internal URL	Report incident to external companies	Unblock blocked IP
Access internal DNS logs	Analyse PDF file	Block data transferring by content pattern	Revoke authentication credentials	Unblock blocked port
	Analyse registry key	Block port external communication		Unblock blocked process
	Analyse domain name			

SANS

MGT551 | Building and Leading Security Operations Centers

31

RE&CT Based Assessment of SOC Capabilities

Many SOC teams have harnessed the power of the organized data in the MITRE ATT&CK Matrix to evaluate their detection capabilities against adversary activities. In a similar way, the ATC RE&CT matrix, combined with tracking in the Navigator application, can assist in evaluation of incident response action capabilities.

By walking through each of the incident response stages and marking the items which your SOC needs to have capabilities to perform, each response action can be given a score and saved on a layer of SOC capabilities, the same way you might want to do it for ATT&CK Tactics and techniques. This gives you a way of granularly verifying that your SOC has the atomic actions it needs to succeed in a fast response against attacker action. You could even build a custom wiki (or use the pre-built Confluence page provided by RE&CT) and attach the procedures for how to perform each response action so that any analyst, no matter the experience level, can accomplish that action. A great way of fighting the "tribal knowledge" problem in a SOC!

Reference: <https://atc-project.github.io/react-navigator/>

IR Communications

- Effective IR requires technical *and* social processes
- Communicating effectively in an incident response can be a major challenge
- Plan IR comms to ensure the team knows:
 - Who gets what information
 - How much detail to share
 - Ways to validate info for accuracy and completeness
 - Communications methods
 - Responsible parties for providing info
 - What kinds of ongoing updates are necessary in what intervals

IR Communications and Collaboration

Communicating during an incident, especially if there are multiple teams or individuals in multiple locations support the response effort, can be one of the most challenging parts of the response. As a SOC Lead or Manager, part of your job is ensuring everyone is moving in the right direction, understands what has been done and needs to be done, and that all stakeholders are informed every step of the way. In larger organizations, you may also act as “traffic cop” to shield the team from constant interruption and distraction so that they can do their work while capturing important information that needs to make its way out of the SOC to key individuals.

In a 2016 study [1] conducted by George Mason University, the US Department of Homeland Security, and others, researchers found that incident response is “an intense social process” that requires integration of both technical and social processes. Furthermore, the study found that the failure to share information, successfully collaborate, poor listening, and lack of communication are all social failings that impair the effectiveness of incident response teams. Establishing communications charters, plans, and/or protocols during an incident can help teams minimize these failures and execute a more effective response. These process artifacts provide useful guidance on:

- Recipients for key information and ongoing updates
- How much information and how much detail should be shared at each phase of the response
- Mechanisms by which posted or shared information is checked for accuracy and completeness
- Communications methods that will be used during the response
- Individuals or teams responsible for providing information
- What kinds of ongoing updates are necessary and on what interval (usually dependent upon the criticality of the incident)

[1] https://www.incidentresponse.com/wp-content/uploads/GMU-Cybersecurity-Incident-Response-Team_social_maturity_handbook-updated_10.20.16.pdf

Keys of Good IR Communications (1)

Focus Areas

- Status updates
- Coordination
- Protect revenue/value generation
- Safeguard reputation and brand perception
- *Remember: incident response is not crisis management*
 - Good practice to have separate channels for crisis communications and incident response communications
 - Multiple conference bridges, e-mail distributions

Keys of Good IR Communications (1)

For the SOC, focus areas of good incident response communications are status updates, coordination, value preservation, and reputation. Communications out of the SOC destined for management should address one or more of these things; getting into too much technical detail that does not speak to any of these items is a good way to get sidelined or create more questions that may be difficult to answer. Remember: be brief but complete and avoid technical jargon. Putting your updates into a larger crisis management context is a good way to maintain the right mindset when it comes to communicating with the rest of the organization during an incident. Finally, remember that incident response is not crisis management

Keys of Good IR Communications (2)

1. Be clear

- Investigations are difficult to investigate, tougher to explain
- Stay on 5th grade reading level
- Avoid attribution and jargon

2. Be timely

3. Take responsibility

- Don't name-drop vendors
- Be human

4. Make comms part of your training and exercises



Keys of Good IR Communications (2)

The keys of good IR communications are to be clear, be timely, and take responsibility. In an information vacuum, an organization's leadership and users/customers/constituents will make assumptions, which can be dangerous in an incident. Only the security team can provide tactical updates based on hard data, and those updates must be easily understood by a non-technical audience. Some SOCs designate a scribe or incident coordinator to distribute non-technical updates based on technical information. Finally, it can be tempting to name-drop vendors either to scapegoat or show progress, but remember to take responsibility for both positive and negative developments. Executives normally aren't interested in your experiences with the products or services you're using or confusing technical jargon.

Leading IR Communications

- Bring order to chaos
- Ensure that the incident is being managed appropriately
- Set up communication channels and steer people towards these channels
 - These channels are the “staging area outside of the burning building”
 - Guard the productivity of these channels ruthlessly
- Collect status information & proposed solutions from SMEs
- Delegate necessary actions required to help bring an incident to closure

Leading IR Communications

As a leader and/or manager, it's your job to bring order to chaos. From a communications perspective, particularly if you are acting in a lead role for the response, this means ensuring response procedures are followed and the incident is being managed appropriately. Delegate technical tasks and collect status updates and other inputs from subject matter experts and other stakeholders in the response. Set up communication channels and steer participants to those channels. But remember, these channels are like the staging area outside of a burning building where firefighters plan their response, triage injured, and perform much of their work- not a place for idle discussion or endless debate! Guard the productivity of these channels as ruthlessly as you can, even if it means shutting down discussion or asking participants to drop or move to another venue.

Preparation Summary

- Strive for prevention, ensure detection
- Defensible network architecture, CIS controls are great references for hardening the environment
- Know where your high-value assets are
- Playbooks can be derived from your incident response plan, historical incident trends
 - Playbook goal: Repeatability without too much restriction



Preparation Summary

In this section, we discussed the need to strive for prevention but design for detection. There are many great reference models we can use for these activities, including the CIS controls and Defensible Network Architecture. Above all, know where your high-value assets (devices and users!) are and how they will be covered by your monitoring and preventative measures. Planning is also the stage where you want to design your response playbooks. Historical incident response efforts can inform your playbooks by codifying what works and doesn't and what steps best align to your organization, culture, and priorities. Barring historical reference, we can also look to the RE&CT framework to guide response actions based on best practices. Whatever our approach for developing incident response playbooks, we want structure and repeatability without restrictive, brittle procedures that can hinder an effective response. We'll talk more about testing your incident response and continuously improving; you should plan for those activities to drive additional refinements and adjustments to your playbooks over time. Next, we'll get more tactically-focused by looking at incident identification and containment.

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

Planning, Preparation, and Review

- Incident Response Planning and Preparation

• Investigation

- *Exercise 4.1 – Investigation Quality Review*

IR Execution

- Identification, Containment, and Eradication

- Incident Response in the Cloud

• IR Tools

- *Exercise 4.2 – Planning Responses with RE&CT*

- Crisis Management and Continuous Improvement

- *Exercise 4.3 – Designing Tabletop Exercises*

- Recovery and Post-Incident

- Summary and Cyber42 – Day 4



This page intentionally left blank.

In This Module

- Investigation tactics and techniques
 - Structured analytical techniques
 - Brainstorming and ACH, application for alerts
 - Gaining clarity in analysis tasks through task decomposition
- Investigation quality review
 - How to do it
 - Structured analytical techniques
 - Documentation

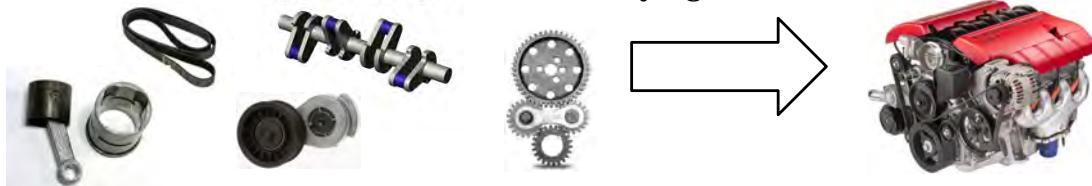
In This Module

This module is all about the investigation process. We will start by covering analyst-specific topics like structured analysis techniques, accounting for bias, and breaking down complex analytic tasks using tools like decomposition and externalization. We'll also discuss how to approach checking that investigations of potential incidents are being done and continue to be executed at a high level of quality.

Investigation Tactics

Investigations involve many bits of information

- Human short-term memory is limited to ~ 7 items
- Therefore, **decomposition** and **externalization** techniques should be used to overcome limitations
- **Externalization:** Getting info out of your head into a visible form
- **Decomposition:** Breaking down a problem into component parts
 - Understanding smaller pieces before trying to understand the whole



SANS

MGT551 | Building and Leading Security Operations Centers

39

Investigation Tactics

During the investigation stage, analysts must thoroughly document the details of the situation and the evidence used in analysis. This can be challenging, though, while trying to keep track of hostnames, IP addresses, usernames, and more, short-term memory will be quickly overwhelmed as the average person can only keep roughly 7 items in their head at once. Since these situations are often complex and difficult to understand, tactics such as decomposition and externalization should be used to assist. Decomposition is the breaking down of a complex problem into more fundamental parts that can be individually understood before assembling them into a whole. Externalization is taking the pieces of data and getting them out of your head and writing them down in physical or digital form. These tactics recommend by Heuer for analysis, help clarify complex situations by making it easy to see how the pieces interact and relate with each other, and break issues down into more easily digestible parts.

Decomposition Techniques for Investigation

Analysts should be **deliberate** and **clear** on actions to take during an investigation

- Use **careful consideration** of questions to answer, data needed
- Should *not* chase immediate intuition of what to do, this often leads to wasted time

Breaking down an investigation - "The Alexiou Principle¹":

1. What question are you trying to answer? (Definition)
2. What data do you need to answer that question? (Selection)
3. How do you extract that data? (Acquisition)
4. What does that data tell you? (Interpretation)

Decomposition Techniques for Investigation

Analysts should also be decomposing investigations during the process to help them assemble the lists of questions they want to answer and data that will be required to answer those questions. One way to do this, called "The Alexiou Principle",¹ suggests that analysts should consider breaking down the big question of what happened in any investigation into a smaller set of more easily digestible questions. If analysts immediately jump into action chasing the first idea they intuitively had, it's very easy to start down the wrong path and waste a lot of time.

The goal of this set of questions is to break the bigger investigation task into "atomic" questions that, if you can answer them, will lead to the conclusion of the larger question they make up. The Alexiou Principle suggests analysts remember the following set of questions to guide them in their day-to-day work. The information in parenthesis was added by the course author to further clarify the steps.

1. What question are you trying to answer? (Definition of the question you're trying to find an answer to in a clear and concise manner)
2. What data do you need to answer that question? (Selection of the appropriate data available to you for answering that question)
3. How do you extract that data? (Acquisition and collection of the data from the systems that store it)?
4. What does that data tell you? (Interpretation of the acquired data to answer the question)

Posting these questions on a wall or somewhere easily accessible in the SOC can help analysts stay on the right track when formulating their plan of attack.

[1] <https://thedigitalstandard.blogspot.com/2009/06/alexiou-principle.html>

Decomposition Techniques for Triage and Analysis

Decomposition of an incident also makes the analysis easier

- Breaks down key pieces of information in documentation
- Use attack models or framework
 - Kill Chain: Write up what happened for each stage
 - Diamond model style: Victim, Adversary, Infrastructure, Capability
 - ATT&CK: Break out each tactic and technique
- Incident Management tools should include these details in writeup

Analysis Writeup Example:

- **Recon:** Targeted due to data breach
- **Delivery:** Invoice-themed phishing email with attached file
- **Exploit:** Word macro, script downloads file from Dropbox
- **Install:** Drops file in user's temp folder, installs service for persistence
- **Cmd. & Ctrl:** HTTPS communication to maliciousc2.com



Decomposition Techniques for Triage and Analysis

Decomposition is best used during analysis to break down the series of steps that have occurred so far and align them to a well-known mental model—the kill chain for example. In the investigation stage, analysts must figure out what happened, and attempt to predict what *might* happen if the attack were to progress. Decomposing their notes into these items using ATT&CK Tactics, the Kill Chain, or the Diamond model can give a frame within which analysts can write up their findings.

For example, an analyst may work an alert where malware has been found on a laptop and decompose their findings into kill chain stages. After working backward through the kill chain from the infection and identifying the details of what happened and using the malware sandbox to simulate future predicted actions, the writeup could be broken into pieces such as the following:

- **Recon:** This user seems to have been targeted due to the inclusion of their email address in the recent data leak at company X.
- **Delivery:** The user received a phishing email masquerading as an invoice with attached file "invoice123.doc".
- **Exploitation:** The user opened the document and enabled an auto-running macro inside which reached out to the internet and downloaded malware from a Dropbox public share.
- **Installation:** Malware XYZ was found on the laptop running from the C:\Users\victim\AppData\local\temp folder.
- **Command and Control:** The malware sandbox uncovered that this malware communicates with server maliciousc2.com using HTTPS protocol.

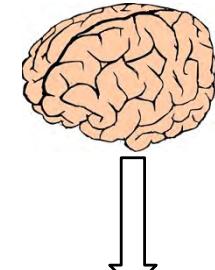
(Note that in a real investigation this would be much more detailed)

Decomposing the incident like this makes it easy to understand exactly what happened for each piece of the attack and would make a quick and simple way to explain what happened in an incident report. Presenting the data like this is much more organized and easier to understand than one giant paragraph purely describing everything that happened.

Externalization

Get things out of your brain only

- Helps understand and see connections
- Physical
 - Notebooks for analyst personal notes
 - Whiteboard walls for collaboration
- Digital
 - For both drawing and sharing
 - Threat intel platform events
 - OneNote, Notion.so, Evernote, etc.
 - Maltego, Visio, etc.



Externalization

Externalization is the other important tactic for overcoming the complexity of the average incident. As the saying goes for analysts, "If you're moving too fast to write things down, you're moving too fast." Analysts should be encouraged to remember this.

Externalization is the process of writing notes *as you work* so that the process could be followed again and what was found is documented in either a text-based list, visual, or other way, so that nothing is forgotten. You should have a variety of options available for analysts for externalization, everything from notebooks and whiteboards for physical writing and collaboration to digital options such as OneNote, Visio, Maltego, or any other organization and visual representation tool. Maltego, in particular (a commercial tool), can be great for writing up incident reports as support for the various entity types built in, and transforms make enrichment of those entities fast and easy so that investigations can be performed directly within the tool.

Investigation and Mindset

- Analysts evaluate and assimilate info through their mental models or "**mindset**"
- Mindsets...
 - Are a product of someone's assumptions and experiences
 - Can cause you to overlook, reject, or missing info
 - **Strongly influence what is and is not perceived**
 - Are further cemented through experience
- Experienced analysts more susceptible to errors due to past success with a given mindset

Investigation and Mindset

One key item for analysts to be aware of when approach any investigation is the fact that their mindset will affect what they see and perceive within the data they are given. Everyone filters everything they see through a set of mental models built from their past experience (or lack thereof), assumptions, and more. These mindsets are what will lead you to either immediately understand and contextualize an attempted attack, or perhaps not understand what you're looking at. In general, building mental models and experience is good, and leads to faster recognition of interesting details in data, the downside is that same built up experience can cause some problems too. When an analyst has seen the same thing for years, they may be quicker to assign an incorrect attribution based on something they've seen in the past, or miss critical data as they do not *expect* to see it (discussed further in later slides). Over the next few slides, we'll discuss some of these common investigation errors as well as the process and structured analytical techniques that can be used to mitigate the risk of these errors, while still keeping the benefits of experience.

One Major Investigation Issues

Inside many analyst's brain:

1. Pick alert and view provided information
2. Develop a theory of what happened
3. Collect evidence to support that theory
4. Move quickly to next alert



Problem

Issues:

- Conclusion developed during incremental data gathering
- Seeking to prove a theory instead of disprove
- No evaluation of other theories compatible with this evidence

One Major Investigation Issues

For most analysts, the investigation process seems intuitive and straightforward—you pick an alert, you look at the data, you decide what it seems like happened and go collect the evidence to support it. You move through, make your determination, and move on to the next item. Unfortunately, this mindset can lead to looking over what truly happened. Why? There are several reasons, the chief one being that process is fueled with confirmation bias.

Step 3 on the page above is the main problem, most analysts will be tempted to go with the theory that the alert most obviously appears to be and may ignore exploring other possibilities. On top of that, the theory you first come up with when seeing the alert is the one your brain is likely to stick with, even if evidence gathered later in the investigation may lead you away from that conclusion. This is not a conscious effort by the analyst to sabotage work, it's just how the human brain works. Once we have one image in our mind from the initial set of data, it's very hard to shake it and any new evidence added after that point tends to get assimilated into the view we already want to believe. What we should be doing is collecting that evidence and looking at everything in totality with “fresh eyes”, unfortunately, this is very hard for people to do.

The other problem is that there is commonly no attempt to *disprove* the theory you first come up with, only to support it. This isn't how investigation works. No amount of confirming evidence can ever *prove* something to be true, only make it less likely false. *Disconfirming* evidence, however, can quickly rule something out, which is why analysts should seek it when trying to figure out what happened.

The Dangers of Experience During Triage

One of the biggest analysis threats: **Confirmation bias**

- Deciding the likely answer, then going to prove it
- Most prevalent in triage and investigation stage
 - Making a call based on limited evidence
 - Analysts start with alerts they've seen hundreds of times before
- Prevention is difficult
 - More experience == more likely it will affect you
 - Repetitive experiences create mental ruts, inhibit creativity
 - Avoiding it is not intuitive

The Dangers of Experience During Triage

One of the biggest threats to accurate triage and analysis is confirmation bias—coming to a pre-made conclusion, then only looking for supporting evidence for that conclusion. Confirmation bias in the SOC is highly likely to show up in the triage and investigation stage because analysts are trying to make a guess about what has and will happen based on limited information. Given that many alerts are repetitive across time, it's easy to assume that the 100th time you've seen an alert, the activity it identified is the same as the first 99 times, but that is not necessarily true of course. What is true, however, is the more times you've seen a given situation, the more likely you are to assume that situation in the future. This occurs because the pathways in our brain for recognizing a given pattern are strengthened every time it is encountered, making those paths more likely to be used in the future—this is often referred to as a “mental rut”. What this means is that experienced analysts, although fastest and most *likely* to be accurate in their job, are also at a higher risk of this type of error due to confirmation bias. These mental ruts can cause them to say, “I've seen this 1000 times before” and go down the wrong investigation path merely because the investigation started with info they're familiar with.

Other Common Biases and Problems

- **Perception Bias**

- We tend to perceive what we *expect* to perceive (and may miss obvious errors)

- **Probability Estimation Bias**

- Probability estimates are affected by how easily you can imagine a scenario

- **Evidence Evaluation Bias**

- **Missing info** – It's hard to judge how to weight what *isn't* there, but perhaps should be
- **Discredited evidence** - After something is proven wrong, our thinking about it may not quickly change

- **Biases in Perceiving Causality**

- Rejection of randomness, accident and errors

Other Common Biases

No matter the type of analysis being performed, threat intelligence, SOC alert, or otherwise, there are some other common biases that often occur that can skew results that you and your analysts should also be aware of.¹

- **Perception Bias** – People most easily perceive what they expect to perceive, leaving otherwise glaringly obvious errors unspotted. While it seems like this wouldn't happen, it is a well-documented effect. If you don't believe it, search YouTube for the videos called the "Selective attention test" or "The Monkey Business Illusion". These are great generalized examples of this phenomenon.
- **Probability Estimation Bias** – The probability assigned by an investigator of something happening has been shown to be highly affected by how easily someone can imagine a given scenario or recall previous similar instances (rather than the actual likelihood). It also can suffer from anchoring to previous estimations of likelihood in a given scenario, moving only slightly (when it should be more drastically adjusted) as new information arrives. Finally, overconfidence is also commonly cited as a potential probability estimation issue, especially with those who have a wealth of experience.
- **Evidence Evaluation Bias** – In an investigation, and especially in relation to investigation of intrusions using multiple data types (logs, PCAP, etc.) it can be very difficult to properly weigh the impact of missing evidence, even if you're aware of the lack of expected information.
- **Perceiving Causality** – While it may seem to an analyst that all events that occurred were well thought out actions taken by an attacker with a plan, the reality is there may be randomness, accidents and errors in the data as well, making it more confusing. If we assume attackers are executing perfectly, and that all data is relevant, it's potentially easier to get confused.

Additional info on these biases can be found in the fantastic free document cited below in the footnote called "A Tradecraft Primer: Structured Analytic Techniques for Improving Intelligence Analysis".

[1] <https://www.stat.berkeley.edu/~aldous/157/Papers/Tradecraft%20Primer-apr09.pdf>

Avoiding Confirmation Bias

Threat Intelligence teams have the same issue

- They make predictions on past/future events with a lack of data
- They lean on **structured analysis techniques** to compensate
- SOC analysts can do the same!

Key tactics to avoid confirmation bias:

1. **Brainstorming** to force consideration of multiple options
2. **Evidence-focused analysis** reveals diagnosticity of each piece of info
3. Focuses on **hypothesis refutation**
 - **Analysis of Competing Hypotheses** is one way to do this

Avoiding Confirmation Bias

If we look around for similar problems in other areas of study, we find that intelligence analysts (whether threat intelligence for cyber security or otherwise) have this same problem. The good news is there's a mature body of knowledge used for intelligence analysis that helps their analysts overcome this problem called structured analysis techniques. SOC analysts can follow the same process used by many intelligence analysts to help avoid bias and gain similar clarity in understanding a situation in which they have less than perfect. These techniques are discussed at length in the free book "Psychology of Intelligence Analysis"^[1] by Richards J. Heuer^[1] who developed and taught many of these techniques at the CIA for decades before releasing the methodology to the public.

The most famous of the methods developed by Heuer is the "Analysis of Competing Hypothesis" procedure or ACH. This method helps analysts avoid bias by enforcing the imagining of multiple possibilities, focusing on hypothesis refutation, and examining each item of evidence individually against each hypothesis (resulting in identification of the most important pieces of evidence). It's also an all-around great teaching tool that helps managers and analysts alike see the problems that occur when proceeding with analysis in the intuitive way. Over the next few pages, we'll dig into an example to show how it works.

[1] <https://www.cia.gov/static/9a5f1162fd0932c29bfed1c030edf4ae/Pyschology-of-Intelligence-Analysis.pdf>

Brainstorming with a Morphological Matrix

Key to success: Quantity leads to quality

- Morphological matrix – an easy way to generate ideas
- Dimensions, concepts, or factors at the top of each column
- Options listed below
- Iterate through columns to create possibilities

Actor	Delivery	Cmd. & Ctrl	Impact
APT	Email	HTTPS	Critical Service
Hacktivist	Web	SSH	Email Admin
Script Kiddie	USB	DNS	VIP Laptop

Brainstorming with a Morphological Matrix

A key initial step in structured analysis is brainstorming. Regardless of the method you use, it's important to consider alternatives to your key theory for avoiding confirmation bias. The key realization is that when it comes to brainstorming, according to Heuer, quantity leads to quality. This is because the hypothesis you seek may not be the one you initially think of, and it's not until you run through all the obvious options that the correct answer might show up.

One of the easiest ways to get the creative ideas flowing is using what is called a morphological matrix to generate permutations of situations that could occur. The process is simple and consists of making a table such as the one on the page above where dimensions or key parts of the thing you're trying to brainstorm are put in the columns at the top, and options for that concept are listed below. Once you have a list for each independent piece you'd like to vary (each column), combinations can be generated simply by mixing and matching a selection from each column.

ACH

1. Identify several **mutually exclusive** hypotheses to consider
2. Make a list of evidence for and against each hypothesis
3. Analyze the "**diagnosticity**" of the evidence and argument
4. Delete evidence and arguments that have no diagnostic value, refine options, reconsider hypothesis
5. Draw tentative conclusions, **try to disprove each hypothesis**
6. Analyze how sensitive your conclusion is to evidence items
7. Report conclusions and relative likelihood of each hypothesis, not just the most likely one

ACH

Here are the full steps of doing a complete Analysis of Competing Hypotheses.

1. The first step is to brainstorm several possible hypotheses. These should be mutually exclusive ideas such that if one is true, others must be false. It is best to come up with as many plausible options as possible, including a hypothesis with deception tactics, if such a thing is a possibility for the given situation.
2. List out all information relevant to evaluating each hypothesis. All evidence and assumptions should be included, including the absence of things that you would expect to see if a hypothesis were true. Assumptions can make a big difference in the judgment made, so they should be explicitly called out here so that others will know that it was included if the analysis is being reviewed in the future.
3. Place all hypotheses and evidence in a matrix (shown on the next slide) and run through each box to note whether evidence is consistent or inconsistent with each hypothesis. If the answer is "it depends", the subsequent judgment for that column will be dependent on that "it depends", and this fact should also be noted.
4. Refine the matrix to eliminate data items that have no diagnostic value and adjust hypotheses as needed. If two need to be merged, or one should be removed or added, this is the time to do so and rerun through each evidence item to update it for the changes.
5. At this point, tentative conclusions can be reached about which hypothesis is most likely based on which has the *least* amount of inconsistency scores for it. The ones with the most inconsistency scores are the least likely options. Remember these are not perfectly weighted rankings, so it will not be an exact science.
6. Analyze how the conclusions were reached. Are there any hypotheses that were ruled out based on a single item of evidence? How confident are you in that evidence? Is it an assumption? If so, it should be noted the conclusions are wholly dependent upon it.
7. Report conclusions on the likelihood of each hypothesis, not just the most likely one, especially if there are key assumptions or data the conclusion hinged on.

DigitalShadows WannaCry Attribution ACH Example¹

Evidence	Evidence Type	Credibility	Relevance	H1	H2	H3	H4
				-13.414	-1.414	-5.0	-3.0
ETERNALBLUE relatively easy to use	DS Assessment	High	Medium	N	N	N	N
Anti-analysis feature usable as kill-switch	Secondary reporting	High	High	I	C	N	N
Samples first appeared in Feb 2017	Primary	Medium	Medium	N	N	N	N
No evidence of phishing vector (untargeted spread)	Secondary reporting	High	High	I	C	I	C
No operator input needed for encryption	Secondary reporting	High	High	C	C	C	C
Victims who paid reportedly did not receive decryption keys	Primary	Medium	Medium	I	C	N	N
Only three BTC wallets produced due to race condition bug	Secondary reporting	High	High	I	C	I	I
Ransom demand 300	Primary	High	High	I	C	N	N

C	Evidence is consistent with hypothesis
I	Evidence is inconsistent with hypothesis
N	Evidence is neither consistent nor inconsistent with hypothesis

DigitalShadows WannaCry Attribution ACH Example

If you'd like to see a great example of ACH as applied to the WannaCry incident, both DigitalShadows and Pasquale Stirparo of the SANS Internet Storm Center created a matrix to evaluate the possibilities of who was behind the attack.^{1, 2}

The page above shows an excerpt from the DigitalShadows evaluation where they use the letters N, C, and I as indicated to track consistency to each different hypothesis.

The hypotheses they used were as follows:

"A sophisticated financially-motivated cybercriminal actor – H1

An unsophisticated financially-motivated cybercriminal actor – H2

A nation-state or state-affiliated actor conducting a disruptive operation – H3

A nation-state or state-affiliated actor aiming to discredit the National Security Agency (NSA) – H4"

Their conclusion was that given all the evidence they had, H2—"an unsophisticated financially-motivated cybercriminal actor"—came out on top with H4—"A nation state or state-affiliated actor aiming to discredit the NSA"—close behind. Although many had attributed the attack to the claimed North Korean-based "Lazarus" group, this led DigitalShadows to ultimately conclude, "*At the time of writing, however, we assessed there to be insufficient evidence to corroborate this claim of attribution to [Lazarus] group, and alternative hypotheses should be considered.*" Of course, these conclusions are based on a rational, but still semi-subjective, point-ranking system.

[1] <https://www.digitalshadows.com/blog-and-research/wannacry-an-analysis-of-competing-hypotheses-part-ii/>

[2]

<https://isc.sans.edu/forums/diary/Analysis+of+Competing+Hypotheses+WCry+and+Lazarus+ACH+part+2/22470/>

Integrating the Lessons of ACH Into Daily Work

- ACH is not necessary for *all* alerts
 - Lessons it teaches must still be understood by everyone
- The biggest issues to avoid:
 - Not **considering other hypotheses** that fit the evidence
 - Not **seeking to disprove** your own theory
 - Careful consideration of most useful **evidence sources**
- When reviewing analysis, ask
 - What other theories did you consider?
 - Which key evidence shows this is the best answer?
 - Did you make any assumptions?

Integrating the Lessons of ACH Into Daily Work

While the ACH process is very thorough, it also takes a lot of time and methodical thinking to get through it. The reality of the situation is that ACH probably won't be used for most investigations. That doesn't mean we can't learn from it, though, and integrate its lessons into everyday life. Where appropriate, instead of doing a full ACH analysis, analysts should at least note the theories they've considered, attempts to disprove their leading theory, and key evidence that their analysis depended upon when making an assessment. If we can take the main lessons of brainstorming multiple alternatives, looking for key evidence differences, and trying to disprove theories, the most common confirmation bias-led errors can be greatly reduced. Of course, when there is a high-stakes situation or a particularly tough investigation, ACH is a great tool that should be used in full format, which has the side benefit of leaving an audit trail of analysis and the evidence used to come to a conclusion.

For additional guidance on how ACH can be applied, see DigitalShadows whitepaper on "Applying the Analysis of Competing Hypotheses to the Cyber Domain".¹

[1] <https://resources.digitalshadows.com/whitepapers-and-reports/applying-the-analysis-of-competing-hypotheses-to-the-cyber-domain>

Keeping Consistent Quality

- How can we be sure we're hitting the analysis mark?
 - Alert queue and SLA pressure incentivizes **speed**
 - Triage and incident response require attention to **detail**
 - Are we being detailed enough? Too detailed?
 - What about our peers?
- **Solution:**
 - Periodic peer and self-review
 - Structured critique methods



SANS

MGT551 | Building and Leading Security Operations Centers

Keeping Consistent Quality

After triaging, investigating and closing alerts over the months and years, you might start to wonder how well you're doing. How can we get feedback on our capabilities and ensure we are first hitting the standard that we should be hitting for investigation quality in the first place and keeping it up over time? As analysts start to see the same situations repeatedly, they more likely to develop an intuitive sense for what they think happened in a situation – which hopefully is correct but may not be. While SOCs usually incentivize speed, investigations also require attention to detail to ensure you aren't only partially removing attackers from the environment. Given these opposing pressures, how can we check ourselves and perhaps compare ourselves to our peers? One solid solution is periodic feedback provided through peer or self-review of your past investigations' structured critique methods.

Premortem Analysis

- **Goal:** Analyze potential failure *before* it occurs
- **Method:** Imagine yourself (or group) in the future learning you were wrong, explain how and why
 - Forces reframe to break mindset
 - **Legitimizes dissent** and group desire for consensus
 - Reduces risk of surprise, and need for post-mortem
 - Use for **conclusion testing, planning, or future prediction**
 - Can be used to **demonstrate overconfidence** in a plan
 - Once people are forced to assume error, making failure modes of the purposed plan of action highlight overconfidence

Premortem Analysis

One structured critique technique that can be used to test the strength of a proposed plan of action or analytical conclusions is the premortem analysis.¹ The goal of this technique is to analyze possible methods of failure *before* they occur to reduce the risk of future surprise, and the need to do a postmortem analysis because something *has* gone wrong. The method for this technique is to either by yourself, or in a group, have a meeting where you imagine yourselves at some point in the future learning that your conclusions or plan of action has gone wrong. Your job is to produce *all* ideas of how and why that occurred, typically in a round-robin everyone speaks type of fashion. Doing this forces everyone to reframe the situation in their mind and breaks you out of mental ruts and groupthink. The expected outcome of this method is a more thorough understanding of the uncertainty of the situation as well as the identification of early warning signs that the plan is not going as anticipated.

This technique can be an outstanding way to break through the issue of groups desiring a fast consensus. Forcing everyone to take a dissenting opinion and suggest how failure could be possible legitimizes dissent that may have gone unspoken due to politics or other group dynamics and gives a voice to potential alternative viewpoints. It can also be used as an outstanding technique to highlight overconfidence in a plan. When people or groups produce a plan of action, they are often extremely confident (overly so) in how well it will likely work. Once you perform a premortem analysis, forcing the reframe and the requirement to list potential methods of failure often highlights potential shortcomings of the plan and allows the creators to take premeditate action to prevent or detect the failure conditions coming to light, allowing them to better control the situation. Note that although premortem analysis does identify that potential problems exist, it does not necessarily highlight which problem or explain how to fix it. The next method, the structured self-critique, can do a more complete job of this part.

[1] "Structured Analytic Techniques for Intelligence Analysis" Heuer & Pherson, 2015, pp.240-243

Structured Self-Critique

- **Goal:** Identify weaknesses in current analysis
- **Method:** Assume the role of an analysis critic, then answer questions from this point of view about potential issues
- **Topics to Discuss:** Uncertainties, analytic process used, critical assumptions, diagnostic or missing evidence, potential deception
 - After discussion, reconsider confidence levels and conclusions
 - Useful for triage and investigation review
 - Great as a follow-on to premortem analysis
 - Focuses in on specific analysis problems and how to fix them



Structured Self-Critique

Another great self-assessment method is structured self-critique. This method is a great follow-on to a premortem analysis because it is more focused on finding the specific problems that may have occurred. For this technique, the method is to have everyone put on their pretend "evil hat" and analyze a conclusion that has been reached by assuming a critical viewpoint and answering questions about the analysis. In this way, each person will be forced to pick apart the process, assumptions, evidence, and other aspects that led to the conclusion and come up with any reason for error.

Questions that should be asked should include things like:

- Were our key sources of evidence reliable?
- Was contradictory evidence ignored?
- Do we have an explanation for missing evidence?
- Were our key assumptions valid?
- Did we seriously generate and consider alternative hypotheses?
- Did the absence of information mislead us?
- Did deception go undetected?^[1]

This technique is like the Devil's Advocacy technique where a single member of the team is designated to play the dissenting role. Although this method can work, Heuer and Pherson say that when only one person is dissenting, the team tends to get more defensive and the technique becomes less productive than having everyone do it.

[1] "Structured Analytic Techniques for Intelligence Analysis" Heuer & Pherson, 2015, p. 244-247

Investigation Quality Review

How can managers verify these principles are followed during analysis? Case review!

Dimensions to consider:

- **Analysis quality:**
 - Do analysis notes show evidence of confirmation bias?
 - Did the assigned person consider all relevant evidence against potential hypothesis?
 - Were any assumptions or tenuous evidence noted as such?
- **Analytical completeness**
 - Were all observables documented correctly? Was all relevant data utilized?
 - Were all kill chain / attack cycle / ATT&CK techniques documented?
 - Was the ticket categorized with all relevant metrics, etc.?



Investigation Quality Review

How can we ensure that analysts are following the principles laid out in this section? One simple method is periodically sampling a randomly selected alert or IR case for quality review!

At a minimum, we are looking to verify that analysts are using a scientific mindset that drives them away from confirmation bias in their analysis conclusions, and also that they have correctly and filled out all necessary information in your information management system as they work. Based on this, review results can be broken into categories of analysis quality and analytical completeness. Analysis quality questions focus on whether the right data was pulled, used, and assessed in a thorough way, free of confirmation, and that all assumptions and conclusions that could be sensitive to future discoveries are documented. Analytical completeness evaluation centers more around the practical aspects of researching all components of the attack and getting them noted into the system to the degree at which your SOC has decided to document findings.

What we are looking to find in these reviews is that analysts are neither moving too fast or too slow. If moving too fast through cases you might expect to find evidence that analysts are documenting the bare minimum information, picking the mostly likely conclusion, finding evidence for it and moving on as quickly as possible to the next case. Doing this in the short term indeed helps them "work" through more cases, but ultimately means quality is low, which means potential missed attacks. Conversely, you also don't want analysts diving *too* deep and writing a book on each case or incident (unless time permits, and that level of documentation is warranted). The goal for a quality review should be to ensure that analysts are hitting whatever target they are given and are continuing to do so over the long term.

Operationalizing Quality Reviews

- Periodic review ensures analysis quality
 - Checklists, spreadsheets, or interactive process
 - Each analyst should get at least one review periodically
 - Identifies potential weak spots in technique or knowledge
 - Those in need of help can be paired with high-scoring analyst to learn
- Focus should be on feedback, not a stressful "analyst ranking"
 - Newer analysts will need more frequent review
 - Should focus on completeness of analysis and process
 - Experienced analysts will need more mindset challenging



Operationalizing Reviews

Given the structured self-critique method, how can we operationalize it so that it becomes part of the culture of the SOC? One way is to perform periodic reviews of a random sample case from each member of the SOC. This review can be once a week, month, or any period desired, but given that feedback is a key component of growth, it needs to happen at some interval. During this review, the case should be read through by a single analyst or group of other analysts from the SOC and notes should be taken about what was and was not done. The questions about analytical completeness and if all aspects of the attacker were found should be assessed as well as the analytic technique used. These reviews can be qualitative or point values can be assigned to each question to get a more objective measure, but if analysts hear back where they can improve from others, the objective should be met.

Aside from quality, there are other benefits of doing reviews. One is that newer analysts can see the technique and thinking process of the more experienced analysts and start to understand more quickly what tools and methods they should use in various situations. Another is that patterns of deficiencies in certain areas can be identified, and those with a need to learn a tool or technique can be paired with those who know it well for efficient on-the-job training.

One word of caution about peer review. The spirit of peer review should be kept light-hearted and purely focused on being a learning tool. If it becomes a stress-inducing "analyst ranking" system, the benefits of it may be overshadowed by the problems it causes. Ensure that all analysts know it is purely for their own learning and will not be used to punish them for anything they aren't yet exceeding at. To this end, you will probably find it beneficial to give newer analysts more frequent reviews than those who have been around longer, and the nature of those reviews will likely be different as well. Newer analyst reviews should focus on if their cases are analytically complete, the right tools were used, and the process they used to produce a conclusion is solid. For more experienced analysts who understand the available tools and analysis process, reviews may be more of a "red team analysis" type exercise that challenges their conclusions and mindset and makes sure they didn't jump to conclusions based on mental ruts.

Alert Triage and Investigation Summary

- Recognizing high-priority items is a must
 - Context and alert triage software make or break it
- Ensure analysts avoid confirmation bias
 - Generate and consider multiple hypotheses
 - Focus on the data that differentiates between them
- Investigation sampling and quality review keeps standards high

Alert Triage and Investigation Summary

In this module, we reviewed both theoretical and practical issues with the triage and investigation step. First analysts must be able to accurately determine which alert is the riskiest. This requires both the information to be available to them as well as the training to read and understand what they are seeing in the alert console. Your SIEM and SOAR will play the biggest part in providing the context while training and experience will build analysts confidence in alert choice.

Once an alert is chosen, analysts must proceed carefully trying to determine what has occurred and if it is a risk or not, without introducing confirmation bias into the equation. The best way to do this is to not rely solely on intuition to give you a single answer to try to prove. Consider multiple options that could be possible and use careful evidence selection to choose which of the valid options could be occurring. This requires analysts know that confirmation bias is a problem, how it can affect analysis, and how to avoid it. It also requires they are familiar with all of the data sources you have collected, and which is available to help with coming to a conclusion. Once evidence has been gathered, analysts should consider how they might *disprove* their hypotheses instead of prove them given that this route is much more efficient for the elimination of options. While working through this analysis, ACH and other structured analysis methods can be used to visualize and decompose the mountain of information that may be involved in a large case.

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

Planning, Preparation, and Review

- Incident Response Planning and Preparation
- Investigation
 - *Exercise 4.1 – Investigation Quality Review*
- IR Execution
 - Identification, Containment, and Eradication
 - Incident Response in the Cloud
 - IR Tools
 - *Exercise 4.2 – Planning Responses with RE&CT*
 - Crisis Management and Continuous Improvement
 - *Exercise 4.3 – Designing Tabletop Exercises*
 - Recovery and Post-Incident
 - Summary and Cyber42 – Day 4



This page intentionally left blank.

EXERCISE 4.1

Exercise 4.1: **Investigation Quality Review**

OBJECTIVES

- Understand how to measure analysis quality
- Develop a repeatable process for identifying errors in analysis
- Select a representative sample of cases to analyze
- Understand which errors can and cannot be eliminated based on their type and category



Exercise 4.1: Investigation Quality Review

Please go to Exercise 4.1 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

Planning, Preparation, and Review

- Incident Response Planning and Preparation

- Investigation

- *Exercise 4.1 – Investigation Quality Review*

IR Execution

• Identification, Containment, and Eradication

- Incident Response in the Cloud

- IR Tools

- *Exercise 4.2 – Planning Responses with RE&CT*

- Crisis Management and Continuous Improvement

- *Exercise 4.3 – Designing Tabletop Exercises*

- Recovery and Post-Incident

- Summary and Cyber42 – Day 4



This page intentionally left blank.

Strategies for Collaborative Problem Solving

- IR can be a disjointed, complex, group problem solving exercise
- Problem solving requires good collaboration
- Set your team up for success:
 - Use pre-briefings or planning sessions
 - Use counter-factual thinking
 - Provide team feedback



Strategies for Collaborative Problem Solving

Even if your SOC is wholly responsible for driving the incident response process, it's likely that your team members will have to work with other groups –and at the very least each other – in order to solve the complex problems and take whatever action the response requires. Conducting pre-briefings, debriefings, and providing focused and timely feedback will improve the team's ability work more effectively together and with others during the IR process. Let's look at each of these activity in more detail:

- **Pre-response briefings or planning sessions:** these sessions do not (and frankly *should not*) need to be lengthy, formal affairs- after all, we have an incident to contain! Focus on establishing a shared understanding of the problem and of the goal or desired outcome of the response effort.
- **Use counter-factual thinking:** We're going to talk about shared knowledge and unique expertise in a moment, but for now it's enough to say that team members don't always realize when they have information no one else has. As someone with a slightly more strategic viewpoint, you may be in a better position to bring up related scenarios or past situations to prompt individuals to share things they might not otherwise consider or evaluate possibilities they may not be looking at. More specifically, consider lessons learned, mistakes you want the team to avoid, and insights they might take from past response efforts that might trigger new details or awareness within the team.
- **Provide team feedback:** Feedback during the debrief or after-action review to highlight successes and failures will help your team understand how it performed during the response, and areas in which they need to improve. Take contemporaneous notes during the response so that you can refer to specific actions or events following recovery.

Combined with tabletop exercises and pre-planning IR communications, these strategies will help shore up any gaps you might have in the social aspects of incident response and ensure that team collaboration supports the process rather than dragging it down.

Improving SKUE

- We strive for diversity in the SOC: backgrounds, skill sets, experiences, specialties
- Team members need to know who knows what to engage various skillsets and collective abilities
- Shared knowledge of unique expertise (SKUE), or *transactive memory*
- Things you can do to improve SKUE: use a knowledge base for incident response activities, cross-train team members, implement job rotation and shadowing

Improving SKUE

Incident response teams require a diverse group of perspectives, skillsets, and backgrounds to support a dynamic security capability. Each team member brings their own unique experiences, specialties, and talents to the table to improve the larger team. This makes shared knowledge of unique expertise, or SKUE, critically important for the team to function effectively as a unit. SKUE is sometimes referred to as "transactive memory," and basically means that all team members must have the same understanding of who knows what in order to work as a team. Focusing on SKUE as a metric decreases the time it takes for team members to identify a skill or knowledge that is needed and go straight to the source. In 80% of the focus groups analyzed in the George Mason study, knowing who had what expertise on the team was among the most important team attributes for team effectiveness in incident response. Knowing what other members know quickens the incident response process. Here are some strategies for optimizing SKUE in your team:

1. Establish a knowledge base specific to incident response tasks and specialties
2. Cross-train team members
3. Implement job rotations or "ride alongs" in investigation and response scenarios

Identification (I)

- An incident is identified when there is an *impact* to your environment
- Requires direct observation and evidence
- High-quality detections, good communication with potential third-party sources (users, other teams, law enforcement, peer groups) is so vitally important



Identification (1)

Many SOC teams struggle with the identification phase of incident response. Sometimes this is a side effect of the escalation process – there may be notifications or other steps the team must take that dis-incentivizes them from calling "incident". In other cases, teams may escalate alerts or suspicious activity to incident status too soon, resulting in lots of wasted cycles and metrics skew. Incident identification happens when there is an *impact* to the environment – a system has been accessed, lateral movement has been observed, data is being prepared for exfiltration, an external third party has observed command and control traffic to your environment. Note the common characteristic in all of these activities – there is evidence demonstrating some negative impact, or evidence of some impending impact. This is why it is *so* vitally important to maintain detections that are high fidelity; without a high degree of confidence that <bad thing> is happening, it is difficult to say for sure whether or not an incident has occurred.

When to Call Incident

- The SOC is often the first place to raise the incident flag
- Analysts must know when it's appropriate to escalate, trigger collaboration
- Use cognitive prompts like the “Five Whys”, mnemonics, adaptive case management



When to Call Incident

Establishing when and under what circumstances an incident response is triggered is an often-overlooked part of the process. This can be especially important if you have relatively junior analysts responsible for taking the first few steps in escalation or must engage other groups to collaborate in a response. It's helpful as part of the preparation phase to define the criteria for escalation, and to ensure your analysts know when it is appropriate to “sound the alarm”. This part of the response process is based almost entirely on your organization's risk appetite and preference for how various elements are engaged in an incident response, so we'll avoid giving specific guidance on when and how you must escalate and start the process. However, we can discuss high-level recommendations that should help you define a customizable process for triggering the response process.

In addition to making sure your team knows when it's appropriate to call incident and what criteria must be met for doing so, it's important to make sure they can apply critical thinking and understanding of your internal process under pressure. In a perfect world, the duty analyst or incident handler can simply refer to the response plan and procedures; however, there may be cases where this reference step is overlooked due to time constraints, stress, or some unique chain of events that isn't explicitly addressed in your procedures. In those situations, it's important to have worked with your team to internalize those critical first steps in a response process. Use cognitive prompts like the **“Five-Why Analysis”**, pioneered by Toyota and commonly used in many companies today. The Five-Whys involves asking “Why?” a particular event is cause for concern or requires a response and then applying the same question five times to each answer. This particular approach tends to be more successful when used in a team setting versus individual decision making, but it can help the team ensure that nothing has been missed in terms of describing the incident triggers. Another strategy is the premortem, which requires an analyst to imagine they have already triggered the response and imagine ways they might fail to take critical steps or capture important information.

Mnemonic devices can be a great way to help analysts follow a simple process or capture key details about an incident in the very early stages of a response. In healthcare, a popular mnemonic is **SBAR**, which stands for Situation, Background, Assessment, and Recommendations (say, this might work for incident response too!). Finally, **adaptive case management** can provide the analyst with a roadmap and set of required actions to follow in escalating an incident versus expecting them to follow general guidelines or generic processes. Normally, adaptive case management is implemented as a case template or built-in playbook in your incident management system.

When It's Go Time - Overview

Once an incident is declared:

- Assign the incident handler / lead
- Collect data from key NSM / CSM data sources
- Understand who owns the asset and what it does
- Identify any compliance and/or safety issues
- Categorize the incident
- Begin remote live-box investigation if required



When It's Go Time

We've prepped our go bag, trained for the real thing, and now the time has come to react! When an incident is declared the detection and analysis stage procedures cover the standard actions that should be performed at this stage in the incident. The incident handler and lead should sync up on the plan of action, locate the details on the affected system, and prepare to contact the owner. If there are any potential compliance or safety ramifications, it's better to bring up that potential early.

At this stage, the incident handler may decide deeper forensic data collection is necessary and start performing live system incident response activities. This could include kicking off forensic agents collecting memory, hard disk image gathering, and more.

Assign the IR Lead

- Primary source of truth on technical elements of the incident
- Facilitates communication and collaboration
 - Establishes communication channels
 - Delegates/supervises response tasks
 - Keeps team on task
 - Panic management
 - Schedules/facilitates post-mortems
- Should be experienced team member with knowledge of IR best practices and your specific IR approach

Assigning the IR Lead

One of the first decisions you'll need to make when initiating a response is deciding who will lead the effort – you'll need a single point of contact to facilitate communication, collaboration, and the various tasks that need to be done. This individual may or may not be a senior member of the IR team, but it must be an experienced team member with knowledge of incident response best practices, organization-specific processes for response and communications, and the ability to make good decisions in a crisis. The IR lead will:

- Serve as the primary source of truth for updates to ops management and/or groups outside of the technical team
- Establish the communication(s) channels the team will use to coordinate the response
- Delegate and supervise technical response tasks
- Ensure the team stays on task by holding them to the pre-established IR process
- Keep the response team and various constituencies calm and focused on achieving near-term goals versus staying in a reactive mindset
- If appropriate, schedule the post-mortem review or at least provide input to that review

Key Roles

- **Subject matter expert(s)**
 - Understands technical systems and their function – in the SOC and in the affected parts of the environment
- **Incident Lead (or Incident Commander)**
 - Brings order to chaos
 - Isn't investigating, analyzing, reviewing, checking, etc.
 - Primary duties are managing and delegating
- **Liaison/Scribe**
 - Captures and disseminates information to interested parties

Key Roles

Key roles in the incident response process are subject matter experts, the incident lead (sometimes referred to as the incident commander), and scribe (sometimes referred to as a liaison or coordinator). **Subject matter experts** are SOC analysts, engineers, or system owners with detailed knowledge of the affected system(s) and available telemetry and security controls. These individuals should be experts in their specific discipline or system, and in certain limited incidents these individuals may also assume the role of incident lead.

As we saw on the previous slide, the **incident lead** ensures that the response is being managed appropriately. Note that in major incidents, the incident lead is normally **not** the one who is executing the technical response or putting “hands on keyboards” to gather information, analyze data, or perform other technical tasks.

Coordinating a large-scale response requires dedicated, undivided attention, strong communication skills, and the ability to enforce process on the fly. This is a specialized role and should be handled by someone who has had training to run incident response in your organization.

Not every incident will require a **scribe** or **liaison**. This role will most likely only be needed during moderate or extensive incidents. The Scribe will collect information and document an incident to ensure key information is recorded for after-action reviews and updates to interested parties are timely, accurate, and complete. Generally, this individual does not require specialized incident response or security knowledge. 3

Evaluate Key Data Sources

Guide analysis by breaking down the process:

- Encourage analysts to have an intentional investigation plan,
- Don't just have them dive in aimlessly – time gets wasted this way

Questions for Preparation:

1. What *specific* question are you trying to answer?
2. What data do you need to answer that question?
 - Auth logs, process creation, flow logs, ...
3. How / where can you get that data?
 - SIEM, IDS console, EDR, ...
4. What does the data tell you, and *not* tell you?



Guiding Analysis During Initial Investigation

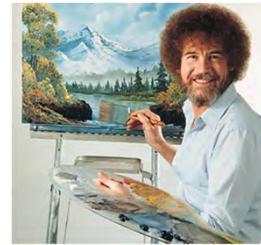
Once an incident is declared, it is highly tempting to go out and haul in all possible evidence and logs and start sifting through them based on your initial ideas. While this can work, efficiency in this stage is key, and there are some methods and mental models that can be used by analysts to focus their data gathering and evidence evaluation.

These questions, on the slide above, are inspired by a blog post called "The Alexiou Principle"^[1], but have some additional detail added to make them more explicit and translate to incident response. You may have heard the quote supposedly from Abraham Lincoln, "If I only had an hour to chop down a tree, I would spend the first 45 minutes sharpening my axe." These questions have a similar theme, and that is - understand and prepare for what you're going to do and choose a clear direction before you just jump in and end up wasting time with a "dull axe" and no plan.

[1] <https://thedigitalstandard.blogspot.com/2009/06/alexiou-principle.html>

Track Down the Asset(s)

- Chris Sanders has written about intentional versus unintentional evidence – same concept applies to asset info!
- **Intentional:** Data purposely created with the intention of auditability
 - Asset management, vulnerability management systems
- **Unintentional:** Created as byproduct of other process
 - Windows event logs, DHCP, web proxy, IT management systems
- The main difference: Intentional is easier to use!



Tracking Down the Asset(s)

There is a great blog post by Chris Sanders that discusses the concept of evidence *intention*.^[1] The concept is this: when doing an investigation, there are a multitude of sources you may consult to try to piece back together the set of events that led to a potential compromise. Some of these evidence sources are written by tools or programs that are writing the log with the full intention of that log being used as evidence. On the other hand, there are also "unintentional" evidence sources—things that happen as a side effect of the user doing something that aren't intended to be used as evidence, but nonetheless still serve the purpose of aiding an investigation. For example, any time a new executable is opened in Windows, a file is written to the hard drive that shows the name of the file and has a timestamp showing when it was run.

This is an extremely useful concept in investigations, but we can also extend this thinking to tracking down assets during an incident. Practically anyone who has responded to an incident or attempted to validate alerts in a SOC has run into the issue of tracking down an asset – many organizations do not have complete or fully up-to-date inventory management systems, or the SOC may not have access to it. So, the question becomes, where else can we look to infer details about where a host is located, who is using it, and what its normal function is? If we can't refer to an asset management system or vulnerability management system, we can try checking unintentional sources of device information – when did it last pull a DHCP lease, and can we get other identifying information from those logs? What about web proxy logs if it's a user device or details from a network management system? Why is thinking about web proxy logs if it's a user device or details from a network management system?

Make sure your team is aware of unintentional sources of asset data so that a rogue or potentially compromised device can be tracked and contained even if intentional data sources aren't available.

[1] <https://chrissanders.org/2018/10/the-role-of-evidence-intention/>

Identify Safety and Compliance Issues

- If you're responsible for incident response, you must understand compliance requirements for your org
- Most regulatory requirements deal with adherence to pre-defined IR processes, impact analysis, internal/external disclosure, documentation, and data retention
- If relevant, include life safety and compliance assessment in your procedures and your exercises



Identify Safety and Compliance Issues

If your SOC is defending an organization involved in a regulated industry, determining compliance with applicable regulations is one of the most important parts of incident response. Staying on the right side of compliance issues can help you avoid fines and penalties, lawsuits, loss of corporate licenses or certifications, and even criminal penalties. At the risk of overstating this requirement, the stakes can be very high for non-compliance. Regulatory requirements usually include formal, well documented incident response processes and team structure. Organizations operating under the Health Insurance Portability and Accountability Act (HIPAA) or the Payment Card Industry Data Security Standard (PCI-DSS), for example, must have a documented security-response plan and a response team; the Federal Information Security Management (FISMA) lays out specific incident management and response guidelines for US federal agencies. Is your team aware of these requirements and adhering to the structure previously defined by your organization? Here are some basic considerations that apply to most regulated industries with regard to incident response:

- Sensitive systems – that is, systems that store or process sensitive data – should be identified; does your incident involve any of these systems or data?
- Does the nature of the incident potentially require disclosure to specific internal or external parties?
- Are detections and your response actions captured in an auditable system of record that includes responder contact information and detailed descriptions of the incident?
- Are your incident response reports and associated logs stored for the required amount of time based on your compliance requirements?

Understanding the implications of compliance and life safety issues, and building that understanding into your response process, will dramatically reduce your organization's exposure to additional risk and keep you out of trouble!

Categorize the Incident

NIST 800-61r2 suggests 3 categories:

- Other popular option – Verizon's VERIS¹

<u>Functional Impact</u>	<u>Information Impact</u>	<u>Recoverability Effort</u>
<ul style="list-style-type: none">• None• Low• Medium• High	<ul style="list-style-type: none">• None• Privacy Breach• Proprietary Breach• Integrity Loss	<ul style="list-style-type: none">• Regular• Supplemented• Extended• Not Recoverable

Incident Prioritization and Categorization

Your incident response policy should contain guidance on how incidents will be prioritized, and one of the ways the priorities may be expressed is through their categorization. There are plenty of options available for categorization, some of the most popular being the NIST 800-61r2 suggestion shown on the slide above, or Verizon's VERIS event recording vocabulary.¹ Which should you pick? Check out the system (or make one up!) that most closely aligns with the items your organization would like to track, and the depth of detail that can be conveyed in that tracking. The NIST 800-61 system is nice because it ranks the various dimensions such as functional impact, information impact, and recoverability, but does not go into great detail. VERIS on the other hand can go into an almost absurd amount of detail, so much so that analysts may find it burdensome to use. (Using only a subset of the VERIS framework can be a solution to this problem.) Feel free to mix and match, taking the pieces that work for you. If you don't need to share your incident metrics in a standardized way outside your organization, you can make any incident recording system that fits your needs.

[1] <http://veriscommunity.net/>

First Actions During IR: Live Box Investigation

Two big categories – **Investigation** and **Response**

For **Investigation**:

- **Gather volatile evidence**
 - Memory forensics and more depend on fast data collection
 - EDR and SOAR tools can be used to automate collection
- **Time bound** the incident if possible
 - Use evidence to find the first moment of activity
- **Identify the initial point of entry / infection**
 - Look for connections to/from that system at that time

First Actions During Incident Response - Investigation

Activity required during an incident can be broadly grouped into two categories – solving key questions about what happened, the "investigation" and what to do about it, the "response". Each of these activities have an enormous list of tasks to undertake, but which ones should be addressed *first*?

One key activity during initial incident response is to get the baseline parameters of the incident established and preserve any volatile evidence. Since these details may help you crack the case, but may also disappear very quickly, it's important to kick off any IR tools or tasks you have that can preserve memory, running processes, or anything else that might hold a key clue in your investigation. Having EDR and SOAR tools can be a big help here.

Another item is to quickly move to find the initial point of infection, and time bound the entire incident if possible. While at the start you know of at least one affected asset or user, that doesn't mean they were the first. The SOC should quickly move to find exactly when the known user was affected, and then correlate other logs and activity at that point in time to see if they were the initial victim, or secondary to other activity. Finding this answer helps time-bound your investigation so that your team can hone-in quickly on the logs that matter and ignore everything previous to the initial infection point.

Data Acquisition Considerations

When deciding on live box investigation, consider:

- Remote versus local acquisition
- Cost to obtain based on tools required
- Format you need the data in
- CLI or GUI acquisition tools
- Full dump or runtime/key artifact investigation



Answers will make a big difference in cost, type data you'll get, skills required, and the time it will take to obtain and analyze it

Data Acquisition Considerations

Early in the response when your team is considering whether to acquire live data, there are several considerations that should factor into your decision. As a lead or manager, it's important to understand these considerations as your recommended approach will have major implications on the time to investigate, technical challenges in getting and sharing the data, and what kind of analysis your team will be able to perform. These considerations are:

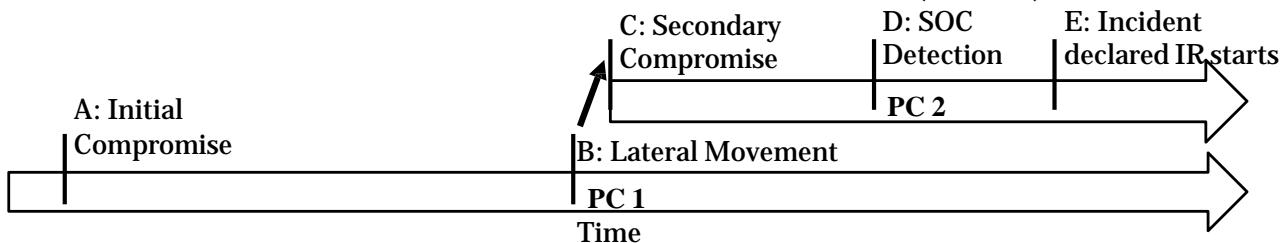
- Do you have physical access to the target system?
- Will data acquisition require expensive, purpose-built hardware and software?
- If you're acquiring memory, what format do you need it in?
- Will you be using the command line or a tool with a graphical user interface to acquire the data?
- Do you need a full physical memory dump or just a snapshot of key artifacts like running processes, network connections, users, etc.?

Understanding these key questions will help you, as a manager, understand what exactly you're asking for and ensure that the results will align to your team's technical capabilities and expectations. We'll come back to this topic later today when we discuss different tools and utilities that you might use to conduct live analysis.

Key IR Moments and Metrics

Points of interest for IR and metrics generation:

- Time of *initial compromise*, and time of compromise *per machine*
- Times of *interaction* between all infected machines
- Time to *detection* after initial compromise (A to D)
- Time to detection after compromise per machine (C to D)
- Time to incident *verification* from SOC Detection (D to E)



Key IR Moments and Metrics

When finding your initial point of compromise, it's common to begin working backwards from an alert you received (point D above) and realize you're looking at a secondary infection (PC 2 above) resulting from a previous infection on a totally different machine (PC 1). The most common workflow then would be to move from the known detection (point D), to find the point of infection for that machine (point C). Once C is known, network and host data should be scrutinized to see how the attacker accessed that machine and compromised it. If this was a multi-machine attack, you should have evidence that points to lateral movement (point B) above on the previously infected machine, and the same process can be repeated, revealing the true point of first infection (point A above). Once the "real" first point of access is found, this is where the SOC can time bound its search. Don't stop there, however; look for evidence of both PC 1 and PC 2 connecting to an as-yet-unknown PC 3. Once the pivoting tactics are exposed, finding each additional effected machine becomes easier. Additionally, these data points also can, in aggregate, make for incredibly useful metrics on the speed of your investigation and detection.

As a SOC, your goal is to get from point D to point E as quickly as possible (if an incident did truly occur), and also to gather evidence from points A, B, and C before it disappears. Additionally, your goal as a SOC in general is to also minimize the time between A and D as well, doing so will ensure volatile evidence does still exist and that minimum damage was inflicted before the SOC starts taking action. The ideal SOC can go from A to D, and D to E in effectively zero time, having an automated detection go off at the initial compromise that is so high fidelity that the team trusts the alert (without a need for in-depth investigation) and can move to immediate containment and remediation actions. Any factor in this diagram that can move you closer to that ideal is a worthwhile improvement to chase.

First Actions During Incident Response - Containment

In many IR scenarios, **containment** is one of the first actions:

- **Network-based**

- Block all infected machine communication to the internet
- Block all infected machine communication on the internal network

- **Endpoint-based**

- Can / should a malware process be killed?
- Should the host be isolated?
- Watch and learn?

- Don't forget to consider **OPSEC!**



First Actions During Incident Response - Containment

If we consider the incident response cycle (PICERL – Prepare, Identify, Contain, Eradicate, Recovery, Lessons Learned), we get to the second answer of what must initially occur during IR – containment. Of primary importance in many cases is to "stop the bleeding" and prevent the situation from becoming any worse. Since often the situation is getting worse by the progression of an infection trying to multiple over the network communicate with its command-and-control infrastructure, or even destroy or modify the system the code is running on, containment can often be broken down into stopping those actions. For containing network traffic, consider if an impacted system might need to be cut off from the internet, the internal network, or both. In many cases for non-critical systems the choice is obvious – cut the system entirely off, but in certain scenarios you may have to leave the system operating while preventing only the malware from running or progressing. Alternatively, or in combination with network-based containment methods, host-based containment such as blocking and killing of malicious processes, host-based firewalls, or other containment strategies also should be implemented if OPSEC is not of concern.

Containment Procedures

Goal: Quick, tactical actions to stop attack progression

- Understand the adversary's foothold in the environment
- Set up additional detailed monitoring of infected asset(s)
- Develop a plan of action
- Inform stakeholders of the issue and planned actions
- Apply tactical containment
- Closely monitor the infected system for changes
- Develop a plan for long-term eradication



Containment Procedure

Now that you've identified an active incident and know the details associated with it, it's time to take the first step to disrupt the activity: containment. This step should be thought of as quick, tactical moves to "stop the bleeding" and prevent the situation from getting any worse than it already is. These will *not* be the long-term remediation actions, just what it takes to pause the attacker.

Tread slowly in this stage, you want to be sure that you understand the communication method and protocols being used, as well as any other machines the infected asset may be talking to. Why? There are many malware samples I've personally seen that have backup command and control servers to fail-over to if the primary domain or IP is blocked. You won't know what they are since they will never be used until the first choice fails. This means if you choose to contain something based on domain or IP address, be ready for it to immediately reach back out with another destination.

Containment procedures should involve understanding what you're dealing with to the best of your ability, coming up with a plan of action, letting system owners know what's going to happen (if necessary), then taking that action and monitoring *very* closely for a shift in communication if you didn't fully cut the machine off from the network.

Containment Actions

Every containment option has positives and negatives:

- Physical disconnection
- Logical isolation
 - Isolated VLANs
 - Non-routing IP assignment
- Block by specific IP
 - Network firewalls
 - Null-routing
- Block by domain name
 - DNS firewall
- Block by port / application
 - Next-gen firewalls
 - Web-app firewalls
- Host-based firewalls
 - IP address / Port
 - Single applications
- Host-agent and EDR assisted isolation



Containment Actions

Incident responders will likely have many different options available to them for containment. Their job is to use their knowledge and skill to identify the best possible route to take from what is available. Usually the "best", in this case, means being nearly 100% sure that the activity will successfully be blocked, while ideally causing a minimum of disruption. Some situations, of course, are so important that disruption is not the priority and merely stopping the attack is all that matters.

The slide above lists some of the many common containment techniques from things that are highly disruptive, such as physically disconnecting a device to extremely targeted tactical moves like blocking an individual program running on an endpoint using a host-based firewall. Analysts should have authority or easy access to request any one of these items your organization allows in order to have the most flexibility in containment actions. To gain the authority to make changes in some of these systems, the SOC may have to build trust with the system administrators, but once they see the need and any fears about process are addressed, access should hopefully be granted.

Staying Organized

During an incident be sure to capture:

- Action items and who is responsible for them
- Timeline of events
- Initial leads and outcomes
- Kill chain context for observed events
- Detailed host information
- Sensitive information that may have been accessed
- Response actions taken

Staying Organized

During an incident response, and particularly during the early stages of it, it's vitally important to stay organized and capture all of the data you'll need to prepare for containment and eradication, not to mention the final report. We'll talk about documentation in a moment, but for now bear in mind that we want to track the following details starting as early in our response effort as possible:

- Action items and leads for follow-up and who, specifically, is responsible for them
- Timeline of events so far
- Initial investigative leads and outcomes (the latter is important so as not to leave any loose investigative threads)
- Kill chain context for all findings and observed TTPs
- Detailed information about impacted hosts, including function, configuration, owners, and vulnerabilities
- Any sensitive information or access that may have been compromised
- Response actions taken

In this stage, it doesn't really matter how this information is captured provided you're keeping it in a central location and somewhat organized. Next, we'll discuss documenting some of these key details as you work.

Documenting Incident Findings as You Work

- Use structured analysis techniques and data organization
 - As evidence is found, brainstorm possible attacker actions / motives
- For organizing evidence
 - Link analysis diagram
 - Event matrix diagram
 - "externalization" in general - *get info out of your head and onto a shared medium everyone can see and understand*
- For understanding where evidence leads
 - Analysis of Competing Hypothesis



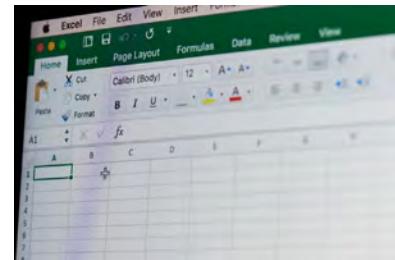
Documenting Incident Findings As You Work

As you start investigating an incident, data will be gathered at high velocity, and organizing findings and theories from the large volumes of data becomes a task of its own. Keep analysts focused on the goal of understanding each step of the attack, and what evidence can be found to support those theories is key to efficiency.

As evidence is found, consider using structured analysis techniques to both explore and understand the data set, as well as interpret and understand what truly happened. Techniques like link analysis and event matrices (covered in the next few slides) can help visually display data and piece together the story of what happened through visual externalization of the evidence. Taking the evidence and generated hypothesis and evaluating it using the analysis of competing hypotheses method can ensure that you are making any basic logical errors, and that your conclusions sit on solid ground.

The Spreadsheet of Doom

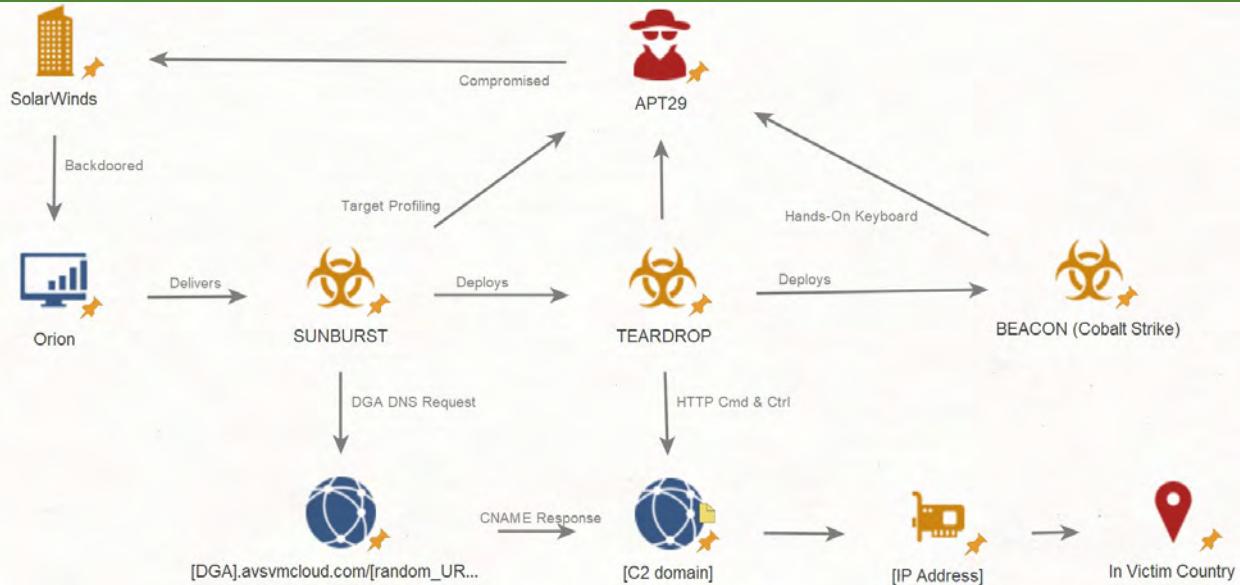
- Don't let perfect be the enemy of good
- Spreadsheets are customizable and everyone knows how to use them
- Supports many different data types
- Exportable in CSV format
- Take note of fields and column names for use in more permanent solutions



The Spreadsheet of Doom

In their book *Intelligence-Driven Incident Response*, Rebekah Brown and Scott Roberts describe what normally constitutes the first attempt at incident tracking using Microsoft Excel. Dubbed “the Spreadsheet of Doom,” such a document tends to get out of control very quickly – it usually contains a mix of indicators, response actions, tabs, and other ever-expanding items. But it’s worth talking about for a few reasons. First, if you’re a new team without a purpose-built repository for tracking and documenting incident response, having a quick and highly customizable solution built on something everyone knows how to use can be the best route to take. We can also take note of how these early solutions are set up, fields they contain, and other features we value so that we can build them into a more permanent solution. Spreadsheets have the added advantage of being exportable as a CSV, which can then be imported wholesale into a variety of different tools and repository. **Bottom line:** don’t discount the humble spreadsheet (of doom)!

Link Analysis



SANS

MGT551 | Building and Leading Security Operations Centers

81

Documenting Intermittent Findings with Link Analysis

As you work through the incident an incredible wealth of detail will be revealed eventually bringing the picture of the full incident into focus. While SOC teams need to keep highly-detailed incident data and observables documented and organized in incident management systems, do not forget about the other audiences for which the incident must be explained. Making a high-level link analysis style diagram highlighting each entity involved in the incident and how it is related to the story can help in this effort. It not only externalizes for the group what is known in a digestible way, but also doubles as a great incident report and presentation tool for those who must be briefed on what is known and what is affected.

In the slide above, a depiction of the main interactions in the SolarWinds breach from late 2020 are documented in Maltego Casefile. This includes the attacker, victim company, the affected software, and how the malware progresses through different stages, communicating with multiple command and control domains and more. There are no hard rules as for what to include and not include, the diagram should be tailored to the audience and its intended purpose labeling both nodes of various types, and edges describing the interactions between them.

Event Matrix Chart

Link analysis meets a timeline:

System	Mar 7	Mar 10	Mar 15	May 13	May 14-Jul 29	Jul 30
Application Servers	Vulnerability disclosed, exploit found	Initial recon begins	<ul style="list-style-type: none"> IDSSignatures installed Vuln scanning misses vulnerable system 	System Exploited, Web shell installed	Data staged in web accessible folder for exfiltration	Breach Discovered
File Share Server				Unencrypted credentials stolen from config database		
Databases				Credentials used to access 48 databases		

Event Matrix Graph

An alternative method for displaying what happened in an incident aligned by both time and the asset on which the event occurred is the event matrix chart. Unlike the link analysis method, event matrices can be useful for tracking a chronological set of events that happened during the attack. Nearly everyone will default to making at least one timeline view of an incident, but the event matrix can take it one step further. The event matrix takes the common timeline and separates it into one timeline per entity involved in the breach and shows them all together (done with a table on this slide). Each entity gets their own "swim lane" for actions that occurred there, and arrows can be used to demonstrate lateral movement and activity across entities. The benefit of the event matrix chart is that it shows relationships between affected devices and illustrates the "story" of the compromise better than a single timeline. Event matrices can even lead to prediction of additional evidence that may not have been found yet through discovering gaps in the timeline or story, which if detailed enough, can lead investigators to new theories and data sources.

The slide above shows a simple demonstration event matrix for the Equifax breach in 2017, one of few breaches for which there is detail recorded and publicly available, due to the US Government congressional report¹. In this chart (which could be much more detailed if we knew every single hostname and weren't worried about it fitting on a slide) we show the application servers which were exploited using the Apache struts vulnerability by the attackers via the internet, as well as used for exfiltration. It also lists the file server which was accessed to steal credentials, and the database servers, which contained the data the attackers came for. Glancing at this chart gives a quick way to understand the key assets involved, the actions that occurred on those assets that lead to the problem, and the order in which the incident advanced – exactly the type of chart that might need to be shown at a higher-level, non-technical meeting. Whether technical and highly-detailed, or high-level activity, the event matrix chart is one of the forms that can help you not only organize but communicate what is known about a breach in progress.

[1] <https://republicans-oversight.house.gov/wp-content/uploads/2018/12/Equifax-Report.pdf>

Physical IR Preparation: Your IR Go Bag

Gathering IR hardware and software (for those doing in-person IR)

- Prep the "go bag" for jumping into action immediately

Hardware and tools to include:

- High-power laptop
- External hard drives / USB drives (forensically wiped)
- Notepad/journal
- Write blockers
- Network taps, cables, small switch
- Bootable Linux LiveCD
- Gold Build Image



Physical IR Preparation: Your IR Go Bag

A key piece of incident response specific preparation is the creation of the "go bag", a ready-to-go kit for the most common incident response tasks that you can pick up and use at a moment's notice. This page lists some of the common items you should stock in your kit. In a nutshell, it should prepare you to go and grab forensic images, packet captures, rebuild systems, and anything else you can envision yourself doing on the scene of an incident.

Once created, remember to keep the tools up-to-date and also not to use it as a place to "borrow" things from (and then forget to put them back).

IR Go Bag Software

Be ready to perform any of the following

- **Live and dead-box disk acquisition**
 - Tools: FTK Imager¹, SIFT², Paladin³
- Local and remote **memory acquisition** and analysis
 - Tools: Pmem tools⁴, F-Response⁵, Volatility⁶, Rekall⁷
- **Network acquisition** and analysis software
 - Tools: NSM distro like SecurityOnion⁸ or Wireshark, Zeek, Suricata, NetworkMiner, etc.



IR Go Bag Software

In addition to the hardware in your kit, there are several types of software you should keep readily accessible as well. The most common tasks for incident response will revolve around the acquisition and analysis of hard drive images from both "live" and turned off (dead-box) machines, memory acquisition from running systems, and capturing network traffic in an out-of-band fashion. This slide is just a partial list of some of the most popular tools for accomplishing these tasks.

- 1 <https://accessdata.com/product-download/ftk-imager-version-4-2-0>
- 2 <https://digital-forensics.sans.org/community/downloads>
- 3 <https://sumuri.com/software/paladin/>
- 4 <https://github.com/Velocidex/c-aff4/tree/master/tools/pmem>
- 5 <https://www.f-response.com>
- 6 <https://github.com/volatilityfoundation/volatility>
- 7 <https://github.com/google/rekall>
- 8 <https://securityonion.net/>

Malware Analysis (1)

A highly complex and time-consuming activity!

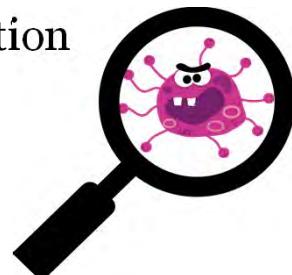
Methods:

1. Automated analysis with a malware sandbox

- Great for high-volume, automated analysis
- *Most of the time* provides enough information

2. Manual analysis performed by hand

- For malware with anti-analysis features
- Gets you the best answers, but slow



Malware Analysis (1)

Malware analysis is a critical capability for your SOC to have available. When you undoubtedly encounter malicious file samples emailed in from spam, downloaded by users, or found on USB sticks, the SOC will need to be able to examine them and extract indicators of compromise in a safe and efficient way. The approach to this should be to take the easiest route first because malware analysis is a highly complex and specialized task.

One method that is available to all teams is automated analysis. There are plenty of solutions on the market for malware sandboxing which enable a SOC to upload a file sample and receive a report on what happens if that file is run. While these systems are highly efficient, automatable, and do not require a high level of skill for analysts to use, they don't always work. Several variables go into this. Many pieces of malware are designed to detect if they are running in a sandbox, so if your sandbox is obvious to the malware, you may see no results. Other times malware is in a format the sandbox does not support. If you upload a zip file that contains a word doc with an embedded executable object, for example, your sandbox must know how to support automatically unzipping the file, opening the doc, and double-clicking that executable (or allow you to manually interact with it during analysis). If this doesn't happen, the sandbox will report no results. Therefore, the quality of output you get from a malware sandbox will depend on a number of factors both within and outside of your control.

When malware sandboxes aren't able to get the job done you must fall back on manual analysis. Manual malware analysis is a sub-specialty of information security and not all teams will have someone available who can do it (which means you may need to identify a group you can outsource it to). While not easy or inexpensive, manual malware analysis is sometimes the only way to get the answers you need. For example, if you experience a targeted attack with highly complex malware that no one has ever seen before, it's likely the attacker will build in self-defense features that will make analysis difficult. Extracting the IOCs you need from the malware to scope your incident timeline and identify victims may rely on your ability to take apart the malware. This means manual malware analysis capability, although rarely needed, can become incredibly important. At a minimum, have a plan for what you will do if you encounter this situation. If you have analysts with the experience and interest in malware analysis, SANS FOR610: Reverse Engineering Malware is an outstanding course that will teach the skills required to tear apart malware at the assembly level and step through it with a debugger at the lowest levels.

Cuckoo Sandbox

- A free malware analysis solution
- If you do not have a malware sandbox, use this!
- Features
 - Wide OS support (Windows, Linux, macOS, Android)
 - File types: exe, scripts, MS Office, PDF, email, websites
 - Optional internet connection: fake, VPN, TOR
 - Decrypts SSL, captures packets, applies IDS signatures
 - Memory forensics captured
 - YARA support



Cuckoo Sandbox

If you do not already have a commercial malware sandbox available, Cuckoo sandbox (<https://cuckoosandbox.org>) is a free option that is truly outstanding. This open-source software does the same thing that commercial sandboxes do—accept samples from a web-based interface or automatically through an API and runs them through a full dynamic analysis, recording the results.

Cuckoo sandbox comes with the same professional features that many commercial sandboxes offer including support for multiple operating systems and file types, controllable internet connection options, IDS signatures applied to network traffic, YARA signature support for files, SSL decryption, memory analysis, and more. It will even allow analysts to manually interact with the virtual machine through a safe web interface in case there are specific actions that need to be taken to trigger the infection. This is a feature that not even all commercial tools offer. In my experience, Cuckoo holds its own against any commercial malware sandbox and even if you already have one, it can make a great supplement in case a backup or alternative is needed.

Eradication

Goal: Fully removing the attacker from the environment

- May not be as straightforward as you expect
- Consider what you know about the attacker before making a move

Eradication Strategies: Decision factors:

- | | |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <ul style="list-style-type: none">• Ignore• Automated removal (AV)• Surgical Removal• Watch and Learn• Wipe and rebuild | <ul style="list-style-type: none">• Targeted / Opportunistic• Length of compromise• Criticality of asset / user• Intel / Attacker TTPs• Attack style• Your role in attacker's goals |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|

Eradication

After containment comes the eradication stage where your goal is to quickly kick the attacker out of the environment. For many attacks, you have the standard set of options ranging from ignore it (for adware and other non-issues) to wipe and rebuild, for when you want to be very sure the infection is gone. Some may question why you might choose any option other than wipe and rebuild with the philosophy that once something is infected, it can never be trusted again. While, personally, I do agree with this sentiment, there are situations where less than wipe and rebuild might be called for. Depending on how much information you have, the type of attack you're dealing with, and the criticality of getting an asset back into service, there are situations which may arise where zero downtime is the priority, and surgical removal of a virus might be the better option. In this case, it may even be a medium-term step while you wait for a service window to do the full wipe and rebuild.

The procedure eradication is not always as straightforward as it may seem, however. Advanced attackers are resilient and depending on some of the factors listed above, you may actually want to wait on eradication and take the "watch and learn" approach, choosing to delay eradication.

Watch and Learn

But *should* you contain/eradicate incidents immediately?

- **Commodity malware:** Yes—go ahead
- **Targeted malware:** Maybe...
 - Consider OPSEC – attacker now knows you're on to them
 - Will change tactics, techniques, tools
 - May spread deeper into the org, make things worse
 - Response must be carefully decided within context of the incident
 - Each situation will require a unique consideration



Watch and Learn

You've picked your containment or eradication method, is it time to let it loose and knock the adversary out of the environment? Not so fast ... There are actually some situations where preemptive containment before the situation is fully understood can make the situation worse!

If you are dealing with commodity malware and you're quite sure of it, then you can likely go ahead. How do you know this? If you can find references to the malware you have on sandbox and virus collection sites, mentioned on blogs, or otherwise publicly known, you probably aren't dealing with a highly advanced attacker. In these cases, they probably aren't trying to crawl through your environment and cleaning up any machine with the infection will get rid of the problem.

If you might be dealing with a targeted attack however, the "watch and learn" approach, many times, will be more appropriate. This strategy is to *very* closely watch the infected asset and look back into history to put the story together of how it became infected before making any drastic moves. If you act too quickly, it's highly likely the adversary has multiple other points of entry into your environment and will now be tipped off to your detection. Once they know you're on to them, it's highly possible they'll change tactics, spread to more machines, or go silent, making you think you've won the battle, even though you truly haven't. Do adversaries do this in the real world? Of course they do! Why wouldn't they?! Even penetration testers and Red Teams use tactics like this to make sure they don't lose access to the target environment.

Good Decisions vs. Good Outcomes

- **"Watch and learn"** is sometimes necessary!
 - You may have to explain yourself very carefully to management
 - US-CERT has your back¹ - "*preemptive blocking is a common misstep*"
- Distinguish a **good decision** with a **good outcome**
 - You must separate the "smart move" from the outcome of that move
 - The safe bet doesn't always work out, and the unsafe bet doesn't guarantee loss
 - When judged appropriate, watch and learn is a **good decision**
 - It is considered best practice, and the safest move in many situations
 - It does **not** guarantee a **good outcome**
 - It may go wrong, that doesn't mean you made a bad decision



Good Decisions vs. Good Outcomes

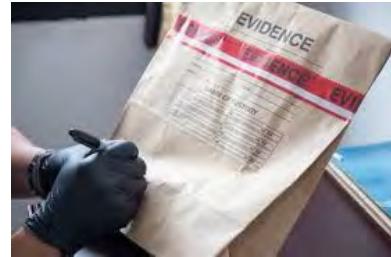
While "watch and learn" can seem like a hard sell to management and people who haven't dealt with advanced attackers before, rest assured it is a best practice in situations where it is appropriate. Even the US-CERT guide¹ lists preemptive blocking as one of the common missteps in incident response. Of course, if someone's safety is in imminent danger, do not watch and learn, but in other scenarios, this is a strategy you should consider.

Just because watch and learn can be best practice, however, doesn't guarantee it will work out as hoped. In tough decisions like this, it's important to remember the difference between a good decision and a good outcome. The "smart move" doesn't always pay off, and the unwise move doesn't always have bad consequences. Consider gambling for example: the house always has the advantage, which means playing is always a "bad decision", but sometimes you end up ahead! Watch and learn is the reverse, it is a defensible, best practice "smart move" in many cases, but that doesn't guarantee it won't backfire. Be prepared to back yourself up with why your watch and learn strategy was still a *good decision*, even if it doesn't work out as planned. People who make good decisions should never be faulted for bad outcomes - you chose the best option given the information you had, but it's still a gamble, and the dice just didn't fall the way you hoped.

[1] <https://www.first.org/resources/papers/conf2016/FIRST-2016-108.pdf>

Preserving Evidence

- When dealing with digital evidence, the following principles apply:
 - The process of collecting, securing, and transporting digital evidence should not change the evidence.
 - Digital evidence should be examined only by those trained specifically for that purpose.
 - Everything done during the seizure, transportation, and storage of digital evidence should be fully documented, preserved, and available for review.
- Digital evidence must be documented, secured, labeled, and preserved
- Ensure chain of custody, authorized personnel, responsible parties are defined in your incident response policies and procedures
- Logs, incident reports, and written policies are all admissible evidence



Preserving Evidence

Following strict evidence preservation guidelines is good practice even if your incident isn't likely to end up in court or reach law enforcement. Engaging with law enforcement can be a hot-button issue for a variety of reasons, but in the course of your duties you may encounter scenarios where that becomes likely. In these scenarios, it's important to have well-documented procedures that include things like evidence preservation should you need to provide that evidence to law enforcement. Adhering to high standards when it comes to evidence preservation can have the added benefits of protecting the organization from loss in an insurance claim, lawsuit, or regulatory violation.

For further information regarding computer and network forensic requirements for law enforcement, see *Electronic Crime Scene Investigation: A Guide for First Responders* and *Forensic Examination of Digital Evidence: A Guide for Law Enforcement*, which are both available at <http://www.ncjrs.gov/>.

Identification, Containment, and Eradication Summary

- This is your chance to contain the damage!
- Bring receipts – network data, host data, timeline of events
- Stay organized, document everything
- Consider data acquisition strategy carefully
- Be ready to perform additional data capture and analysis when required – too late to think about these capabilities when an incident has occurred

Identification and Containment Summary

Identification and containment are some of the most crucial phases of the incident response process as this is your opportunity to minimize the damage to your organization and gather volatile evidence that may only be available for a short time. Also, remember that you will now likely be involving other stakeholders in the process, so backing up your assertions with hard evidence – key network and host data – is critical so as not to waste cycles or misidentify the incident. Capture those key pieces evidence, and your analysis of them, in detailed notes that are accessible and editable by all participants in the process. Also, now is the time to consider gathering additional evidence from affected hosts, so consider your data acquisition strategy carefully. Failure to guard your credentials when connecting to compromised hosts can potentially give the attacker much more access than they already have, and failure to exercise care when collecting evidence can corrupt important data. Finally, have that “IR go bag” ready in advance, because forensic analysis capabilities get much more difficult (not to mention expensive) to build out during an incident versus during the preparation phase.

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

Planning, Preparation, and Review

- Incident Response Planning and Preparation

- Investigation

- *Exercise 4.1 – Investigation Quality Review*

IR Execution

- Identification, Containment, and Eradication

• Incident Response in the Cloud

- IR Tools

- *Exercise 4.2 – Planning Responses with RE&CT*

- Crisis Management and Continuous Improvement

- *Exercise 4.3 – Designing Tabletop Exercises*

- Recovery and Post-Incident

- Summary and Cyber42 – Day 4



This page intentionally left blank.

Incident Response in the Cloud

- Incident handling best practices still apply
- Focus on layer 7: applications, APIs, access and identity
- Design a collaborative response approach, including infrastructure teams and vendors
- What do you want to be able to do?
 - ❑ Commands you want to run
 - ❑ Telemetry you want to generate
 - ❑ Artifacts you want to collect

Incident Response in the Cloud

As many of our organizations race towards IT “transformation”, understanding how to accomplish our detection and response goals in cloud infrastructures will quickly become a necessity (if it isn’t already part of your scope). Remember that most of the incident response best practices we’ve discussed still apply – the general process and procedures should remain the same. What will differ from your on-prem systems is the kind of data you’ll collect and the means by which you’ll get it. In dealing with infrastructure or software as a service, you’ll be dealing primarily with applications, application programming interfaces (APIs), and identity as key assets. Getting the visibility and the data acquisition capabilities you need will likely require you to rely on various partners and service providers; as part of your IR planning process, you’ll need to decide what commands you want to run, telemetry you want to access, and artifacts you’ll need to collect in the event of an incident. You don’t want to be working these details out with your vendors when an incident has already occurred!

In this section, we’re going to discuss some of these preparatory steps, reference frameworks you can use to track your capabilities and your progress, and some high-level best practices for a few of the major cloud service providers.

Preparation (2)

- Turn on logging and, if possible, VPC flow capture
- Encrypt data at rest
- If possible, separate access roles by system or business line
- Consider third parties that may require access in the event of a compromise
- Ensure incident leads and SMEs are trained to deal with cloud-specific incidents
- Build security-related failure scenarios into chaos engineering or simulation efforts

Preparation (2)

Good preparation is key for cloud incident response, just as it is for responding to intrusions on-premises. As with on-premises systems, we want to assume a detection-oriented posture which means establishing the right visibility into cloud systems and networks (where possible). Your ability to generate this telemetry will depend upon the capabilities of your cloud service provider (and perhaps your account with that service provider), but strive for host visibility via log collection APIs or purpose-built cloud logging utilities such as AWS' CloudTrail. If available, generate network traffic records using VPC flows or virtualized host intrusion prevention systems. Segmentation is also an important concept in the cloud, though it is normally implemented in user roles versus controls at layers 2-4. Credential compromise is a major risk in cloud environments, so try to keep system or business lines separated by user access to limit damage should access credentials become exposed or compromised. Finally, ensure that your team is trained to deal with incident response in cloud-specific scenarios; build these scenarios into your training and exercises and document related procedures in your incident response plans.

MITRE ATT&CK Cloud Matrix¹

- A great way to self-check your capabilities
 - Prioritization based on intel, purple team testing to check!

Cloud Matrix

Initial Access	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Collection	Exfiltration	Impact
5 techniques	5 techniques	1 techniques	5 techniques	4 techniques	9 techniques	2 techniques	4 techniques	1 techniques	4 techniques
Drive-by Compromise	Account Manipulation (3)	Valid Accounts (2)	Impair Defenses (2) Modify Cloud Compute Infrastructure (4)	Brute Force (4) Steal Application Access Token Unused/Unsupported Cloud Regions Steal Web Session Cookie Unsecured Credentials (2)	Account Discovery (2) Cloud Infrastructure Discovery Cloud Service Dashboard Cloud Service Discovery Network Service Scanning	Internal Spearphishing Use Alternate Authentication Material (2)	Data from Cloud Storage Object Data from Information Repositories (2) Data Staged (1) Email Collection (2)	Transfer Data to Cloud Account	Defacement (1) Endpoint Denial of Service (3) Network Denial of Service (2) Resource Hijacking
Exploit Public-Facing Application	Create Account (1)								
Phishing (1)	Implant Container Image								
Trusted Relationship	Office Application Startup (6)								
Valid Accounts (2)	Valid Accounts (2)								

SANS

MGT551 | Building and Leading Security Operations Centers

95

MITRE ATT&CK Cloud Matrix

Beyond data collection, the team must be trained in the specific tactics of cloud attackers and know what an attack looks like so that they can be detected. The MITRE ATT&CK Cloud Matrix is a great source for this type of information. It even breaks tactics and techniques down based on CSPs, with specific detail for AWS, GCP, Azure, Office 365, Azure AD, and SaaS.

As always with MITRE ATT&CK, the matrix should be evaluated considering threat intelligence on your specific attackers and their preferred TTPs. Prioritization of techniques to test and prepare for based on intelligence and applicability to your environment can guide your SOCs preparation for cloud-based incident response. As we will discuss later in the course, it is *highly* advised you not just assume detections will work and the team will understand the data produced during cloud IR, a purple team and eventually red team test should be regularly scheduled to ensure you are where you think you are.

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

Identification (2)

- Most cloud detections will be driven by anomaly-based analytics and user account activity via log data; other sources of alerts:
 - Billing activity
 - Threat intelligence
 - Third party cloud security tools
 - CSP monitoring/notification
- Up-to-date inventory and configuration information, including public/routable IP addresses, user roles and access, and system function will make validation much easier
- Expect a high number of false positives, especially in environments where there is a lot of churn

Identification (2)

Detection in the cloud relies heavily on behavioral analytics and user account activity. Since you've already instrumented these items (right??), be sure to build them into your triage, validation, and tuning processes.

Having access to up-to-date inventory and configuration information will make validation much easier as the ephemeral nature of this infrastructure makes false positives likely. Your primary source of cloud telemetry will be system logs collected via cloud utility or logging API, but other detection sources may include:

- Billing activity – unusual or unexpected spikes in usage charges may indicate malicious activity or abuse
- Threat intelligence – awareness of specific interest in your service provider or your cloud infrastructure by a specific threat actor or campaign
- Third party cloud security tools
- Cloud service provider monitoring/notification

If you're just getting started in monitoring your cloud infrastructure, you are likely to encounter a lot of anomalies and false positives. Don't get frustrated! Work closely with your infrastructure support team(s) to better understand how the platforms are being utilized and where churn is likely as systems are torn down and rebuilt. Doing this challenging work early can help you baseline your environment more effectively and cut out the noise later in the process.

Containment and Eradication

- Remember containment considerations from earlier?
Becomes a bit more complicated now
- What is available depends on nature of your cloud implementation; incidents can occur in:
 - **Service domain** – heavily reliant on CSP's tools and capabilities, will impact your CSP account, permissions, billing, etc.
 - **Infrastructure domain** – relies on interaction at the OS level, potentially CSP APIs to gather data and make access changes
 - **Application domain** – relies on cloud tools and automated forensics capabilities; more about protecting data and access
- Use **ephemerality** to your advantage



Containment and Eradication

Of course, containment is a more challenging proposition when you only have control over a portion of the impacted system. In an entirely on-prem enterprise, the SOC has a variety of containment options from layer one (physical) all the way up to layer seven (application). Many of those lower-level responses are no longer available to you as a tenant on shared infrastructure. Understanding the containment options you *do* have based on your organization's cloud deployment model is an important part of your incident response planning. In the same way that you might conceptualize your internal network as various layers – network, host, application, etc. – you can conceptualize your cloud environment as a service domain, infrastructure domain, and application domain. This concept of incident domains is described in detail by Amazon here:

<https://docs.aws.amazon.com/whitepapers/latest/aws-security-incident-response-guide/incident-domains.html>.

Essentially, incidents can occur within each of these domains, and as with our on-prem “layers” we must have detections and response actions defined for each of them. Your **service domain** will be heavily dependent on which cloud service provider you have and what tools they have made available for monitoring your service account, permissions, costs, and other elements. Detection and response in the **infrastructure domain** will likely rely on the APIs your CSP exposes to query available data and make changes. Finally, the **application domain** deals primarily with software deployed to your cloud infrastructure. Detection and response in the application domain will probably leverage many of the same cloud-native toolset that your infrastructure teams use to analyze and deploy code. Keep in mind that one of the main advantages to using cloud infrastructure is its *portability* and *ephemerality* – characteristics you can use to your advantage. Work closely with your infrastructure teams and site reliability engineers to classify security incidents within the application and infrastructure domains as potential failure conditions that may be remediated by re-deploying or rolling back infrastructure. These may not be your first choices for incident remediation but they are options that may be available to you in this environment that aren't in your on-premises environment.

Incident Response in AWS

- Use firewalls and security groups
- Use managed IAM policies
- Assign policies to groups instead of users
- Enforce least-privilege/deny by default
- Robust suite of security monitoring tools and data sources:
 - AWS VPC flow logging
 - CloudTrail and S3 access logs
 - Purpose-build monitoring tools like GuardDuty, Detective, Security Hub, and Amazon Macie.
 - Custom monitors such as Route 53 health checks and CloudWatch alarms.
 - Optionally you can use CloudWatch agents to log Windows Events, Linux syslog logs, and application logs to Amazon CloudWatch

Incident Response in AWS

Amazon's web services infrastructure has a relatively robust set of capabilities for security monitoring and response, but like other elements of AWS it can quickly get complex. Start with the basics of firewalls and security groups for your AWS assets. Much like you would in an on-premises Active Directory environment, use policy templates to assign privileges to groups instead of users to minimize access creep and inadvertently broad access. Amazon has a rather large suite of security monitoring tools and data sources, including (but not limited to) VPC flow logging for network monitoring, log monitoring in CloudTrail and S3 access logs, purpose-built security tools like GuardDuty, Detective, and SecurityHub, and custom "monitors" like CloudWatch alarms. If you're so inclined, you can also log to Amazon CloudWatch the same way you would a SIEM using CloudWatch logging agents.

Amazon has published a comprehensive incident response guide at:

<https://docs.aws.amazon.com/whitepapers/latest/aws-security-incident-response-guide/aws-security-incident-response-guide.pdf>

Incident Response in Azure

- Monitor via Azure Security Center, Azure Sentinel, or export via CSV:
 - Security alerts and recommendations
 - Secure score (per subscription or per control)
 - Regulatory compliance data
- Network data available via Network Watcher
 - May be analyzed via Azure Traffic Analytics
- Hybrid cloud workload monitoring including servers, data, storage, containers and IoT with Azure Defender

Incident Response in Azure

As with many things Microsoft, your detection and response capabilities in Azure will be somewhat dependent on your investment in Microsoft's ecosystem. At a basic level, you can monitor Azure telemetry by streaming events to Azure Security Center or Azure Sentinel (if you're using those platforms), or by exporting it via CSV for ingestion into the monitoring platform of your choice. This telemetry includes security alerts and associated recommendations, scoring data, and compliance data for your Azure infrastructure. Network telemetry is available via Azure Network Watcher, which is designed to monitor assets in the infrastructure domain such as Virtual Machines, Virtual Networks, Application Gateways, Load balancers, etc. (but not for platform or web analytics data). Azure Defender provides endpoint-type security for cloud servers, data, storage, containers, and connected devices.

For more information on incident response in Azure, start here: <https://docs.microsoft.com/en-us/azure/security/benchmarks/security-control-incident-response>

Incident Response in GCP

- Cloud Logging for real-time log management and analysis
- Google Cloud Operations Suite (formerly StackDriver)
 - Cloud monitoring collects data from Google Cloud as well as AWS and many other sources
- Security Command Center
 - Asset inventory, discovery, search, and management
- Chronicle SIEM and Web App and API Protection (WAAP)

Incident Response in GCP

Most of Google's incident response capabilities for their cloud environment is heavily focused on what it refers to as "data incident response", and it relies heavily on a combination of Google's own internal monitoring and analytics. Potential detections include data from Google's Cloud Logging API, Google's Cloud Operations Suite (which can ingest data from a variety of different sources outside of GCP), and Google's Security Command Center which focuses on more compliance-oriented telemetry. Finally, Google provides a SIEM capability (for a price) in its Chronicle platform and protection and telemetry for web and API infrastructure in its Web App and API Protection (WAAP) product.

A more comprehensive White Paper detailing Google's cloud security capabilities and incident response can be found here: https://services.google.com/fh/files/misc/data_incident_response_2018.pdf

Cloud Incident Response Summary

- Differences from on-prem IR can be compounded depending on deployment model, vendors, service providers
- Make sure you know what you have and what telemetry is available (and useful)
- Train your incident leads and analysts on investigating and responding to incidents in the cloud
- Different approaches and tools for different CSPs

Cloud Incident Response Summary

Your telemetry and response capabilities in the cloud will be almost completely reliant on your cloud service provider, your licensing level, and the domain in which you operate. Make sure you know what your organization has and what telemetry is available in as many cloud domains as are relevant for the services you're using. A good first place to check for this information is accounting, where (hopefully) most of the billing for your cloud services is handled. Train your SOC – especially your incident leads – on investigating and responding to incidents in the cloud as these actions will likely come with their own tools and visibility. Recognize that there are different approaches and different capabilities from one CSP to the next, and you may not enjoy the same level of access or visibility in each one. In the next section, we will switch back to focusing on systems to which we *do* have full access as we discuss some of the most prevalent and effective incident response tools.

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

Planning, Preparation, and Review

- Incident Response Planning and Preparation

• Investigation

- *Exercise 4.1 – Investigation Quality Review*

IR Execution

- Identification, Containment, and Eradication

• Incident Response in the Cloud

• IR Tools

- *Exercise 4.2 – Planning Responses with RE&CT*

- Crisis Management and Continuous Improvement

- *Exercise 4.3 – Designing Tabletop Exercises*

- Recovery and Post-Incident

- Summary and Cyber42 – Day 4

This page intentionally left blank.

IR Tools Overview

- IR data acquisition is increasingly about live response due to large, distributed environments
- Quick access to key data at network and host layers is critical
- Host-based tools can be agent-based, or leverage existing utilities like WMI and PowerShell
- Cloud, mobile, virtualization technologies make this more complex
- You'll need the right tools and skillsets to acquire and interpret the data



SANS

MGT551 | Building and Leading Security Operations Centers 103

IR Tools Overview

Incident response capabilities can vary widely from one organization to the next. It might be part of a SOC function, a separate team within the organization, or be outsourced entirely to a third party. The IR function might include forensic capture and analysis, or it might be more oriented to restoring systems as quickly as possible. IR toolsets likewise can incorporate a wide spectrum of functionality from forensic capture to real-time monitoring. In general, incident response tools are increasingly focused on live data (that is, non-persistent and real-time data) in environments where cloud computing and distributed infrastructure make more resource-intensive activities like disk acquisition difficult. Whatever the tool or mechanism, getting quick access to key data like network communications, running processes, file listings, user actions, and other events at the host and network layers is required to fully scope and respond to an intrusion. At the network layer, this usually incorporates some combination of full packet capture and summary data. At the host layer, tools can be agent-based, or they may leverage existing utilities like Windows Management Instrumentation and PowerShell. Whatever your team structure and focus, incident response requires the right tools *and* the skills to operate those tools and interpret the data they produce. Given some of the unique challenges in data acquisition, chain of custody, and data analysis during incident response, this is probably a different (though perhaps overlapping) skillset than what you'll find in "front-line" security monitoring and analysis functions.

In the next few slides, we'll talk about some of the more prevalent tool types used in incident response and situations in which they generally come into play.

EDR

- Endpoint detection and response: "SIEM" for the host layer
- Great for forensic analysis and threat hunting
- Better investigative detail than A/V or HIDS
 - Example: suspicious file is identified and isolated, one-click search for that file's hash across the environment
- Most have active features that can terminate processes, implement network blocks, isolate files
- FireEye, Crowdstrike, MS Defender, Wazuh

Endpoint Detection and Response

Endpoint detection and response is a term first coined by Anton Chuvakin and defined as, “the tools primarily focused on detecting and investigating suspicious activities (and traces of such) other problems on hosts/endpoints.” Like some of the other terms we discuss in this section (namely, NDR and XDR, which we’ll get to, next), this term encompasses not only a set of tools but capabilities that may include multiple technologies and data sources. In this respect, you can think of EDR as a kind of Security Information and Event Management (SIEM) platform for the endpoint that incorporates system logs, memory, file analysis, intelligence, and other data into a robust detection engine. The common thread in EDR solutions, and where they tend to expand upon more traditional host-based controls like host intrusion detection and antivirus, is visibility. Unlike tools designed to identify specific threats or TTPs, EDR solutions tend to focus on providing rich telemetry and a wide range of active mitigations to help defenders identify and respond to all kinds of malicious or suspicious activities, regardless of how they show up in the data. FireEye HS, Crowdstrike Falcon, and Microsoft Defender are all examples of commercial EDR platforms. Wazuh is an open source EDR that includes intrusion detection, log analysis, file integrity monitoring, and active response capabilities among many others.

NDR

- What we used to call anomaly-based network intrusion detection, with some NSM sprinkled in
- Can be useful for large-scale data analysis, encrypted traffic monitoring
- Flow data and layer 7 metadata can be retained for a long time to support forensic analysis at network layer
- Darktrace, Vectra, ExtraHop, Fidelis

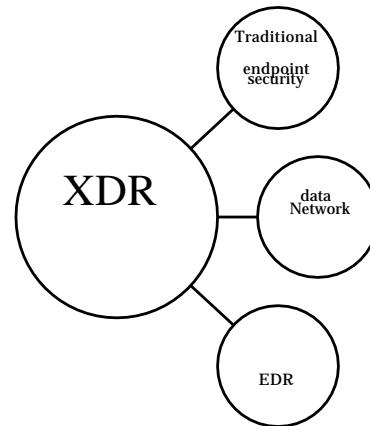
Network Detection and Response

Network Detection Response, or NDR, is a class of security technologies that is defined by Gartner as “primarily use non-signature-based techniques (for example, machine learning or other analytical techniques) to detect suspicious traffic on enterprise networks. These solutions are generally very similar to older network security monitoring platforms that produce flow, statistical, and alert data based on network traffic capture. These platforms often leverage automated statistical analysis techniques to identify anomalies and potentially malicious behaviors, and many now incorporate automation and orchestration features to facilitate a rapid response to that activity.

Awake Security, Blue Hexagon, Bricata, Cisco, Corelight, Darktrace, ExtraHop, and Fidelis Cybersecurity are examples of commercial NDR vendors.

XDR

- Gartner defines as “*a SaaS-based, vendor-specific, security threat detection and incident response tool that natively integrates multiple security products into a cohesive security operations system that unifies all licensed components*”
- Collects and correlates data from servers, email, cloud workloads, and endpoints
- Intended to improve upon EDR by incorporating features such as cloud and network data sources
- Cisco, Palo Alto, Microsoft, FireEye, VMWare, most major A/V vendors



Extended Detection and Response

Extended Detection and Response, or XDR, is a term coined by Nir Zuk of Palo Alto Networks in 2018[1]. Gartner defines XDR as “*a SaaS-based, vendor-specific, security threat detection and incident response tool that natively integrates multiple security products into a cohesive security operations system that unifies all licensed components*.” XDR expands upon EDR solutions by providing visibility across an organization's endpoints, network, and cloud workloads. “What a minute,” you might say, “isn't that what I have a SIEM for?”. XDR vendors differentiate their solutions by remaining focused on the endpoint, incorporating traditional controls like anti-malware and file integrity monitoring, and enhancing their telemetry with other data sources. The advantage of consolidating this data at the host layer from an incident response perspective is more effective triage and faster, more surgical response actions. Many large software and hardware vendors, such as Microsoft, VMWare, Palo Alto Networks, and Cisco claim XDR products, and traditional enterprise security companies including FireEye, TrendMicro, and McAfee have added XDR products or services to their overall security platforms.

[1] Kerravala, Zeus (2018-09-06). "[EDR is dead! Long live XDR!](#)". InsiderPro. Retrieved 2020-10-26.

Guarding Your Credentials: The Threat

WARNING: Logging in to a potentially compromised system risks making things *much* worse



- Credentials may be stolen and used for lateral movement
- **Interactive** logons
 - PSEXEC, RunAs, RDP, are most dangerous
- *Any* login can be potentially key logged
- Attacks may steal password, hash, or Kerberos TGT
 - Allows them to impersonate the security team on additional systems!

Guarding Your Credentials: The Threat

Although it may be tempting for analysts to immediately remote into a machine to take control, probe, and see what's happening during an incident, this can be a *terrible* idea if you aren't extremely careful.

In Windows, specifically, many of the ways that are available for remote administration leave credentials in memory that can be stolen by adversaries who are also present on the machine. There are different ways of logging in—interactive and noninteractive. Interactive logins are mostly what they sound like: They are the login types where you will interactively use the machine on the remote end—this includes sitting at the keyboard as a normal login, RDP, VNC, PSEXEC, RunAs logins, and more. Noninteractive logins are logins such as mapping a drive on a remote file share. It is the interactive logins you must be careful with as these are the ones that typically store the credential used to login in the memory of the remote machine in a way that attackers can steal it. If they do get a hold of your analyst's password hash, Kerberos Ticket Granting Ticket, or even their plaintext password (which is sometimes possible, especially on pre-Windows 10 machines), you can bet they'll use it. This would enable attackers to easily pivot to their next point of interest within your organization, leaving the SOC with an even bigger and more embarrassing problem.

Guarding Your Credentials: The Fix

Credential Protection Options for remote access:

- **RDP Restricted Admin Mode¹**
 - User must be in local administrator group of remote system
 - Enables pass-the-hash logins, but this may not matter*
 - Most secure method, unnecessary with best practice AD design
- **Windows Defender Remote Credential Guard²**
 - No pass-the-hash, uses Kerberos, RDP group membership only
 - Pass-the-ticket attack exposure during ticket lifetime
- **LAPS – Use a unique local admin password for login**
- **PowerShell Remoting**
- **Access facilitated via installed agent (EDR for example)**



Guarding Your Credentials: The Fix

How do we connect to a potentially infected system in a secure way that will not endanger our credentials?

While this is a very technically deep topic with many caveats, there are some methods for doing so that are safe (or safer) in most situations. Here are some of the options:

- For any system with an up-to-date version of Windows 7 or Server 2008 R2, Restricted Admin Mode is available to connect to remote systems over RDP in a way that will not make the credentials available in the remote system's memory. While this is great, it does have the minor downside of enabling pass-the-hash style logins via RDP and requires users be in the Local Administrators group of the remote machine. While the pass-the-hash issue sounds bad, it's not something attackers couldn't do over SMB anyway in many scenarios (if the port is available) and still prevents attackers from moving beyond that single system itself. There is a valid argument to be made that in organizations where strict Active Directory tiering has been implemented this would not be necessary in the first place, but many organizations aren't quite there yet. The restricted admin mode feature is disabled by default but should be utilized where appropriate and enforced by Group Policy where possible, especially for environments where strict best-practice Active Directory tiering hasn't been implemented.
- For a Windows 10 version 1607 and Server 2016, another option is Windows Defender Remote Credential Guard which allows you to safely connect to remote systems using your normal Single Sign-On credentials. This method prevents pass-the-hash attacks and only requires users be in the Remote Desktop Users group instead of local admin, but only supports the credentials used for single sign-on as a connection method and is compatible with Kerberos only. While this method is an improvement, service tickets granted to the user who connects are still potentially vulnerable to theft and reuse during the lifetime of the ticket. Refer to the documentation² for additional technical details.
- All organizations should be using unique local administrator passwords for every single machine in the environment. To facilitate doing this in a manageable way, Microsoft has created LAPS—the Local Admin Password Solution³ tool. If you connect to a machine with credentials that are only valid on that single machine, then damage exposure is limited. If the system is already compromised, you can assume the attacker has those credentials anyway, and if they don't, they won't help them achieve any additional access anyway.

- Where PowerShell Remoting is enabled, connection via this method uses Kerberos and WinRM with a noninteractive login that will not expose credentials⁴. PowerShell remoting is an incredibly powerful feature for incident response allowing connection to many different machines for mass querying of data and more.
- If you have an EDR agent, chances are it can access the data and take the actions you need without needing to use a traditional login at all. This is likely the best option for those who have this capability. In this case, the agent has been installed with the permissions required to take the action and all actions are requested centrally via the software, keeping losing credentials out of the realm of concern in most cases.

Once you have collected the items of interest, if there was malware involved, there's a good chance you'll now have a sample of it that needs to be analyzed. Let's take a quick second to discuss malware analysis since this is a task that, when done thoroughly, can significantly fast track the scoping of an incident, but only if it is done efficiently and correctly.

1<https://social.technet.microsoft.com/wiki/contents/articles/32905.remote-desktop-services-enable-restricted-admin-mode.aspx>

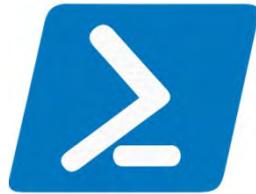
2<https://docs.microsoft.com/en-us/windows/security/identity-protection/remote-credential-guard#comparing-windows-defender-remote-credential-guard-with-other-remote-desktop-connection-options>

3 <https://www.microsoft.com/en-us/download/details.aspx?id=46899>

4 <https://www.sans.org/blog/the-power-of-powershell-remoting/>

PowerShell

- Remoting capability via WinRM
- Great for data collection, analysis, mitigation actions
- Do things like:
 - Check for and terminate malicious process
 - Remove persistence mechanisms like scheduled tasks, registry run keys, services
 - Check for files, get hashes



PowerShell

The remote management and command execution capabilities provided by PowerShell (thanks to Windows Remote Management) can make for a robust rapid response capability without the need for third party tools. In his excellent blog post on PowerShell Remoting and Incident Response[1], responder Matthew Green describes six advantages in using PowerShell for incident response:

1.Data collection: PowerShell has access to WMI, COM, .NET as well as to the Windows API, which means you can grab all sorts of useful data in support of an investigation. This includes files on disk, registry artifacts, log and configuration data, volatile processes, and network information.

2.Analysis: PowerShell is an object-based scripting language that you can use for statistical analysis in support of live response, baseline comparisons or timeline analysis.

3.Supportability: PowerShell is just one consumer of the Windows Remote Management service, which is heavily entrenched as a Windows administrative tool and unlikely to go anywhere anytime soon.

4.Performance: Since you can run collection and analysis actions on each target machine in parallel, PowerShell is great for fast remote operations at large scale.

5.Agentless: Since it's using Windows' built-in WinRM capability, PowerShell remoting does not require any additional software or agents to be running on the target machine(s).

6.Cost: Provided you have the skillsets to use and customize PowerShell remoting for incident response use cases, its low cost makes PowerShell an *extremely* attractive option. Windows Remote Management has been available since PowerShell 2.0 and Windows 7 through to the most recent incarnation in Windows Management Framework (WMF) 5.1. WinRM is enabled by default in Windows Server 2012 and 2016 but, as you'll see below, simple to enable back to Windows 7 running PowerShell 2.0.

[1] https://mgreen27.github.io/posts/2017/01/12/PowerShell_Remoting_IR.html

PSEExec

- Part of Microsoft's Sysinternals suite
- Provides telnet-like remote command execution, returns results to local console
- Used by admins, responders, and (unfortunately) adversaries
- Can be used in conjunction with other third party scripts and utilities
- It's free!
- *Warning: May leave credentials open to theft*

PSEExec

PSEExec is part of Microsoft's Sysinternals toolsuite that performs remote process execution. It's commonly used by administrators, incident responders, and adversaries alike for remote script execution, and it's immensely popular because it provides this capability reliably at a low cost (free!). In incident response scenarios, PSEExec is often used in combination with other third-party scripts and utilities to run those utilities on target machines and get the resulting data – for example, dumping the machine's volatile memory for offline analysis. While PSEExec may be convenient, be aware that it is one of the tools that may leave credentials open to theft. If you choose to use PSEExec, be sure to check if your version of Windows, and the method in which you use PSEExec leaves your password in a vulnerable state. When in doubt, use an alternative known safe method.

WMI

- Provides a flexible query language to query WMI object instances
- Almost all actions in Windows generate a WMI event – robust data source for investigation and real-time alerting
- Also, very popular among adversaries in all attack stages



```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\marka> wmic:root\cli!list startup
Caption
Command
HPEU_Host_Launcher
C:\System.sav\util\HpeuhostLauncher.exe
OneDrive
C:\Program Files\OneDrive\OneDrive.exe
EADP
C:\Program Files\OneDrive\OneDrive\OneDrive.exe
com.squirrel.Teams.Teams
C:\Users\marka\AppData\Local\Microsoft\Teams\Up
OPENVPN-GUI
C:\Program Files\OpenVPN\bin\openvpn-gui.exe
Discord
C:\Users\marka\AppData\Local\Discord\Update.exe
Gitter
C:\Users\marka\AppData\Local\Programs\Gitter\G
GoogleChromeAutoLaunch_0547ED9A09B3858A36B4D14FF68A4447
C:\Program Files (x86)\Google\Chrome\Application\47.0.2526.106\chrome.exe
Spooler
C:\Windows\system32\spooler\spooler.exe
CiscoMeetingDemon
C:\Users\marka\AppData\local\WebEx\ciscowebex
SecurityHealth
C:\Windows\System32\Security\HealthSystray.exe
RtkAudService
C:\Windows\System32\RtkAudService64.exe" -bac
ExtTiltPenSrvc
C:\Program Files\ELAN\ExtTiltPen\ExtTiltPenAgent
LogiPresentation
C:\Program Files\Logitech\LogiPresentation\Logi
WindowsDefender
C:\Program Files\Windows Defender\MSASCui.exe

wmic:root\cli>
```

Windows Management Instrumentation (WMI)

Windows Management Instrumentation enables a user to query WMI object instances, classes, and namespaces using Windows Query Language (WQL), which is similar in syntax to database query languages. Many defensive utilities like Sysinternals' Autoruns tool rely on WMI for remote data collection and active response actions like file removal. Because almost all events in the Windows operating system generates a WMI event, it's an extremely useful source of data for tracking attacker actions. One of the most powerful features of WMI from an attacker's or defender's perspective is the WMI event, which can be used to respond to nearly any operating system event and generate a real-time notification.

OSQuery

- Operating system framework that organizes OS artifacts into SQL tables
- Enables fast, flexible searching across a large set of endpoints
- Functionality can be extended via API, data parsed and presented via SIEM and other tools

```
SELECT DISTINCT processes.name, listening_ports.port, processes.pid  
FROM listening_ports JOIN processes USING (pid)  
WHERE listening_ports.address = '0.0.0.0';
```

OSQuery

OSQuery is an operating system framework, open sourced by Facebook, which exposes an operating system as a relational database. As an incident responder, this allows you to access operating system data via SQL queries and organizes forensic artifacts like running processes, kernel modules, network connections, browser plugins, and more into SQL tables. The advantage of using OSQuery for incident response, in addition to its open data model and fast performance, is that it has a much lower bar for learning search syntax than other forensic and endpoint analysis tools since it uses SQLite as its query engine. OSQuery is cross-platform and runs on Ubuntu, CentOS and Mac OSX as well as Windows. It also includes an API which allows you to develop custom extensions like Ben Bornholm's CommunityID app¹.

To manage deployments of OSQuery agents, or "fleets", the Kolide Fleet control server was developed as an open source project. Unfortunately, the Fleet project has been retired in lieu of focus on Kolide's commercial endpoint security product. The files and documentation for Fleet is still available here: <https://github.com/kolide/fleet>.

[1] <https://holdmybeersecurity.com/2020/02/11/creating-my-first-osquery-extension-to-generate-communityids-with-osquery-python-on-windows/>

Kansa

- **PowerShell-based incident response framework:**
 - Written by Dave Hull (<https://github.com/davehull/Kansa>)
- **Modular scripts that can be run remotely via SOAR orchestration**
- Collects all the previously mentioned system info
- Returns information in well-formatted PowerShell objects
- Modules for:
 - Netstat, autoruns, DNS cache, ARP, processes, services, users, prefetch, userassist, WMI, handles, and other advanced pieces of data

Kansa

One of the most commonly referenced tools for data gathering for live incident response is Dave Hull's **Kansa** framework.¹ This is a live incident response data-gathering tool built completely out of PowerShell scripts that can be separately deployed by a SOAR engine to the device in question, pulling back the output for evidence. It even supports JSON output that could be pulled directly into your SIEM or ticketing solution with ease! There is no need to reinvent the wheel on many of these actions; your SOAR tool should simply be giving you a platform to easily chain a bunch of automated events together and to take action based on their output.

Although Kansa is a great project, it does require that PowerShell remoting access is available for the remote machine. If you cannot do PowerShell remoting, the **CIMsweep** project² is an alternative that uses Windows WMI commands instead.

1 <https://github.com/davehull/Kansa>

2 <https://github.com/PowerShellMafia/CimSweep>

Velociraptor

- Endpoint collection tool inspired by Google's GRR and OSquery
- Uses SQL-like query language called VQL
- Optimized for performance even in very large environments
- Artifacts = saved queries
- Many forensic analysis modules included



Velociraptor

Velociraptor is an endpoint collection tool developed by Michel Cohen, who also served as the lead developer for Google's GRR endpoint forensics tool. Velociraptor was created to simplify the GRR architecture and provide a robust query language (VQL) and open source collection framework to facilitate rapid response and forensic data gathering. Like OSQuery, Velociraptor has a flexible API and was built for high performance in environments with large numbers of endpoints. To simplify forensic data gathering, complex queries are templated in "artifacts" - text files, written in YAML, that include the raw VQL human readable descriptions, and parameters allowing query customization. These artifacts can be combined with a variety of included low-level modules for parsing prefetch files, raw registry access (for AMCache analysis), ESE database parsing (facilitating SRUM database forensics and Internet Explorer history analysis), SQLite parsing (for Chrome and Firefox history) and much more to create an extremely powerful forensic analysis capability. It can be deployed as a standalone (internally-hosted) architecture, in the cloud, or in non-persistent triage mode.

You can read more about Velociraptor here: <https://velociraptor.velocidex.com/velociraptor-e48a47e0317d> and download it here: <https://gitlab.com/velocidex/velociraptor>

OSSEC / Wazuh

- OSSEC is an open source, multi-platform host intrusion detection system
 - Can also ingest syslog events from network devices
- Wazuh is an open source EDR that began with a fork of OSSEC & added:
 - OpenSCAP support
 - New WUI on top of Kibana 5 and integrated with the RESTful API to monitor configuration of the manager, rules and status of the agents.
 - Improved log analysis and FIM capabilities.
 - Modules manager that will allow future integration of other tools
 - Can read host logs, do FIM, compliance monitoring, active countermeasures



OSSEC/Wazuh

OSSEC is an open source Host based Intrusion Detection System that performs log analysis, integrity checking, Windows registry monitoring, rootkit detection, real-time alerting and active response. It runs on most operating systems, including Linux, OpenBSD, FreeBSD, Mac OS X, Solaris and Windows, and interprets system events via various rules and decoders. Wazuh is another open source project, built on a fork of OSSEC, that provides capabilities more akin to modern EDR solutions. More specifically, Wazuh includes the same functionality as OSSEC plus openSCAP integration, the Wazuh graphical user interface (WUI), native integration with AWS and Azure, improved log analysis and FIM capabilities, compliance monitoring, and a module manager to facilitate integration with other tools.

KAPE

- Kroll's Artifact Parser and Extractor, created by Eric Zimmerman
- Enables rapid collection and processing of forensic artifacts
- Two main functions:
 - File collection
 - Processing using one or more utilities
- Comes with default set of targets and modules for common forensic operations

KAPE

Kroll Artifact Parser and Extractor (KAPE) is a triage program that will target a device or storage location, pull relevant forensic artifacts, and parse them for analysis. Because of its speed, KAPE allows investigators to find and prioritize the more critical systems to their case. Additionally, KAPE can be used to collect the most critical artifacts prior to the start of the imaging process. While the imaging completes, the data generated by KAPE can be reviewed for leads, building timelines, etc.

KAPE serves two primary functions: 1) collect files and 2) process collected files with one or more programs. By itself, KAPE does not do anything in relation to either of these functions; rather, they are achieved by reading configuration files on the fly and based on the contents of these files, collecting and processing files. This makes KAPE very extensible in adding or extending functionality.

KAPE modules deal primarily in *targets* and *modules*. Targets are basically collections of file and directory specifications for useful artifacts in a forensic investigation. Modules are defined using simple properties and are used to run programs for additional processing and analysis. These programs can target anything, including files collected via the target capabilities as well as any other kinds of programs you may want to run on a system from a live response perspective.

KAPE is free to download and use, but it requires an enterprise license when used on a third-party network and/or as part of a paid engagement.

Memory Forensics

- Sources:
 - live memory, VM snapshots, RAM crash dumps, Windows hibernation files
- Dumping tools
 - Windows - Winpmem, Moonsols dumpit, Redline, .vmmem files
 - MacOS - Osxpmem
 - Linux – Linpmem
 - Containers - Sysdig
- Analysis
 - Volatility
 - Redline
 - Rekall
 - Plugins for detection/investigation functionality



Memory Forensics

Random Access Memory (RAM), or volatile memory, has become an extremely important element of incident response. It's here that malware authors can hide from disk-oriented detection mechanisms like Antivirus and HIDS, using system binaries or memory injection to execute code without writing persistent artifacts to disk for blue teamers to find. The most common source of memory data is the live data of a running system, but there are other sources we can use in incident response as well. Virtual machine snapshots may contain memory data, though you may need special tools to analyze them. Crash dumps resulting from system problems may contain useful information, as can Windows hibernation files, where RAM contents are stored when a system goes into sleep mode.

Tools for memory analysis generally have two types of features: acquisition or dumping of volatile memory, and the actual analysis of its contents. Regardless of your tool set, it's vitally important to capture memory *first*; this data will be lost on system reboot, and time spent analyzing the device without capturing memory will result in a less-than-pristine memory image as your actions and normal system activities create additional artifacts there.

On the analysis side, some of the more popular utilities are Volatility, a fork of Volatility called Rekall, and Redline. Whatever tool you use for memory analysis, the goal is to identify anomalies among what otherwise appears to be normal system activity. Both Volatility and Rekall support the use of various plugins to make this process easier – for example, there are plugins to generate process lists and detections for known malware from a memory image.

Malware Analysis (2)

- Online analysis
 - VirusTotal, CAPE, Joe Sandbox, hybrid-analysis
 - Remember OPSEC!
- Offline analysis
 - Cuckoo, Sandboxie
- Static analysis
 - YARA, FLOSS
- Lots of these included in forensic live CD distros

Malware Analysis (2)

Here are some of the most widely-used tools for conducting static and dynamic malware analysis. Online services include VirusTotal, JoeSandbox, and Hybrid Analysis. Remember your OPSEC when using these online platforms! Use the search functions to avoid tipping off attackers that you have discovered something made specially to target your organization. Also remember that these online services are hosted and managed in various countries, so check with your acquisitions and/or legal team before signing contracts to pay for these services. Where offline and/or on-prem analysis is concerned, Cuckoo sandbox and Sandboxie are generally the most common choices for Blue Teams. Finally, YARA and FireEye's FLOSS (FireEye Labs Obfuscated String Solver) are extremely popular free utilities for static analysis. Many of these tools are included in forensic live distributions, which we'll talk about next.

SIFT

- SIFT Workstation is a group of free open-source incident response and forensic tools built into a Linux-based VM appliance
- Highly customizable, contains more than 200 tools and plug-ins
- Includes utilities for file system/registry/memory analysis, network investigation, live response, and more

SIFT

SIFT Workstation is a powerful collection of tools for examining forensic artifacts related to file system, registry, memory, and network investigations. First released by forensics expert and SANS Fellow Rob Lee in 2007, SIFT is freely available for download as a virtual machine (VM). The appliance can run on Linux natively or any Windows installation that includes the Ubuntu subsystem, and it is extremely customizable – users can string together various commands and utilities for a wide variety of use cases.

You can download SIFT Workstation from here: <https://digital-forensics.sans.org/community/downloads>

FLARE

- Open sourced, customizable Windows-based live distribution for incident response and forensic analysis
- Includes debuggers, disassemblers, decompilers, static and dynamic analysis utilities, network analysis and manipulation, web assessment, exploitation, vulnerability assessment applications, and many others



SANS

MGT551 | Building and Leading Security Operations Centers

121

FLARE

FireEye's FLARE virtual machine, first released in 2017, has become a favorite for incident responders, malware reversers, forensic investigators, and researchers. This Windows-based live distribution comes with a variety of analysis tools and a highly automated installation process.

You can download FLARE from here: <https://github.com/fireeye/flare-vm>

REMnux

- **A Linux Toolkit for malware analysis**
 - Your one-stop-shop for analyst malware reversing needs
 - Built for static and dynamic analysis, fake internet services and more
- All common malware reverse engineering tools, ready to go
- Maintained by SANS FOR610 author Lenny Zeltser
- A great addition to analyst workstation setups
- **Suggested setup**
 - Host – Linux-based operating system
 - Virtual machines: REMnux + Windows, and more



REMnux

An incredibly useful Linux distro as well as a huge time-saver for any team looking to get into malware analysis and reverse engineering is the REMnux distribution (remnux.org). REMnux is a free Linux distro with all the commonly used, Linux-based malware analysis tools installed and ready to go and serves as an amazing jumping off point for analysts who need to quickly take apart a file of nearly any type, dynamically reverse-engineer code, perform memory forensics, explore packet captures, analyze malicious documents and more.

The easiest way for analysts to get started using REMnux is to download the virtual appliance and simply load it into their virtual machine software of choice. An example set up for a malware analysis workstation for an analyst involving REMnux might be a separate (not company network attached) PC running Linux as a host operating system (to reduce the chance of infection), with virtual machines for analysis like REMnux running alongside Windows machines that can be the victim operating system for running samples. In this setup, snapshots can be used to take a clean baseline, load samples into one or both virtual machines, perform the analysis, then reset them to clean when complete, all while keeping the whole machine safe and separate from the corporate network.

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

Additional IR Team Training and Reference Material

In-depth Training for incident responders

- **Immediate** training options

- **SEC504:** Hacker Tools, Techniques, Exploits, and Incident (GCIH)
- **SEC503:** Intrusion Detection In-Depth

- **Intermediate** specific skills

- **FOR500:** Windows Forensic Analysis
- **FOR518:** Mac and iOS Forensic Analysis and Incident Response
- **FOR508:** Advanced Incident Response, Threat Hunting, and Digital Forensics (GCFA)
- **FOR572:** Advanced Network Forensics – Threat Hunting, Analysis and IR (GNFA)
- **FOR610:** Reverse-Engineering Malware (GREM)

- Books

- *Applied Incident Response* – Steven Anson
- *Blue Team Handbook: IR Edition* – Don Murdoch
- *Blue Team Field Manual* – Alan J. White



SANS

MGT551 | Building and Leading Security Operations Centers

123

Additional IR Team Training and Reference Material

For those looking to staff up a brand-new incident response and forensics capability within their SOC or hone the skills they have, here are some training resources and references guides for the team to keep close at hand.

Immediate training wise there is SANS SEC504 and SEC503. These two courses are some of the most popular and oldest courses at SANS. SEC504 focus on incident handling and understanding the attacker perspective to make students better defenders, SEC503 focuses on teaching defenders to dive deep into network traffic to help those who may need to scrutinize captured traffic from affected systems.

Intermediate level, specific skills training is also available for nearly any operating system and cloud platforms as well through the SANS Forensics and Cloud curriculum course offerings. Linux, Mac, Windows, Network, Cloud, malware analysis and more are all covered in their own 6-day courses going into even further depth for those doing true forensics.

Books

- If you'd like a reference guide from a top-tier incident responder on the tools and processes that an incident response team should follow, check out the book "Applied Incident Response"¹ by SANS Certified Instructor Steve Anson. The book does a great job of covering the technical details of which tools to use and when and is backed up with the knowledge sourced from years of experience in the trenches.
- For those looking for a quick reference manual to use in the heat of battle, the first volume of the blue team handbook and blue team field manual

[1] <https://www.amazon.com/Applied-Incident-Response-Steve-Anson/dp/1119560268>

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

Planning, Preparation, and Review

- Incident Response Planning and Preparation
- Investigation
 - *Exercise 4.1 – Investigation Quality Review*
- IR Execution
 - Identification, Containment, and Eradication
 - Incident Response in the Cloud
 - IR Tools
 - *Exercise 4.2 – Planning Responses with RE&CT*
 - Crisis Management and Continuous Improvement
 - *Exercise 4.3 – Designing Tabletop Exercises*
 - Recovery and Post-Incident
 - Summary and Cyber42 – Day 4



This page intentionally left blank.

EXERCISE 4.2

Exercise 4.2: **Planning Responses with RE&CT**

OBJECTIVES

- Identify potential response actions based on threat model and detection planning
- Develop incident response playbooks for high-impact incidents
- Document response capabilities using RE&CT Navigator
- Review RE&CT artifacts and templates



Exercise 4.2: Creating, Classifying, and Communicating Your Metrics

Please go to Exercise 4.2 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

Planning, Preparation, and Review

- Incident Response Planning and Preparation

- Investigation

- *Exercise 4.1 – Investigation Quality Review*

IR Execution

- Identification, Containment, and Eradication

- Incident Response in the Cloud

- IR Tools

- *Exercise 4.2 – Planning Responses with RE&CT*

• Crisis Management and Continuous Improvement

- *Exercise 4.3 – Designing Tabletop Exercises*

- Recovery and Post-Incident

- Summary and Cyber42 – Day 4

This page intentionally left blank.

Breach Happens

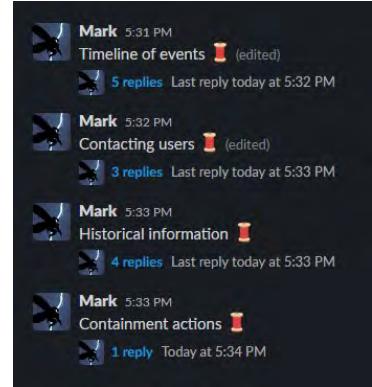
- While having zero cyber incidents is the ideal, they are becoming much more routine
- Doesn't let you off the hook
- The *response* can still be horribly fumbled
- If an incident occurs, you still must still...
 - Show that you were/are using best practice (data encryption, password hashing, etc.)
 - Be transparent in what happened - what was taken, and how
 - Be timely and complete in your announcement
 - Discuss what you are doing to handle it, prevent it happening again, and how you will fix the damage that did occur

Breach Happens

As much as we'd all like to imagine we will never be in this position, in all likelihood at some point in your career you will find yourself working an incident in which customers, law enforcement, or even the media may be notified. These extra competing pressures further complicating the situation and, based on the long history of both good and bad breach responses, can either inspire trust from customers, or make things *much* worse.

Real-Time Collaboration

- Documentation during an incident
- Consider OPSEC & data privacy
- Should be archivable
- Accessible to all participants
- Information should be organized, so new participants can get to the info they need



*Threaded Slack comments,
organized by activity*

Real-Time Collaboration

We'll get to the incident report in a moment, but during the response we'll need a place to store findings, actions taken, and other notes where other team members can see them. Case notes in a ticketing system may be sufficient for this, but it can also be handy to leverage existing real-time collaboration platforms like chat or online documents for capturing this information. The team will need some repository or communication channel that can be quickly stood up and taken down following an incident, with features to organize actions and notes so that newcomers can quickly find the information they need, and the tool(s) must be accessible to all key participants during the response effort.

Some examples might be:

- Threaded comments and/or custom channels in Slack or another chat platform (as shown in the screenshot above)
- Online office documents
- Incident management system

Although chat and online office documents may provide the best way to allow access for participants not on the security team, there may also be OPSEC or privacy issues in sharing potentially sensitive incident details in these channels. Whatever solution you use, text should be archivable or otherwise available for reference when it comes time to write the incident report.

Case Study: 2017 Equifax Breach

- **Summary:** Millions of US citizens credit data was stolen due to unpatched vuln. on a web server that led to a data breach
- **Response issues:**
 - Six weeks from discovery to notification
 - Different statements of impact on mobile vs. desktop website¹
 - Incident info website was broken, lack of info, also had certificate errors
 - Equifax offered their own products - "CreditLock" and "TrustedID Premier" subscription monitoring services as solutions to the breach
 - "Check if you're affected" site returned standard response to both real and fake info
 - Equifax Twitter account accidentally referred users to phishing site
- **To make it worse - Equifax executives sell millions of dollars of stock before breach announcement, two were sentenced, one is sent to prison for four months for insider trading^{2,3}**

Case Study: 2017 Equifax Breach

In 2017, Equifax servers were breached by advanced threat actors leveraging a webserver vulnerability (Apache Struts CVE 2017-5638) which had not been patched. The attackers were able to steal personal data on approximately 145M US Consumers, as well as hundreds of thousands UK and Canadian citizens. The attack itself started in July and was announced to the public in September. As detailed on the slides above, there were a number of preventable mistakes that contributed to making the situation worse for Equifax than it needed to be, including things like confusing messaging, poor communication with affected consumers, broken breach info websites and more. A potential takeaway here is to make sure everyone is clear on what channels are being used for communication and what is being communicated before information is posted.

As a side note, while the breach situation was already far from ideal, matters were made worse when some executives at Equifax used the information about the breach to sell stock before the announcement was made. Here's an excerpt from the US Department of Justice sentencing press release on then CIO of Equifax Jun Ying, who was one of those executives that was accused of, and plead guilty to insider trading:

"On Friday, August 25, 2017, Ying texted a co-worker that the breach they were working on "sounds bad. We may be the one breached." The following Monday, Ying conducted web searches on the impact of Experian's 2015 data breach on its stock price. Later that morning, Ying exercised all of his stock options, resulting in him receiving 6,815 shares of Equifax stock, which he then sold. He received proceeds of over \$950,000, and realized a gain of over \$480,000, thereby avoiding a loss of over \$117,000. On September 7, 2017, Equifax publicly announced its data breach, which resulted in its stock price falling."³

While being CIO during a data breach surely isn't a situation anyone wants to find themselves in, this move landed Ying a completely avoidable four months in prison, a year of supervised release, restitution charges of \$117,000, a \$55,000 fine, and untold additional company reputation damage.

1 <https://krebsonsecurity.com/2017/09/equifax-breach-response-turns-dumpster-fire/>

2<https://www.theverge.com/2019/6/29/20056655/jun-ying-equifax-breach-jail-time-insider-trading-department-of-justice>

3 <https://www.justice.gov/usao-nwga/pr/former-equifax-employee-sentenced-insider-trading>

Case Study: 2019 Wipro Breach

Summary: Attackers gain leverage Wipro systems to attack their customers by leveraging their privilege access to their networks

- 15-Apr-2019: Brian Krebs leaked info about potential breach from multiple anonymous sources, publishes blog post¹
 - Wipro requests several days to investigate, doesn't address concerns in the response
- 16-Apr-2019: COO claims errors in Brian's story in an investor call
 - Says it was only a few employees that were phished, and was already handled
 - Brian phones in, asks details on where he was wrong, receives a non-answer¹
 - Later Brian is told a "zero-day" was used, no details on product or zero-day are given to back up this claim
- 17-Apr-2019: Wipro admits phishing incident led to hack by 'state-sponsored actor'
 - Claims they came from their own team when they actually came from an affected customer who found the attack `_(ツ)_/``

Case Study: Wipro 2019

Starting around March of 2019, Wipro (a large IT outsourcing firm) experienced a widely published cyber attack. The attack is reported to have involved phishing of 100's employees and using an otherwise legitimate remote access tool called ScreenConnect to remotely control Wipro systems¹. While that level of access would have been bad enough, what the attackers were truly after were Wipro's customers. Leveraging ScreenConnect and pre-created lookalike domains for command and control such as `(ns3.microsoftonline-secure-login[.]com)`, attacks leveraged Wipro's privileged access to customer environments to access customer networks and systems. While the attack and its effects were no-doubt high impact, for this discussion, we're focusing on handling of the breach, which was fumbled in several preventable ways.

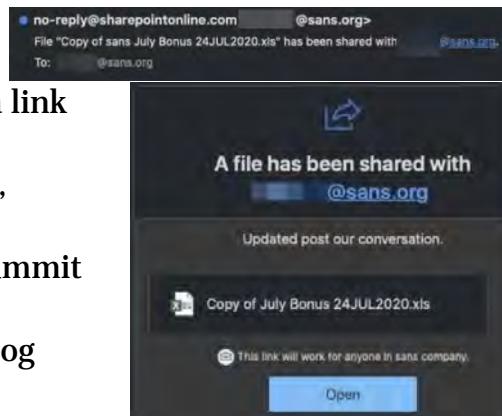
Take a look at the details above, if your organization were potentially affected by this breach, how would you feel about this response? What if you were a potential customer evaluating Wipro for services in the future, would it influence your decision? How do you think the breach response may have affected Wipro's reputation within the industry?

[1] <https://krebsonsecurity.com/2019/04/wipro-intruders-targeted-other-major-it-firms/#more-47453>

Case Study: SANS 2020 Data Breach

Incident Timeline:

- **24 July 2020** - Successful phishing email with a link led to a malicious Office 365 OAuth app
- User was tricked into consenting for inbox access, attackers set up auto-forwarding rule for email
- Some email to that employee contained PII on Summit attendees
- **6 August 2020** – SANS discovers app through log review for suspicious activity
- **Key findings:**
 - This is an increasingly common "**consent**" attack
 - No malware, no credential exposure, and bypasses MFA
 - Was *not* a targeted attack (based on employee list and other orgs with similar incidents with highly similar IOCs)



SANS

MGT551 | Building and Leading Security Operations Centers 131

Case Study: SANS 2020 Data Breach¹

As proof that no company is safe from attack, even those that know the most about them, let's discuss the mid 2020 data breach that occurred at SANS. While this breach wasn't nearly at the scale of the other case studies, it does give us an opportunity to discuss the way it was handled, and what we hope was acceptable response given the situation.

Here's a brief summary of the incident:

- **24 Jul 2020¹** – A phishing email with a malicious link was sent to 17 employees. One non-IT employee clicked on the link, which took them to a site where they were socially engineered to give the attacker's malicious Office 365 app access to their email. An auto-forwarding rule was set up to send all incoming email with key words back out to the attackers.
- **6 Aug 2020¹** – SANS identified the forwarding rule through log review / hunting for suspicious inbox rules and begins incident response.
- **Incident / Impact Details¹** – Through the incident response process, SANS investigation team found that during this time, 513 emails were forwarded to a suspicious external email address. Most emails were "normal" non-sensitive communications, but some contained PII data (email, work title, first/last name, work phone, company name, industry, address, country) for attendees of a SANS Summit event. After continuing to investigate, log correlation showed that at the time of the email forwarding rule creation, a suspicious (and unique to that employee) Office 365 app named "Enable4Excel" was used, and that its installation was linked to previously mentioned phishing email.

Through investigation and forensics on the affected machine and user account, it quickly became clear that SANS had a data exposure event on their hands and needed to send a notice to those who were affected.

[1] <https://www.sans.org/webcasts/data-incident-2020-technical-details-webcast-116375>

SANS Response and Lessons Learned

- Immediately notified all users of data exposure and all details
- Posted link for info: [sans.org/dataincident2020](https://www.sans.org/dataincident2020) which answered key questions clearly:
 - What is known for certain?
 - Are there additional details on what occurred?
 - What information was disclosed?
 - How are we investigating further?
 - What are we doing to prevent this from happening again?
- Webcast¹ given clearly stating incident details
 - "Oversharing" - Timeline, IOCs, details, and high/low-level lessons
- Site giving incident IOCs posted for reference / help to others²

SANS Response and Lessons Learned

In response, SANS utilized the otherwise unfortunate situation as an opportunity to demonstrate and model the type of response we (and likely most people) would like to see from any organization in which customers data is breached. This included clear, detailed communications with those affected, admission of mistake that were made, and oversharing of incident details.

Beyond the email communication, every effort was made to share both the lessons learned and technical details of the attack with those in the industry, in an effort to help others from falling victim to the same style of attack (especially considering the attack was found to be similar to many other orgs experiences and not targeted at SANS specifically). This was done through:

- A webcast¹ from leadership and technical experts giving everything known about the timeline, details, and lessons learned from the attack
- A landing page with key questions answered for general information on the incident (link in the slide above)
- A technical incident information page for technical security practitioners²

For those interested, some of the lessons learned given during webcast are listed below.

Technical Lessons Learned¹:

1. Increasingly remote workforce is introducing new challenges that all orgs must face
2. Finding the right balance between acceptable risk and sustainable costs is a struggle for every organization – SANS included
3. Outsourcing to cloud providers and services lead to reduced control and visibility (and experience)

Tactical Lessons¹:

1. Challenge with awareness is limiting what you teach people, you can far too easily overwhelm your workforce. You can't teach every possible attack, focus on common / shared indicators. Example: One indicator is if after clicking you are requested to enable permissions – likely it's an attack. Include screenshots in training so it's easier to identify.
2. Use tools like Slack / Microsoft Teams for more personal, responsive interaction between security and the workforce.

Strategic Lessons¹:

1. Security awareness is nothing more than another control to manage risk, it is part of an extension of all your controls
2. Train not only your workforce on how to interact with technology, but train your security team on how to interact with your workforce
3. Emphasizes the need for *continuous* training of workforce, like vulnerability management (patching), you can *never* slow down

1 <https://www.sans.org/webcasts/data-incident-2020-technical-details-webcast-116375>

2 <https://www.sans.org/blog/sans-data-incident-2020-indicators-of-compromise/>

Crisis Management Process (1)

- Three functions:
 - Recovery
 - Readiness
 - Response
- Like IR, best if planned prior to your crisis (and far less expensive)



Crisis Management (1)

Crisis management is a related, but wholly separate, function to incident response. Whereas incident response from the SOC's point of view is focused on things like identification, isolation, and eradication, there is a larger effort that must be managed across business functions to manage impacts to the organization beyond technical concerns. While the SOC normally does not run this process, it is a key participant and should be both aware of and a driving force behind the crisis management process.

Crisis management has three main components, according to Deloitte's Cyber Crisis Management lifecycle[1]: Recovery, Readiness, and Response. As in incident response, these functions work most effectively if planned and practiced *prior to* the incident!

[1] <https://www2.deloitte.com/content/dam/Deloitte/global/Documents/Risk/gx-cm-cyber-pov.pdf>

Crisis Management Process (2)

- Interacts with incident response, does not overlap
- Cross-functional planning and support that starts with *Readiness*
- Long- and short-term communication is key
- Can be a long process



Crisis Management Process (2)

Crisis management can be a lengthy process that continues long after the Incident case is closed out in the SOC. The security team's role in this process changes over time:

1. The SOC is a key driver in the readiness process, taking steps to mitigate risk and inform the business of any residual risk that is either accepted or unmanaged
2. Response occurs in real-time when an incident happens, and the SOC is merely a participant – security must provide timely updates to inform the larger business strategy and response.
3. Recovery may include non-technical (and even non-security) initiatives to restore trust in the organization or its brand. Security is a key element here as well, advising and informing based on lessons learned and any new efforts or re-tooling in the wake of the incident.

Crisis Management Functions

- **Governance**

- Defines decision framework and cross-functional plan
- Determines who does what, and specifies a business lead who can coordinate activities outside of the technical response

- **Strategy**

- Outlines the escalation process to manage and coordinate IT, ops, business recovery
- Describes how to best liaise with law enforcement and regulatory bodies
- Aligns response efforts with security and IT

- **Technology**

- Balances immediate need and long-term remedies
- Facilitates workarounds and stop-gap measures

Crisis Management Functions

Governance decision framework determines when and how crisis management processes are invoked, as well as where the crisis management process provides inputs to and takes outputs from incident response. This latter element is key in ensuring that business processes do not trample on or unduly influence technical incident response and making sure that crisis management processes aren't invoked too soon or too late.

Strategy defines in more concrete terms exactly how escalation will work, including ways the organization should work with external groups like regulatory bodies and law enforcement and internal groups like security and IT.

Crisis Management Functions(contd.)

- **Business operations**

- Removes barriers like inefficient or ineffective processes
- Provides surge support and resource re-allocation

- **Risk and compliance**

- Anticipate requests from external parties
- Analyze impacts and exposures, including risks from short-term solutions and workarounds

- **Remediation**

- Balances the recovery efforts with business operations
- Prioritizes new technology, budget updates in wake of the incident
- Prepares for increased regulatory (and possibly consumer) scrutiny

Crisis Management Functions (contd.)

Technology is one area of overlap between crisis management and incident response. It's helpful to remember that some technology solutions – like those that block or isolate threats – may be temporary measures, while others – like new security controls – may be longer term. Keeping in mind which measures are temporary workarounds or stop gaps is key to avoiding technical debt taken on during an incident and later forgotten. The business operations function of crisis management serves to remove "red tape" and other organizational barriers to successfully recovering from an incident. This may include things like resource re-allocation on a temporary or permanent basis.

Finally, the risk and compliance function of crisis response serves to anticipate requests for information from regulatory bodies and other third parties. This is a critical function in crisis response where privacy issues or other regulatory concerns might come into play. This function also analyzes residual risk introduced to the environment from temporary workarounds – for example, shutting off a portion of the network to contain an intrusion.

Crisis Communications

- “What you say when everything goes wrong.”
-Scott Roberts, DFIR Summit 2015
- “Everything” usually equals:
 - A breach
 - A vulnerability
 - A DDoS
- Plan to be a key participant, not an owner or decision maker



Crisis Communication

According to Scott Roberts in his DFIR Summit 2015 talk on the subject, crisis communications is "what you say when everything goes wrong." In cybersecurity terms, "everything" usually means a high-profile breach, vulnerability, or distributed denial of service. We often enjoy a level of anonymity in the SOC, doing most of our work "behind closed doors". But in a major breach, the SOC will become a hotbed of activity with eyes from all over the organization (and outside of it) focused like the Eye of Sauron on everything you're doing. Many teams make the mistake of planning communications during incident response under the assumption that they will operate largely unimpeded. However, anyone who has responded to a major incident will tell you that at least half the battle is managing communications and near-constant interruptions from people looking for updates, asking questions, and inserting themselves into the process.

In fact, when an incident rises to the level of an organizational crisis, the SOC normally assumes more of a supporting role. Crisis communications is an important component of crisis management, so it's important to understand how the SOC plays into this process as a participant. Let's dive into the fundamentals of good communications in an IR scenario, which will feed into the organization's larger crisis management efforts.

You can view the slides from Scott's fantastic presentation here: <https://sroberts.medium.com/crisis-communications-for-ir-the-preso-bdce57113e3d>

Tabletop Exercises

Tabletop exercises are a great way to test your crisis management plan!

- Also tests playbooks, procedures, response, communication, decision making, and more

To run a tabletop exercise:

1. Define the target audience, objectives, items being tested, outcomes
2. Create a realistic, pre-defined problem scenario, or use an example!¹
3. Gather stakeholders and run the scenario
4. Inject twists and turns, have a facilitator take observations as you go
5. Debrief – objective review, lessons learned, assign action items



Table-Top Exercises

A high-value activity that can be performed to test your newly formed SOC without a lot of planning or overhead is a table-top exercise. These exercises involve talking out a tricky scenario that you might encounter in order to give those involved practice in how to respond in case of the real thing.

It's tempting to think that everyone will instinctually know how to act when an incident occurs but, unfortunately, that is just not the truth. In the moments of chaos that follow a potentially high-impact incident, people forget whom to talk to, what to do, and which process to follow, and often all of it goes out the window with potentially disastrous results. There's a reason we do fire and tornado drills, as the saying goes, "The body cannot go where the mind has not gone first." Until you've at least *thought* your way all the way through a scenario, it's highly unlikely that, in the real situation, you will act in the optimal way.

To run a successful tabletop, there needs to be a few simple considerations taken care of first.

1. Consider who you want the test to involve, and which processes and procedures will be tested. What are the expected outcomes or objectives of this test?
2. Once you know what you'd like to test, craft a realistic scenario (or borrow one from a guide like the CIS tabletop exercise document linked below¹) that can be read aloud to the group expected to respond. These can involve as much detail as you'd like.
3. Clear out the time in everyone's calendar and gather everyone in a room. Tabletop exercises don't have any strict definition on how long they should last, it could be anywhere from 15 minutes to hours. (One note, if you do plan a long and detailed scenario involving minor interaction with those outside the SOC, you don't have to have minor players sitting there the whole time. One tactic I've seen successfully used is to warn minor players about the tabletop and tell them that, for example, they may get a scenario related email throughout the day requesting some theoretical action be taken, which they must respond to as they normally would. Put **TEST** in the subject header to specify this is a tabletop-exercise related email and not the real thing).

4. As the scenario runs, if designed as such, inject new information, "discoveries" of false info, or plot twists into the room, and ensure whoever is facilitating is observing and noting what happens and any key reactions in the group.
5. Once complete, run a short debrief session. Cover what went well and what didn't, outcomes for each objective, and what was learned. Before you go, assign all necessary action items with a deadline (and follow up) to ensure the improvements needed that you identified are not lost.

[1] <https://www.cisecurity.org/wp-content/uploads/2018/10/Six-tabletop-exercises-FINAL.pdf>

Designing a Scenario

- Are you drilling the organization or the security team?
- **Start at the end:** what are you worried about?
 - Specific threats, actors, failures
 - Poor coordination
 - Crisis comms/reputational damage
- Develop your central "plotline" first: beginning, middle, and your ideal ending



Designing a Scenario

Developing tabletop scenarios can be a great opportunity to get creative in training your team or organization, but it's easy to go overboard or focus on situations that sound exciting at the expense of learning opportunities for the group. Our scenario must be realistic but should also give participants ample opportunity to communicate, make decisions, and work together to put our response process through its paces and identify improvements we need to make. Think about the participants and objectives you've captured – are you trying to train the operations team, the larger security team, or multiple groups that include non-technical staff? We want to avoid having other groups around waiting for the SOC team to walk through its analysis and incident response procedures if the exercise is more operationally-focused. On the other hand, if the goal is to train multiple groups outside of the security team to work together in an incident response scenario, it may not be necessary to have heavy participation from the technical team. The answer to this question will help you sketch out your scenarios and identify key participants.

When it comes to building or selecting the right scenario, it's often helpful to start at the end and work backwards: what are we most worried about? Are there specific threats we want to make sure we're equipped to respond to? Specific actors? Are there failure conditions we want to make sure the team can work together to isolate and recover from?

Once we know where the team should (hopefully) end up, we can think about our central "plotline" - the incident trigger, the decisions made and actions taken, and the ultimate resolution. This plotline should represent the ideal scenario where everyone involved makes the best decisions possible to reduce damage.

Adding Injects

- Consider major and minor decisions
 - Minor decisions demonstrate adherence to process – no impact on the scenario or its outcome
 - Major decisions become branches on your central plotline
- Operational scenarios will have more major decisions, organizational scenarios more minor decisions
- Major decisions have the potential to change the scenario outcome
 - Normally prompted by **injects**
 - Prune your branches!

Adding Injects

Building decision points into your tabletop will make the exercise more interactive and allow participants to help drive the narrative. Depending on the audience and the goals of the exercise, participants may make major or minor decisions; minor decisions are participant choices that demonstrate or reinforce adherence to some process. An example of a minor decision in a tabletop scenario might be, "At this point, I would notify legal and corporate communications by sending an email to so-and-so with a link to the incident executive summary." This choice doesn't necessarily impact which way the table is going to go or later events, but it demonstrates that the participant knows and is following the process so is still a choice we want to capture.

Major decisions *do* impact the scenario and are normally prompted via injects. An example of a major decision might be, "We're going to continue to monitor and scope the intrusion versus isolating the activity we've seen so far." Since these choices may impact the flow of events, you can conceptualize them as "branches" off of your core narrative. Designing multiple narratives in this "choose-your-own-adventure" style is a great way to keep the scenario engaging and interactive; just take care to "prune" these branches and ensure that they lead back to your core narrative or one of the outcomes you've pre-determined for the exercise!

Short Tabletop Example (1)

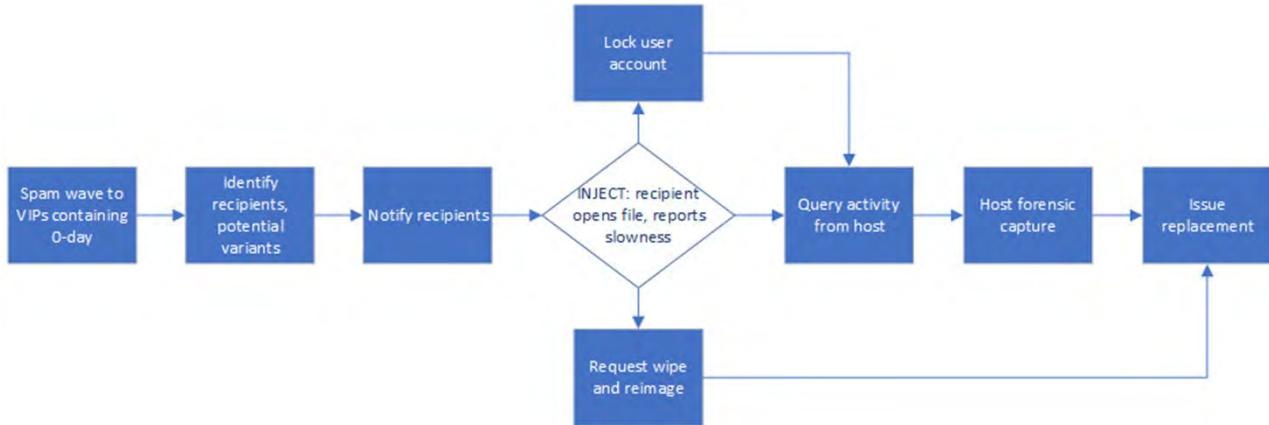
Surprise Targeted Phishing:

- **Objectives:**
 - Test critical phishing incident response
 - Test off-hours incident response capability
 - Test communication with email team
- **Scenario:** *A spam wave of 100 emails to VIPs containing an attached file with a zero-day exploit comes in over the weekend. The SOC receives an alert and presumes it to be a targeted attack that must be dealt with immediately.*
- **Inject:** *An hour later, one of the recipients has self-reported they immediately opened the email and now their PC is running slowly*

Short Tabletop Example (1)

This page has an example of a short operations focused table-top scenario you might run to see how people would respond to a critical off-hours situation. This scenario introduces a stress-inducing situation where key individuals might be at home and forces them to consider how they might contact them. It also tests if they know what actions they can immediately take to contain damage, as well as respond to (at least one) infection. To bring additional realism to a scenario like this, you could have analysts bring a laptop and actually run through the motions they would take to look up phone numbers, search the SIEM for details, and request help from the email team. The closer the analysts are to going through the actual motions they would take in a real situation, the better the training scenario becomes.

Short Tabletop Example(2)



Short Tabletop Example (2)

Here's our short scenario represented as a flow chart. You can see the main plot line running from left to right with various steps you could expect the team to take to scope and contain the intrusion. Note the decision point in the middle, which we'd consider a major decision: do we lock the user account immediately and risk impacting any off-hours work the VIP might do, do we search logs for this host to see what activity may be causing the slowness, or do we request a wipe and reimagine and move to issue the user a new machine? Each of these choices could have very different outcomes depending on what you've identified as your "best case" scenario. This flow chart is a very simple example of what you might create even for this simple scenario, but hopefully you can see how visualizing the narrative and various courses of action can make for a more engaging tabletop exercise.

Crisis Management and Continuous Improvement Summary

- Walk through, document and practice your crisis management process
- "Proper preparation prevents poor performance"
 - Design tabletop exercises of multiple types
 - Run them periodically – "train like you fight"
- Gamify it!
 - Looking to make practice easier and more fun?
 - Check out "Backdoors and Breaches" from Black Hills Infosec

Crisis Management and Continuous Improvement Summary

When it comes to crisis management, having thought through various scenarios and doing proper after-action reviews with supporting documentation on what could have gone better will drive improvement. While you may not be in charge of many aspects of breach handling, you will certainly be a key stakeholder as well as contributor of information on what happened and how the organization was impacted. Preparing your communication plan as well as practicing recovery, readiness and response are key habits to minimize friction during an incident.

While thinking through crisis management is great, practice is even better! Tabletop exercises of various types utilizing scenarios of common, and high-risk scenarios will help prepare the team to jump to action when called upon. While we will show you in a moment how to design tabletop exercises, know that you don't necessarily even have to come up with every scenario on your own. As with brainstorming, "quantity leads to quality" is also true of tabletop exercise scenarios, and card games like "Backdoors and Breaches" from Black Hills Information Security can help spice your tabletop exercises up when you need some additional inspiration.

<https://www.blackhillsinfosec.com/projects/backdoorsandbreaches/>

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

Planning, Preparation, and Review

- Incident Response Planning and Preparation

• Investigation

- *Exercise 4.1 – Investigation Quality Review*

IR Execution

- Identification, Containment, and Eradication

• Incident Response in the Cloud

• IR Tools

- *Exercise 4.2 – Planning Responses with RE&CT*

- Crisis Management and Continuous Improvement

- *Exercise 4.3 – Designing Tabletop Exercises*

- Recovery and Post-Incident

- Summary and Cyber42 – Day 4



This page intentionally left blank.

EXERCISE 4.3

Exercise 4.3: **Designing Tabletop Exercises**

OBJECTIVES

- Walk through pre-planning considerations
- Identify tabletop objectives
- Identify key participants and stakeholders
- Select a scenario and design the exercise



Exercise 4.3: Creating, Classifying, and Communicating Your Metrics

Please go to Exercise 4.3 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

Planning, Preparation, and Review

- Incident Response Planning and Preparation

- Investigation

- *Exercise 4.1 – Investigation Quality Review*

IR Execution

- Identification, Containment, and Eradication

- Incident Response in the Cloud

- IR Tools

- *Exercise 4.2 – Planning Responses with RE&CT*

- Crisis Management and Continuous Improvement

- *Exercise 4.3 – Designing Tabletop Exercises*

• Recovery and Post-Incident

- Summary and Cyber42 – Day 4



This page intentionally left blank.

Recovery

Typical recovery activities:

- Rebuilding from backups
- Re-imaging from scratch
- Replacement of files, applications
- Patching and reinstalling programs
- Restoration of activity
- Validation
- Monitor for signs of repeated attack attempts



Recovery

After you've booted the adversary out of the environment for good, it's time to clean up the mess. Recovery procedures should instruct analysts how to initiate your organization's process for restoring from backups or rebuilding the affected systems (in large organizations, this duty will likely fall outside the SOC). Security's role at this stage is to help facilitate a clean and complete recovery effort while watching for any additional signs of attack or reinfection. Since adversaries love to use multiple backdoors to give themselves numerous avenues into an environment, it's not uncommon to see the recovery stage turn into a second round of detection, containment, and eradication! If this happens, consider the watch and learn technique since you are clearly now dealing with an adversary that puts the P (persistence) in APT.

Post-Incident Activity

Items to take care of post-recovery:

- Writing the incident report
- Feed info back to threat intelligence
- Consider new detection and prevention measures
- Continued extra monitoring for affected assets
- Revise procedures and plans as necessary
- Lessons-learned meeting



Post-Incident Activity

Once recovery has initiated the SOC can start on the post-incident activities. For major incidents with significant impact, this will undoubtedly include the writing of the incident report. Incident reports may seem like they'll only be read once but believe me when I say I've seen organizations ask for them repeatedly for *years* after the fact. Why? Because they contain a list of vulnerabilities that should be fixed, real impact that was caused as a result of mistakes and oversight and are great for justifying all sorts of initiatives throughout the coming years.

Other than the incident report, feedback all tactical and strategic information about the adversaries that was learned throughout the course of the incident, this should help them find additional info and better prepare you for the next time that attacker comes knocking on your door. Write new detection analytics and implement new prevention measures where possible, revise any procedures that were non-optimal during the incident, and continue to monitor the affected assets more closely for signs of reinfection. Finally, and maybe most importantly, book the lessons learned meeting.

Writing the Incident Report

Common report sections:

- Executive Summary
- Incident Timeline
 - Specific devices
 - Overall
- Impact and Response
- Root causes / security failures
- Lessons Learned
- Appendices
 - IOC lists and more

Tips for Success:

- No standard format – pick one and use it across incidents
- Consider your audience!
- Consider 2 versions – with more/less sensitive info, or a redacted version if necessary
- Remember your report may be used for *years* as reference

Writing the Incident Report

For incident report writing there is not a standard industry template for an incident report, but you can either choose a pre-existing one online (such as Palo Alto/Demisto's example available at the URL in the footnote below¹) or craft your own in a format that contains the data you'd like to collect for each incident. Standard sections for a large incident would include things like an executive summary, detailed timeline of the incident as a whole as well as the individual systems involved as necessary, key data on the impact and incident response, root causes and lessons learned. Additionally, while indicators may be mentioned throughout the report, it's best practice to collect all IOCs at the end of the report in a copy/paste friendly format, especially if you plan on sharing the report with threat intelligence sharing groups.

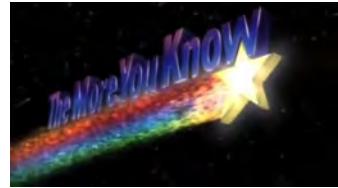
Some additional tips for success - Consider your audience in the report, write it to the level of technicality of the reader. In a large incident, you may have multiple report audiences of various levels of detail you'd like to provide. In those scenarios, having multiple versions of the report with more or less sensitive data, or a redacted version may be called for. Don't forget that in some scenarios, large scale breach reports and other items may be referenced for years by various groups for reference, justification of expenses for controls, and more.

[1] Palo Alto Incident Response Reporting Template:

<https://www.paloaltonetworks.com/resources/whitepapers/incident-response-reporting-template>

Collecting Intelligence

- Every investigation yields new IOCs, techniques, and hints about the attacker's motivation
- Gather these and use references like the ATT&CK Matrix to see where your team was successful in the initial detection and where gaps may exist
- Feed this new information back into your threat intelligence process to better understand your adversary



Collecting Intelligence

During the detection, containment, and eradication phases, it's very likely the team has gathered new information about infrastructure and tactics used by the attacker as well as artifacts they have left behind. This is valuable information that can help you detect the next incident faster and scope it more effectively. Analyzing these items in their entirety can also help your team better understand your adversaries and the TTPs that will likely be used again to target organizations like yours. Maintaining an effective threat intelligence capability means feeding this critical information back into the intelligence analysis process so that your team may better understand the threats you face. Since every incident has its own nuances, it's helpful to map these new artifacts back to standard models like the ATT&CK Matrix. Use this to identify techniques you successfully identified in your investigation as well as techniques your initial detections may have missed.

Implement New Detection and Prevention Measures

- Secondary goal of IR is to make your environment more resistant to future attack
- Revisit your control catalog of choice and identify detection and prevention controls that might have driven faster detection, more effective investigation, or prevention of malicious activities
- Re-baseline your environment and controls, add new services or infrastructure you were previously unaware of

Implement New Detection and Prevention Measures

Reducing the impact of an intrusion as quickly and effectively as possible is, of course, the primary goal of incident response. However, a secondary (and almost equally important) goal is to improve your defenses and make the enterprise more resistant to future attacks. Think back to earlier in the day when we covered control references like NIST SP 800-53 and the CIS Top 20 Controls. Whatever reference your organization uses to track controls implemented in your environment, refer to that accounting and identify areas where controls did not work as intended, or the lack of a control contributed to negative outcomes during the response process. Some examples of these gaps might be detection-related – for example, data your team needed to identify or investigation the incident wasn't available when you needed it – or they may be prevention related. An example of the latter might be that malicious code was ran to run on an endpoint when ideally it would have been stopped and isolated. Also keep in mind that security controls are often complex and imperfectly implemented; it's possible that controls you thought had complete coverage in your environment were missing in a few enclaves or on some endpoints. Perhaps you've discovered some shadow IT or new application infrastructure you were previously unaware of. Make sure that the SOC capitalizes on this learning opportunity to revise or refine your baseline understanding of the environment and improve the enforcement of security policies and process.

Groups for Additional Logging During and After Incidents

- When incidents occur, additional logging may be desired
 - Prepare for this ahead of time
 - Create AD groups and other configs with additional logging enabled
 - Add recently-compromised assets to those groups
 - Consider OPSEC concerns before switch...
- Pre-compromise:
 - Standard auditing policy
- Mid and post-compromise:
 - Enable your additional logging policy
 - Temporarily add centralized verbose logging and additional data



Groups for Additional Logging During and After Incidents

One way to preemptively prepare for the need to monitor a currently or recently compromise asset is by creating special configuration management groups that these compromise machines can be placed into. These special "recently compromised" groups should push a higher-volume, extra-verbose auditing policy to the system which enables collection of data sources you wouldn't normally centralize due to high volume (such as full PowerShell or host firewall logs). The idea is to keep the systems in the special high-volume logging group as long as required until you are comfortable the adversary is not coming back for them as soon as you have kicked them out. Creating these groups ahead of time puts your team in a position to easily keep a closer watch on these systems while they are in a higher state of risk

Revise Procedures and Plans

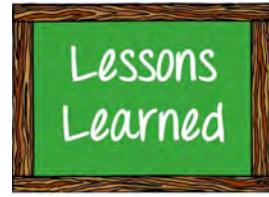
- Slow is smooth and smooth is fast
- Before we can do something quickly, we must do it reliably with no mistakes
- Review evidence and incident timeline
 - What delays or gaps can you identify?
 - Are any of these due to lack of process or deviating from the existing process?
 - Focus training on those items

Revise Procedures and Plans

Sometimes in our haste to return to “normal operations,” we forget to revisit our incident response procedures and plans. Incident response is a major learning opportunity, and the lessons it can teach us extend to our processes as well as our technical controls and knowledge of the adversary. Review your incident response plan and the way you engaged with each other and other teams outside of security; are there things that could have been done better? Are there steps that were skipped or missing from the process which might have resulted in a smoother process? Was there any rework or unacceptable opportunity cost that impacted important functions outside of the response effort? There is a saying in law enforcement and the military that “Slow is smooth and smooth is fast”. Before we can worry about speed, we must focus on executing the process in a repeatable way without mistakes. Once we can do that, we can work on making that repeatable, high quality process work at a faster pace. The same is true with our incident response plans and processes and working out those mistakes and gaps will result in better results and more speed the next time an incident occurs.

Lessons Learned

- One of the most important parts
 - Helps improve next response; ensure it doesn't happen again
- Plan this meeting ASAP!
- Questions to review:
 - What went well?
 - What could have gone better?
 - Were procedures followed, did they work as intended?
 - How can we stop or detect this earlier next time?
 - Was everything communicated in a timely fashion?
 - Were findings properly documented?
 - Was there any unnecessary disruption or other impact? Etc.



Lessons Learned

After the incident is wrapped up and the situation has returned to business as usual, there's one final step to take care of—the lessons-learned meeting. Despite your best intentions and understanding of the usefulness of this step, I guarantee at the time it will still be something you consider if you *really* need to do. Everyone just wants to get back to work, and the temptation will be there, but don't give in! If you want to make your organization's defense better, the lessons learned meeting is the place where everyone is forced to face the failure and discuss how the situation can be prevented or minimized the next time.

Plan the lessons learned meeting as fast as possible after the incident concludes while the details of communication and work done are still fresh in everybody's head. The meeting doesn't have to be long, but does need to answer any questions that can lead you to improving incident response operations the next time around. Ask stakeholders about how the disruption could have impacted them less, how your communication was, and what could have gone better. Check if the team properly documented everything that happened, if procedures were followed, and anything else to help fix the mistakes that might've fallen through the cracks in the heat of the moment. It is this moment where your team learns how to get better, and skipping out on these questions and this meeting would only be harming yourself in the long run.

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

Planning, Preparation, and Review

- Incident Response Planning and Preparation

- Investigation

- *Exercise 4.1 – Investigation Quality Review*

IR Execution

- Identification, Containment, and Eradication

- Incident Response in the Cloud

- IR Tools

- *Exercise 4.2 – Planning Responses with RE&CT*

- Crisis Management and Continuous Improvement

- *Exercise 4.3 – Designing Tabletop Exercises*

- Recovery and Post-Incident

- **Summary and Cyber42 – Day 4**

This page intentionally left blank.

Day 4 Summary

- Investigation tactics and techniques
- Incident response policy, planning, and process
- Preparing your environment and your team
- Designing tabletop exercises to test processes and procedures
- Incident identification, containment, and eradication approaches
- Recovery and post-incident activities
- Incident response in the cloud
- The breach response process and crisis communications
- IR tools
- Measuring and improving incident response capabilities using RE&CT and case reviews

Day 4 Summary

In book 4, we covered incident response from many different angles – from preparation to identification to containment, eradication, and recovery, and applying what we learn to continuously improve the process. We discussed response planning from a policy, process, and training perspective, and technical execution in a variety of different infrastructures and response use cases. We also talked about useful models and frameworks to guide your planning and measurement such as RE&CT, defensible network architecture, and the CIS Critical Security Controls. We designed a tabletop exercise to test our incident response process, planned out response actions using the RE&CT framework, and did some incident response quality control by reviewing a documented investigation.

Understanding how to equip your team to support the full incident response lifecycle will better enable you as a manager to measure and improve its capabilities, even if some incident response tasks fall outside of the SOC's purview. As with most of the other topics we've covered this week, incident response is just one element of security operations, and it isn't "done" when your IR plan and procedures are in place. It is a discipline that should evolve and improve over time along with your team and your organization. Using some of the hands-on tools, approaches, and frameworks we went over in book 4 should give you some great starting points to do that in your own teams.



Cyber42 Simulation

Day 4

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.

551.5

Metrics, Automation, and Continuous Improvement



SANS

THE MOST TRUSTED SOURCE FOR INFORMATION SECURITY TRAINING, CERTIFICATION, AND RESEARCH | sans.org



551.5: Metrics, Automation, and Continuous Improvement

© 2021 John Hubbard and Mark Orlando | All Rights Reserved | G02_02

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

TABLE OF CONTENTS	PAGE
Introduction	of Class ¹
Staff Retention and Burnout Mitigation	4
Exercise 5.1 – Training and Career Development Planning	30
Metrics, Goals, and Effective Execution	32
Measurement and Prioritization Issues	63
Exercise 5.2 – Creating, Classifying, and Communicating Your Metrics	83
Strategic Planning and Communications	85
Analytic Testing and Adversary Emulation	105
Exercise 5.3 – Purple Team Assessment Planning, Execution, and Tracking	124
Automation and Analyst Engagement	126
Conclusion	152



This page intentionally left blank.

Day 5 Overview

- **Introduction**
- **Effective Execution**
- Staff Retention and Burnout Mitigation
- Metrics, Goals, and Effective Execution
- Measurement and Prioritization Issues
- **Continuous Improvement**
- Strategic Planning and Communications
- Analytic Testing and Adversary Emulation
- Automation and Analyst Engagement
- **Exercises:** Training and career development planning, creating SOC metrics, and purple team assessment

Day 5 Overview

Here is a list of topics we will be discussing throughout the fifth and final 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

- **Effective Execution**
- **Staff Retention and Burnout Mitigation**
 - *Exercise 5.1 – Training and Career Development Planning*
- Metrics, Goals, and Effective Execution
- Measurement and Prioritization Issues
 - *Exercise 5.2 – Creating, Classifying, and Communicating Your Metrics*
- Continuous Improvement
- Strategic Planning and Communications
- Analytic Testing and Adversary Emulation
 - *Exercise 5.3- Purple Team Assessment*
- Automation and Analyst Engagement
- Summary and Cyber42 – Day 5



This page intentionally left blank.

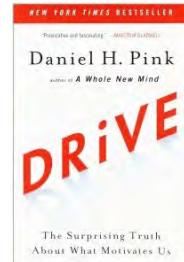
Drive: Cultivating Intrinsic Motivation

What we need in the SOC: intrinsic motivation

- Analysts that love the job and the daily challenge
- But how do we cultivate intrinsic motivation?
- What studies show *doesn't* work: More money

Daniel Pink defined **three factors** in "Drive":

1. **Autonomy** – The desire to be self-directed
2. **Mastery** – The desire to get better at something that matters
3. **Purpose** – The yearning to do what we do in the service of something larger than ourselves



Drive: Cultivating Intrinsic Motivation

One of the most important factors to mind over the long term is your SOC's retention rate. Burnout is a real and serious concern for every SOC, but not all SOCs have the same high turnover rates experienced by some. In this section, we'll describe some of the specific models that researchers have found that describe how burnout occurs, and how we can keep it at bay.

First, let's approach the motivation and burnout discussion from a more general level, however. When we reveal the factors behind what specifically stops burnout in a SOC, the items the researchers found will likely not surprise you but looking beyond them to find *why* those might be the factors is interesting as well.

In Daniel Pink's 2009 book "Drive: The Surprising Truth About What Motivates Us", Pink dives into the research about what keeps knowledge workers, in specific, excited and engaged in their jobs. What he found was that while you do have to pay people a fair salary, money is not the primary factor that makes people happy. He says that people just need enough money to keep money as a non-issue, after that it can actually ruin intrinsic motivation by replacing it with extrinsic rewards. What the research does point to is that there are three main factors that keep folks intrinsically motivated to come into work and solve hard challenges day after day. Those factors are a sense of **autonomy, mastery, and purpose** (the definition which he used for each is shown on the slide above.) While you may have not thought of these three words on your own, they feel very intuitively correct to most who hear them. Of course, we all want to decide on our own work, see the progression and the fruits of our labor, and do it in service of something with meaning. As we continue this discussion, consider whether, at the end of the day, these are ideals that you cultivate in how your SOC operates.

Does This Apply to the SOC?

Yes indeed – consider the following SOC-specific research:

- Sundaramurthy, Sathya Chandran, et al. "**A human capital model for mitigating security analyst burnout.**" *Eleventh Symposium On Usable Privacy and Security ({SOUPS} 2015)*. 2015.
- Studied SOC analyst burnout via anthropological study
- Developed a model to fix the underlying issues, not just symptoms

Conclusion:

"burnout is a human capital management problem resulting from the cyclic interaction of a number of human, technical, and managerial factors. Specifically, we identified multiple vicious cycles connecting the factors affecting the morale of the analysts."¹



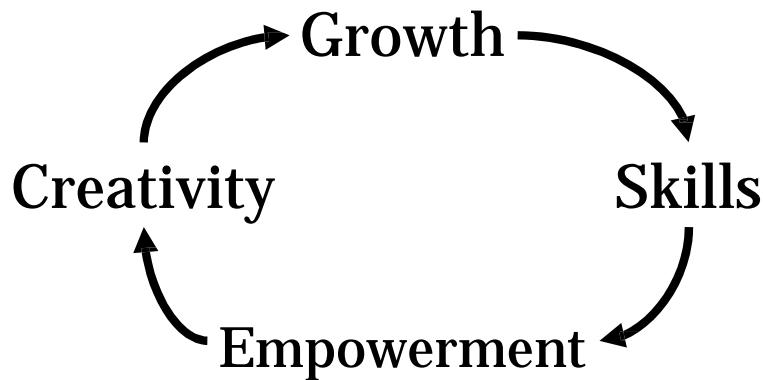
Does This Apply to the SOC?

The next obvious question is if these ideals apply, in specific, to SOC work. While there isn't an enormous body of research on what specifically causes SOC burnout, there is enough to show us that these ideals do line up. In 2015, a group of researchers set out to find the causes of SOC burnout by running an anthropological-style study of a SOC (which they found to be the necessary method after several other failed attempts). After spending six months in a SOC, performing work as an analyst, they took copious notes and were able to turn their findings into a succinct model of what the factors are that can cause burnout and misery in a SOC job. Their paper, listed on the slide above, should be mandatory reading for any SOC manager.

The researchers used Grounded Theory to formulate a model they call the "human capital model for mitigating security analyst burnout", which we will discuss over the next few slides. What they found was that *"burnout is a human capital management problem resulting from the cyclic interaction of a number of human, technical, and managerial factors. Specifically, we identified multiple vicious cycles connecting the factors affecting the morale of the analysts."¹*

[1] <https://www.usenix.org/system/files/conference/soups2015/soups15-paper-sundaramurthy.pdf>

SOC Human Capital Model Factors



Successful SOCs must develop and manage human capital

- Attending to these factors creates a **virtuous** or **vicious** self-perpetuating cycle
- A positive feedback loop

SOC Human Capital Model Factors

Here is the core of the model that the researchers developed—the human capital model of the SOC. In it, there are four factors that must be attended to carefully: Growth, Skills, Empowerment, and Creativity. These four factors are linked in a positive feedback loop such that if one is trending in the wrong direction, the next factor is likely to follow. This means that once one of these factors is ignored, a vicious cycle can be created in your organization which leads to the burnout and retention issues that so many SOCs face. This connection is likely why so many SOCs suffer with keeping analysts—just one mistake in this loop can poison the whole cycle. The good news is it also works in the opposite direction, once things start to trend positively, those successes feed back on each other and can cause rapid growth in the SOC as well. Over the next few slides, we'll discuss the factors in the human capital model and how to tend to them and then expand the model out into the full set of factors the researchers identified.

Growth

- **Definition:** Increasing intellectual capacity of analysts
- **Key drivers:**
 - Variety—in job tasks, incident types worked, etc.
- **Common problems:**
 - Scoping analysts' jobs too tightly (only ticket evaluation and escalation)
 - No time to learn on the job, swamped with ops activity
- **Recommended actions:**
 - Scope jobs to contain a variety of tasks that keep analysts learning
 - Confirm analysts have time to exercise *creativity*, which brings growth
 - Eliminate mundane activity with automation

Growth

First up is growth. This factor is about maintaining analysts that are constantly getting better and more capable of doing a variety of tasks—variety being the keyword here. The biggest driver of growth is analysts that are constantly given a variety of experiences and situations with which they have a chance to learn and improve. This can become a problem in teams where the scope of the job is too small (such as if tier 1 analysts only qualify alerts, then pass them up the chain) or if analysts have no extra time on the job to learn how to take on anything new. To ensure the growth variable is tended to, the research suggests that analysts should be given a wide variety of tasks that will give them a high chance of encountering new situations. Automation should be used to eliminate mundane work where no growth will occur and no value is being added by having the analyst do the work manually. In addition, as a feed-in from the previous step in the human capital cycle, giving analysts creative freedom to explore new solutions to problems leads to growth.

Skills

- **Definition:** Development and continuous improvement of analysts' skill-set required to do their job
- **Key drivers:** *Growth* opportunities
 - On-the-job – Tabletop exercises, Purple Team, penetration tests, real incidents
 - Peers – Presentations, hands-on exercises, mentors
 - Formal – Paid training and workshops
- **Common problems:**
 - No on-the-job exercises to train analysts with their own tools
 - Inability to do formal training – insufficient time, budget, or approval
- **Recommended actions:**
 - Strive to (at the right time) provide formal and informal training wherever possible
 - Refer to previous discussion on the Continuous Learning Model

Skills

Skills were defined as the development and continuous improvement of analysts' skill set required to do their job. Notice it's not just development at one point in time, but a *continuous* process that is required for this step. Analysts that feel their skills are no longer growing may start to look externally for new opportunities where they can unleash their potential, so keeping the learning experiences flowing is one clear way to maintain a happy and engaged workforce.

Training doesn't have to be fancy or expensive to meet this requirement. Training can come from:

- On-the-job and daily experiences – tabletop exercises, Purple Team exercises, penetration testing and Red Teaming, and actual incidents. Any time the analyst is doing their job, they are honing their skill if new aspects are introduced to the attack.
- Peer-directed training – through mentorships, team presentations or hands-on exercises, this can be a useful supplement to daily on-the-job type skill improvement.
- Formal training – the most focused of the three, formal training should be provided where possible, especially for skills that may not be present in-house. Refer to the previous discussion on formal training for guidance on where, when, and how to best provide it.

Training is one item that many teams can struggle to either make the time or get the budget for. To help ease these issues, be sure to make a big deal out of the value gained from any training exercise, and give analysts the time to seek out new skills they have interests in.

Empowerment

- **Definition:** Enabling analysts to do their jobs efficiently
- **Key drivers:**
 - Trust in SOC *skills* to wield potentially dangerous power in a safe way
- **Common problems:**
 - New teams not trusted with capabilities
 - Politics issues, siloed teams not giving the SOC access
 - Previous mistakes lead to mistrust of the SOC
- **Recommended actions:**
 - Allow analysts to create threat detection analytics and see impact of work
 - Slow building of trust with mistake-reducing, peer-reviewed process
 - Clear communication about the need and impact of empowerment

Empowerment

Empowerment was defined as the perception by analysts that they are able to do their job efficiently (without unnecessary process and painful workarounds). The key item related to analyst empowerment is typically the trust of the SOC. If management believes they are a strong and knowledgeable team, it is likely the SOC will be trusted with the ability to make changes to the environment that could break things, but this also allows them to react quickly and decisively during an attack. An empowered SOC is a strong SOC that is enabled and trusted by management to wield these kinds of powers and knows how to use them in a safe way.

How does a SOC reach this status? By building and showing they have built the *skills* to be trusted with them, which, of course, is fed by their *growth* capability.

The most common issues in the empowerment area are:

- New teams that have not yet had a chance to prove themselves and teams that have made trust-eroding accidents in the past—in this case, I recommend working on a slow but sure plan to show that you can be trusted to not make mistakes, and use a mistake-limiting and peer-reviewed process to gain trust in the meantime. If, for example, your firewall group doesn't want to give the ability to the SOC to push blocking rules directly out of fear of self-inflicted DoS, ask if you can do it given SOC management or peer-review approval for each instance to limit the ability for one person to make a mistake.
- Politics issues, team silos – Sometimes there is fear within a group that if a SOC takes over some of the capabilities, they may be looked at as no longer necessary. Working through these types of problems is often about building mutual trust with the other team and ensuring they understand why you need the access that you do, and why speed is of the essence.

Creativity

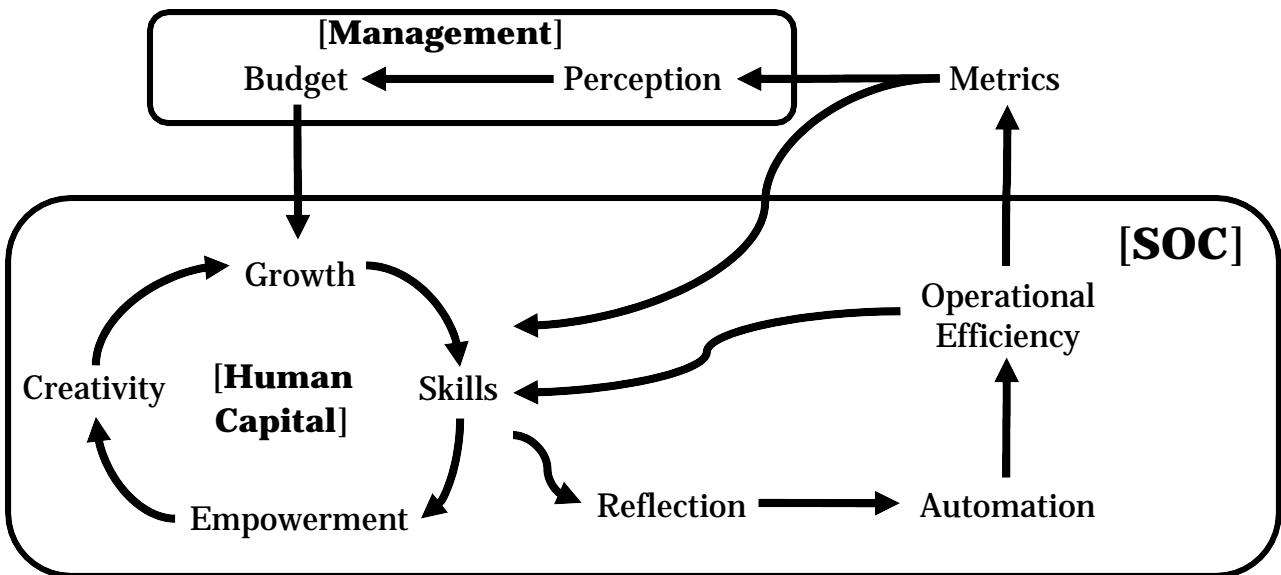
- **Definition:** Ability to handle novel operational scenarios
- **Key drivers:**
 - Empowerment to solve challenges in new and unique ways
- **Common problems:**
 - Over-prescribed procedure and workflow
 - Overly detailed playbooks that make analysts feel "on rails" or like robots
 - Lack of time for creative pursuits / lack of automation
- **Recommended actions:**
 - Free time to consider and pursue potential improvements
 - Encourage and reward learning of new technologies and solutions to old problems

Creativity

Creativity is the final piece of the puzzle and one that can be most difficult for many SOCs to achieve. Giving analysts a creative outlet often means letting go of the reigns a bit and allowing analysts free time to pursue improvements or take on novel situations in ways you haven't used before. If, for example, you encounter an incident where there is no playbook that fits what needs to happen, are analysts *empowered* to do whatever they need to do regardless? Or, do they have to work within a defined set of procedures that may restrict them.

In order to foster creativity, analysts should be given a bit of slack when triaging and investigating alerts, as well as to look for ways to make the team better and more efficient. Analysts that succeed in these endeavors should be praised—not only has your team become better from their discovery, but they also completed the cycle by using their activity to foster *growth* as well.

The Full Model¹



The Full Model

Now that we've discussed the four factors, it's time to complete the rest of the model. The researchers noted, "To mitigate analyst burnout, SOCs have to pay special attention to the interaction of human capital with 3 other factors: automation, operational efficiency, and metrics." This full model shows that there is not one, but 3 loops that can be made here, *all* of which are positive feedback loops.

- The smallest human capital cycle as just explained
- The second loop through reflection, automation, and ops efficiency feeding back to the human capital cycle
- The largest loop through the human capital cycle, through automation and ops efficiency, finally fed into the metrics and perception of management, which feeds back to the human capital cycle through budget.

[1] <https://www.usenix.org/system/files/conference/soups2015/soups15-paper-sundaramurthy.pdf>

Reflection, Automation, and Operational Efficiency

Reflection



Automation



Operational
Efficiency

Additional required items for burnout avoidance:

- **Reflection**

- Periodic review of procedures to find ops bottlenecks
- Must be *empowered* and *incentivized* to do so

- **Automation**

- Elimination of repetitive tasks
- Implemented with SOAR, EDR, SIEM, scripts, etc.
- Analyst and developer tool co-creation

- **Ops Efficiency**

- Leveraging all resources to respond to threats quickly

Reflection, Automation and Operational Efficiency

The first step outside of the human capital circle is toward the automation factor. But between the two lies reflection, something found to be necessary for automation to occur in the first place. The paper defines this reflection as time to review procedures and locate operations bottlenecks that can be improved. In addition, not only must analysts be given explicit time for reflection, but you must also empower and incentivize them to actually take that time to do so.

Automation in the SOC is one of the best ways to improve efficiency and overall happiness of analysts, which is likely why SOAR tools have become such a market hit. The idea is to find all repetitive tasks that don't require manual human decisions and facilitate scripts, your SIEM, EDR, or SOAR to take those actions for you as soon as they can. Not only is automation good for morale, but it's also an outstanding creative outlet for analysts, making it a double hit of positive effect for the SOC. When analysts are given time to develop new automated tasks, they get to do work outside the normal operational tasks they do, and when they're finished, their job becomes easier!

Naturally the more automation can be developed, the better the operational efficiency will become. This is why automation leads to it as the next step in the researcher's model. In the paper, on the operational efficiency factor, the researchers noted that not only are teams who are more operationally efficient happier with their job, but being happy with your job, in and of itself, causes you to perform better—meaning this connection to the human capital cycle is actually a bi-directional connection.

"The direct causality from human capital to operational efficiency indicates the obvious fact the highly skilled and creative analysts make operations efficient. Human capital also affects efficiency in operations through automation via reflections. The resulting automation accelerates operations — especially in case of highly repetitive tasks. ... On the other hand, the benefits of resulting from operational efficiency in turn create a positive influence on the analysts."¹

[1] <https://www.usenix.org/system/files/conference/soups2015/soups15-paper-sundaramurthy.pdf>

Budget, Metrics, and SOC Perception

Metrics



Perception



Budget

• Metrics

- Crucial in communicating the SOC's value
- Shows management SOC ROI

• Perception

- Driven by the metrics presented
- Can become falsely low if using bad metrics

• Budget

- Funds the right technology, pays for training, helps the SOC operate efficiently
- Directly feeds human capital

Budget, Metrics, and SOC Perception

The operational efficiency that is attained will certainly improve human capital, but that's not all, it should have a clear impact on the metrics being reported to management as well. The thing about metrics is they are the main communication channel that informs the business if they are getting the ROI they expect out of the SOC or not, which means anything that affects business can have an enormous impact on the group.

In an ideal situation, since improved operational efficiency will have a clear net positive effect on metrics by improving consistency and shortening response times, the perception of the SOC will also be improved. (This, of course, assumes you have selected metrics that will convey these items.) When the perception of the SOC improves, the budget provided to the SOC stays flowing or even increases. That money then flows back into the group as investments for better tools, training, and more that will all add to the virtuous cycle of building human capital. This largest positive feedback loop may be the most important of all—without the backing of management (which is fed by the metrics the SOC provides), the SOC may cease to exist altogether, making all efforts to improve the other factors useless.

Additional Findings on Metrics

Other interesting research findings:

- Defining good operational metrics was *hard*
- Found analysts were more receptive to hearing feedback from management when they *believed* in the metrics
- Metrics should show meaningful *effort*
 - Example: Spending time qualifying a false positive, writing new analytics
- Bad metrics masked problems, lead to negative consequences
 - Example: Not showing when more people are needed in the SOC, analysts became over-burdened with no way to quantify it
- It was hard to quantify analyst time usage due to use overlap

Additional Findings on Metrics

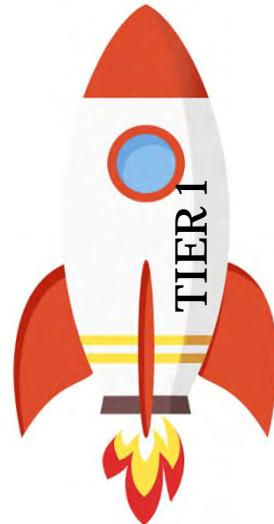
There were some other details the researchers found on metrics that are worth discussing as well:

- There was a struggle to define good, representative, operational metrics. This is a struggle experienced by nearly every team. Later on, we will review some methods for determining the right metrics for your group.
- Analysts were more receptive to feedback on their performance if they believed in the metrics being used and agreed that they were fair.
- Metrics should not just show incident data, but also meaningful *effort* put in by the SOC. One example given in the paper is that of an analyst that said when they spend time investigating a false positive, it showed up as wasted time in the metrics. Is this really wasted time? No, the situation must be figured out, it just didn't turn out to be an attack (which is a good thing!) Not every day will bring an incident, and metrics that don't show effort may leave the organization wondering, "Do we really need to pay for all of this? What is the SOC doing when not dealing with an incident?". Metrics around effort can help display this additional effort.
- Bad metrics mask problems and can lead to negative consequences. The example given is that of a team that badly needed more personnel to help out, but the metrics didn't show it. This had the effect of analysts being overburdened and claiming they needed help, but with no data to back it up. If left in this situation for too long, burnout will undoubtedly result due to overwork.
- Attempts to quantify what analysts spend their time on were difficult. The example was given of a SOC that tracked time spent on operational tasks vs. time spent on projects and other work. The analysts thought certain things were operational tasks or had operational impact and, therefore, were partially operational while management had a different definition, leading to failure of tracking the time in a meaningful way.

Burnout Mitigation Advice for New Analysts

Focus areas:

- **Growth:** Put them on the express elevator
 - Limit "false ceilings" (restricted tool or data access)
 - Focus on giving a **variety of tasks**
 - Watch out for alert "cherry picking"
 - Get them ready to be a tier 2+ as quickly as possible
 - Give stretch goals for learning at the limit of their comfort level
- **Skills:**
 - Provide the right training when ready
 - Give them practice on your *actual* equipment



Burnout Mitigation Advice for New Analysts

Given this model, hopefully, you have started to think about some ideas on how to improve the day-to-day life in your own SOC. To add to the recommended fixes already present, here are some experience level-specific recommendations and areas of focus to consider that, in my experience, can lead to long-term enjoyment and retention of employees.

For new analysts, the biggest focus areas tend to be growth and skills. Many people come into information security with a burning desire to devour every bit of information they can get their hands on. If they step into their new job full of inspiration and find themselves limited and immediately maxing out on what they can do, you can be sure they won't be there very long.

In order to optimize the experience for new folks, above all, make sure they are learning everything as fast as they are interested in having it thrown at them (but not so much they feel overwhelmed). Focus on giving new people a variety of tasks and adding additional ones as soon as they master what they have on their plate. You'll know it's time to start pushing them toward something new if they start looking bored or you notice them always taking on the same types of alerts—"cherry picking" them, as it's called, because they have become so familiar with how to deal with that one specific situation.

Provide on-the-job, peer-led, and formal training when the time is right and, ideally, have them learn on the actual tools used for the job when possible. This all ensures analysts are on the right track to bring them toward a tier 2 level job (if you have a tiered SOC) as quickly as possible and they are brought up to speed as quickly as they can handle it.

Burnout Mitigation Advice for Experienced Analysts

Focus areas:

- **Growth:** Keep it moving
 - Never let them find the "ceiling" of what they can learn at your org.
 - If there is no one left to train them, provide additional outside training
 - Give them special assignments to exercise new skills
 - Provide opportunities for out-of-group rotation
- **Creativity:** Give them time and an outlet, and get out of the way—these people know what they're doing
 - Continuous improvement efforts (they know best what needs fixing)
 - Automation and integration
 - New tools / scripts / software



Burnout Mitigation Advice for Experienced Analysts

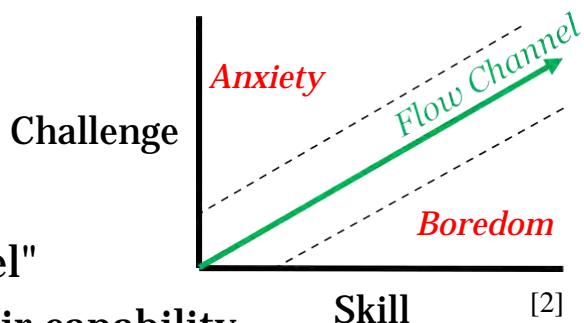
Keeping experienced analysts is similar in nature to a newer analyst, but also much harder. These folks have seen it all, and it's harder to hit them with a continual variety of tasks. While growth is an excellent thing to continue to focus on where possible, it can become more difficult to provide to top-tier analysts. When your "best" analyst looks around and sees no one else for them to learn from, what will they do? Will you provide them with outside resources to keep them growing, or will they start looking for other jobs, sensing that they've found the peak?

To mitigate this risk, the creativity factor can be leaned on to create new and more difficult challenges for these analysts to take on. With their high level of skill, you should be able to give them a vague description and give them the autonomy to take on the task in their own way. In other words, give them orders and move out of the way—these are people you can trust. Not only that, but they're also the best at what they do and are most likely to be able to make massive improvements in your operations since they know the ups and downs inside out. Leverage this power and ability to work on their own to produce new tools, automation, and any other process improvements you can assign.

Optimal Tasks for Growth

How to steer people in the right direction:

- Know what your analysts are
 - Uncomfortable with
 - *Too* comfortable with
 - Interested in learning
- Get them into their "flow channel"
- Stretch people to the *edge* of their capability
- "Deliberate practice" – *Anders Ericsson, Peak*



Optimal Tasks for Growth

With the guidance given to us by the human capital model, how, in everyday terms, can we steer people in the right direction? One way is to gear the tasks they take on toward the edge of their capabilities—tasks that are not so easy they're boring, but not so hard they're stressful. Tasks like these tend to put people in the flow state (or "flow channel" as popularly described in multiple books by Mihaly Csikszentmihalyi)—engaged, and challenged, and, of course, growing and building skills.

Doing this successfully, of course, implies understanding what each of your employees is capable of, interested in, and finds easy and difficult. Having regular one-on-one meetings can help keep the lines of communication open on what they're enjoying and want to do more of, and if they are becoming complacent or doing tasks that have become no longer a challenge. For additional information on quick learning and the state of flow, check out the idea of deliberate practice in the book "*Peak*" by Anders Ericsson¹ and "*Flow: The Psychology of Optimal Experience*" by Mihaly Csikszentmihalyi².

1 <http://peakthebook.com/index.html>

2 <https://www.pursuit-of-happiness.org/history-of-happiness/mihaly-csikszentmihalyi/>

Thomas Gilbert's Behavior Engineering Model (BEM) ¹			
	Information		Motivation
	1. Data	2. Resources	3. Incentives
Environment (external)	Does the individual know what is expected of them?	Do people have the right tools for performance?	Are adequate financial incentives that are contingent upon performance available?
	Do people know how well they are performing?	Are tools and materials designed to match the human factors of performance?	Are nonmonetary incentives available?
	Are people given guidance about their performance?		Are career development opportunities available?
	4. Knowledge	5. Capacity	6. Motives
	Do people have the skills and knowledge needed to perform as expected?	Is performance scheduled for times when people are at their best?	Has a motivation assessment been performed?
	Is well-designed training that matches requirements of performance available?	Do people have the aptitude and physical ability to perform the job?	Are people willing to work for the incentives? Are people recruited to match the realities of the job?

SANS

MGT551 | Building and Leading Security Operations Centers

19

Thomas Gilbert's Behavior Engineering Model

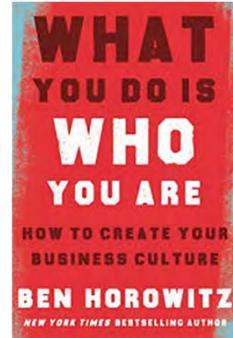
A useful model for understanding human behavior from the field of Human Performance Technology is Thomas Gilbert's Behavior Engineering Model¹ (BEM) shown on the slide above. This model has direct relevance to managers for employee performance management and preventing burnout. When trying to understand why someone is not living up to expectations, the approach should be to try to analyze the situation and consider both the person's environment and individual motivations. The BEM is meant to help you do just that by breaking down the factors that may be causing the issue.

The BEM is split into six sections with three being external factors of the environment, and three being internal factors of the individual. Within each section, there is a further sub-specification that can help you narrow in on why an employee may be not meeting expectations. External factors include things like whether they know what is expected of them, if they have the right tools for the job, and if they are being offered the proper incentives. Internal factors include items such as having the knowledge to perform the job, if their performance is scheduled for the point where they're at their best (think working midnight vs. the day shift), and if their motivations are in line with the incentives being offered.

[1] <http://hpt2014.weebly.com/gilberts-bem.html>

Building the Right Culture (1)

- Culture is "how your [team] makes decisions when you're not there"
- Not something you can design or force, but something you shape
- Culture will reflect the sensibilities of its leader(s)
- Starts with recognizing your own flaws and limitations
- Pick the virtues that will help your team accomplish its mission



Building the Right Culture (1)

In his book "What You Do Is Who You Are," venture capitalist and CEO Ben Horowitz frames the challenge of building a culture not as something you can design or force, but something that must be shaped and exemplified. Though Ben sums up culture as "how your business makes decisions when you're not there," building a culture is much more complex than trying to make the team do what you want when you aren't around. Think of the teams you've seen or been a part of that were really effective. Were their leaders effective? Highly technical? Hard-working? Understanding of the business challenges from a security perspective? If the answer is no, consider how long those teams remained highly effective (probably not very long).

The culture of your SOC team will reflect your sensibilities as a leader. That means the first step in building the SOC culture you want is knowing yourself, identifying your own flaws and limitations, and seeking out team members, colleagues, or mentors that compliment or balance those limitations. Maybe you are a highly skilled analyst but struggle with some of the managerial aspects of leading a SOC. Maybe you are a good leader but aren't as technical as you'd like to be. Understanding these limitations will help you determine what skills or abilities must be rounded out within your team, either through hiring, mentoring, or seeking mentorship yourself.

Building the Right Culture (2)

Signs you have a culture problem:

1. The wrong people are quitting too often
2. The team is failing at strategic priorities (identifying attacks, resisting intrusions, meeting SLAs, etc.)
3. An employee does something that shocks you (not in a good way)

Possible culture breakers:

- The Heretic – difficult to turn a negative person around
- The Flake – finding the source can be tough but necessary
- The Jerk – sets a bad precedent and discourages others from contributing
- The "Prophet of Rage" - perfectionists who don't work well with others

Signs You Have a Culture Problem (2)

If you have done the work of defining and measuring good metrics, but you still feel like the team isn't operating at a high level, you may have a culture problem. Here are some signs that the team may be off track despite good processes, good people, and good measures:

1. The wrong people are quitting too often. It's common for people to leave for new opportunities – especially in this industry – but if your team experience higher-than-average turnover? This can have significant operational and cost impacts and create a snow-balling effect of dropping morale. If you've built your team based on people you think are a good fit for the culture you want, but they are still leaving, you may not have the culture you think you have.
2. If user/culture satisfaction is low or you're missing your SLAs, but can't identify any obvious operational reasons for that, some of the issue may come down to a culture that doesn't encourage the kind of accountability and work ethic you need.
3. An employee does something that shocks you – like intentionally skirting process or ignoring evidence – this may be evidence that your culture enables this sort of thing. When a team member surprises you in this way, investigate why – was it an aberration or part of a larger pattern?

As a manager and leader, it's your responsibility to help shape the culture you want. This may mean identifying team members who are disruptive and taking action, even if they are technically proficient or solid contributors. In "What You Do Is Who You Are," Ben Horowitz breaks down a few of the personality types he's encountered that can disrupt an otherwise positive team culture: The Heretic, The Flake, The Jerk, and The Prophet of Rage (taken from a song by Public Enemy). You may recognize some of these personality types. Failing to recognize these disruptive behaviors and take action can be every bit as damaging to team morale and a positive culture as the behaviors themselves. Let's talk about what we can do to address them.

Addressing Bad Behaviors

A common pitfall in maintaining good culture is failing to correct bad behaviors; take these actions to fix what may be broken:

1. Give feedback on effects, not behaviors themselves
2. Recognize that you can't change personalities
3. Focus on things your team can do to improve



Addressing Bad Behaviors

Addressing toxic behavior is a skill that doesn't always come naturally, and many managers handle it poorly (or not at all). Here are some things you can do to address behavior that is preventing the team from enjoying a positive culture:

1. Give feedback on behaviors' negative impacts, not the behaviors themselves. Break your comments down into three parts:
 - Here is what I observe
 - Here is the immediate result it has
 - Here is the effect it has on the team
2. Recognize that you can't change someone's personality or behavior, and not all personalities are compatible. You may have to work with the individual(s) in question and the larger team to help them be more understanding or accommodating, *or* the individual just may not be the right fit.
3. Focus coaching in one-on-one feedback sessions on things you want your team member to improve, emphasizing things they can do to reduce negative impacts on the team.

Your Job as a SOC Leader

Be like Bounce!

1. Remove wrinkles
2. Absorb static
3. Translate the unknown
4. Stay fresh
5. Career bounce



SANS

MGT551 | Building and Leading Security Operations Centers

23

Your Job as a SOC Leader

Cybersecurity executive Jon Check wrote about how the seemingly unrelated chore of doing laundry, and the function of dryer sheets, has influenced his thinking about leading teams. He recommends the following five leadership practices that managers and executives should integrate into their teams:

1. **Remove wrinkles** - Remove unnecessary practices that keep employees from achieving their goals and a high level of job satisfaction, like needless bureaucracy, repetition, excessive meetings, and other factors that can contribute to a stressful or frustrating environment.
2. **Absorb static** - As a security manager, you will experience similar frustrations with corporate governance or leadership ; avoid bringing those frustrations into your communications with your team. It is your job to absorb the "noise" and help your team focus on their mission and day-to-day challenges.
3. **Translate the unknown** - Office communications should be relatively straightforward. But just like dryer sheets may have multiple languages and symbols on the box, sometimes we communicate in different ways. Good leaders will take the time to explain concepts and expectations clearly, ensuring all parties understand. Supplementing that communication with one-to-ones and ongoing dialog will help you avoid miscommunication.
4. **Stay fresh** - Encourage your employees to maintain a healthy balance between work and other aspects of their lives, and to deviate from their normal daily routine when SOC tasks allow. Help your team members preserve their personal time by minimizing off-hours communications and taking their allotted time off to recharge.
5. **Career bounce** – much like washing and drying cycles, too much repetition can lead to undesirable results. It's your job to help your team members grow through new assignments, projects, or new roles. Support them in these endeavors, even if it might mean that you will ultimately lose out on their skills and expertise.
6. Unlike a single use dryer sheet, your management skills must be effective time and time again. Applying these principles will help your team stay happy and focused and lead to a more effective and efficient SOC.

You can read more about Jon's "Bounce" approach here: <https://www.linkedin.com/pulse/laundry-isnt-chore-its-leadership-training-jon-check>

Management Debt

- Popular forms:
 - Putting “two in the box”
 - Matching competing offers
 - No performance management or feedback process
 - Keeping people on tasks and projects they dislike



SOC Management Debt

Some of you may already be familiar with the concept of technical debt: additional work caused by choosing a more convenient near-term solution instead of a better, though perhaps more challenging, long term solution. In his book “The Hard Thing About Hard Things¹,” Ben Horowitz coins the phrase “management debt” to describe a similar concept in managing people. Ben writes from the perspective of leading startups, but these concepts are relatable to managing security operations teams as well. While taking on some management debt is unavoidable, incurring too much can result in management bankruptcy – a lack of social or management “capital” required to effectively run the team.

The most common forms of management debt are putting “two in the box”, matching competing offers, lacking performance management and feedback processes, and keeping people assigned to tasks they don’t like. Again, there are circumstances where these decisions might be purposeful and even necessary. But it’s important to understand that making them comes with a cost that will eventually come due.

[1] <https://www.amazon.com/Hard-Thing-About-Things-Building/dp/0062273205>

Two in the Box

- Most often applies to team or task leads in ops
- More than one employee who fits in the same role
- Payment comes due when things fall through the cracks, confusion about who is leading the task or team



Two in the Box

Here's a scenario: you have two fantastic engineers on your team. One of them knows the infrastructure backwards and forwards and can handle any issue that comes up. The other can anticipate change and ensure the environment evolves to adapt. Both are ready to be the Engineering Team Lead in charge of a team of security engineers. You don't want to kill their motivation and there is tons of work to be done, so you promote both to "co-leads". What's the issue here?

For one, who is ultimately responsible for Team Lead tasks? Who do the other security engineers go to when they have a problem or a question? Having two people fill a role meant for one person may help you avoid a tough decision right now, but the payment will come due when the team is impacted by this lack of clarity in roles and responsibilities.

Matching Offers

- Analyst gets an offer for more than you're paying any other team member
- Word gets out, chances are high the analyst will ultimately leave anyway
- Payment comes due when morale dips and other competing offers start coming in



Matching Offers

This is an EXTREMELY common scenario in our industry: an analyst comes to you with a fantastic offer from another company. In fact, it's more than anyone else on the team is making, but you can keep the analyst if you're able to match it. Do you?

The answer, like so much else, is – *it depends*. This may be a critical team member that you can't afford to lose (is your team mission oriented or talent oriented?), so matching their offer might be necessary to maintain your capabilities. But here is how that is likely to play out: teams talk, and you should assume that at some point other team members will discover the increase – especially when their colleague who got an amazing offer has suddenly decided to stay with the team. Now, the rest of the team has been given the impression that this is a viable method of getting their own increases, so they start interviewing. Before you know it, your retention problem has gotten exponentially worse and you'll have a lot of explaining to do when the next budget planning cycle rolls around.

Finally, compensation is often something that can attract people, but it usually won't keep them long term. Matching competing offers should be considered a temporary stop gap, not a long-term retention strategy.

Lack of Performance Management Process

- No time or desire to implement
- Don't want things to get "too corporate"
- Teams execute best when everybody is on the same page, constantly improving
- Payment is due when performance suffers and you don't know why or struggle to correct

Lack of Performance Management Process

There can be a number of reasons why a team might lack formal performance management processes, but constructive feedback isn't possible without setting clear guidelines and expectations for individual team members. A lack of formal processes and feedback is common in high-performing teams where individuals are empowered and trusted to make the right decisions. However, constructive input is necessary even for the most skilled and experienced staff. Seniority is no guarantee of great performance, and payment comes due on this kind of management debt when someone's performance begins to decline, and you have no idea as to why.

Measuring performance and providing timely, direct feedback is fundamental to good communications and a healthy team environment, not to mention the best way to get a handle on issues before they become larger problems for the team.

Keeping People on Tasks and Projects They Dislike

- Someone may be great at something they hate doing
- Sometimes necessary; not every task can be fun
- Payment comes due when the person you were relying on gets fed up and leaves



SANS

MGT551 | Building and Leading Security Operations Centers

28

Keeping People on Tasks They Dislike

As awesome as blue teamwork can be, it's not always sunshine and rainbows. As a lead or manager, you'll sometimes have to ask someone to do a task or project that isn't very fulfilling or engaging but must be done. Certain compliance tasks, documentation, and user support are examples of things that don't always excite us as defenders and often can't be automated. In these situations, you may find that certain team members have a talent for getting the job done. It can be tempting to simply let these individuals continue getting the job done, but keep in mind that ability does not always equal job satisfaction.

Conducting one-on-ones with your team members can help you identify tasks where performance might be stellar, but work satisfaction is low. Identifying commonalities across the team will let you know where you might need to re-engineer a process or at least rotate responsibilities.

The payment for allowing this kind of management debt to accrue comes due when someone you were relying on to do The Thing No One Likes leaves unexpectedly, possibly taking key knowledge and skills with them.

Staff Retention and Burnout Mitigation Summary

- Burnout is a **human capital management problem**
 - Optimize for **Growth, Skills, Empowerment, and Creativity**
 - Also: **Automation, Operational Efficiency, and Metrics**
 - Each form a positive feedback loop!
- New analysts should be learning as fast as possible
- Experienced analysts can use ...
 - Special challenges to continue growth and creativity
 - Outside training to prevent stalling out skill progression
- SOC management debt will eventually come due
- If things have already gone bad, find the root cause fast and fix!

Staff Retention and Burnout Avoidance Summary

This module may be one of the most important in the class as it deals with how to keep your employees engaged and excited at work. This is, of course, not only good for the organization, but more important for the mental health of everyone on the team. No one wants to work in a place where everyone is miserable and the team is constantly changing due to turnover. While these techniques won't necessarily fix issues when people don't want to do this type of work, they *can* prevent those who would otherwise enjoy it from having a bad experience.

If you have issues with employees who genuinely just aren't interested in the job, rotations to engineer, threat intel, other security groups, or even into other IT groups can help save good talent for the company before that person walks out the door. Burnout has causes that have now been discovered through the highlighted research. The hope in sharing this model with you is that you will read the cited full paper and consider how to implement these ideas in your own SOC. With commitment, the factors that influence burnout can be minimized to give ourselves, our teams, and our organizations the best possible chance at success.

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

- **Effective Execution**
 - Staff Retention and Burnout Mitigation
 - *Exercise 5.1 – Training and Career Development Planning*
 - Metrics, Goals, and Effective Execution
 - Measurement and Prioritization Issues
 - *Exercise 5.2 – Creating, Classifying, and Communicating Your Metrics*
 - Continuous Improvement
 - Strategic Planning and Communications
 - Analytic Testing and Adversary Emulation
 - *Exercise 5.3- Purple Team Assessment*
 - Automation and Analyst Engagement
 - Summary and Cyber42 – Day 5



This page intentionally left blank.

EXERCISE 5.1

Exercise 5.1: **Training and Career Development Planning**

OBJECTIVES

- Use the SOC-CMM model to measure knowledge management and training in your SOC
- Inventory technical skills and knowledge within your team
- Identify training objectives to fill gaps
- Develop a training plan for SOC staff
- Measure progress towards learning objectives



Exercise 5.1: Purple Team Assessment Planning, Execution, and Tracking

Please go to Exercise 5.1 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

- **Effective Execution**
- Staff Retention and Burnout Mitigation
 - *Exercise 5.1 – Training and Career Development Planning*
- **Metrics, Goals, and Effective Execution**
- Measurement and Prioritization Issues
 - *Exercise 5.2 – Creating, Classifying, and Communicating Your Metrics*
- Continuous Improvement
- Strategic Planning and Communications
- Analytic Testing and Adversary Emulation
 - *Exercise 5.3- Purple Team Assessment*
- Automation and Analyst Engagement
- Summary and Cyber42 – Day 5



This page intentionally left blank.

Metrics Motivation

How do we know where we are / where we're going? Metrics!

"You cannot manage what you cannot measure"

- Peter Drucker

- SOC metrics are confusing, difficult, and high stakes
- But we *must* get them right to...
 - Show an ROI, justify budget
 - Validate we are operating within expected parameters
 - Keep on track to hit new initiative objectives
 - Spot attack trends
 - See areas for improvement



Metrics Motivation

How do we know where the SOC is and where it is going? Metrics of course! As legendary management consultant Peter Drucker said, "You cannot manage what you cannot measure", which means that unless we know how to measure our SOC, we likely do not have any true control.

The problem is that nailing down a solid set of meaningful and actionable SOC metrics is difficult. Metrics are one of the most debated and discussed topics in many Blue Teams as they have such a profound impact on how the SOC is perceived and how it operates. It is a challenge we must rise to regardless of the difficulty because they are how our group will be judged. These measurements must be used to show a SOC ROI, ensure security is operating within expected parameters, keep our new initiatives on track, and more. Let's dive in and see how we can tackle this extremely important problem in a clear and repeatable way.

In This Module

- Due to their complexity, we'll take a different approach
 - Not just going to give you a list of metrics
 - Going to cover the concepts of what makes a good metric and why
 - Learn to *derive* which metrics we should use
- In this module:
 1. Metrics, KPIs, and OKRs—the different types of measures
 2. Measuring daily operations vs. new initiatives
 3. Using good metrics to guide and drive execution
 4. The how and why behind metrics
 5. Practical metric collection issues

In This Module

While some sources may throw a list of security metrics at you, we want to be more comprehensive than that here. Instead of just throwing a list of measurements at you without explaining why they're important, we're going to cover the concepts behind which measurements we might be interested in. This way, we'll be taking the "teach you how to fish" instead of "give you a fish" approach to metrics. The goal is, by the end of this module, to not only be able to come up with metrics that could be useful to you, but also know how to derive *why* they are good metrics and how to tie them to the objectives and goals we have for the SOC.

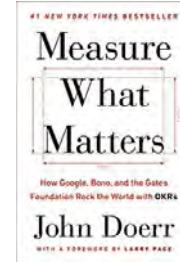
To cover this, in this module, we'll start with introducing some of the most popular goal management and execution frameworks out there, and the measurements that are generated and used in the course of applying them. We'll talk metrics vs. Key Performance Indicators vs. Objectives and Key Results, execution strategy, and using metrics to succeed at new initiatives, as well as what makes a good metric and how to make collection easy.

Standing on Shoulders of Giants

For this piece, we'll lean on two books and systems

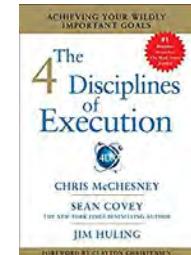
1. "Measure What Matters" by John Doerr

- About execution via Objectives and Key Results (OKR) system
- Invented by Andrew Grove at Intel
- First described in "High Output Management" (1983)



2. "The Four Disciplines of Execution: Achieving your Wildly Important Goals" by McChesney, Covey and Huling

- Separating "important" from "urgent" in daily operations
- How to *execute* on those important priorities
- Selecting and using the right metrics to lead you to success



Standing on the Shoulders of Giants

The two books that this section heavily relies on are "Measure What Matters", by John Doerr and "The Four Disciplines of Execution: Achieving Your Wildly Important Goals", by Chris McChesney, Jim Huling, and Sean Covey. From these two books, I've extracted the core messages that are applicable to SOC metrics, and we will use those pieces to help us come up with the best metrics and strategies for running the SOC in the best way possible.

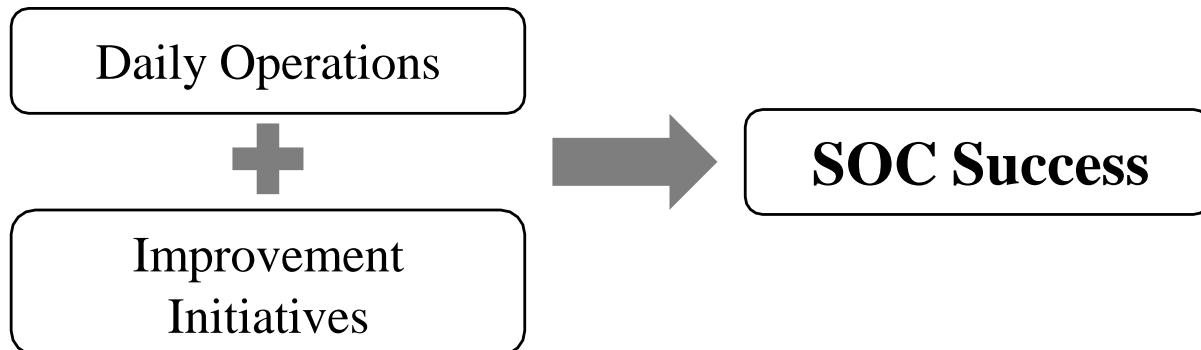
"Measure What Matters", released in 2012 is a book that lays out the OKR system, a famous framework for setting and hitting objectives that is used in major organizations such as Google, Amazon, Facebook, Intel, and more. This book, as well as the supporting reference info we will cover, introduces the concepts of metrics, KPIs, OKRs and how they all relate to help you understand whether your daily operations are on track as well as if your new initiatives are headed in the right direction.

The second book "The Four Disciplines of Execution" is a personal favorite in the realm of setting priorities and creating a system to ensure those priorities are followed through on. The system itself is great, but in terms of this section, it also contains key guidance on which metrics to use and how to employ them to guide your team to victory in execution. It also sets a great precedence on separating "urgent" from "important" task—a problem that plagues many SOCs and holds them back from ever improving. This is another all-around great read that I recommend to all managers.

Two Components of SOC Success

A SOC has two main activities: **ops** and **improvements**

- Both require different mindset for measurements and tracking
- Both compete for your time



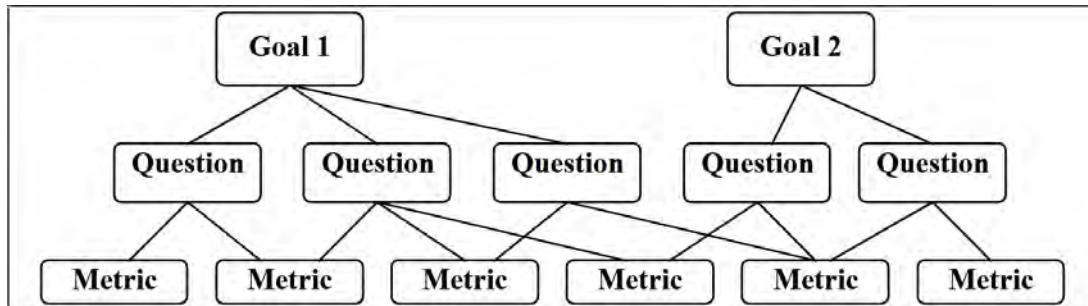
Two Components of SOC Success

Before we jump into the OKR and 4DX introductions, there is one quick division to point out. Throughout this section, we will be making a division on the types of activities a SOC needs to do. One of those things is maintaining daily operations. In other words, keeping everything you've already established in normal working order. The second, and competing priority is finding time to do things *better*. As much as every SOC wants to do this, it is a constant struggle. For success, we need to tend to both of these pieces, and each has their own mental models and measurement types that go along with them. Keep these two items in your head as we progress through the rest of this module.

Key To Metric Usefulness: Top-Down Goal Alignment

One key aspect of successful metrics:

- **Top-down derived metrics** for goal alignment
- Goal-Question-Metric¹ (**GQM**) is one system for this
 - You will see the GQM system in action in the metrics exercise



Key To Metric Usefulness: Top-Down Goal Alignment

Regardless of the specific system you use to derive your metrics, there is one bit of advice that is nearly universal about metrics—they must directly flow from, or be tied to, an objective or goal. When deciding what to collect, being able to tie the collection of that metric directly to a "why" is a technique to help guarantee the usefulness of the metric. If your metrics serve as a measure that shows whether you are operating within expected parameters, or measure your progress towards a specific goal, then it is very clear that having that number assists your ability to stay on track and is worth your time to generate.

One system for explicitly tying and deriving metrics from a goal is the Goal-Question-Metric system, which you will get a detailed introduction to, and a chance to try, in the next exercise. In the GQM system, the idea is to first decide on what goals you need to hit, what questions can tell you if you've hit that goal, and the data needed (metrics) to answer those questions. If your chosen metrics answer a question that answers whether you are meeting your stated objective, then there is a direct line as to why each is useful. Before we get too far into the system however, we will first take a moment to define metrics and associated concepts in a bit more detail.

[1] <https://www.cs.umd.edu/users/mvz/handouts/gqm.pdf> (<http://mgt551.com/gqm>)

Metrics

Metrics: A tool used to measure something

- Components:
 - **Current Value** – The value the metric has at this moment
- Calculation method should be clear since methods can vary
- Example SOC metrics:
 - Alerts in the alert queue
 - Percent of devices monitored
 - Infections in the last seven days
- **Purpose:** Measuring a business process
 - Notice: No reference point or context required to meet this definition



Metrics

First up are some definitions to make sure everyone is on the same page with the terms we'll be using. Metrics is a term that is thrown around a lot, but for the purposes of this discussion, we want to define it clearly. Metrics are a tool used to measure something—a quantifiable measurement often used to track the status of some business process. When it comes to metrics, everyone should clearly understand the way the metric you are reporting is calculated. In some cases, organizations may use the same name for a metric but calculate it differently, so it pays to be clear.

The only property of a metric is the value that metric has at the current time. For a SOC, some metrics you could calculate are the number of tickets in the queue, percentage of devices monitored, or the number of infections you've seen in the last seven days. Notice these are pure measurements with no context or benchmark to say whether they are good or bad (of course we know what we *want* the answer to be in some of these cases, but it is not inherently part of the concept of a metric itself). The purpose of the metric is purely to measure something, ideally in a quantifiable way.

Key Performance Indicators (KPIs)

KPIs: A tool to keep track of how a key area is performing

- Measures the **key area** using a metric and a **target value**
- Components¹:
 1. Metric
 2. Current Value
 3. **Target/Threshold Value** – the desirable bounds of the metric
- Performance measurement that *evaluates* an ongoing process
 - Unlike a metric, tells you if you have a problem, but *not* how to fix it
- **Purpose:** Tells you if you are **maintaining the status quo**
 - Used for **important daily operational metrics**
 - A way to monitor for "**business as usual**"



Vehicle KPIs

Key Performance Indicators (KPIs)

A "key performance indicator" (KPI) as described in the OKR system is often confused with a metric but is not the same thing. A KPI takes the **metric** and **current value** and builds on it by adding a **target** or threshold that metric should fall within, as well as the assumption that the metric itself is measuring something important for performance.¹ In other words, a KPI is a measurement of an ongoing important process—something that would be part of your daily operations.

The goal of a KPI is to tell you whether you are still within the expected parameters for that metric and if you are maintaining the status quo or need to course correct. The best real-life comparison would be the gauges on your car dashboard. Each individual gauge is a calculation of some metric to which the needle is pointing to the current value. What makes this more than a metric though is that dashboard gauges also usually have the bounds of "normal" included within them. As the picture on the page above shows, the oil temperature and pressure gauge are a KPI because you can see if the temperature is getting too hot or the pressure is too high or low. They're both metrics of ongoing processes with bounds and thresholds of acceptability outside of which you may need to take action.

[1] <https://www.perdoo.com/resources/the-anatomy-of-a-kpi/>

Potential SOC Operational Metrics & KPIs

Think by SOC process area:

- **Collection** - Source data coverage (percent or by count)
 - Grouped by data type, subnet, system type, count, EPS, outages, etc.
- **Detection**
 - False positive / true positive count & ratio grouped by data feed, signature, etc.
- **Triage / Investigation Times**
 - Alert count, time to acknowledge, assign, contain, eradicate, recover
- **Incident Response** - Incident categorizations
 - VERIS (actor, actions, asset, attributes)
 - US-CERT Incident categories observed, impact and more
- Remember: "*Metrics are like lightsabers – they can be used for good and for evil*"¹ – Carson Zimmerman

Potential SOC Operational Metrics & KPIs

This slide lists some of the potential metrics or KPIs you might consider collecting for your SOC. Again, when picking metrics and KPIs, consider a question that it's important to know the answer to, and pick the metric that brings you that answer. Don't just blindly pick metrics because they *can* be collected, they should have a purpose and help the team monitor for issues or answer important operational questions.

When choosing which items to collect, consider each stage of the SOC process. As we'll discuss more in an upcoming slide, it's important to get samples of how each step in the SOC process is functioning.

[1] <https://www.youtube.com/watch?v=RwO-uT2jh6E>

Threat Intelligence Metrics

- How can we measure threat intelligence feeds?
- Problems with threat intel measurement
 - Don't know what *isn't* included (false negatives)
 - Difficult to measure timeliness and completeness
- (Goals) The ideal feed has...
 - Accuracy of items on the list
 - Timely IOCs updates / additions
 - Useful context provided
 - Focus on your industry, threats, attack types/groups
 - Usable, ingestible format

Threat Intelligence Metrics

Developing metrics around threat intelligence application is another area that many SOCs are interested in. The problem, however, is that due to the messiness and overlapping nature of data and coverage in this space, coming up with meaningful and representative measurements can be difficult. For example, one of the key qualities of good threat intel is its completeness and context provided with tactical level IOCs.

Taking the approach of starting with our goals, we can first clarify what we want from a threat intelligence feed, and then look for ways to either measure these items directly, or measure the effects that they would have. The ideal feed for example, would be one that contains some information on all relevant attackers and attacks, is updated very frequently, provides ample context to qualify alerts, and is highly accurate. On top of that, it should be delivered in a format that is easy to ingest with the tools the SOC utilizes, making it easy to ingest and apply.

Threat Intelligence Measurements

- **Ideal analyst experience**
 - Threat intelligence alerts of malicious activity, analyst finds it accurate, receives useful context, and can attribute it
- **Per-feed, goal-focused threat intelligence metrics:**
 - **Feed Size** - Number of data items per feed, more is better (assuming constant accuracy)
 - **Accuracy** - % true positive for IOC matches, change over time
 - **Completeness** - For *all* incidents that occurred, how many matched with an item from that feed?
 - **Timeliness** - Time from attack to matching IOC being added to the list (may be zero or higher for retroactive detection),
 - **Context** - Time from match to alert confirmation / attribution (proxy for context)

Threat Intelligence Measurements

While measuring threat intelligence can be difficult quality and completeness can be difficult, we can still attempt to define some quantitative measures that may serve to at least correlate with the quality of a given threat intelligence feed. Given our goals of completeness, accuracy and speed of updates, here are some ideas for ways you can assess the usefulness of your threat intelligence feeds. (Note that to use these methods, each alert that is triggered based on a match will need to be tagged with the threat intelligence feed that it came from.)

- **Feed Size** – The number of the items in the feed itself. Holding other variables steady such as the accuracy, a feed with more malicious items to match against is better than one with fewer.
- **Accuracy** – For each time an alert was produced via a match from a given feed, how many times was it truly an indicator of an attack? Remember while 100% seems like the goal here, this number needs to be put into the context of the size of the list. A threat intel feed with one accurate item could easily be 100% accurate, but still not very useful.
- **Completeness** – Though it is difficult to measure false negatives in a confident way, we can make an attempt by looking at malicious things that happened that *didn't* match. Since the ideal threat intel feed would match 100% of the malicious activity that happens in the environment, doing the reverse of the above accuracy measurement and calculating the count of percentage of incidents that occurred that matched a feed might give an interesting result and highlight feeds that are better than others (understanding that no feed will likely ever approach 100% on this metric).
- **Timeliness** – Since the ideal threat intel feed has items to match on *before* you experience an attack using them in the environment, a SOC could look at detections from a list and put them into two camps. One being alerts that fired because an IOC on a list matched an in-progress attack, the second being the number of matches that occurred after the fact (implying the attack occurred, and the IOC was added later on, leading to a retroactive detection).
- **Context** – While it's very difficult to measure a qualitative aspect like context numerically, one proxy for this might be the speed at which analysts can move from the "alert acknowledged" stage to the "confirmed false/true positive" stage, or even perhaps to the "alert attributed to a threat group/attack tool/campaign" state. More context should make this happen more quickly, but there are many other factors that could affect this as well, depending on the environment and automation technology in use, a more creative idea may be needed for this.

Objectives and Key Results

A strategic goal management framework:

- Designed to define and measure **advancements** via **initiatives**
- **Objectives:**
 - The goal to be achieved or destination state
 - Example: "minimize successful phishing"
- **Key Results:**
 - How you will know you're making progress towards the objective
 - **Specific** and **measurable**
 - Example: "Less than 3 phishing infections per week"
- **Initiatives:** Activities undertaken to achieve key results
- **Key Results Components¹:**
 - Metric, Current Value, Target value +
 - **Start value** – The metric value at the beginning of a timeframe

Objectives and Key Results

Objectives and Key Results, unlike metrics and KPIs are parts of a goal-management framework which is described in the "Measure What Matters" book's OKR system.

Objectives in this system are the goal to be achieved or the end state that is desired. They do not have to be quantifiably described and often are not, that is left to the Key Results. An example of an objective may be "minimize successful phishing attempts in our organization."

Key Results, on the other hand, assist us in getting to the objective by answering how we know we're making progress. These must be *quantifiable* and *specific* measurements that tell us that we're making progress toward the stated objective. An example of a key result for our previous phishing example objective could be, "Reduce phishing incidents to less than three per week". A key result is similar to a KPI, but not quite the same in that a Key Result requires a **metric, current value, target value**, and **start value¹**.

The key difference to understand here is that Key Results are temporary and used exclusively for *new initiatives*, not daily operational measures like KPIs. While a metric or KPI could play a part in a key result, they are not the same thing since Key Results are about getting from a start value to a certain destination within a finite amount of time, whereas KPIs are meant to be a continual measure.

For an excellent in-depth explanation on developing OKRs and KPIs, see the excellent blog post from Perdoo referenced below¹.

[1] <https://www.perdoo.com/how-to-write-great-okrs/>

The Interaction of Metrics, KPIs, and OKRs

- **Metric:** Infections via phishing per week
 - Current value = 10
- **KPI:** Have less than five infections via phishing per week
 - Metric = Infections per week, **Target = less than five**
 - Current value = **10**, desired "status quo" was violated
 - You *could* use the OKR system to help fix the issue
- **Objective:** Minimize infections via phishing
- **Key Result:** Bring infections from phishing per week from 10 to less than five
 - Metric = Infections per week, Start value = 10, Current value=10,
 - Target value = five or less
- **Initiatives:**
 - Better spam filtering controls
 - User awareness training



The Interaction of Metrics, KPIs, and OKRs

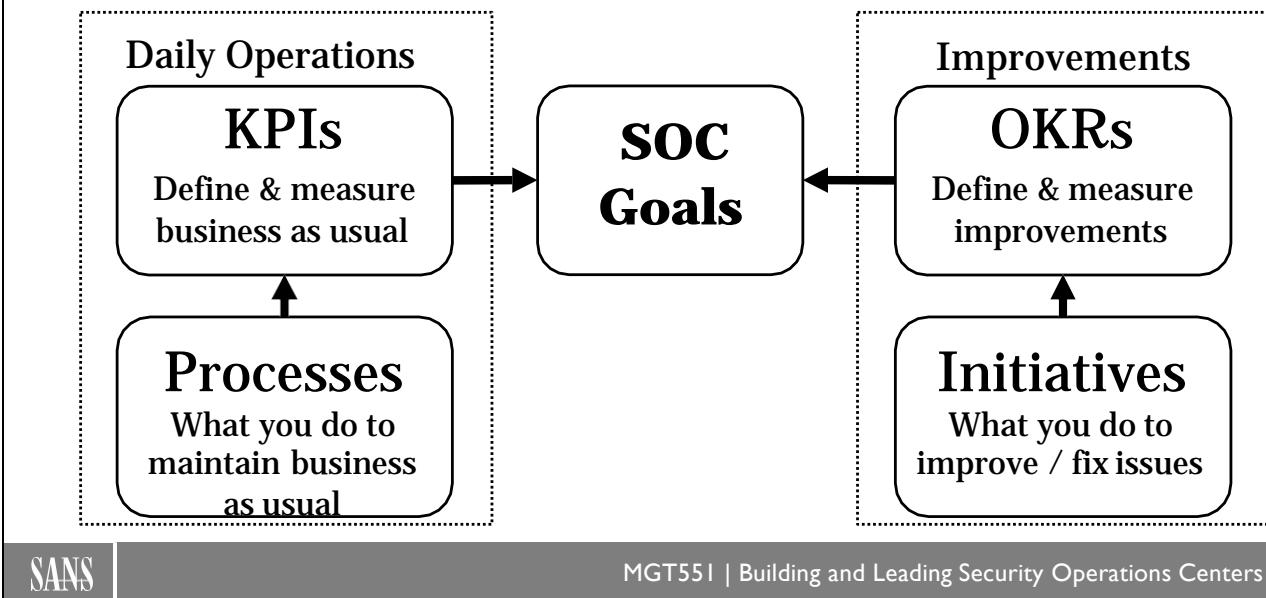
How do all of these items work together in the form of daily operations and new initiatives? Here's an example:

Let's say you decide to track infections that occur due to phishing per week. That number itself is just a metric. Since it is a key performance item for your SOC (because your business is worried about email attacks), you decide the acceptable level of incidents per week is five based on the long-term average you've seen. Anything more than that would be an indication that something important has changed—either you are getting worse at blocking spam, or the spam is getting better at tricking people, either way, you need to act!

While the KPI continues to be tracked as one of your daily operational KPIs, the most recent week shows that you've surpassed the allowed threshold of five and it has doubled to 10 incidents! What to do? If you follow the OKR system you may decide to make an objective and key result around fixing the issue. The objective could be, "minimize infections via phishing", and a key result for that objective might be, "bring infections per week from 10 down to less than five". Along the way, you would track the current value each week until the number has been brought back down. In short, a KPI that becomes out of line could be the starting inspiration for developing OKRs to pull the KPI back into line, but the two serve different purposes.

[1] <https://www.perdoo.com/resources/kpis-okrs-the-goals-that-drive-business-success/>

The Relationship of KPIs and OKRs to SOC Goals¹



SANS

MGT551 | Building and Leading Security Operations Centers

45

The Relationship of KPIs and OKRs to SOC Goals

Put another way, here is a visual of how both KPIs and OKRs contribute to success in the SOC. KPIs cover the "are we operating as expected?" side of measurement via analyzing the day-to-day processes we run. OKRs are there to guide us to the goal with any improvements that we are making. In the parlance of the Perdoo blog which we've been referencing for its outstanding and clear explanations of these terms, OKRs define and measure the *initiatives* that we undertake in order to achieve the key results we have defined. This chart is a SOC-centric modification of one of Perdoo's excellent charts in a blog post on OKRs and KPIs that, again, is suggested reading to go deeper on this topic.¹

[1] <https://www.perdoo.com/resources/kpis-okrs-the-goals-that-drive-business-success/>

Step 0: Consider Your Goals

What kind of SOC do you have?

- National/HQ SOC
- Component or subordinate SOC
- Managed service provider



What kind of value is expected based on your business and/or charter?

- | | |
|-------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------|
| <ul style="list-style-type: none">• Clearing alerts• Producing intelligence• Consuming intelligence | <ul style="list-style-type: none">• Responding to incidents• Servicing users• Implementing new controls |
|-------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------|

Before We Start: Consider Your Goals

Our metrics will tell a story and we want that to be a story in which our audience is invested. You'll soon see that there are a lot of metrics we *could* measure – but which ones are the most appropriate for the type of SOC we have? Metrics should drive a decision or demonstrate value – if servicing user requests is part of your SOC's charter, your metrics should measure time and quality of that service. If you are part of a managed service, your metrics should likely reflect the alerts you're handling on behalf of your customer(s), incidents reported, and customer interactions closed. If you're a national or HQ-level SOC, your metrics may be more focused on campaign analysis and intelligence gathered from subordinate SOCs. If you are a component or subordinate SOC, your metrics should reflect more tactical operations and contributions to security of the enterprise.

Mitre's [Ten Strategies of a World-Class Cybersecurity Operations Center](#) by Carson Zimmerman gets into more detail about the types of SOCs and comparative focus areas you might need to consider in your metrics:

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

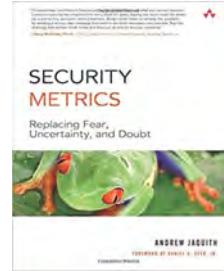
Step 1: Select Your Metrics

For you: There are a lot of metrics you *could* collect

- Some may be usable in KPIs, some may not

The definition of a good metric, according to "Security Metrics" by Andrew Jaquith:

1. **Consistently measured**, without subjective criteria
2. **Cheap to gather**, preferably in an automated way
3. **Expressed as a cardinal number or percentage**, not with ordinal or qualitative labels like "high" or "red"
4. **Expressed using at least one unit of measure**, such as "hours" or "dollars"



Step 1: Select Your Metrics

Now that we're familiar with metrics, KPIs, and OKRs and how each can help us operate and improve the SOC, how can we bring this information to our own organization? The first step is categorizing the data you have or would like to have. What metrics are you currently collecting and which type of measurement are they? Can you divide them into metrics, KPIs, and OKRs, and organize how each is helping you?

If you'd like to improve on what you have, it likely will start with collecting additional metrics. This is where we leave the "what" and "why" of measurements and begin on the "how" part. What makes a metric something that is useful and good to collect?

In the book "Security Metrics", by Andrew Jaquith, the four criteria listed on the slide above are key requirements for any metric. In addition, he says it should ideally be "**Contextually specific**" – relevant enough to decision-makers so that they can take action.¹ On the flipside, Jaquith says *bad* metrics are metrics that fail to exhibit these qualities. For example:

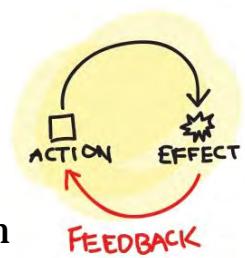
- Metrics that rely on the judgement of humans (unless very strict guidelines for judgement are provided)
- Non-cardinal number metrics such as stoplights. If your data has more than three states it can be in, it's unlikely that reducing the metric to that level of simplicity is doing you any favors.
- Metrics that are painful, expensive, and burdensome to produce – This could be due to the lack of automation for producing them, or metrics that rely on questionnaires and evaluations that would annoy subjects if given too often. The fallback for the later situation often is to sample less frequently, if there are long periods of time between when the metric moves, this might be ok, but quick moving metrics will simply be under sampled giving you a poor result.²

1 <https://www.amazon.com/Security-Metrics-Replacing-Uncertainty-Doubt/dp/0321349989>

2 Ibid.

Metrics Rate

- The **frequency** of metrics production matters!
- **Goal:**
 - Effortless metrics gathered at *fastest useful rate*
 - Minimize the delay between measurement and reaction
- Remember the OODA loop: The more/faster you observe, the faster you can react and adjust
 - **Faster problem detection means easier fixes**
- **Key to success:** Match your metrics sample rate to the rate of what you're trying to measure



Metrics Rate

Speaking of the rate at which we should gather metrics, remember the conversations we had about systems dynamics and the OODA loop? Since the SOC is a complex system we're trying to constantly optimize, many metrics will require frequent sampling of their state to truly understand what is going on. Without frequent measurements, the signs that we are off track may not be sensed until it is too late to take easy action to correct. In this way, the sampling rate of metrics is directly tied to your ops tempo capabilities!

As Andrew Jacquith said, good metrics should be gathered as effortlessly as possible in an automated-fashion way and on a short timescale. Doing so for quick-moving metrics gives us the shortest OODA loop and quickest reaction time. Your goal should be to minimize the delay between measurement and reaction to that measurement. If you only look for spam email waves once a day, for example (an event that can start and end by the minute), your reaction time will be severely hampered, taking up to 23 hours 59 minutes at worst to realize that something bad has occurred. Match your metrics sampling rate to the timescale at which the thing you're trying to measure may occur.

Step 2: Identify Your KPIs

- From your metrics, select...
 - Which relate to **key areas of operation**
 - What the **targets/threshold** should be for each (historical/peer data)
- The *metrics* may be similar for many orgs
 - Phishing incidents, monitoring percentage, time to respond, etc.
- The *targets* will be very different
 - Based on SOC requirements
 - Based on specific issues or areas of focus, etc.
- **These KPIs are your daily operational measures**
 - Metrics are given context by the limits and thresholds
 - Metrics without a clear threshold/target are questionable

Step 2: Identify Your KPIs

From the metrics you've chosen to collect, some of them are likely candidates for becoming KPIs. If that metric relates to a key area of your operation you need to watch (think something similar to the gas gauge in your car) and you know the bounds at which you would need to take action, that metric is a candidate for a KPI. These numbers can be put on a dashboard, alerted on, or any other method that it takes to ensure they stay within the correct parameters. They represent the key data that you will want to have available on a near-constant basis and tells you that everything you do monitor is within the operational norms.

While the metrics collected by the SOC may have some overlap between organizations (incident counts, timing for reactions, etc.), the targets for the KPIs are likely to be wildly different. These will be highly organization-specific and based on the specific requirements, environment, and risk tolerance of that organization.

Organizing Operational Measures

Each KPI should be documented with a

1. Measure: How to calculate the metric

- Should include time scale – "alerts per week" instead of "alerts"

2. Target/Threshold: Where it should fall to be "normal"

3. Source: Where is the data coming from?

4. Frequency: How often will it be sampled?

- **Remember:** Frequency of sampling should be rational compared to the rate of the event occurring
- Wrong frequency - Major incidents per hour
- Correct Frequency – Major incidents per month/year

Organizing Operational Measures

Similar to use cases, you should consider documenting the KPIs and metrics you have selected to collect in some sort of organized database or spreadsheet with an entry similar to your use case database that explains the metric or KPI, and why it's important. Fields you may want to track are:

Measure: How the underlying metric itself is calculated.

Frequency: The timescale at which the metric is or should be collected and why.

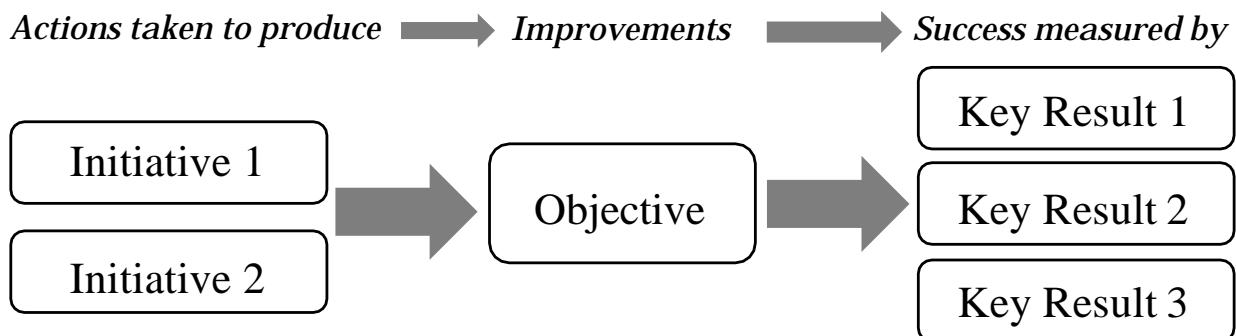
Target / Threshold: The reasoning for selecting the target or threshold you have chosen, and the historical data used to generate it or policy from which it is derived.

Source: What data is required in order to produce this metric? This is great for seeing which tools and data sources are giving you valuable measurements.

Step 3: Creating OKRs

Write your improvement goals as objectives:

- List how you will measure / know you succeeded (key results)
 - Remember the key results must be **specific** and **quantifiable**
- Enumerate the *actions* you are taking to get there (initiatives)



Step 3: Creating OKRs

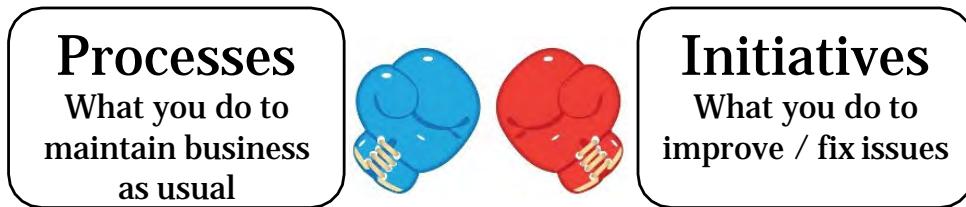
Next, it's time to list any Objectives and Key Results for SOC initiatives that are underway. Consider all current and future projects and, if appropriate, convert them into OKR form. To do this, decompose your goals into the following key items to clarify exactly what the end state is, how you will get there, and how you will know you've achieved it:

- Objectives: The goal to be reached.
 - Example – Minimized incidents caused by phishing attacks
- Initiatives: The actions you are undertaking to reach your goal.
 - Examples – Implementing better spam filtering, teaching users not to open documents from email, etc.
- Key Results: How you will measure that you have successfully met your objectives.
 - Example – Phishing-based incidents should drop to below five per week

Remember the goal here is to clearly state exactly what you are doing (initiatives) to meet your goal state (objective), and how you will define success that you've gotten there (key results). Breaking down projects in this way helps everyone understand all the moving pieces and should make clear how the success of the initiatives are tied to the completion of the objective.

Competition Between Ops and Improvement

- OKRs and KPIs are now defined ...
- **Next challenge:** Tackle both at the same time
 - Time must be split to ensure both are on track
 - **Problem:** Your team will always be pulled toward ops tasks at the expense of your initiatives



Competition Between Ops and Improvement

Now that you have both OKRs and KPIs defined, you will have to deal with the natural competition between the two. Operational tasks and their related KPIs are what must happen to "keep the train on the tracks", but without improvement, SOC weaknesses will never change. This tension naturally exists in every team due to resource time restrictions.

What happens as a result? People are inherently drawn toward operational tasks as they are always seen as the most immediate and important need. While that may be true in some cases, only working on day-to-day tasks gets you caught up in the whirlwind of daily firefighting, potentially at the cost of long-term improvement. What can we do?

The Four Disciplines of Execution (4DX)

A formula for high-performance execution

- Helps you see past the whirlwind to what is *truly* important

Principles:

1. Focus on the *Wildly Important Goal* (WIG)
2. Act on the *lead* measures
3. Keep a *compelling* scoreboard
4. Create a cadence of *accountability*

A recipe to drive continuous improvement



The Four Disciplines of Execution (4DX)

An excellent system and framework to help ensure you are not getting sucked into daily firefighting at the cost of achieving improvement initiatives is from the previously mentioned book "Four Disciplines of Execution" (4DX) by McChesney, Covey and Huling.¹ Throughout many years I have used the lessons from this book at work and in my personal life to help prioritize goals and ensure success in the face of daily distraction and believe me when I say, it absolutely works.

The principles from the book are relatively simple and form a potent and proven recipe for driving continuous improvement:

1. Focus on the "wildly important goal": This is all about defining and achieving the goal that will push you to the next level of excellence (new SOC initiatives).
2. Act on the *lead* measures: This principle explains the different types of metrics we may take, and which ones we should focus most on to achieve success.
3. Keep a *compelling* scorecard: Keeping a visible scorecard to help each team member stay emotionally invested and know if they are "moving the needle" in the right direction.
4. Create a cadence of *accountability*: Engaging each person on the team in a way that maximizes commitment to the team and aligns their actions with what it takes to achieve the goal.

Throughout the next few pages, we'll briefly describe each item in more detail.

[1] <https://www.franklincovey.com/the-4-disciplines.html>

I. Focus on the WIG (Wildly Important Goal)

WIG – "The **most important objective that won't be achieved **without special attention**"**

- Often **quadrant 2** "not urgent, important" activity
 - Automation, process improvement
- To define a WIG, identify:
 1. Where you are now (starting line)
 2. Where you want to be (finish line)
 3. By when (deadline)

		Urgency	
Importance		High	Low
	High	1	(2)
	Low	3	4

1. Focus on the WIG (Wildly Important Goal)

The first principle in 4DX is focusing on the "wildly important goal" (WIG). The definition of the WIG in this system is "the most important objective that won't be achieved without special attention". To define a WIG, the 4DX system defines the process almost identically to using an OKR—you need a starting line and finish line that defines success, and a timeline of when you want to arrive there.

In the case of the SOC, these WIGs are the initiatives that represent extra work that is super important for progress, but without the commitment to get it done, will never actually go anywhere. One way to think about this is placing items you do day to day on what is often called the "Eisenhower matrix". The matrix asks you to consider all your to-do items on separate axes of urgency and importance. Daily activities we run into are nearly all urgent feeling, which is why they tend to take precedence over improvement initiatives. But what is *really* more important—answering that email *right now*, or working on a process improvement that will make your SOC better for all of time? I can tell you which one probably *feels* like you should do it first, but you also probably know what the correct answer is—improvements. **Ranking your to-do items on the Eisenhower matrix helps remind you of the fact that not everything that demands your attention actually matters, and that if you are to achieve your WIG, you sometimes must actually delay the urgent and not-important tasks for the non-urgent and important tasks. If you remember nothing else from this section, this is the one key insight you should remember, as I truly believe it can single-handedly significantly improve your ability to execute.**

Identifying Your Wildly Important Goal

Step 1. - Identify your WIG

- Ask: "What *one thing* we can do such that by doing it everything else will become easier or unnecessary"
 - The ONE Thing by Gary Kelley & Jay Papasan
- Examples:
 - Improve network and host visibility
 - Minimize compromise from phishing
 - Drive down infections that must be responded to
 - Improve analyst efficiency with automation

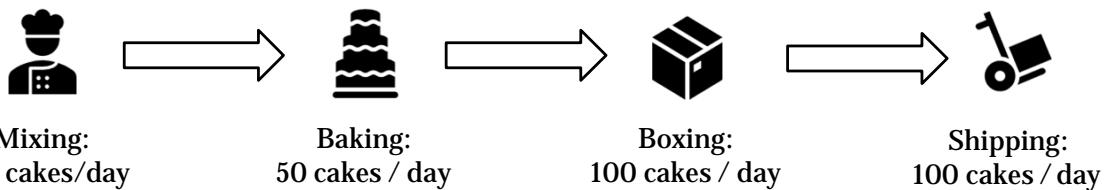
Identifying Your Wildly Important Goal

Now that we've defined a WIG, how do we decide what the most important goal actually is? We must highlight one or two because as the saying goes, "if you have too many priorities, you have none". Maybe you already have a number of initiatives underway, maybe you don't have any. How do you get from either state to find the *most* important goal? Another idea from a highly useful book, "The ONE Thing", by Gary Kelley and Jay Papasan is to ask yourself what they refer to as "the focusing question". That question is, "What *one* thing can you do, such that by doing it, everything else will become easier or unnecessary?". In the face of multiple potential priorities, asking this question can help determine which should be number one based on the fact that doing it will help everything *else* go more smoothly. As you might be thinking, SOC process improvements clearly fall into potential answers to this question, which is exactly the point.

Which Process Step to Improve? Find Your Bottleneck

Theory of Constraints: In any process there is a **bottleneck**

- Any improvement outside the bottleneck is an **illusion**
- Improvements pre-bottleneck cause a pile up at the bottleneck
- Post-bottleneck improvements are pointless, steps remain starved



- Above example: Mixing, boxing, and shipping faster won't help
- To ship more than 50 cakes/day you **must bake cakes faster**

Which Process Step to Improve? Find Your Bottleneck

When looking at your processes and collected metrics, how do you know which one to focus on to make the most improvement in the system? Using what is called the "theory of constraints" and a SOC process model, we can come up with where to make improvements to ensure the entire system works better. The theory of constraints is a system dynamics-based methodology that gives us some important statements about improving the performance of any system:

- In any process there is a "bottleneck", at least one item that limits the system from going faster
- Improvements made "upstream" from the bottleneck will not provide any advantage, they will simply cause more items to pile up at the bottleneck
- Improvements made downstream from the bottleneck will be ineffective at improving the systems production rate as they will simply cause steps post-bottleneck to become starved of resources faster
- Therefore, the only way to improve the system is to increase throughput at the bottleneck

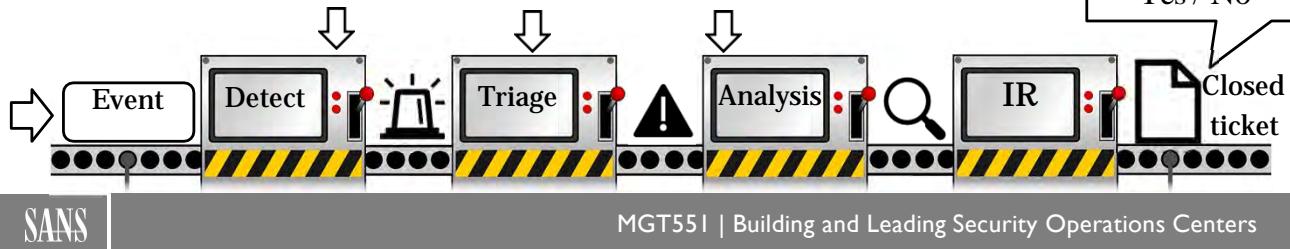
When it comes to alert triage and investigation, looking at your SOC like a production line means the theory of constraints applies as well. If you can find the bottleneck in your system and eliminate it, your team will become faster at moving alerts through the system. To identify it, look for the step where the bandwidth is lowest, and items tend to pile up. Then, consider whether that pileup is a result of equipment, people, or policy and find a way to alleviate the issue!

In the example bakery system on this page, all but one step works at 100 cakes/day. You could increase mixing speed, but that would just cause more cake batter to pile up ready to be baked. You could increase boxing and shipping rates, but those steps would just have nothing to box or ship more quickly. The only way to ship more cakes per day is to address the bottleneck—baking.

Where to Take Measurements

- Short answer: At every useful stage of the process
 - You wouldn't measure an assembly line by only measuring the final step ...
 - Don't sample your SOC only at the end of the process
- Use intermediate process measurements
 - Measure all—collection, detection, triage, investigation, and IR
 - Remember, garbage in, garbage out

Measure these steps too



Where to Take Measurements

To find your bottleneck, and just to have good coverage of your process with metrics in general, remember that you need to have measurements from multiple steps in the process. As an illustration of this concept: Do you think Ford could make cars if their only measurement was "Does the car work?" at the end of the assembly line? Of course not, they want to make sure that everything is working well along each step of the assembly process and likely take measurements to show that, and find errors before they echo all the way down the line. Like this example, the SOC is also a process with multiple steps, and to have a complete view of what is going on as well as be able to locate the source of issues, likewise you must take measurements along the collection, detection, triage, analysis and IR process as well. This should include times taken for triage steps, coverage of data collected, detection count and accuracy, etc. Having a more granular view will help you see problems at the actual step where it manifests, and zero-in to solve it as quickly as possible.

2. Act on the Lead Measures

Which type of metrics do you collect?

- **Lag Measures** track the success of your goal
 - Driven by what has *already* happened
- **Lead Measures** track activities that drive a lag measure
 - To achieve a goal, *you must focus lead measures*

Example: Weight loss

- Lead Measure – Diet and exercise
- Lag Measure – What the scale reads

2. Act on the Lead Measure

The second principle of 4DX is the idea of whether a metric is a "lead" measure or a "lag" measure. A common issue with many metrics is that they are what can be considered a "**lag**" measure—a metric that measures something that has already happened. While lag measures can be informative, they aren't often the best type of metric to collect as they have no predictive power, they can only tell you what has *already* occurred.

The other type of metric is called a "**lead**" measure. These are metrics that measure the input to a process and drive what may ultimately be measured as a lag measure. These metrics have predictive power because they measure the *input* to a process, which will then determine the output. If the output determines whether you hit your goal or not, then controlling the lead measures (the input which determines the output) is truly what matters most, since lead measures can be directly acted upon.

An example: Suppose you are trying to lose weight and you step on the scale every morning to see how successful you are. The weight you see is a lag measure—it measures your goal (did you lose weight or not) but there's nothing you can do about it at the moment, it's just the output of what you've already done. While tracking your weight answers the question about whether you got to your goal, it doesn't help get you there, for that you need a lead measure. In this case, the lead measure would be a metric of the things that cause you to lose weight—what you ate and how much you have or have not exercised, these are variables that can be directly acted upon to change your outcome. If you can manage to control the input variables in a positive way, you have no choice but to achieve the goal you are aiming for. When the inputs are controlled, the outputs (lag measures / goals) must follow. Therefore, in weight loss or in the SOC, lead measures should be identified and focused on for influence because controlling them is what truly controls whether you hit your goal or not.

[1] <https://resources.franklincovey.com/the-4-disciplines-book-videos/the-4-disciplines-of-execution>

3. Keep a Compelling Scoreboard

Why: Engagement!

- People do their best when they're emotionally engaged
- More engagement when people know the "score"
- Shows connection to lead and lag measures

How:

- Create a "player-centric" scoreboard
- Designed *for* and *by* the players
- See progress in real time

3. Keep a Compelling Scoreboard

The third principle in 4DX is keeping a compelling scorecard. What is a compelling scorecard by 4DX definition? A dashboard or visible measure of sorts that shows a "player-centric" (or SOC employee centric in our care) view of progress design by and for those who are being scored on it. While progress on the project may also be reported outwards to additional parties, the scorecard in 4DX focuses on what those involved need to see to gauge their contributions. The idea of the scoreboard is to see progress towards the goal in as real time as possible and give people continuous feedback on the impact of the work they are doing.

Why is a scoreboard so important? It is one of the magic formulas to create engagement in employees who otherwise might be distracted and pulled in by every other day-to-day item that is calling upon their limited attention. Scoreboards keep these improvement initiatives and projects front and center and show the connection of daily actions to the lead and lag measures.

4. Creating a Cadence of Accountability

Discipline 4 is how to play the game:

- A regular, frequent meeting, 20 minutes or less
- Deliver a commitments report:
 1. Did I meet last week's commitments?
 2. Did they move the scoreboard?
 3. What will I commit to this week?

Choose your commitments based on the following:

"What are the most important things I can do this week that will have the biggest impact on the scoreboard?"

4. Creating a Cadence of Accountability

The final principle in 4DX is creating a cadence of accountability. The authors of the 4DX book suggest you cultivate feelings of personal accountability and responsibility for the initiatives by implementing a regular and frequent, but short meeting. In this meeting, employees answer the question, "What are the most important things I can do this week that will have the biggest impact on the scoreboard?" Once decided, teammates go through and do a commitments report. This consists of answering three main questions.

1. Did I meet last week's commitments?
2. Did they move the scoreboard?
3. What will I commit to this week?

Why This Works

- Discipline 1-3 create a winnable game...
- Discipline 4 facilitates engagement and commitment
 - People are more likely to commit to their *own* ideas rather than directions given to them (think autonomy and creativity)
 - Commitments to team members go beyond job performance, and become a *personal promise*
- Team sees their positive impact on the WIG
- Seeing the impact drives morale and engagement

Why This Works

There are 2 key ideas behind the 4DX system's success. First is that you must create a winnable game to set your team up for success. This is the aim of items 1-3 (Focus on the WIG, Act on Lead Measures, Create a compelling scoreboard). The second key idea is facilitating engagement and commitment, which is the goal of item number four. Employees are much more likely to respect and hit deadlines set for themselves that they have determined on their own versus orders from someone else. In addition, when teammates are making commitments to each other, the feelings go beyond being a business task and feel more like a personal promise, making those on the team more likely to deliver on those commitments and be engaged and invested in doing so. The 4DX system is set up so that teammates can immediately see the positive impact their actions have toward achieving the WIG and seeing those achievements continuously improves morale. Combined with the strength of personal commitments, the 4DX system strives to set the team on a course to success and inspires them to continue this powerful forward momentum.

Metrics, Goals, and Execution Summary

Measurement types:

- **Metrics** = A measurement with a current value
- **KPIs** = Measurements plus a target/threshold
 - Best used for daily operational numbers
- **OKRs** = Measurements plus a defined start, end, current value
 - Define initiatives, goals, and measurements for success in short term project

4DX process helps execute on initiatives

- Emphasizes finding and focusing on **WIG**
- Encourages use of **lead measures**
- Utilizes **scoreboard** and **commitment** meetings for engagement



Metrics, Goals, and Effective Execution Summary

In this section, we focused on the how and why of metrics in order to help you understand what makes a good measure for your SOC versus what doesn't. Key takeaways here are to consider the measurements that you're taking and whether they fall into the Metrics, KPIs, or OKRs camp, and whether organizing them like this might help you see them in a new and more effective way. Metrics is an enormous discussion that could fill an entire course, and a tricky one as well because what might be the favorite measure at one organization may be irrelevant to another. The key to creating good, meaningful metrics for operations is to first consider the problems you're trying to solve by tracking that item, and how you will react if given information that a measurement is outside of expectations or its normal range. Metrics for projects are a whole different space where the goal is often to define how close you are to completing. For these items, the OKR system is a great way to separate the story you're trying to communicate into the goal, the actions you're taking to get there, and the measures that will tell you that you have arrived.

While having and measuring improvement projects is great, validating people actually take the time to work on them and produce those awesome measures is a totally different challenge. This section covered the Four Disciplines of Execution system as a way to confirm your team continues to execute on one-off initiatives in the face of the daily emails, phone calls, and other disruptions that will always plague our work life. This clever system helps us create a winnable game and engages and inspires team members to follow through, ensuring success and continual progress in the SOC. Progress which is then hopefully reflected in the improvement of daily metrics that can be used to show the business how effective the SOC is, and the ROI it is producing.

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

- **Effective Execution**
- Staff Retention and Burnout Mitigation
 - *Exercise 5.1 – Training and Career Development Planning*
- Metrics, Goals, and Effective Execution
- **Measurement and Prioritization Issues**
 - *Exercise 5.2 – Creating, Classifying, and Communicating Your Metrics*
- Continuous Improvement
- Strategic Planning and Communications
- Analytic Testing and Adversary Emulation
 - *Exercise 5.3- Purple Team Assessment*
- Automation and Analyst Engagement
- Summary and Cyber42 – Day 5



This page intentionally left blank.

The Importance of (Correct) Prioritization

- In information security, **prioritization** is everywhere
 - Risks
 - Alerts Severity
 - Vulnerabilities (CVSS scores)
- Therefore, it's incredibly important we understand how to do it correctly
 - Information security commonly makes miscalculations
- In this module, we'll dive into some common errors



The Importance of (Correct) Prioritization

One concept that repeatedly appears throughout information security is prioritization. We prioritize risks to address, alerts by severity, vulnerabilities by CVSS scores and more. Since we rely so heavily on getting priorities right to ensure we are addressing the truly most dangerous items first, it's important that we have a quick discussion on some of the errors that are made and some of the poor logic that is commonly introduced around calculation of these priorities in information security.

Levels of Measurement

Four measurement type classifications¹:

1. **Nominal** – Named (ex: colors, flavors, towns)
2. **Ordinal** – Named & Ordered (ex: satisfaction ratings, education level attainment)
3. **Interval** – Named, ordered, and has units of even spacing (ex: temperature)
4. **Ratio** – All the above, plus units referenced with "absolute zero" (ex: dollars, mass)



Levels of Measurement

The first concept we need to discuss is what is called the various levels of measurement, as in, what are the different ways things can be ranked, ordered or measured in different scenarios.

- **Nominal** – The most basic measurement is the nominal measurement. A nominal measurement is more of a label than anything else, it doesn't denote order, category, or allow comparison to anything else. An example would be colors or flavors.
- **Ordinal** – An ordinal measurement is one in which the items being measured can be compared to others in a way that denotes lower to higher, or at least a spectrum of one extreme to the other. Examples would be education level (high school, university, post-graduate degrees, etc.) or SANS course evaluation scores. ;)
- **Interval** – Interval measurement is where things start to get more precise. Items that can be measured as an interval measure have a name, order, and some numerically even unit of spacing between them such that they can be quantitatively compared. An example would be a temperature measurement, you can compare 10 degrees to 20 degrees and there is a meaningful definition of 1 degree such that you know exactly how far away they are from each other, unlike in an ordinal measurement. Note that the point set as "zero" is arbitrary in these scales, which is a defining difference between intervals and ratio measures.
- **Ratio** – Ratio measurements have all the properties of the previous categories, but also can be referenced by a true zero point, enabling addition, multiplication and more, and the production of a meaningful result of doing it. Examples would be dollars, object mass, and plenty more. We know what zero means, we know $\$10 * 5 = \50 , and that makes meaningful sense.

Levels of Measurement and Allowed Operations

Depending on the type of measurement, only certain operations are allowed¹:

<u>Incremental progress</u>	<u>Measure property</u>	<u>Mathematical operators</u>	<u>Advanced operations</u>	<u>Central tendency</u>
Nominal	Classification, membership	=, ≠	Grouping	Mode
Ordinal	Comparison, level	>, <	Sorting	Median
Interval	Difference, affinity	+, −	Yardstick	Mean, Deviation
Ratio	Magnitude, amount	×, /	Ratio	Geometric mean, Coefficient of variation

Levels of Measurement and Allowed Operations

Given the levels discussed on the previous page, here is a summary of what type of mathematical and sorting operations are allowed for each level of measurement.

- **Nominal** – Nominal measures can only be grouped and compared as equal or not equal. The only central tendency calculation possible is the mode.
- **Ordinal** – Ordinal measures can be said to be greater or less than (but not how much), they can be sorted, and a median can be calculated due to the ability to order the items being measured.
- **Interval** – Interval measures can be numerically compared with each other, and numbers can be added or subtracted from each other since they are quantitative. Means and deviations can be introduced since measurements are now numerically described.
- **Ratio** – Because ratio measures are quantitative and have a meaningful zero-point referenced, interval measures can be multiplied and divided, and central tendency measures can be geometric means and coefficients of variation.

Risk Matrices and Why You Shouldn't Trust Them

- Surprisingly, there is **no evidence that they work**
 - See book Douglas Hubbard's book "*How to Measure Anything in Cybersecurity Risk*"¹ for details
 - Research has found they can be "**worse than random**"
 - Do produce "analysis placebo" – makes you feel better but provides **no measurable improvement**
- Why? Consider levels of measurement
 - What type of measurement are they?
 - What operation do we use to combine them?

		Probability		
		Low	Med	High
Impact	High	Low	Med	High
	Med	Low	Med	Med
	Low	Low	Low	Low

Risk Matrices and Why You Shouldn't Trust Them

One of the first, and perhaps the most surprising problems we run into when considering how levels of measurement affect prioritization is that of the incredibly common risk matrix. We've all seen one of these at some point, with an axis denoting likelihood, and another axis denoting the potential severity if an event occurred. Finally, we have the matrix of colors with green to red showing when the two variables combine, how much we need to be concerned with that level of risk and label it from "very low" to "extreme".

Given the new perspective of levels of measure, consider what type of measurement the is a likelihood measure? What type of measurement is a severity measure? Both axes in this case are **ordinal** measures. Then consider what is happening when we create the matrix – we are performing a **multiplication-like** operation on these ordinal measures. (If you were trying to do multiplication on numbers in the axes for example, this layout would make perfect sense.) If you don't see the problem yet, consider what the result of "likely" multiplied by "significant" is – nothing, you can't multiply words. Therefore, the "high" risk label the graph assigns to this result is also meaningless. Uh oh!

For a *much* longer and more in-depth discussion of this error, see the fantastic book "*How To Measure Anything in Cybersecurity Risk*"¹ by Douglas Hubbard and Richard Seiersen. It's a real eye-opener to some of the problems in cybersecurity prioritization and measurement and leads down a rabbit hole of what does and doesn't make sense in the processes we very commonly used to make decisions.

[1] <https://www.howtomeasureanything.com/cybersecurity/>

Why Don't Risk Matrices Work?

Partial reasons:

- Use imprecise "gut feel" evaluations of 1-5
- People are biased and inconsistent at ratings across time
- Research shows people estimate numbers very poorly
- Even if right, matrix introduces range compression and noise

But the biggest reason:

- **Multiplication of ordinal scales**
- A "type error", it just can't be meaningfully done

		Probability		
		Low	Med	High
Impact	High	Low	Med	High
	Med	Low	Med	Med
	Low	Low	Low	Low

Why Don't Risk Matrices Work?

There are multiple reasons why research seems to show risk matrix-style ranking is ineffective. Some of the reasons are listed on the slide above, include:

- "Gut feel" instead of a clearly defined standard used for high, medium, low
- Individual bias across people and time leads to inconsistencies in ranking
- People repetitively have shown to be poor estimators of many situations
- Matrix approach itself leads to range compression and noise within the measurement

For all of these reasons, inaccuracies start to build and multiply upon each other, and that's before we consider the biggest reason – that *this approach clearly violates the mathematical operators that are allowed for the ordinal level of measurement*. Against what reference point does "high impact" x "med probability" = "yellow" (medium) risk? What is medium defined as in this case? Is it relative to some standard? Also, is that risk *really* the same as a "high probability" x "medium impact" event, which is also labeled as medium risk? While we can certainly say that a high impact event is worse than a medium impact event, because this is an ordinal scale, it is the combination of two ordinal scales in this way that leads things to go "off the rails". You can see how these are at best, *very* broad, imprecise measurements that could easily lead you astray.

It's probably not a shocker to you that these are imprecise, you might be thinking "yeah, but they at least correlate to some reality of risk, which is better than nothing, right?" Well, that is where surprising research has weighed in and said that, in some cases, risk matrices can be "worse than useless" and actually return worse performance than random assignment¹. Yikes! The cited study abstract concludes "These limitations suggest that risk matrices should be used with caution, and only with careful explanations of embedded judgments."

[1] "What's wrong with risk matrices?", Louis Anthony Cox Jr., <https://pubmed.ncbi.nlm.nih.gov/18419665/>

What About Your Alerts?

Risk matrices aren't the only violation...

- How about **alert prioritization**?
- Which alert would you address first?
 - Low priority, high severity? = "*Medium urgency*"
 - Critical priority, low severity? = "*Medium urgency*"

Assigned Severity

Assigned Priority	Assigned Severity					
	Informational	Unknown	Low	Medium	High	Critical
Unknown	Informational	Low	Low	Low	Medium	High
Low	Informational	Low	Low	Low	Medium	High
Medium	Informational	Low	Low	Medium	High	Critical
High	Informational	Medium	Medium	Medium	High	Critical
Critical	Informational	Medium	Medium	High	Critical	Critical

What About Your Alerts?

Another place this exact same problem shows up is in many security appliances and tools. When alerts are generated, or exploit attempts are correlated with the vulnerabilities of the victim machine, two variables of severity and priority of the situation may be assigned by the tool. In the example above, (pulled from the Splunk Enterprise Security documentation on how the Urgency field is calculated for notable events), a third field is generated based on the combination of the priority and severity variables. Given that we know assessments of a person for the same data will vary over time, and the assessments of multiple experts also widely vary, how much can we trust that these labels were applied to our assets in a standard way, let alone the dubious combination of ordinal measurements that occurs in these types of matrices.

This example is not to beat up on Splunk, nearly every vendor has a version of this, it is to demonstrate that this method is used pervasively throughout the industry in multiple tools and locations from vendors far and wide. If we cannot trust it, or at least understand its limitations, this may lead to poor conclusions and sub-optimal actions taken by our SOC analysts.

[1] <https://docs.splunk.com/Documentation/ES/6.4.0/User/Howurgencyisassigned>

But We Use Numbers!

- **Questions:**

- Where did those numbers come from?
- How did you map to those numbers?
- Can you quantitatively describe the difference between a 2, and a 1?



- You *might* just have an ordinal with a numeric name

Example of mapping words to numbers:

- Low priority attack on a critical asset: $1 \times 9 = 9$
- High severity attack on unimportant asset: $9 \times 1 = 9$
- Lowish priority attack on less important asset: $3 \times 3 = 9$

Yes, you did math, but they *still* aren't directly comparable

SANS

MGT551 | Building and Leading Security Operations Centers 70

But We Use Numbers!

One somewhat sneaky version of this is when numbers are assigned to what otherwise be an ordinal scale, masking the truth and making it harder to realize you're still violating the principles of measurement theory. For example, in a risk matrix we could assign low=1, med=2 and high=3. If we have a high probability x high impact you *could* calculate this to be $3 \times 3 = 9$ on the risk scale. But is this situation really 9 times worse in some way than a low probability and low impact event ($1 \times 1 = 1$)? You'd probably say no, in fact, doing this may have led you in a VERY wrong direction.

The same thing is true of alert prioritization, if you assign numbers from 1-10 for example and use the multiplication of those numbers to then rank which alerts analysts need to attend to first, are you really getting the intended result? Consider the example on the slide. Hopefully, the follies that we so often encounter are becoming clear at this point, and why it's worth visiting how many places this error shows up.

What About CVSS Scores^{1,2}

CVSS scores also have some issues:

- Uses descriptor to number conversion for exploitability multiplication
- Explanation and justification for weightings *not* given
- Research showing accuracy is not provided
- Eventually map numbers *back* to ordinal scale for low/med/high

Attack Vector	Metric Value
Network	0.85
Adjacent	0.62
Local	0.55
Physical	0.20

$$\text{Exploitability} = 8.22 * \text{Attack Vector} * \text{Priv. Required} * \text{User Interaction}$$

One Final Example: CVSS Scores

One final violation of measurement theory that is commonly referenced in information security that may lead to sub-optimal decisions – CVSS scores. While there is clearly a need to express the severity of a vulnerability so that teams know which ones they must address, the approach the algorithm for CVSS scores uses also makes the same mistakes we have discussed.

CVSS has more than one issue. One is the method of measurement itself, the second is in how companies use the measurement of CVSS scores to prioritize patching, which has been shown to be the wrong approach. On the first issue of the measurement, CVSS scores are calculated using the addition of an Impact and Exploitability score, each of which is created by the multiplication of several other factors such as the Attack Vector. Exploitability, for example, is calculated as:

$$8.22 \times \text{Attack Vector} \times \text{Attack Complexity} \times \text{Privilege Required} \times \text{User Interaction}^{[1]}$$

This very similar to a risk matrix-style calculation, but with more variables. The values for the attack vector metric are given on the slide above. Given we know that the Attack Vector variable is at best an ordinal measurement ("network" is assumed to always be worse than "adjacent", which is worse than "local" or "physical"), what is the meaning of these numbers? This should lead to questions in your head like is a network exploitability issue really 4.5x worse than a physical exploitation opportunity (0.85 vs 0.20)? How were these numbers arrived at, are they based on anything? Unfortunately, the CVSS literature does not clearly explain or substantiate their calculations. To make things worse, once the 0-10 calculation is performed, the number is then converted back to an ordinal scale rating of the vulnerability from "low" to "critical" – words that are imprecise on their own. All this means, at best, we're hoping these numbers will correlate with some truth that will steer us in the right direction.

What about the second issue though? Using it for patching prioritization, let's see what the data has to say about this approach.

1 https://www.first.org/cvss/v3-1/cvss-v31-specification_r1.pdf

2 CVSS v3.1 Score Calculator <https://nvd.nist.gov/vuln-metrics/cvss/v3-calculator>

What is CVSS?

According to the specification from FIRST¹:

- CVSS measures "severity of a vulnerability **relative to other vulnerabilities**"
 - Not a measure of the **risk of exploitation**
 - Does not give **exploit availability** in most cases (temporal score)
- Patching must be aligned with exploitation risk
- Therefore:
 - Should be a **factor** for prioritization of patching
 - Should not be the **only factor** for prioritization



What is CVSS For?

Making the CVSS problem worse is the second issue - prioritizing patching based on calculated CVSS Base Scores. Although many companies are tempted to use CVSS Base Score as a *risk* measurement, this is not actually the intended use of the number and using it as such in fact has been shown to work poorly. This is not the fault of the CVSS system but rather a common misunderstanding by those who use it. In the documentation on CVSS3.1, FIRST states that CVSS is for measuring the "severity of a vulnerability relative to other vulnerabilities". Notice this is *not* the same as measuring the *risk* of exploitation (which is what we truly care about for patching).

CVSS theoretically addresses the need to assess risk of exploitation as well by adding an optional (but often unprovided) "Temporal" calculation that modifies the base score. When present, the Temporal score considers whether an exploit is available that works well, if a patch is available, and how confident we are in these answers. Once again however, the Temporal Score is calculated based ordinals but is converted into numbers, and that Temporal Score number ultimately becomes a multiplier to the Base Score to make it higher or lower. If the unexplained multiplication of a bunch of ordinal measurements converted to numbers is beginning to seem a bit unscientific to you, you are in good company.¹

[1] <https://www.first.org/cvss/specification-document>

Patching Based on CVSS Score Alone

How well do CVSS scores correlate to patch importance?

- Researchers showed in a BlackHat 2013 presentation^{1,2}:

"Our analysis reveals that...

- (a) **fixing a vulnerability just because it was assigned a high CVSS score is equivalent to randomly picking vulnerabilities to fix;**
- (b) **the existence of proof-of-concept exploits is a significantly better risk factor;**
- (c) **fixing in response to exploit presence in black markets yields the largest risk reduction."**

- Is this surprising?
- Do you agree?
- How do you prioritize patching?



CVSS – Maybe Not What You Think It Means

How well do these numbers correlate to reality of exploitation when used in this manner? Research presented in 2013 at Blackhat gives us the answer. Researchers Luca Allodi and Fabio Massacci from the University of Trento in Italy crunched the numbers and said:

"Our analysis reveals that

- (a) fixing a vulnerability just because it was assigned a high CVSS score is equivalent to randomly picking vulnerabilities to fix;
- (b) the existence of proof-of-concept exploits is a significantly better risk factor;
- (c) fixing in response to exploit presence in black markets yields the largest risk reduction."²

This is quite the set of statements, pause for a moment here and consider how this may influence what you've been doing in the past.

1^{https://infocondb.org/con/black-hat/black-hat-usa-2013/how-cvss-is-dossing-your-patching-policy-and-wasting-your-money}

2 ^{https://www.youtube.com/watch?v=_laE5mp1NI8}

Patching: What DO We Do Then?

- From Carnegie Mellon SEI Research¹
 - "CVSS is designed to identify the technical severity of a vulnerability. What people seem to want to know, instead, is the risk a vulnerability or flaw poses to them, or how quickly they should respond to a vulnerability. If so, then either CVSS needs to change or the community needs a new system."
- The correct way to patch:
 - **High CVSS score AND publicly available exploit**
 - (or with "Exploit Code Maturity" Temporal Score available and still high)
 - Consider Attack Vector (AV) and Priv. Required (PR) - exploit may not work
- Why?
 - Most exploited vulns are medium/high, but the reverse is not true
 - Prioritizing patching vulns that no one can exploit is a waste of time
- This was found to be the best path to TRUE risk reduction

Patching: What DO We Do Then?

In addition, the Carnegie Mellon Software Engineering Institute paper previously cited has concluded "CVSS is designed to identify the technical severity of a vulnerability. What people seem to want to know, instead, is the risk a vulnerability or flaw poses to them, or how quickly they should respond to a vulnerability. If so, then either CVSS needs to change, or the community needs a new system".

The message here is clear: CVSS scores are being improperly used in many organizations and are not designed to answer the question most of us truly want answered. What the Blackhat presentation does show, however, is how to get closer to this answer. They found the most effective way to use CVSS for true *risk* reduction is to prioritize not all high CVSS scores vulnerabilities, but only ones that that score highly AND are known to be available on the black market for use in exploit kits followed secondarily by vulnerabilities that have proof-of-concept code in the wild (the "ExploitCodeMaturity" factor of the Temporal Score). Patching vulnerabilities that meet these two specific conditions is what was truly found to reduce to most residual risk, while patching things that only have a high CVSS score only does not accomplish the intended goal.

The easiest way to summarize this is the following: Most (~90% as of when the research was done) of the attacked vulnerabilities out there have a high or medium CVSS score, but there are many high and medium CVSS score vulnerabilities which have no exploit publicly available. If you patch ALL high and medium scored items with equal urgency, you will be wasting your time as many of them will not and cannot be utilized by most attackers. Therefore, prioritize items that are both known to be attacked *and* a high score to maximize your efficiency.

[1] "Towards Improving CVSS", Carnegie Melon University Software Engineering Institute, J.M. Spring, E. Hatleback, A. Householder, A. Manion, D. Shick,
https://resources.sei.cmu.edu/asset_files/WhitePaper/2018_019_001_538372.pdf

What's Going On Here?

Problem 1: Qualitative labels like "unimportant", "important", "critical" are assigned a **number**

- Give the impression data is interval/ratio, when **ordinal**
- Ex: Alert severity, CVSS, event likelihood and impact

Problem 2: Calculations based on those numbers violates the allowed measurement level operations

- Ordinals cannot be meaningfully multiplied (or added!)
- Unjustified formulas used to designate importance

What's Going On Here?

The problem with all these situations is twofold. First, the individual ranking of "important" (an *ordinal* measurement) was confusingly assigned a numeric value of 8 which gives the false impression that it is an interval measurement. What does that "8" mean though? It's clearly not an interval measurement, otherwise that would imply that asset is "twice as important" as an asset ranked 4 (note this is issue also true of the alert severity "10" ranking). Therefore, assigning numbers steers analysts in the wrong direction from the start.

The second problem is that you are violating the principles of measurement when you take these falsely and meaninglessly numbered ordinal measurements and multiplying them together. While your SIEM can show you that the overall important is "80", what you're *really* doing is multiplying "important" x "high severity", which has no mathematical meaning or answer, especially one that can be compared to other combinations. Not only can ordinal measurements not be meaningfully multiplied; interval measurements can't be multiplied either! That means this method is actually 2 measurement types away from being valid, yet somehow, we have convinced ourselves this is an approach that leads to a meaningful output. On top of this, it is assumed that even if the numbers were anchored to a numerical reality, multiplication somehow would be the right way to do this. Why should it be multiple as opposed to summed, multiple with a weighted average, exponentiated? For this to be valid and meaningful, the approach we would have to have an explanation and a mathematical anchoring in data that showed it was meaningful in steering analysts towards the best choice. The violations don't stop at alert prioritization though, let's look at another example of this problem.

Why Do These Methods Feel Like They Work?

Analysis placebo effect creeps in for several reasons:

1. An unstated, underlying assumption

- We assume there is an unknown, numeric variable that the ordinal numbers weakly correlate with
- If true, vague guessing could be better than nothing
- Example: Race finishing rank *does* correlate with finishing time
- *Possibly* true for infosec, but *very* inaccurate to the point of near uselessness

2. The mere-exposure effect

- We frequently see something, so it feels like it must be correct – not true of course

Why Does These Methods Feel Like They Work?

When using ordinal type variables in a measurement, given that the categories are in a known order, one might think the values at least *weakly* correlate with some sort of unknown to us, but numeric, increasing, interval-style value. For example, the rank of runners finishing a race does indeed correlate with the interval variable of the time it took them to run the race, therefore the rank is at least *weakly* correlated with finishing time. Therefore, if estimating a runner's finishing time in a race, knowing their rank might indeed help. If this relationship were true for *all* ordinal measurements, perhaps the estimations would indeed help improve our assessment, but in many cases, this isn't the best way to approach the problem.¹

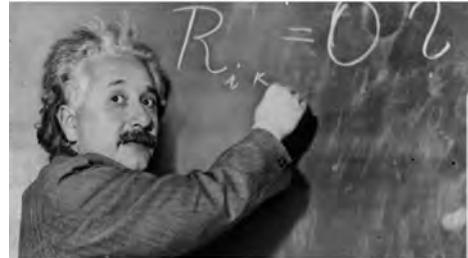
If the variable you're attempting to measure – risk for this discussion, does indeed have an underlying interval component, the best approach is to design a way to measure it directly. Instead of using an ordinal as an estimation with unknown accuracy, we could make our estimations quantitatively. For example - predicting the actual likelihood on a percentage scale and assessing possible damage in dollars, this would produce a *much* more meaningful result. This is what Douglas Hubbard suggests as a replacement technique in his book "How To Measure Anything in Cybersecurity Risk" and gives the specific methods for doing. This technique works, even with the incredible difficulty uncertainty inherent in estimating such numbers.

The second reason that this seems like a valid method, in the author's opinion, is that we are just so used to seeing it done. Of course, however, this has no bearing on if it works or not. The tendency of people to develop a preference for things they are familiar with is called the "mere-exposure effect" or the "familiarity principle". The same principle drives us to trust products we see more advertising for or trust a political candidate that we hear about more frequently. Does that mean that product or person is the best for the job? No, it only means they had the most funds to purchase ways to get your attention. When it comes to risk matrices and other ordinal based rankings used for multiplication or addition, although the industry loves the technique and we see it often, it has no bearing on the validity of the method.

[1] <http://jeromyanglim.blogspot.com/2009/10/analysing-ordinal-variables.html>

One Final Problem...

How I think I look explaining cyber risk



How I actually look



One Final Problem...

One final potential issue with this more rigorous approach is that of appearing to over-complicate what seems to have been working. Remember, the mere-exposure effect has led to years of people approach the problem in one way, and those who do it may be resistant to change due to the lack of perceived issues. Combine that you are telling someone their approach is wrong and starting to throw around words like ordinal measurements and quantitative analysis, and they might start to think you've lost your mind.

Exercise restraint, and a slow, simplified approach when introducing new ideas into a process that may have not been considered incorrect. It may take some careful, measured explanations and reference to outside materials before coworkers come around and start to see how these approaches remove clear problems and ultimately bring more objective methods into security prioritization and measurement. Over the next few slides, we'll look at some practical and simple ways to improve the issues we raised.

Recommendation I – Picking the Right Scale

- **Use the right measurement scale**
 - If measuring on something with interval/ratio variable, design it that way from the start
 - Example: impact in **dollars**, likelihood in **percentage**
 - But how do I do that??
 - Numerical estimation is something people can be trained in and improve on very quickly - "Calibrated Probability Assessment"
- **Show your work!**
 - You must justify that the math being done actually produces results!

Recommendation 1 – Picking the Right Scale

So, what are the recommendations for starting to solve these issues? One item is to reconsider potential mismatches in levels of measurement. If we are wanting to perform meaningful addition or multiplication on alerts, risk, or otherwise, we must use interval or ratio type measurements. If we are measuring the chance that something will happen pick a time frame and estimate the percentage likelihood it will occur within that period. If we are measuring what ultimately converts into dollars of impact, use dollars from the start. If this feels difficult and like you're wildly guessing, consider how much it bothers you when the scale was 1-5 or low to high, is this really different? Douglas Hubbard describes his book "How to Measure Anything in Cybersecurity Risk" how employees can be trained in numerical estimation to help calibrate the ability to meaningfully put numbers to these difficult situations, this skill is called "Calibrated probability assessment", and can be both easily learned and rapidly improved.

Once interval and ratio scale measurements are implemented, don't forget that any calculations performed on those measurements still needs to be vetted, validated, and explained, unlike the unjustified equations of the CVSS score scale.

Recommendation 2 – Drop the Numbers

Using a true ordinal measurement?

- **Drop the numbers!** Give up "if greater than CVSS Base Score 8 – patch"
- There are multiple ways to make an 8, which leads to sub-optimal prioritization
- Researchers from Carnegie Mellon SEI suggest this approach :
 - "*We suggest that the way to fix [the CVSS] problem is to skip converting qualitative measurements to numbers. CVSS v3.0 vectors could be mapped directly to a decision or response priority. This mapping could be represented as a decision tree or a table.*"¹
- Create a mapping of ordinal values to expected responses – example:

CVSS Attack Vector	Exploit Available?	Expected Response
Network	Yes	Drop everything, patch now!
Network	No	Patch ASAP
...
Physical	No	Patch when convenient

Recommendation 2 – Drop the Numbers

A second recommendation for those measurements that are truly ordinal can't be stated numerically is to drop using numbers for decision making and make decisions on the valid ordinals instead. With CVSS for example, instead of patching based on the dubious base score number, instead consider the factors that go into the Base Score calculation, which *are* valid. Using factors Attack Vector and Privileges Required enables you to make a mapping or decision tree for what to do when a vulnerability can be exploited over the network vs. only locally, or exploitation requires no permissions vs. high permissions. Using this approach would shift policy from something like "patch anything with a score over 8 first" to a "if the vulnerability has a High rated impact on confidentiality, or integrity, Privileges Required is None, and Attack Vector is Network – patch first." This second approach is decision tree based directly on the ordinal measurements and removes the dubious math that serves to obscure the true nature of the vulnerability. This is the course of action recommended in the research from Carnegie Mellon Software Engineering Institute on the problems with CVSS scores and how to fix them.²

Mapping actions directly based on the combined ordinal categories avoids measurement type errors we might otherwise make and clarifies what is truly the intended priority in scenarios that would be ambiguous when calculated based on false numbers. The ability to individually filter alerts in our tools based on both "alert severity" and "asset priority" for example, as well as map the combined answers to a priority (instead of calculate it) will keep responses more consistent and not allow our tools to overstate conclusions that should not be made.

[1] https://resources.sei.cmu.edu/asset_files/WhitePaper/2018_019_001_538372.pdf

Recommendation 3 – Investigate Quantitative Risk Analysis

Quantitative risk analysis:

- See Doug Hubbard's book¹ on quantitative methods for risk assessment
- His suggestions for quick improvement include:

Instead of	Do this Instead
Using 1-5, low-high scales for likelihood	Estimate probability of event occurring within a set time period
Using 1-5, low-high scales for impact	Estimate 90% confidence interval for monetized loss
Likelihood vs impact matrix for risk	Use "Loss exceedance Curves" generated with Monte Carlo analysis
Colored risk matrix areas (green, yellow, red)	Comparing risk tolerance to loss exceedance curve, prioritizing actions

Recommendation 3 – Investigate Quantitative Risk Analysis

For risk matrices, Douglas Hubbard recommends several quantitative analysis methods in his book with the research to back up their effectiveness. In chapter 3 for example, he lists the items shown on the slide above as quick improvements to the unscientific risk matrix approach. He shows how these methods are much more practical and lead to numerically-based assessments, and with training on estimation for those using them, can rapidly lead to much higher quality results.¹ Moving from measures like an "1-5" or "low-high" estimate to using numeric statements allows risk calculations methods from industries that excel in these predictions (like insurance), and bring much needed rigor and definition to our evaluations.

While those numerical quantitative analysis methods are far beyond the scope of this course, realizing the existence of the problem and the direction of the solution is the first step towards improvement. For information and excel worksheets to calculate loss exceedance curves as mentioned above see the example worksheets provided at [howtomeasureanything.com/cybersecurity/](http://www.howtomeasureanything.com/cybersecurity/).

[1] [https://www.howtomeasureanything.com/cybersecurity/](http://www.howtomeasureanything.com/cybersecurity/)

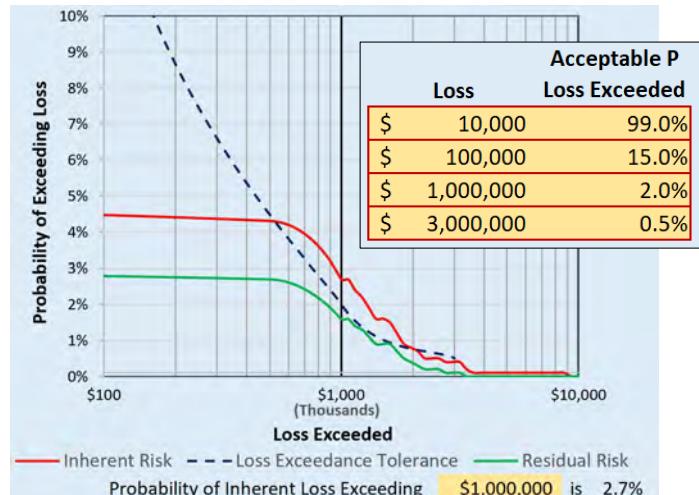
Example: Major Ransomware Attack Risk Estimation

Option 1: Traditional

- Probability = Low
- Impact = High
- Risk = Low

Probability				
Impact	Low	Med	High	
	High	Low	Med	High
	Med	Low	Med	Med
	Low	Low	Low	Low

Option 2: Quantitative



SANS

Building and Leading Security Operations Centers 81

Quantitative Risk Calculations

Which statement would you rather use to evaluate your risk of ransomware attack?

1. We estimate a **low likelihood** and **high impact** of a human operated ransomware event, which means the risk is green (**low**)
2. We estimate that there is a **5% chance** of a major ransomware event happening to this company within the **one year**. If that were to happen, we estimate within a **90% confidence interval** that the **impact** will be **between \$250k and \$3M** and a compensating **control with 50% effectiveness cost \$250k**.

With method 1, you would follow whatever procedure you lay out for low-level risks and apply compensating controls or not based on your decision.

With method 2 however, you can perform a Monte Carlo analysis, simulating many trials of the next year using the given likelihood and cost estimates, and see what the average impact is in each. From the aggregate data, you can create a "loss exceedance" curve, showing simultaneously the likelihood of that costs in that year will be above any given value with and without compensating controls, and how those expected costs compare to your risk tolerance. (This chart was generated by placing these numbers into the worksheet at howtomeasureanything.com/cybersecurity.) From the above, we can see for example that there is a 4% risk of an impact greater than \$700k, a 1% risk of a loss of roughly \$1.08M, etc.. Assuming the risk tolerance profile shown in the table, which sets the tolerance to probability of impacts at various levels, purchasing the compensating control brings the risk from above tolerance (top solid line) to an acceptable range (lower solid line). This makes it easier to decide if you should purchase the control, the chart shows a clear undesired level of risk of impact between roughly \$500k and \$1.1M if no controls are implemented.

Clearly the second method is a much better approach and can be much more easily used to justify purchases and actions. Even though the numbers were estimated, combining available industry data (average ransomware payment surveys) with calibrated assessments from multiple people, quantitative analysis produces much more detailed, actionable, and meaningful estimate of the full spectrum of potential impact. If needed for a high-level presentation you can still simplify the decisions and levels of risk, but now the meaningful data behind those decisions are present and driving the decisions in the background.

Measurement and Prioritization Issues Summary

- Understand, and use the correct measurement types
 - Consider your ordinals vs. interval vs. ratio type data
 - Avoid falling for common measurement type issues
 - Make data numeric (interval and ratio) where possible
 - Use decision trees and guidance where numbers cannot work
- Be prepared to convert data for presentation
 - The numbers exist, but are presented qualitatively
- There *is* power in communicating a clear message
 - But behind the scenes, your work should include more detail
- **Bonus:** If asked, you have a knock-out explanation!

Measurement and Prioritization Issues Summary

In closing, since prioritization plays such an important role in so many facets of day-to-day information security, it's important we use as objective data as possible. While the estimation and imprecise methods of the past are still commonly in use today, hopefully this section has opened your eyes to some of the issues with them and can guide you towards improvement. In short, be aware of the levels of measurement that are really available for a situation (ordinal vs. interval vs. ratio) and consider what you are doing with that data, and if that is a valid operation for that type. When it's not possible to use meaningful numbers, direct decisions based on categories can be defined instead to help those making patching or alert prioritization decisions.

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

- **Effective Execution**
- Staff Retention and Burnout Mitigation
 - *Exercise 5.1 – Training and Career Development Planning*
- Metrics, Goals, and Effective Execution
- Measurement and Prioritization Issues
 - *Exercise 5.2 – Creating, Classifying, and Communicating Your Metrics*
- Continuous Improvement
- Strategic Planning and Communications
- Analytic Testing and Adversary Emulation
 - *Exercise 5.3- Purple Team Assessment*
- Automation and Analyst Engagement
- Summary and Cyber42 – Day 5



This page intentionally left blank.

EXERCISE 5.2

Exercise 5.2:

Creating, Classifying, and Communicating Your Metrics

OBJECTIVES

- Derive which metrics you should (and should not) be collecting
- Use the "GQM" system to map goals to collected metrics
- Explain the factors that make a chosen metric useful to measure
- Classify and gain insight into the nature of your metrics
- Suggestions for effective presentation and communication of metrics

Exercise 5.2: Creating, Classifying, and Communicating Your Metrics

Please go to Exercise 5.2 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

- Effective Execution
- Staff Retention and Burnout Mitigation
 - *Exercise 5.1 – Training and Career Development Planning*
- Metrics, Goals, and Effective Execution
- Measurement and Prioritization Issues
 - *Exercise 5.2 – Creating, Classifying, and Communicating Your Metrics*
- Continuous Improvement
- **Strategic Planning and Communications**
- Analytic Testing and Adversary Emulation
 - *Exercise 5.3- Purple Team Assessment*
- Automation and Analyst Engagement
- Summary and Cyber42 – Day 5



This page intentionally left blank.

Strategic Planning and Communications Overview

- Always have a (long-term) plan!
- Where is the SOC going? Evolving capabilities, improving metrics, new ways of supporting the organization, and maturity
- Telling the right story
- Communicating your story, and your strategy, effectively



Strategic Planning and Communications Overview

Managing a SOC isn't about being the best analyst, engineer, or responder in the group. Much of the job is being able to rise above tactical considerations and ensure the entire operation – people, process, and tools – are moving in the right direction as a single combined set of capabilities. Security is an often messy and unpredictable discipline, so this forward (planned) motion normally doesn't just happen on its own. It requires careful planning, execution, and communication with your stakeholders and constituents about where you are today and where you're going in the future. While the work we do is sometimes sensitive, you don't want the people writing the checks to wonder what you're doing for the organization or what improvements you plan to make for the budget you request each planning cycle. Having a plan and telling the right story is a big part of gaining buy-in from your team and your organization, and that's what we're going to talk about in this section.

Building a Strategic Plan

- Start with a vision statement and key themes:
 - “Our vision is to enable our users to save lives by offering them peace of mind and freedom from work disruption.”
 - Key themes are availability, transparency, and a focus on positive user experiences
- Figure out what, specifically, they need from you and you from them
- Make sure you’re aligned to organizational vision and priorities
- Refine and improve over time

Building a Strategic Plan

As defenders, we sometimes get so focused on technical details and tactical, day-to-day operations that we sometimes forget to look at the “big picture” of defending the enterprise. As a leader or manager, this is not something you can afford to do. Even if your management has not requested one, it’s vitally important that you always have a long-term plan to address gaps, make improvements, and add capabilities to the SOC. This plan may change or have priorities rewritten by senior leadership, and that’s ok if it helps the SOC meet the needs of the organization. The point is that you, and the team, have a “north star” that guides your metrics, engineering projects, and the value that your team brings to the organization.

Graham Kenny, writing for Harvard Business Review, recommends starting with a vision statement (hopefully laid out in your SOC charter) and then following this high-level approach to developing a strategic plan:

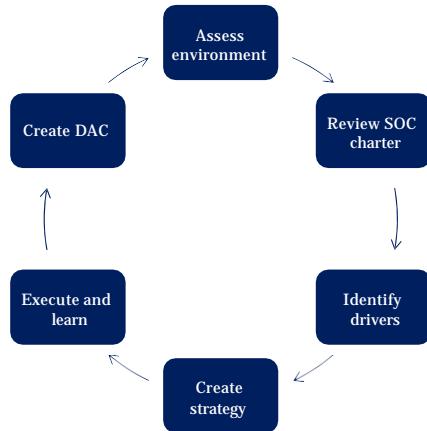
1. **Recognize your dependencies and key stakeholders:** this is also hopefully included in your SOC charter and is more about defining stakeholder roles that specific individuals or groups. Are they business owners? Users? Partners? Customers?
2. **Identify your “target” stakeholder:** who is the person you are really working for, whose requirements come before everyone else’s? Is it a consumer? The business owner?
3. **What do your stakeholders want** and need from you?
4. **What do you need** from your stakeholders?
5. **What is your organization’s position** on your stakeholders’ requirements? For example, if you’re at a public sector agency and the SOC’s primary goal is protecting citizen data, what is the agency’s position (and ultimately commitment to) that goal? How is it willing to prioritize, innovate, and invest in keeping that data safe, and by extension how willing is it to support you in fielding the best possible SOC to help make that happen?
6. **Continuously improve** and refine your plan to ensure it aligns to stakeholder requirements, in the correct priority order, and in a way that reflects organizational objectives around those requirements.

In this section, we’re going to talk about implementing this kind of approach in the SOC and some key reference models and skills we’ll need to make it happen.

Future Planning and Strategy

Easy to get lost in the "fog of war" - how do you take a distanced view and continue in the right direction?

6- Step Process:



"Perception is strong and sight weak. In strategy it is important to see distant things as if they were close and to take a distanced view of close things"

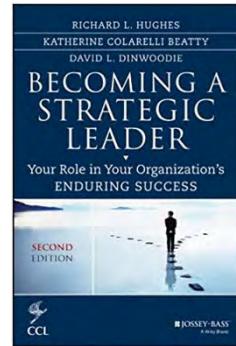
—Miyamoto Musashi, legendary Japanese swordsman and philosopher

Future Planning and Strategy

We saw a high-level process for developing a strategic plan, but now let's talk about a more refined 6-step approach to developing a strategy specific to the SOC and its mission. Building a strategy can be a fun and challenging thought exercise for "what could be," but it's not just about brainstorming all of the fun technical things we want to do. We need to maintain DAC: direction, alignment, and commitment with our organization and the constituents we serve. Focusing on those elements and ensuring our strategy reflects them will keep our key stakeholders bought in to what the SOC is doing and where it's going, which is extremely important if you want to get the support and investment you'll need to get there. Here's the process at a high-level: assess the current environment and what the SOC is capable of today, review your charter and needs of your key stakeholders, identify drives (including stakeholder support and input) that will help you fulfill your charter now and in the future, create a plan to translate that support into action and capability via your people, process, and technology, execute that plan, and monitor to ensure you're achieving DAC. Let's walk through each of these.

Assess the Environment

- Think back to day one: how does your organization create value? Consider market forces, competitors, merges and acquisitions, or new innovations.
- Think back to day three: how is the threat landscape changing?
- Review your KPIs and OKRs: is it time to raise the bar?
- *Example: we are acquiring an e-commerce business with a large, complex cloud infrastructure*



Becoming a Strategic Leader by Hughes, Beatty, and Dinwoodie discusses strategy in a volatile, uncertain, complex, and ambiguous environment – sound familiar?

Assess the Environment

In the first phase of strategic planning, as described in the book Becoming a Strategic Leader, we can conceptualize the SOC as our own unique business unit. We'll set aside the fact that our SOC actually *costs* money rather than making money and think instead about the value it creates. How does it compare against other security capabilities in similar organizations or in the same market? How does this align with the threat landscape as it relates to us today? Think back to the threat research and defensive planning we did back in book one. Are our KPIs and OKRs aligned to those goals? Are our target values and major initiatives tied to the outcomes we want? You should be able to trace most of the core capabilities of your SOC and the direction they are going to the answers to these questions.

Review SOC Charter

- Your SOC charter (hopefully) includes the team's mission and vision
- Reflects input from cross-functional leadership represented in the steering committee
- Is the SOC meeting the needs of its constituency, based on all of these points of view?
- Are there aspirational elements of your mission that you can make more demonstrable?
- *Example: our charter does not currently include defending cloud assets or a focus on availability*



Review the SOC Charter

Turns out a lot of the foundational work we have done is paying off! We discussed the SOC charter in book one as well, and now it's time to refer to it to make sure the technical direction of our SOC is aligned to the needs of our constituency. Make sure that your charter reflects the needs of the organization as it exists today and as it will exist tomorrow – think about initiatives like IT transformation and corporate acquisitions – and ensure that your strategic goals in the SOC reflect your mission as laid out in your charter.

Identify Drivers

- Also known as *key success factors*
- Different from objectives and wildly important goals, but addressing drivers correctly will bring you closer to them
- Do you need to revisit your team, processes, or tools based on any of the internal or external factors you considered in step one?
- *Example: We need to assess and deploy cloud security controls and revisit our monitoring and incident response processes*

Identify Drivers

We talked about wildly important goals in our section on metrics and key success factors, or *drivers*, are slightly different. These are basic functions, tools, processes, or capabilities that will position your team to achieve your goals. They might be enablers or dependencies, but whichever category they fall into, once you commit to a plan you don't want to miss your goals because you lacked some key ingredient to success. This is especially true when management dictates a strategic goal to you as the SOC manager. Get into the habit of identifying key drivers for each goal *before* you commit and make your plan.

Create Strategy

- *Strategy is "the pattern of choices we make to position ourselves for superior performance over time."*
- *Example: to become a cloud-oriented SOC, what vendor relationships, standards, architecture, and telemetry do we need to build and/or learn? How will we define success in this new environment?*

Create Strategy

The next step in strategic planning is to devise the strategy itself. We can define *strategy* as “the pattern of choices we make to position ourselves for super performance over time”. More simply, we want to do well and we want our SOC to do well. The question is, how do we define success? And how will we make it happen. Here’s an example of this question expressed in a more specific operations context that, when answered, could describe a strategy for a cloud-first SOC.

Execute and Learn

- Ensure tactics (daily operational decisions) align to strategy by sharing with your team and getting buy-in
- Monitor your KPIs and OKRs to identify trends and patterns that indicate success or the need to adjust
- *Example: we have integrated new cloud controls into our monitoring infrastructure, but our time to identify metrics have taken a nose-dive and aren't coming back up*



Execute and Learn

“Stop trying to hit me and hit me!” Matrix fans will remember how ready Neo was to try out his new kung-fu skills on his mentor, Morpheus. But Neo quickly learned that understanding concepts and theory does not an expert make. Only by trying, learning from mistakes, and improving does Neo finally begin to gain the upper hand. We can all agree that execution in the SOC is pretty much exactly like kung fu in a virtual dojo (Right? Anyone?), in that we must use the KPIs and OKRs we have laid out to measure and adjust over time based on results. Monitoring your metrics isn’t just a way to check day-to-day operations; it is a way to ensure your SOC remains on the right path towards your larger strategic goals.

Create DAC

- Create *Direction, Alignment, and Commitment*
- Embrace polarity, or both-and, thinking:
 - "I need to be the expert" versus the need to learn
 - "We need to be successful" versus the need to hit KPIs
 - "We need to accomplish the mission" versus team morale
- Make connections and be accessible
- Beliefs drive behavior – build and personify a positive culture within the team
- *Example: "bringing our identification KPI back up for the new cloud environment will take time and effort, but these are the things we must do to get there"*

Create DAC

Finally, our strategy must create *DAC*: direction, alignment, and commitment. As an operational leader, you will be in a unique and sometimes challenging position to make this happen within your teams. Consider these types of “polarity” thinking:

- The need to be the expert versus the need to learn
- The need to be successful as a team versus the need to hit your metrics targets
- The need to accomplish the mission at all costs versus the need to keep the team engaged and motivated

Any operations leader will tell you that these are often diametrically opposed points of view and that leaders often struggle to decide which one is “right”. The answer is that both could be right depending on the situation. Your job is to keep these directions in balance and recognize when certain scenarios may require one way of thinking over the other.

Measuring SOC Maturity with SOC-CMM (1)

- The **SOC Capability Maturity Model** (SOC-CMM) was created by Rob van Os to measure SOC characteristics and features
- Based loosely on CMMi
- Measures technical capability and overall maturity
- Consists of 5 domains and 25 "aspects"
- Continuous security model where aspects can grow in maturity independently

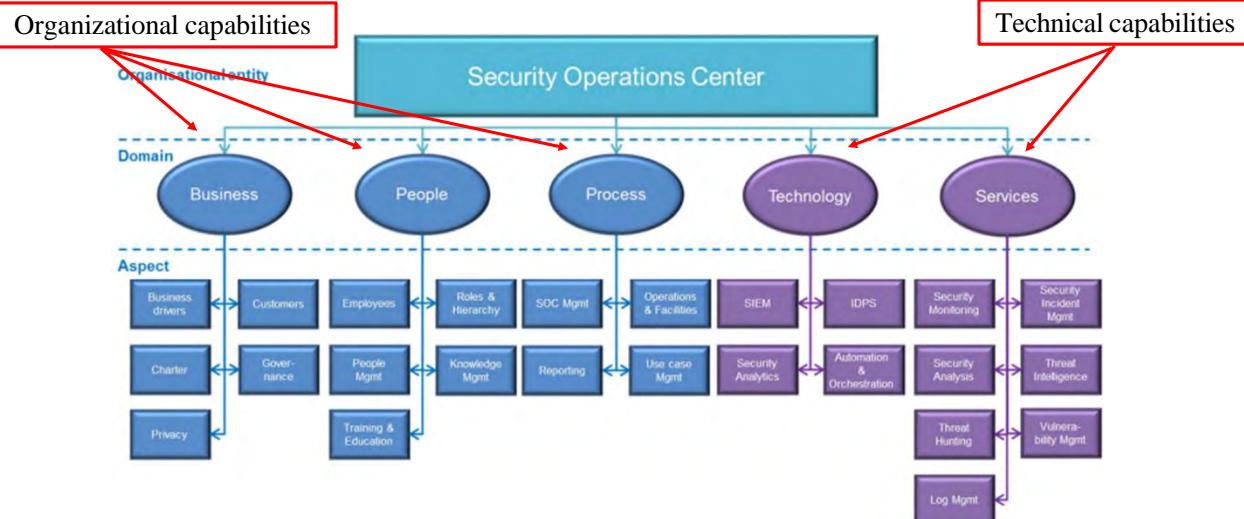


Measuring SOC Maturity with SOC-CMM (1)

Now let's talk about a more tangible tool for strategic planning: measuring and improving SOC maturity with the SOC-CMM. The SOC-CMM was created by Rob van Os for a Master's thesis research project. You can think of it as a reference model for measuring the technical capabilities of a SOC and the maturity of its processes. Based loosely on Carnegie Mellon's CMMi framework, SOC-CMM measures SOC capabilities on two axis: technical capability and overall maturity. This is an incredibly useful feature, since some SOCs will have lots of tools and technologies but few mature processes to govern them (or many processes but lacking technical capability).

The SOC-CMM covers 5 domains – Business, People, Process, Technology, and Services – and 25 aspects within those domains. The maturity scale for these domains ranges from Non-existent to Optimizing. Technical capability is measured in the Technology and Services domains on a scale ranging from Incomplete to Managed. You can read more about SOC-CMM and download related resources created by Rob at <https://www.soc-cmm.com/>.

Measuring SOC Maturity with SOC-CMM (2)



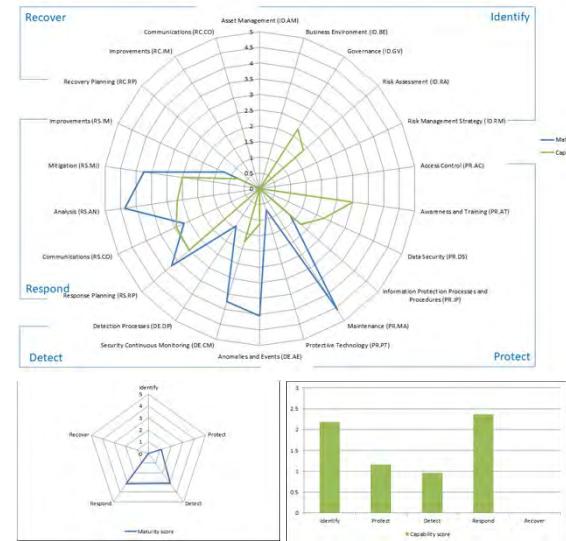
Measuring SOC Maturity with SOC-CMM (2)

Here are the SOC-CMM domains and aspects visualized. Note that technical capability is only measured within the Technology and Service domains, but maturity is measured across all domains. Technical evaluation covers control types, such as IDPS and SIEM, as well as governance, support, documentation, and specific functionality. Organizational capabilities cover aspects like SOC skills, knowledge management, and team composition.

Rob has created a free self-assessment tool to help conduct an evaluation using this model, which is aligned to the NIST Cyber Security Framework and can be used for either a full assessment or "quick-scan" assessment to demonstrate progress. The tool is built in Excel and provides some nice visualizations as well as results mapping to NIST CSF. You can download that tool at <https://www.soc-cmm.com/downloads/latest/>.

Measuring SOC Maturity with SOC-CMM (3)

- Maturity and capability scoring
- Alignment to NIST Cyber Security Framework (CSF) capabilities
- Possible to score highly on maturity but lower on capability – for example, lots of tools and processes but little improvement or analysis



Measuring SOC Maturity with SOC-CMM (3)

The SOC-CMM tool provides some useful scoring aligned to the NIST Cyber Security Framework (CSF) and some nice visualizations, as shown in this slide. Again, SOC-CMM measures capability and maturity on two independent scales, meaning an organization could score very highly on one and lower on another. Based on the graphs in this slide, this organization has lots of tools, processes, and procedures, so scores highly on the capability scale. However, they score relatively low on continuous improvement, performance measurement, and repeatable hunting and analysis functions. In this way, the organization can see progress for the investments they've made, but we can guess where they should focus their efforts next.

Storytelling in Security

- Security can seem like a mysterious black box of processes, technology, and jargon
- In the worst case, security can seem like a needless roadblock
- No matter how good your technical delivery is, success often comes down to telling a good story!



*"There's nothing in the world more powerful than a good **story**. Nothing can stop it. No enemy can defeat it."*
-Tyrion Lannister

Storytelling in Security

As shocking as it may be to those of us who have spent a significant amount of time in security, cyber defense is not always top-of-mind for business leaders and executives. Once you've accepted this hard truth, you might be able to also admit that security can seem like a mysterious process that comes at a great cost without providing a lot of tangible value. In the worst case, security can be seen as a needless roadblock that infrastructure and application support teams can simply do for themselves. The bottom line here is that no matter how good your team is, and how many quiet, uncelebrated successes you have had defending your enterprise, your success will likely come down to telling a good story about what it is you all do in your cold, dark security dungeon all day. Constructing a narrative about SOC capabilities and successes is an often overlooked, but **extremely** important part of being an operations leader.

Constructing a Narrative

Tips:

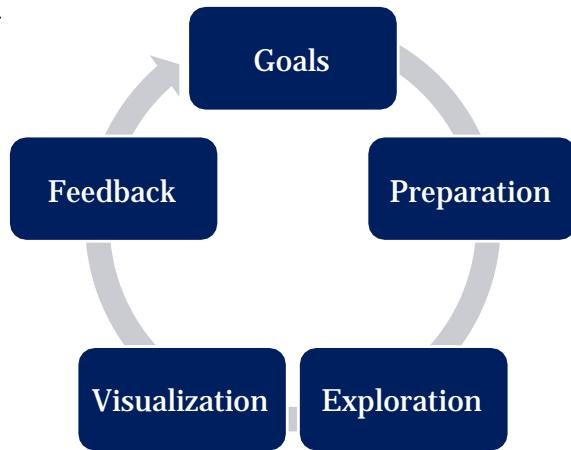
- Know your audience
 - Executive? Technical? Business-oriented? People-oriented?
- Don't rush
- Make it matter
- Cover less for greater impact
- Gift or an ask?

Constructing a Narrative

Good storytelling starts with the right narrative and an understanding of your audience. Assume you have been asked by your boss to give a briefing on the SOC to an audience of business leaders from across the organization. Many of us would throw together a summary of core processes and maybe include our charter. Others might even have a “SOC Presentation Deck” that they dust off for precisely this purpose. The challenge here is that no two audiences are the same, and we must understand that audience in order to tell them a compelling story. What do they care about? What motivates them in their professional lives? Your SOC briefing, and any other discussion of your team’s capabilities, should be told through this lens. Other considerations in constructing a narrative are to cover less for greater impact – this is usually counter-intuitive for those of us who enjoy a decent amount of technical detail. Finally, consider the goal of your narrative – are you giving a gift (i.e. educating your audience) or are you asking for support, investment, participation, etc.?

Communicating Visually (1)

- Most efficient path to human understanding
- Hack the cognitive system!
- Basic 5-step approach to data visualization:



Communicating Visually (1)

So, now we know our audience, we have our rough narrative in mind, and now it's time to think about how we are going to tell our story. Visual communication is the most efficient path to human understanding [1], and it can be a powerful tool in conveying progress, challenges, status, and other SOC narratives. In order to effectively communicate complex technical concepts and data to a wider audience, we'll need to hack the cognitive system through visual communication!

Consider this approach, written about by Balaji Balakrishnan in his white paper *Security Data Visualization* [2]:

1. **Identify your goal:** Think about what you want your audience to take away from your report or presentation. Consider the audience's technical knowledge, background, area of interest, and what kind of response you're looking to elicit. While data visualization may help us identify trends or events of interest in our investigations, we want our reporting to be goal-driven and not data-driven. In other words, we want our visualization to tell the story we want versus leaving it up to interpretation.
2. **Prepare the data:** "Cleansing" the data is an important step in developing quality visualizations. Much like our SIEM must parse, index, and normalize data to prepare it for searching, we need to be sure the data we're using is complete, consistent, and available in a common format.
3. **Explore the data:** Here's where we draw on our analysis expertise to formulate questions or hypothesis and look for evidence that disconfirms those theories. This process may require some free-form queries, pivoting through our data sources, or enrichment to gather additional context.
4. **Visualize the data:** Find the best type of presentation based on the data type and the message (more on that in a few slides)
5. **Gather feedback:** Solicit feedback from your audience to ensure you are telling the story you intend.

Whether you're working your team to create SOC dashboards or reports for management, adhering to best practices for visual communication is a must. This is especially true when your audience lacks the technical knowledge or context to consume what can often be very complex technical information. Next, we'll discuss some best practices for visual communications and some of the underlying concepts behind strong visualization and dashboard design.

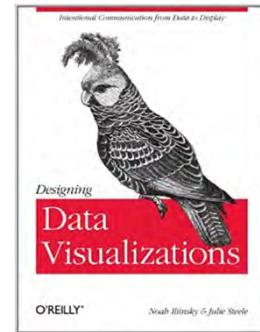
1.Jay Jacobs and Bob Rudis, *Data Driven Security: Analysis, Visualization, and Dashboards*, 2014.

2.<https://www.sans.org/reading-room/whitepapers/metrics/security-data-visualization-36387>

Communicating Visually (2)

Tips:

- Know your audience
 - Executive? Technical? Business-oriented?
People-oriented?
- Don't rush
- Make it matter
- Cover less for greater impact
- Gift or an ask?



Communicating Visually (2)

When designing visualizations for reports, dashboards, and other content, it's a good idea to keep your audience in mind. Are they technical? Do they have any understanding or context for what you're trying to communicate? In security, we often make the mistake of assuming everyone knows (or cares) about what we do – painting a compelling picture that speaks to our audience may be the best way to overcome a lack of understanding or interest. This is not a process we want to rush. We must make sure we're asking the right questions of our data so that the answers are evident and relevant. Does an executive outside of the technical organization need to know which kill chain stages show up most often in our incident response metrics, or might they be more interested in the number of incidents that has impacted their division or the corporate bottom line? We also need to consider focus – recall that we have already discussed the limits of short-term memory. Selecting fewer items for our visualizations will not only help the reader or audience retain the information more easily, but it will also draw our attention more rapidly to the things that really matter. Finally, consider your goals for visual reporting – are you trying to inform, surprise, or get something from your audience (buy in, investment, decisions, etc.)? These factors will drive how you employ the various components of visual communication, which we'll talk about on the next slide.

You can read much more about these best practices in [Designing Data Visualizations: Representing Informational Relationships](#) by Julie Steele and Noah Iliinsky.

Components of Visual Communications

- **Position:** information processed via series of movements around the screen, focusing on obvious features first
- **Shape:** we're designing for the screen or paper (2D mediums)
- **Length:** good for communicating quantitative and other precise data
- **Size:** imprecise way of communicating specific values; implies values relative to each other
- **Color:** subjective, interpreted relative to the surrounding environment (ex: darker = emphasis)

Components of Visual Communications

When someone looks at dashboard, infographic, or other visualization on the page or screen, their eyes are naturally drawn to the most obvious features first. From there, a series of small movements called *saccades* shift their eyes around the screen to ingest other visual information. This is sometimes referred to as *flow* in the visual field, and it plays an important role in the **position** of various elements we want to use. Also keep in mind that we are building for two-dimensional mediums here: paper, whiteboards, screens. Adding three dimensional objects to our communications might seem cool but will only clutter the visual field and degrade our audience's ability to digest the information we want to convey – stick to simple, flat **shapes** in your dashboards and visualizations.

Length is a component we can use to convey quantitative and other precise data; we most often see this element employed in line charts or bar charts, where quantitative data can be overlayed with categorical data. **Size** is best used for relative values, where you want the audience to understand one value is larger or smaller than another but the actual numbers or quantities are less important. Finally, **color** can sometimes be used for emphasis or highlighting with the caveat that *not everyone can see color or will interpret it the same way*.

Building Effective Dashboards

- Don't pack the dashboard
- Make important items obvious
- Don't try to "show your work"
- Avoid excessive framing, chart variation
- Choose colors and fonts wisely
- Dashboards build trust
- "So what?"



If you recognize this, you may be getting old.

Building Effective Dashboards

There are many nuances to building good visualizations into your tools or for management reports, but there are a few tips that are universal:

- Don't pack the visual field with various shapes, graphics, and bits of information; you want to limit the viewer's eye movement so that they can focus on the elements that are most important or most fundamental to the story you're trying to tell
- Highlight important items – things requiring decisions, action, or follow up – through color and placement
- Don't try to "show your work" by breaking out the underlying data and computations behind your visualization; a well-designed dashboard should communicate what the viewer needs to know without explaining itself
- Avoid excessive framing for your graphical elements and use of different kinds of charts – simpler and cleaner is better
- Choose colors and fonts wisely; the wrong choices could impact the visual flow of the screen or page and confuse the reader about where their focus should be

Above all, remember that dashboards are meant to communicate information or tell a story. The answer to the question, "so what?" should be evident to the audience. If your management or your team must guess why a given piece of information or element has been included, it's time to go back to the drawing board. Equally as important is trust. Have you ever been presented with a report or dashboard that you know is missing key data or showing incomplete or inaccurate results? These discrepancies result in an immediate lack of trust in the source of the information or the underlying data model. Whether you are communicating up your management chain or to your own team, make sure that the information and its presentation tells a realistic, trustworthy story.

Strategic Planning and Communications Summary

- Communicating is a BIG part of your job as a SOC Manager
- Know where you are and where you're going – always have a strategy and a plan to execute it
- Involve your team and your stakeholders in the process
- Tell a good story



Strategic Planning and Communications Summary

Whether you like it or not, communicating is now a major part of your job as an operations leader. You're responsible for a wide variety of technical tasks, details, people, tools, processes, requests, and other independent variables that must somehow coalesce into something that moves your organization towards demonstrably better security. Having a strategy that ties these elements together, guides your daily execution, and that you *regularly share* with others will position you and your team for success. Don't underestimate the power of a good story, and never underestimate the power of a good story told repeatedly. Your SOC strategy and the narrative that accompanies it will evolve over time and that's perfectly ok – the important thing is to give your team and your constituents something they can connect with and get behind.

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

- Effective Execution
- Staff Retention and Burnout Mitigation
 - *Exercise 5.1 – Training and Career Development Planning*
- Metrics, Goals, and Effective Execution
- Measurement and Prioritization Issues
 - *Exercise 5.2 – Creating, Classifying, and Communicating Your Metrics*
- Continuous Improvement
- Strategic Planning and Communications
- Analytic Testing and Adversary Emulation
 - *Exercise 5.3- Purple Team Assessment*
- Automation and Analyst Engagement
- Summary and Cyber42 – Day 5

This page intentionally left blank.

In This Module

In this section:

- Analytic coverage checks and visualization
- Atomic analytic testing
- Commercial security testing solutions
- **Red Teaming** vs. **Purple Teaming**
- Penetration testing vs. adversary emulation
- Assessment methodology

In This Module

To wrap up the course, this final module will cover one of the most important pieces of SOC improvement—analytic testing and SOC assessment. Throughout this section, we'll discuss how to prioritize and visualize your current detection coverage and objective measure and improve it. Once the SOC has gone through this exercise, you should not just assume it will all work as planned, testing will be necessary. To that end, we'll discuss the testing of SOC capabilities through automated unit testing as well as things like Red/Purple Teaming, and adversary emulation. We'll close up the course with recommendations on how to decide when you're ready to test your defenses, the different methods available, and how the results can be tracked over time to show the continuous Blue Team improvement your SOC should be striving for.

Important Question: How Do We Know Our Analytics Work?

2 Questions:

1. Are we confident our analytics will catch the **right** thing?
2. Will it catch **every instance** of bad activity?
 - What's worse than a false positive? A **false negative!**

You tracked techniques, wrote analytics, awesome...

- But how do you know they actually work?
- ...and will continue to work
- **Answer: Continuous analytic testing!**



Important Question: How Do We Know Our Analytics Work?

Although false positives are annoying, at least they are mostly harmless. False *negatives*, however, are a much bigger issue, and one the SOC must have a strategy to avoid. One of the trickiest problems to deal with after an analytic is created is ensuring that it works and will continue to function as the environment and endpoints change. There's nothing worse than thinking you should be able to catch an attack then finding out later that your analytic failed to alert, and an adversary was able to leverage the blind spot to intrude into your network!

How do we prevent this scenario? Analytic testing, not just at the point of creation, but on a continuous basis! In the past, continuous analytic testing would have meant having not only an incredible amount of knowledge on which attacks were being used, but also knowing exactly how to execute them, and how to script up a large number of tests. Fortunately, the industry has now solved this issue and provided tools that give us both the attack technique knowledge we need (MITRE ATT&CK) and software to test implementations of each technique.

Analytic Testing Methodology

1. Run commands that look like attacks on a **test** machine
 - Remember to run different versions of the same attack
2. Collect info from endpoint/network during the "attack"
3. Did the attack...
 - Set off an alert? Great!
 - Create log data, but not set off an alert? Verify your analytic!
 - Slip by without anything happening? Time to change logging setup!
4. Repeat as often as necessary

The tricky part: Doing this repetitively in an **efficient** way

Analytic Testing Methodology

The basic method for individual analytic testing is straightforward—first you run commands that look like an attack on a known test endpoint. Afterward, check the logs and network traffic that were produced and recorded by your security tools. Be sure to try the same attack in different ways, if applicable, to verify all of them are caught. Did the attack set off an alert? Great! Time to move on the next one. If there was no alert, was there at least evidence recorded of the attack? You may just need to adjust the alert conditions. Did the attack fully slip by without creating any visible mark? If so, it's time to adjust what gets logged so these attacks do not get by from a real attacker.

The tricky part about this is that most organizations have many different analytics and performing these types of tests manually each time would create an enormous burden on the SOC. Fortunately, multiple frameworks and tools (listed on the next page) have been created to take care of much of this for you. Each contain a list of individual command line commands that are representative of attacks in the ATT&CK matrix. Using attack simulation tools allows the team to run these types of tests on a continuous basis and keep constant tabs on the status of their alerting capability. The goal should be to use these frameworks on a regular basis to perform checks and establish all techniques included have associated analytics that are functioning properly, and to identify any critical visibility gaps that need to be fixed.

Analytic Testing Tools

Automated / Scripted Assessment

- **Atomic Red Team¹** by Red Canary
- **RTA²** by Endgame
- **Metta³** by Uber
- **CALDERA⁴** by MITRE
- **Infection Monkey** by Guardicore
- **Purple Team Attack Automation** by Praetorian

Commercial

- Breach and attack Simulation (BAS)
- Continuous Security Validation (CSV)

Manual Assessment

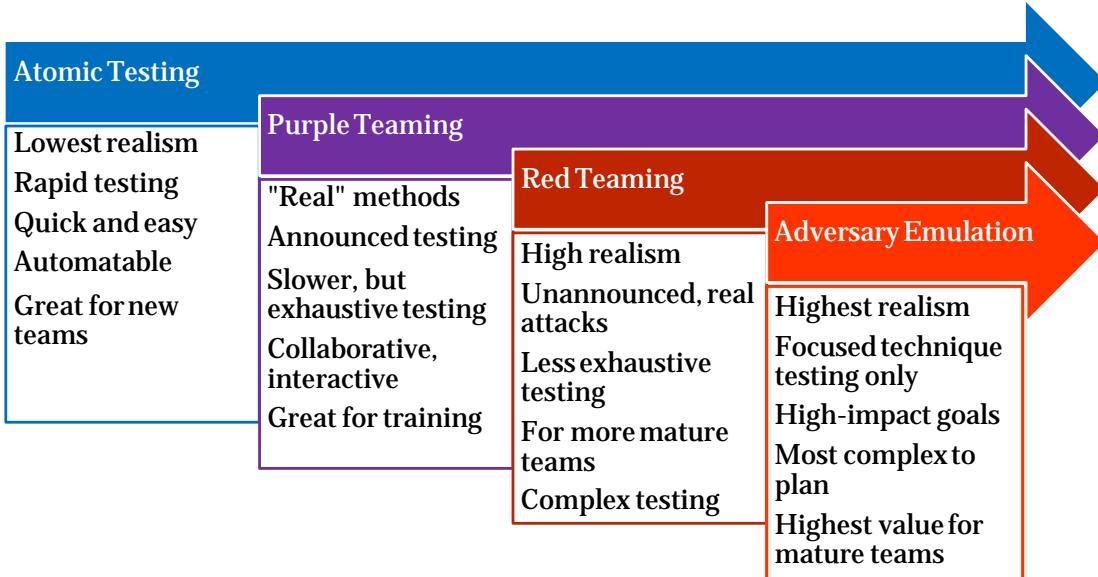
- Metasploit / Armitage
- Cobalt Strike
- PowerShell Empire
- Covenant

Analytic Testing Tools

To solve the continuous analytic testing problem, multiple open-source tools have been created by a number of organizations. Some of the most popular are listed on this slide. The tools themselves can be roughly divided into two styles—tools meant for automated/scripted execution, and tools meant for Red Teamers and penetration testers to use in a more interactive way. You've likely heard of many of the manual testing tools such as Metasploit and Cobalt Strike, but it's the automated and scriptable tools we're interested in—things such as Atomic Red Team by Red Canary or RTA by Endgame. These are designed to be run in a repetitive fashion, as we have been discussing here, and can help your SOC run checks repeatably in the most efficient way possible. For details on these frameworks, see Erik Van Buggenhout's presentation from Pen Test Hackfest Europe Summit 2019⁶ titled "Automated Adversary Emulation Using CALDERA".

- 1 <https://github.com/redcanaryco/atomic-red-team>
- 2 <https://github.com/endgameinc/RTA>
- 3 <https://github.com/uber-common/metta>
- 4 <https://github.com/mitre/caldera>
- 5 <https://www.guardicore.com/infectionmonkey/>
- 6 <https://www.slideshare.net/erikvanbuggenhout/adversary-emulation-using-caldera>

Self-Assessment Progression



Self-Assessment Progression

This slide shows the spectrum of different types of tests and the pros and cons of each. Smaller teams and teams just starting out may want to focus on atomic testing as it is generally easier to automate and it's not a large time sink. As your team becomes more advanced and surer of their capabilities, however, progressing into Purple Teaming, Red Teaming and, ultimately, specific adversary emulation is a great way to ramp up focus and realism. Adversary emulation, as described in this course, is considered a specific type of Red Teaming, the difference being whether the Red Team's activities are meant to mirror an existing actor or not. In the next slides, we'll further explain the methods and benefits of each of these types of tests.

Purple or Red?

Which style is more appropriate for you?

	Purple Team	Red Team
Uses realistic threat TTPs	Yes	Yes
Blue Team Interaction	Constant	Limited (if caught)
Goal	Improve current state	Assess current state
Most value when	SOC is newer	SOC is mature

Purple or Red?

There are multiple options for running a self-assessment (and a whole spectrum of options in between), but in general, testing can be broken down as either a "Red Team" or "Purple Team" exercise. Both Red and Purple Team activities require the use of realistic threat actor TTPs, but like pen testing vs. adversary emulation (discussed on the next slide), the engagement style and goals are different. In both cases, the goal is to measure the difference between what "is" and what "should be".

A Purple Team exercise is a great first step beyond atomic testing and has both the Red and Blue Team working hand in hand (and we will see a detailed example of how these work in our final exercise). These tests are done with both teams working simultaneously, perhaps even sitting at the same table the entire time. Since the Blue Team will be fully aware of the attacks, they can look at the logs recorded and alerts that fire after each one, noting whether the analytics and security tools worked as expected. In this type of test, the goal is on *rapid improvement*. Multiple attack techniques are tried in rapid succession and as each attempted technique is either prevented, blocked, or missed, the Blue Team can take note and turn around quick fixes for each item. These types of assessments have an enormous value if the SOC is newer and is unsure of their detection capacity but are also useful to perform on a regular basis. They are best used as a way to verify operation of analytics and build confidence that a real attack can be caught.

Red Teaming, on the other hand, is often done more in the style of a penetration test. The Blue Team generally will not know the attack is coming, which leaves the Red Team to try to avoid the prevention and detection controls that are in place and not tip off the SOC. Because of this methodology, the Red Team exercise is naturally more focused on an *assessment of the current state of affairs*. These types of tests are most helpful when the SOC is at a more mature phase and has good reason to believe they stand a chance of detecting and defending against a real attacker.

Penetration Testing vs. Red Teaming/Adversary Emulation

As described by SANS Purple Team course author Erik Van Buggenhout...

Penetration Testing:

- Identifies and exploits vulnerabilities on a (series of) system(s) to assess security
- Focused on a specific scope, typically an application or network range
- Primarily tests prevention, typically less focused on detection

Adversary Emulation:

- Assesses how resilient an organization is versus a **certain adversary / threat actor**
- Focused on the execution of a **scenario** (typically defined by a number of flags)
- Typically **tests both prevention and detection** (so is less valuable if there is no Blue Team)



Penetration Testing vs. Red Teaming/Adversary Emulation

You may have heard about Red Teaming and adversary emulation before (they are the same when the Red Team is pretending to be a specific adversary) and wondered how it is different from typical penetration testing. Erik Van Buggenhout, lead author of both SANS Purple Team courses (SEC599 and SEC699), makes the distinction this way: While pen tests focus on identifying potential vulnerabilities and exploits on a series of systems in a network, adversary emulation is focused on checking how vulnerable your organization is to a *specific threat actor*. The key differentiator in adversary emulation is the execution of a crafted scenario (often with the goal of acquiring specific "flags") that is based on threat intel and represents attacker goals, tools, tactics, and techniques you would expect to see from your actual enemies. Penetration testing often uses standard pen testing tools and focuses on a more generic set of goals such as domain admin access or data theft. It is more centered around exploring the art of the possible. Both items are necessary and bring value to the organization, but their focus is different.

Adversary Emulation: The Ultimate Goal

- Great for once you have built confidence...
 - Threat intel should identify the TTPs of your adversaries
 - These form potential **real attack chains**
 - Emulating relevant chains creates **adversary emulation**
- Attack chaining is supported in multiple tools
- Aligns tests to most probable attacks
 - Further focuses analytic testing effort
 - Builds team confidence against "real" attackers



Adversary Emulation: The Ultimate Goal

Once your team has built confidence using atomic testing, Purple Teaming, and perhaps even some penetration tests or targeted Red Team assessments, it's time to crank the realism up to the maximum with "adversary emulation". This is a specific type of Red Teaming assessment where the attacking team will do their best to leverage your threat intel on threat actor groups and form their attacks to look like a *specific* threat actor. This approach is often called "adversary emulation" and is a great goal to strive for as a SOC matures its self-assessment and testing capabilities, while it is obviously more complex, tools such as MITRE's CALDERA can help automate portions of adversary emulation testing.

Purple Team Methodology: Planning

Step 1 – Attack Planning

- Select attack techniques to test—leverage ATT&CK
- Prioritize TTPs based on threat intelligence
- [Red team] Prepare tools/methods to run each test
- [Blue Team] Get familiar with all data sources
- Create multiple variations on each technique
 - Ex: Prepare to deliver malicious files of many formats, stage malicious files on good sites, bad sites, and shortened links
- Block off schedules—half day or full day, if possible

Purple Team Methodology: Planning

Perhaps you've decided you want to run your first Purple Team test, fantastic! How should you proceed? The test can be broken down into 3 general phases: planning, execution, and findings analysis.

In the first phase the Red Team (using info based on threat intel) should plan what specific tests will be run. This is not just creating a general list but involves ensuring, for each test you want to run, that the tools and tests are ready to go and can be easily deployed. For example, do you have the ability to create and send malicious test files in rapid succession? Perhaps you want to even prepare attacks and malicious files ahead of time. The goal here is to ensure attacks can be run one after another without pauses during the actual test, as this will keep the test running smoothly and wasted time to a minimum.

Remember, since this is an adversary emulation exercise, choices should be prioritized based on the expected attack techniques a real attacker would use and, this is key, *multiple* methods of performing each technique should be planned. An example: Perhaps you want to test your resistance to phishing attacks to assess your resistance to the "Spearphishing Link" and "Spearphishing Attachment" ATT&CK techniques. Don't just plan to send a single malicious email for each technique, send multiple emails of varying types. For the malicious links test, use multiple types—links to "good" sites like OneDrive/Dropbox hosting publicly accessible malicious files, links to files on suspicious looking, uncategorized sites, and links to link-shortening services that will redirect you before getting to the intended destination. How well do your defenses deal with each? For malicious files, don't just use one type of attachment, run through the gamut of common malicious email attachments: doc, xls, docx, xlsx, bat, ps1, js, hta, vbs, etc. This way you will have a comprehensive assessment of all different varieties of the same ATT&CK technique "Spearphishing Link" "Spearphishing Attachment".

Purple Team Methodology: Test Execution

Step 2- Test Execution

- (If possible) Gather everyone in the same room
- [Red Team] Try each technique / method in the list
- [Blue team] Investigate visibility and outcome of each attack technique / method
 - Bonus: Experience analysts can teach new analysts where to look
- Record results: **blocked, detected, or not detected**
 - Including nuance where required (ex: "detected on servers only")

Purple Team Methodology: Test Execution

When it's time to perform the test, ideally everyone from both teams can sit in the same room. If everyone is not co-located, depending on team size, a separate conference call for each team plus a group chat between the two teams to announce when attacks are run might be the best solution (one large conference call could make collaboration difficult). Be sure to set the ground rules that this is not an adversarial environment, but one where both teams have the same goal of validating and testing the security of the organization toward the goal of rapid improvement.

Once the test is started, the Red Team can start throwing each planned attack, in succession, at the environment. Each attack should be notated with the time of execution and any other pertinent details (we will describe a software platform for doing this in a moment). Notating attack times helps the Blue Team most efficiently locate the moment in the time where they should see evidence of the attack in the logs. If nothing is seen, it is clear the attack was missed.

While the Red Team is attacking, the Blue Team should be coordinating and looking for evidence of each atomic test throughout all security tools. The result of each attack should be recorded in the Purple Team results tracking system as either "blocked", "detected", or "not detected". Any qualifiers on these terms should be noted as well, such as if an attack was blocked against servers but only detected on desktops for example.

This section of the test is key because it provides double benefits. Not only does it test analytics and monitoring capability, but it's also a golden opportunity for training. Any new analysts that are either not familiar with the environment or are new to information security should be watching over the shoulder of an experienced analysts to understand what a "real" attack looks like in the environment, and what the source of data for evidence of each attack would be. The value of this portion of the test cannot be overstated, training with "real" attacks in your real organization environment is one of the most realistic and, therefore, best training opportunities you will come across!

Purple Team Methodology: Analyze Findings

Step 3 – Analyze Findings

- **Dedicate time** to interpret results, prioritize, and implement fixes
- Questions to answer:
 - Which attacks were not detected and why?
 - Which attacks could be blocked that were only detected?
 - Did any analytics fail to fire?
 - Can you assemble a full attack path for your organization?
 - Which techniques should be prioritized to fix?

Repeat periodically, report improvements over time



Purple Team Methodology: Analyze Findings

Once the test is done the work is far from over, this is where the most important piece begins—analyzing the outcomes, prioritizing and fixing problems, and reporting results. In this stage, the Blue Team should be analyzing the outcome of each attack that was missed and finding the root cause as to why the analytic or monitoring solution didn't meet expectations.

Some of the main questions you are looking to answer are:

- What attacks weren't seen at all?
- What was detected, but perhaps could have been blocked?
- Were there any analytics you expected to work, but failed to trigger? Can you figure out why?

Given that the Red Team tried for a technique in every stage of the kill chain, was there at least one method that succeeded in each? If so, you now have a full attack chain with specific techniques that could be used to successfully complete a full-scale breach of your organization, and you should prioritize remediation of those items given this new insight. This is the stage where findings can be rolled up, prioritized, fixed, and reported on. Do not skip the reporting stage—this is where you can show upper management the fruit of your efforts and hopefully improve outputs of multiple Purple Team tests across time. This is one outstanding way to demonstrate the value in Purple Team adversary emulation activities and objectively show that the Blue Team is improving over time ... a critical need for staying funded as a SOC.

Purple Team Tracking Tools:Vectr

Status: Completed

Red Team Details

Name: Credential Dumping with Mimikatz

Description: Dump the password hashes for local and domain user accounts. Identify Mimikatz spawned by PowerShell. Multiple indicators including download string, PowerShell launched in bypass mode, and DLLs loaded by Mimikatz.

Technique: Credential Dumping

Phase: Credential Access

Operator Guidance: Invoke Mimikatz in-memory within Cobalt Strike beacon.

Attack Start: 08/07/2018 09:47:51 status changed to InProgress

Attack Stop: 08/07/2018 09:47:52 status changed to Completed

Source IPs: Linux VM: 10.30.20.115

Blue Team Details

Outcome: Not Detected

Was the event source logged? Yes

Outcome Notes: No detection activity observed for in-memory Mimikatz execution run from a remote beacon on the compromised host. While EDR alerts exist for suspicious PowerShell usage, this execution did not use Mimikatz binaries on disk or PowerShell one-liners to download and run the PS1 payload.

Tags: #

Rules:

Detection

Prevention

EDR: Endpoint Protection

Detection Time: 08/07/2018 09:47:58 outcome changed to Not Detected

Expected Detection Layers: EDR, Endpoint Protection

SANS | MGT551 | Building and Leading Security Operations Centers | 117

Purple Team Tracking Tools: Vectr

A Purple Team exercise will generate a lot of data that has enormous value to the organization, but without a meaningful way to organize it and interpret the results, the value of the exercise can become lost. One of the best free tools for organizing Purple Team exercise results is Vectr¹ by Security Risk Associates. Vectr is perfect for documenting and interpreting the results of Purple Team tests because it not only helps record the results of individual attack techniques, but also charts them over time across multiple Purple Team tests and maps results back to the MITRE ATT&CK framework.

[1] <https://vectr.io/>



Vectr Output

Vectr breaks down each test result into Blocked, Detected, and Not Detected and can visualize results in aggregate as well as do historical trending across tests. This helps your organization go from raw test results to prioritized action and technique coverage mapping in the minimum amount of time and produces an objective record of how the Blue Team is improving over time. In addition, Vectr will look at attacks from each stage of the kill chain and ATT&CK tactics and create a useful diagram of a full attack path (assuming one is found) that could be used to successfully infiltrate your organization.

Breach and Attack Simulation (BAS)

- **Problems that remain:**
 - Pen testing / red teaming / purple teaming is a "*point in time*" assessment
 - Environments constantly change
 - Any change can break collection/detection in unidentifiable way
- **Breach and Attack simulation help solve these problems**
 - Automate and scale testing of defenses
 - Remove "*point in time*" testing issue
 - Goal is to provide a *continuous, real time* risk assessment
 - Automates tests to demonstrate controls are working
 - Helps prepare and show compliance for audits

Breach and Attack Simulation (BAS)

Each testing method has a specific problem it's best at solving, and while the authors are a huge fan of purple team testing as a training and manual testing exercise, even purple teaming has its downsides. The biggest issue with individual, manually completed testing is that it's a "point in time" assessment that only gives a snapshot of your capabilities. Since environments are in a state of constant flux, the results of your test may be immediately out of date if you work in a highly dynamic environment. Any change made can cut off a data flow, break an analytic, or blind you in some other unexpected way. Therefore we have a new problem to solve – one of testing immediacy and relevance. The solution to this problem of course, is finding a way to retest as much as possible, as frequently as possible, and this is where automated breach and attack simulation tools come in. The tests may not be as comprehensive or realistic, but they *are* fast and continuous, which means you get a constantly updated view of your security posture with minimal work. Over the next few slides, we'll take a look at how these relatively new tools in the security market work, and how they can complement the types of testing we've covered earlier.

How BAS Works

Setup:

- Identification of risks within the environment
- Integration with existing security tooling and intel

Test Operation:

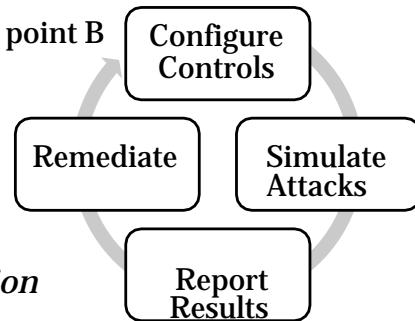
1. Real attacks are simulated through "sensors"

- Realistic malware traffic sent over network from point A to point B
- Suspicious activity, files, activity, generated on endpoints
- Cloud APIs assessed for overly permissive config

2. Confirmation of results

3. Prioritized results reported to blue team

- Repeat *as often as possible through automation*



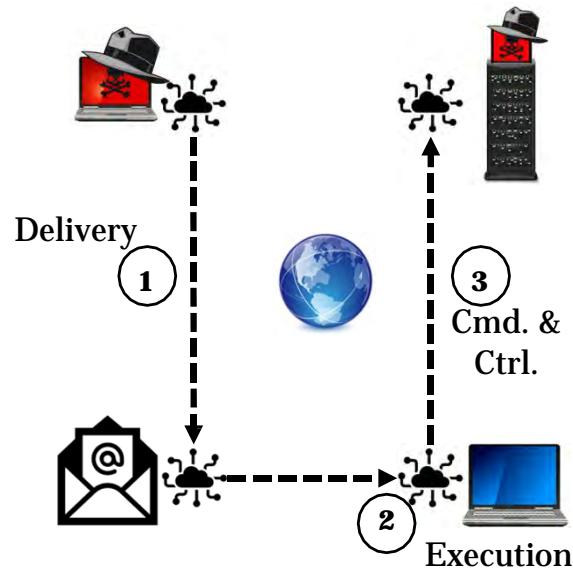
How BAS Works

To setup a BAS solution you first must go through a few steps. The two necessary pieces of information a BAS must have in order to function are integrations with your existing security tools (so it can see the results of its actions), and guidance as to which types of attacks and tests you would like to run, ideally fed from good threat intelligence. The security team will then need to set up the specific types of tests to be run, the locations (agents) those tests will be run from, and the frequency at which the tests will occur. Once the integrations are set up and the test scenarios are selected, you are ready to let your BAS solution loose!

During operation, your BAS solution will be simulating attacks throughout the kill chain stages and MITRE ATT&CK Techniques. Utilizing both endpoint and network attack vectors your BAS will be hard at work in the background, ensuring that there are no false negative detection surprises, and that any problems are quickly identified. The higher the rate of repeated testing, the faster that issues can be spotted, but there may be diminishing returns or resources constraints that keep you from literally continuously testing. Therefore, tests are commonly run once a month or even once a week, or day with a higher frequency of testing being preferable when the environment is undergoing rapid change. Completed tests should output a highly actionable report complete with a description of the problem, evidence, and remediate advice. Results are often ranked with various risk measures to help prioritize corrective actions. Remember that while detection of a problem is a feat on its own, it's useless without the capacity and intention to respond to those findings.

BAS In Action

1. Platform sends malicious email to a tool-controlled test account
2. Agent on test system runs file that was sent in email and records results and progression
3. Agent generates simulated malware traffic to attacker
4. Platform reports each independent test, and results



BAS In Action

Here's an example of a multi-step simulated test that might be run by a BAS solution on a daily, weekly or monthly basis. The attack consists of multiple steps within the kill chain – delivery of a malicious email, execution of a malicious program, and command and control from that program back out to the internet. Running this scenario in your environment would test multiple network and endpoint-based controls, give you a comprehensive view on if the attack would work.

To simulate the attack, the BAS solution would orchestrate the 3 separate pieces happening independently, but when the results are put together, it's easy to see whether the whole chain of events would work or not. First, the BAS solution would send a malicious email (perhaps of many different types) to an email address set up for testing, with all the same corporate protections available to a normal user. The solution can check the inbox to see what happened to each malicious email it sent, and whether it was blocked, flagged, or got through. The next piece would be the agent installed on a representative test system running an executable that's as close to the real thing as possible without actually endangering the network. If the virus would typically install a persistence mechanism, create a new service, and modify firewall rules, the BAS vendor should have a program that does all those things as well. The BAS platform watches output from endpoint detection tools and checks which activities are block, alerted on, or allowed, same as the email. Finally, the agent will initiate traffic outbound from the endpoint, over the organization's network, to a BAS owner test server that can receive it and simulate a real command and control channel. The results of this test – allowed, alerted, or blocked, are known as well and the totality of action that was possible is then fed into a report that goes to the blue team. You can see how a system like this can save TONS of time, as well as help facilitate continuous testing and remediate for priority items.

What to Look for In a BAS Solution

Choosing the right vendor involves finding...

- Complete, trustworthy test results
 - Data feed availability and health measurement
 - Threat intelligence backed testing
 - Full attack cycle validation with multi-step simulation
 - Realistic attacker activity – network, endpoint and more
- Relevant, flexible, and actionable testing
 - Up-to-date attack methods
 - Multi-environment support
 - Clear and actionable report output



What to Look for In a BAS Solution

The benefits of breach and attack simulation or continuous security validation (CSV) as it's sometimes called, are clear, however the problem that remains is picking the best solution for you. In the author's opinion, two of the most important factors to consider when choosing a solution are the ability to deliver *complete*, and trustworthy testing results, as well as relevant, flexible testing with actionable output.

On the complete and trustworthy factor, consider the following items:

- Can the solution give you an accurate status of the health of your data feeds? Tests may work for parts of the network where data is available, but what if something has gone offline, can it warn you?
- What type of threat intelligence is baked into the platform? Can it help you determine what the most relevant and dangerous threats are that your organization might face? Does the vendor specialize in threat intelligence?
- Can the solution not only test single controls, but multi-vector attacks all chained together with multiple test cases per step?
- How realistic are the attacks? How does the vendor deal with need to test against real viruses, but not actually put your network in danger?

Additionally, security validation solutions should have the customer relevant data consistently available for testing in flexible ways. The results of these tests should also be presented in a clear and actionable way to your security team and management.

You should also ask any potential vendors:

- How long does it take from the release of a new virus, exploit, or otherwise, until you are able to test for it with their tool? Is it a quick turnaround or might you be waiting so long that you would be leaving your network vulnerable during the window you would expect a newly discovered exploit to show up?
- Do they support all the environments and operating systems your organization uses? Windows, macOS, Linux/Unix, cloud platforms and APIs, SaaS solutions and more?
- How much detail is included in the output of a report? This is a great area to have a bakeoff against multiple vendors. Set up a desktop or server with a set of known problems, and see which solution best guides you towards what the problems are, and how to fix them.

SOC Assessment Summary

Analytics should and must be tested

- Manual tests should be used at time of creation
 - Verify all known instances of attack are caught and false positives are non-existent or minimized
- Adversary Emulation can be used for blind testing
 - Automated continuous unit testing
 - Adversary emulation
 - Purple Teaming / Red Teaming / Penetration Testing
- ATT&CK Navigator helps organize and prioritize

SOC Assessment Summary

When it comes to analytics, the last thing you want is to be caught by surprise by realizing they have been bypassed and an attack is now underway. Analytic testing ensures that everything is working as the SOC intended and that no recent infrastructure actions, or otherwise, have created side effects that could cause you to miss an important alert. Fortunately, the industry has jumped to solve this challenge by creating many unit tests as well as fully automated adversary-emulation frameworks that you can use for free. On top of these automatable tests, manual adversary emulation should be used to give different and potentially blind tests as a double-check for effectiveness. In the earlier stages of forming a SOC, Purple Team testing may be a better use of money as a collaborative testing style works as an analytics test as well as training. When a SOC believes it has a reasonable chance at resisting intrusion, Red Teaming and penetration testing can be used for a "real scenario" unannounced style test.

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

- **Effective Execution**
- Staff Retention and Burnout Mitigation
 - *Exercise 5.1 – Training and Career Development Planning*
- Metrics, Goals, and Effective Execution
- Measurement and Prioritization Issues
 - *Exercise 5.2 – Creating, Classifying, and Communicating Your Metrics*
- **Continuous Improvement**
- Strategic Planning and Communications
- Analytic Testing and Adversary Emulation
 - *Exercise 5.3 - Purple Team Assessment*
- Automation and Analyst Engagement
- Summary and Cyber42 – Day 5



This page intentionally left blank.

EXERCISE 5.3

Exercise 5.3: **Purple Team Assessment Planning, Execution, and Tracking**

OBJECTIVES

- Plan a Purple Team assessment for your SOC
- Organize your Purple Team strategy into assessments and campaigns
- Use threat intelligence to guide testing and track results
- Identify your most and least effective security tools
- Learn how to objectively demonstrate SOC improvement over time



Exercise 5.3: Purple Team Assessment Planning, Execution, and Tracking

Please go to Exercise 5.3 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

- Effective Execution
- Staff Retention and Burnout Mitigation
 - *Exercise 5.1 – Training and Career Development Planning*
- Metrics, Goals, and Effective Execution
- Measurement and Prioritization Issues
 - *Exercise 5.2 – Creating, Classifying, and Communicating Your Metrics*
- Continuous Improvement
- Strategic Planning and Communications
- Analytic Testing and Adversary Emulation
 - *Exercise 5.3- Purple Team Assessment*
- Automation and Analyst Engagement
- Summary and Cyber42 – Day 5



This page intentionally left blank.

Automation

Why We Automate

- Better quality
- Consistent processes
- Time savings
- Better governance
- Reduces burnout and keeps it fun



Automation is a process, not a project!

SANS

Building and Leading Security Operations Centers 127

Automation and Security Operations

One of the most common process improvements we can make in the SOC is automation. "Automation" is a broadly-used term in technology, and there are many different kinds of automation we might apply to improve various SOC functions. Before we talk about an automation strategy or tools, it's important to remember that automation is a process, not a project. It's a way to reduce friction in the SOC and improve quality, consistency, speed, and governance. Automating repetitive tasks and augmenting creative work your analysts are doing will also reduce burnout and make the SOC a more fun and engaging place to work. Remember that, like any process improvement, we want to understand the need and the impact of automation before we decide to spend time and resources deploying it – we want to avoid automation for the sake of doing so and simply looking for excuses to spend time developing scripts, tools, or playbooks. That said, the next few slides will discuss opportunities to improve SOC workflows via automation and approaches we might take to do so.

Types of Security Automation

- **Robotic process automation (RPA):** automating repetitive tasks, sometimes integrating with IT infrastructure using application programming interfaces (APIs), using pre-defined rules and logic
- **Integrated automation, aka *orchestration*:** bringing together independent systems in order to reduce complexity and streamline multiple automated processes
- **Conversational automation:** Natural Language Processing that supports human-like interactions
- **Cognitive automation:** automating tasks and workflows using unstructured data

Types of Security Automation

There are many different types of automation we might employ in a SOC, but these are the most common: robotic process automation (RPA), integrated automation, otherwise known as *orchestration*, and conversational automation. RPA enables individual processes to occur automatically with minimal human interaction.

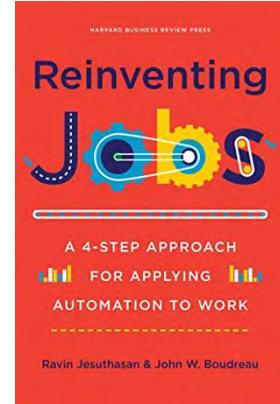
Integrated automation strings multiple automated processes together into a single set of actions. Conversational automation, which is often used in conjunction with RPA, enables more human-like interactions with our tools using chat bots and other messaging mediums. Cognitive automation is a subset of artificial intelligence that deals with analyzing and processing unstructured data for the purpose of simulating human judgement and decision-making.

Security orchestration, automation, and response (SOAR) tools focus primarily on the first two automating types: RPA and orchestration. Security teams can automate discrete tasks or string multiple tasks together in a playbook or workflow without needing to write code or API integrations (although most SOAR platforms allow for code-level customizations). Conversational and cognitive automation have emerged in more limited use cases but are gaining traction for

5 Step Approach to Automation in the SOC (I)

Modified 5-step approach to applying automation, based on Reinventing Jobs:

1. Identify the opportunity
2. Deconstruct the task
3. Consider return on investment
4. Evaluate automation type
5. Evaluate automation impact



We don't need to use this for everything, but it's helpful for major initiatives

5-Step Approach to Automation in the SOC (1)

In their book Reinventing Jobs, Ravin Jesuthasan and John Boudreau propose a repeatable process for applying automation to streamline manual work. We can build upon the concepts they describe to devise a repeatable, 5-step process to applying automation in the SOC. While we don't necessarily need a process like this to write a simple script or automate a local/individual task, having a structured process is helpful for larger efforts that require more significant resource investments.

The process is:

1. Identify the opportunity – what problem are we solving? The previous examples we reviewed all had specific problems they were designed to address.
2. Deconstruct the task – what is the nature of the task we're automating?
3. Consider the return on investment – what are we saving in terms of cost, effort, or time?
4. Evaluate automation type – what kind of automation is the most appropriate based on the answers to #2?
5. Evaluate automation impact – what measurable results do we expect to see?

We'll discuss each of these in more detail in the next few slides.

5-Step Approach to Automation in the SOC (2)

Step 1: Identify the opportunity

- What is it?
- Historical context
- Related events or alerts
- Why do analysts want it automated?
 - Common first step(s) in validation or analysis
 - Tedious or time-consuming
 - Repetitive and predictable



5-Step Approach to Automation in the SOC (2)

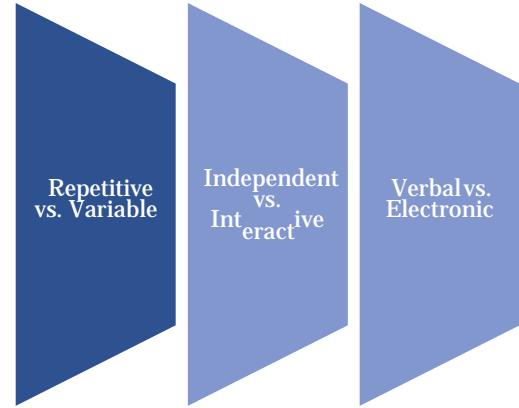
The first step is identifying the thing you want to automate. This could be a single task, a function (such as data enrichment), or a series of tasks in a workflow – let's call it "task x". Automation is a fun and exciting prospect, especially when so many tasks in the SOC can get repetitive and onerous. But automation isn't free; every bit of engineering effort costs something in terms of financial resources, staff, time, or a combination of all three. To make sure we're getting the maximum value out of our automation, we want to look at historical context – what has been the cost of doing "task x" manually? Are there specific scenarios, events, or requests where the manual workflow is necessary? How often do those occur?

Finally, we must get buy-in from the team. Dictating an "improvement" that no one asked for is a recipe for failure at worst and shelfware at best, so make sure there is sufficient demand for the improvements you want to make!

5-Step Approach to Automation in the SOC (3)

Step 2: Deconstruct the Task

- Remember decomposition!
- What are the benefits, dependencies, and difficulty levels of each subtask?
- How big do we want to go?
- Can we re-use any of the sub-tasks, integrations, or automations?



5-Step Approach to Automation in the SOC (3)

The second step is deconstructing the task, which will help us pick the right automation approach. Decomposition is a problem-solving technique we have discussed a few times already, and it comes in handy in process improvement as well!

First, determine whether the task is:

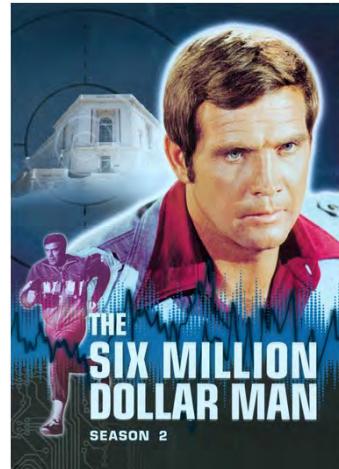
- *Repetitive or variable*: are the inputs, actions, and outputs the exact same every time, or do they change based on the scenario?
- *Independent versus interactive*: does the task or workflow include a decision point or require analyst input, or can it be automated without any human interaction?
- *Verbal or electronic*: Does the improvement require verbal input (such as, for example, in a call center or user support scenario) or strictly electronic inputs?

While the task or workflow you want to automate may at first seem like a single improvement, it may contain several prospective improvements that will require different types of automation. Breaking down the tasks and classifying them in this way will help you select the right tool(s) for the job or prioritize certain tasks in a workflow over others.

5-Step Approach to Automation in the SOC (4)

Step 3: Consider Return on Investment

- **Incremental value:** we will see strategic value added every time we improve this task; for example, faster response times
- **Constant value:** every improvement adds *some* value; for example, improving morale
- **Negative value:** diminishing returns for every improvement we make beyond a certain point; for example, reducing false positives



5-Step Approach to Automation in the SOC (4)

The next step to automating SOC tasks is to consider your return on the investment it will take to automate the task. Some automations provide **incremental value** – that is, there will be major gains in time saved, resources saved, etc. *every* time you improve the process. An example of this would be reducing response time through various automated processes. **Constant value** means that every time you improve a process through automation, things will get somewhat better. An example of this would be automating case notes and other documentation, which theoretically has a constant value in terms of improving morale with every improvement you make.

Negative value means that at some point in the improvement process you are putting in more effort than you are saving. An example of this would be tuning out false positives – as we saw in our discussion of the false positive/false negative balance, false positives can never be 100% eliminated if you want to identify all anomalous (and potentially malicious) activity. Therefore, “improving” the false positive ratio will have negative gains at the point where you start missing true threats.

5-Step Approach to Automation in the SOC (5)

Step 4: Evaluate Automation Type

- What is the most appropriate automation type for each subtask?
 - RPA
 - Orchestration
 - Cognitive automation
 - Conversational automation
 - Micro-automation or user-level automation



5-Step Approach to Automation in the SOC (5)

The fourth step in automating SOC workflows is to evaluate the type of automation best suited to solve the problem. Remember that not all SOC automation is (or should be) achieved with SOAR tools. There are many cases – desktop automation for analysts being one – that *micro-automations* like scripts or macros can handle more effectively. Other tasks, like prompting analysts for playbook steps or taking natural language inputs, might be better for conversational automation such as a chat bot. Broadening your thinking about what tools you might use to automate various tasks will help you right-size the investment you must make and give you many more options.

5-Step Approach to Automation in the SOC (6)

Step 5: Evaluate impacts

- Do you expect to substitute something a human would do or augment what a human would do?
- What measurable improvement(s) has your new automation brought?
- Are these improvements evident in your metrics?



5-Step Approach to Automation in the SOC (6)

The final step of our 5-step approach is to evaluate the impacts of the automation we have applied. You want to be sure that the work you have invested in automating a task or a workflow is having the intended impact – does it substitute or augment something one of your analysts would otherwise do manually? What does the return look like in terms of time, effort, or a better result? Is the newly-automated task or workflow reflected in your operational metrics in the form of shorter time to identify, time to respond, or other measure? Don't forget to also include the ongoing management effort in your evaluation. Does your new automation require time to update, fix bugs, or service new feature requests? These are important questions to answer to ensure that your team isn't simply automating for automation's sake.

Next, we'll look at a few specific examples of automation in the SOC and tools we might use to build those automations.

Example: Automating SOC Workflows with SOAR

Typical categories for SOAR:

- **Enumeration and Enrichment (IP, Hostname, Hash)**
 - Using internal tool APIs
 - On external data
 - Resolved by SOAR framework
- **Incident Response**
 - Blocking actions
 - Sample gathering
 - Cleanup
- **Alert and Case management**



Example: Automating SOC Workflows with SOAR

For automation and orchestration, many of the use cases revolve around a small set of categories. One of those categories is enumeration and enrichment. In these use cases, the SOAR tool is using data already obtained to perform a lookup and pull in additional values. This could be an IP address associated with a domain, a virus check based on a hash, resolving a hostname via DNS inside the network or otherwise. Where these enrichment requests go can be broken down into lookups to internal tools and APIs, pulling information from external sites and data, and information the SOAR tool can make the request directly to resolve (such as DNS or NETBIOS lookups).

Another common category is response actions. The SOAR tool can be used to speed up ticket response and containment times by automatically grabbing malicious file samples or taking action to block hashes, domains, or IPs by interfacing with the organization's security tools directly.

The final category is alert and case management. Items in this category tend to be in the form of "moving text around and auto-submitting fields, so analysts don't have to." SOAR platforms may be used to correctly fill out observables, hostnames, notes, or other details in a ticketing system, or use that data to fill out tickets to other groups within the organization on different ticketing platforms. These actions usually revolve around moving data around instead of making analysts copy and paste it manually. In addition, some SOAR platforms contain their own dedicated alerting and case management systems! Whether or not it is acting in that capacity in your environment, the key item it should be solving is automating actions where you might ask yourself, "why do I have to do this manually?"

Automating Alert Validation with SOAR (cont.)

- **Problem:** pulling contextual data can be time consuming, but without it we might ignore low-priority events of interest
- **Approach:** write a custom playbook to pull a variety of contextual data based on alert fields

The screenshot shows a SOAR tool's alert details page for a "Potential Malware Download - Blocked" alert. The alert is marked as "Open" and was created by "Splunk". The "Other Related Data" tab is selected. The "Alert Information" section includes fields like Severity (Low), Alert ID (2019-094435628), Source IP (10.10.4.251), Destination IP (144.91.69.195), Domain (144.91.69.195), Alert Date (2019-09-25 23:28:56), Mitigated by Controls? (Yes), Alert Data Source (Bluecoat | Suspicious), and Device Type (Laptop | Dell | Win 10 x64 | 1809). The "Enriched Information" section provides context such as Resolved IPs and Domains, Related Alerts (4), Related Incidents (1), Related Events (5), and Device Owner (VIP | John Davis). A red box highlights the "Related Alerts" and "Related Events" sections.

Automating SOC Workflows with SOAR (cont.)

This screenshot was taken from Cybersponse, a commercial SOAR tool. In this example of automation with SOAR, we have an alert that might not be high priority - "Potential Malware Downloaded – Blocked" - but may be something we want to review depending on the context. In this case, we have developed a custom playbook to pull event context such as device type, user, vulnerability information, as well as related alerts and incidents. By reviewing this contextual data, we might be able to identify risky behaviors or other threats not mitigated in what is otherwise a relatively innocuous alert. Pulling all of this data manually would probably require us to log into multiple different tools and run various queries, so we have likely saved a lot of time in addition to making a more actionable alert!

Example: Automating Alert Enrichment with ChatOps

- **Problem:** validating alerts can be time-intensive when it requires contacting users



- **Approach:** use a chat bot to prompt users in order to validate specific alerts, authenticate them, then append the response to the alert presented to the analyst

Automating Alert Enrichment with ChatOps

Including users in an automation loop may sound daunting, but in some use cases it can be a tremendous time saver. Most security analysts have felt the frustration of having to chase down users and confirm that they did, in fact, run a sensitive command or download a sensitive file that triggered a security alert. The ability to gather and present this information as context to the alert would be huge; the question is, how might we automate that?

Defined Networking CEO (and former head of security operations at Slack) Ryan Huber wrote an excellent blog post describing how his team at Slack used conversational automation to query users following certain alerts – in this example, the use of administrative commands. The bot authenticates the user and captures their response and then returns that in a way that can be presented to the security analyst. This enables users to interact with Slack's security infrastructure in a conversational way and provide the security team with valuable information they would normally have to gather manually.

You can read Ryan's full post at <https://slack.engineering/distributed-security-alerting/>.

Example: Automating Report Processing with Mitre's TRAM

- **Problem:** reviewing threat reports for actionable intel can be time consuming and highly manual
- **Approach:** train a tool to recognize techniques from MITRE ATT&CK in unstructured data sets



From "TRAM: An Easier Way to Map ATT&CK" by Jackie Lasky and Sarah Yoder

Automated Report Processing with MITRE's TRAM

Mitre's Threat Report ATT&CK Mapper (TRAM) tool is a great example of cognitive automation. TRAM provides a streamlined approach for analyzing unstructured data – in this case, threat reports – and extracting ATT&CK techniques.

Here's how it works:

1. TRAM trains itself using Procedure Examples from the ATT&CK website
2. Terms are normalized and tokenized into a "clean" data set that is easier for a computer to interpret
3. TRAM uses a variety of supervised learning methods, notably Logistic Regression via Python's Sci-kit library, to build a model for predictive analysis so as to recognize techniques within a sentence
4. TRAM applies the models to new data sets, i.e., whatever reports you give it to analyze

When TRAM identifies a technique in a report, it highlights the text and shows the predicted technique in a dialog box for an analyst to either accept or reject. This feedback further refines the model and enables TRAM to identify techniques more reliably. While it is still early in its development, the ability to pull important data from unstructured text is a big win for teams who have to extract actionable data from threat reports on a recurring basis.

You can download TRAM here: <https://github.com/mitre-attack/tram>. You can view the presentation from which the above slide was taken here:

<https://www.slideshare.net/attackcon2018/mitre-attckcon-20-attck-updates-tram-jackie-lasky-and-sarah-yoder-mitre-193677940>

Automation Risks

- **Lack of adoption:** does your team *want* task x to be "improved"?
- **Lack of staff re-training:** does your new tool or automation require training to maximize its benefit to the team?
- **Ongoing support:** what kind of ongoing engineering support does your new tool or automation require? Would staff turnover put your new automation at risk?
- **Trying to do too much:** don't "boil the ocean"!

Automation Risks

Of course, any time we build a new capability or make improvements, there is a cost and subsequently some risk. Let's say that you, as a newly-minted technical leader in the SOC, come across a new open-source automation project in which you see a lot of potential. You're so excited to see it in action that you commission a new task to deploy and configure it in your SOC. In fact, you're so excited that maybe you skip some of the analysis and evaluation you would normally do with any project that requires engineering resources. A common pitfall is lack of adoption; maybe your team isn't as excited about this new project or doesn't see the value that you do. Another risk is that the new tool or automation isn't fully incorporated into your team's workflow and training, making it difficult for analysts to get full value out of the project. Ongoing support is another potential issue – what kind of maintenance and upkeep does your shiny new project require in order to continue providing value. Finally, doing too much is something we have all likely been guilty of at some point (or will if you continue in your career as a technical leader). Try to focus on solving discrete problems well instead of automating entire multi-step workflows or complex tasks. Little efficiencies can add up in a big way.

Part of the fun of a technical leadership role is to look at new and emerging technology, and not all improvements will pan out. That's ok if you are willing to learn from those experiences and make the occasional tough call to pull the plug on something that hasn't brought the efficiencies you'd hoped. The key is to apply the same criteria, measurement, process rigor to automation that you would any other SOC engineering effort.

Process Improvement

Automation is a major example of improvements we might make in our SOC workflows. But how do we know what needs improving?

Consider each SOC function a "project":

- Each project needs to have a clear process of inputs and outputs
- Evaluate the project without pre-determined solutions
- Focus on reducing variation to make it easier for junior/inexperienced analysts
- Project needs to be approached with awareness of inputs to the process and how they might vary, ways we might control and/or eliminate outcomes we don't want

Process Improvement

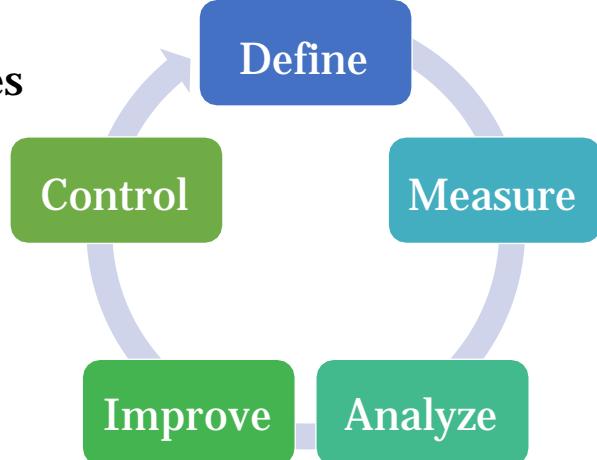
Focusing too much on process improvement can lead to "analysis paralysis," but trying to implement improvements like automation without analyzing the problems can lead to wasted cycles and ineffective solutions. The fundamental question here is, "what really needs improving that may benefit from automation, additional technology, or new processes"? To answer this question in a logical, repeatable way, we can look to Six Sigma - a quality improvement method designed to increase performance and decrease process variation.

Before we dive into that, let's conceptualize the "thing" we want to improve. In order to make our operation a bit more Six Sigma-friendly, consider each SOC function – alert triage, investigation, incident response, intelligence analysis, etc. - a "project" with clear inputs and outputs. The project must be something we can analyze without a pre-determined solution ("I want to implement SOAR, now let's figure out where we can plug in some playbooks"). A key element of process improvement under Six Sigma is reducing variation, so let's focus on reducing variation in whatever process we're trying to improve to make things easier on the team. Think about making a process template or playbook for that process as simple and with as few one-offs as possible. Finally, whomever is doing this process analysis needs to be experienced enough to understand the inputs and outputs and how they might vary; for example, "we need to improve lateral movement detection, and here are all of the ways lateral movement might look different in our environment".

Now that we have a sense of the kinds of things we might focus on improving in the SOC, let's take a look at a simple process that can get us there.

DMAIC Process

- Component of Six Sigma
- Uses data to improve processes
- Measures existing processes, identifies target outcomes, root causes preventing those outcomes, and solutions
- We can implement a "DMAIC-lite" approach



DMAIC Process

The DMAIC process is a component of Six Sigma used to improve processes. DMAIC can get pretty complex, with a variety of different processes, tools, and deliverables used at each stage. For our purposes, we're going to apply a "DMAIC-lite" approach to getting the results we want.

At a high level, it works like this:

1. Define the problem to be solved or result we want to achieve
2. Identify metrics we can use to measure the current process and its outcomes
3. Analyze the process to identify root cause(s) preventing us from achieving the ideal result
4. Create or implement a solution for that root cause
5. Ensure sustainability through ongoing measurement and management of the new (improved) result.

This all sounds pretty high level, so let's use an example to illustrate the process.

Example: "Response time in the SOC is too slow"

Define	We're taking too long to respond when incidents occur
Measure	Mean time to respond: time from when an alert fires to when an analyst initiates the first response action; goal is <30 minutes
Analyze	Currently, MTTR is >1 hour for high priority incidents due to heavy alert load, high false positive rate. Gathering contextual data to differentiate between false positives and true positives is manual and time-intensive.
Improve	Enrich key alerts in the SIEM with additional context, promote high fidelity detections to higher priority level to make them stand out.
Control	Newly improved alerts go into high-priority queue where they are reviewed and escalated more quickly; MTTR, now at 18 minutes, is monitored weekly.

Example

Let's take a relatively common complaint about response times in the SOC: "response time in the SOC is too slow". First, recall that we need to boil this statement down to a discrete process with clear inputs and outputs. In this case, we know that there is a monitoring and analysis process that takes in alerts and other signals and produces incident discoveries/responses as an output. We can further distill the problem statement to "we're taking too long to respond when incidents occur," based on our average response time measurements. The target value for that measurement is less than 30 minutes for high priority incidents, but it is currently over an hour due to heavy alert load and false positive rate. In this hypothetical scenario, we determine that taking potentially high-fidelity detections and enriching them in our SIEM to support faster validation will address most of the underlying problem. Making that improvement has the result of reducing our mean time to respond to 18 minutes – a metric we'll keep tracking to manage the process moving forward.

A quick side note about variation, which we mentioned a moment ago but didn't discuss in this scenario. In a real SOC, this problem very likely would not be this simple to address. Consider your own environment – do you have just one set of inputs to your analysis and incident response process? Probably not. Part of this improvement approach is reducing that variation as much as possible so that we can control the outcomes and ensure the results we want. Following the response time example, if you have ten different inputs to your alert escalation process, try to identify ways you can feed those into a common input – say, a ticket queue – to make the analysis and escalation process as consistent as possible. This will reduce outliers that can skew your data and disrupt any improvements you're trying to make.

Automation Summary

- Take a logical, repeatable approach to process improvement
 - what problem are we trying to solve and what impact will that have?
- Automation is a process – not a project or destination you "get to" by deploying a single tool or platform
- Repeatable, agile process helps us avoid wasted time
- Consider priorities - not all improvements have equal value
- Can you quantify your return on time/cost investment?

Automation Summary

Many practitioners who have spent time in security operations are rightfully excited about the promise of automation. Analysis can be repetitive, sometimes laborious work, and taking those repetitive tasks off our analysts' plates so they can focus on more creative pursuits is an attractive proposition. But automation is not without its risks, and in fact can be a huge resource drain if not managed well. By taking a logical, repeatable approach to improving SOC tasks and workflows through automation we can avoid wasted cycles and unclear value propositions. Remember that not all improvements have equal value and know when to shift your strategy, toolset, or the problem you want to solve based on your observations. Finally, know that automation (like security) is a process – not a destination. Simply throwing an automation platform into your environment and looking for ways to utilize it will probably not give you the results you want. Identify the problem, select the best tool for the job, and measure, measure, measure!

Engineering for Engagement

- Security tools don't always prioritize engagement and usability
- Creates a **problem sensitivity** challenge
- Even empowered analysts sometimes have trouble with sustained attention and focus in a dynamic and unpredictable environment like the SOC
- Approaches that can help:
 - Gamification
 - Habit-forming technology
 - Deep learning features

Engineering for Engagement

Earlier in this book we talked about the SOC Human Capital model, and how as managers we must strive to maintain a virtuous cycle in which our teams are empowered to be creative and make improvements to stay challenged and grow professionally. This is an important part of our management approach; however, keeping analysts happy and engaged doesn't stop with good process and the right support. We must "engineer engagement" in our SOC toolset, which can be a challenge when most of the tools don't prioritize user experience and engagement.

In the George Mason study on social maturity in CSIRT teams, researchers found that 96% of respondents reported that Problem Sensitivity – "the ability to tell when something is wrong or likely to go wrong" - was important for identifying and mitigating a cybersecurity incident. This is a common challenge, or question, in security operations; we all want to believe that we can quickly identify and escalate incidents whenever they occur, right? Unfortunately, that is easier said than done, and organizations spend millions of dollars every year on tools and technology that promises to make that possible.

But let's look at the underlying mental mechanics of problem sensitivity in the SOC. According to the GMU study, problem sensitivity requires sustained attention and maintained focus; that is, analysts must be able to concentrate without distraction while performing their various duties over time.

In this section, we're going to talk about modifications we can make in our toolset to keep the team motivated and engaged.

Gamification

- Applying game-style mechanics such as point scoring and competition to encourage engagement
- Allows analysts to be creative, earn status, stay motivated, turn challenges into achievements
- Requires:
 - Understanding of gaming mechanics (!)
 - **Behaviors we want to encourage**
 - Strategy
 - Budget/time
 - Buy in from team and management

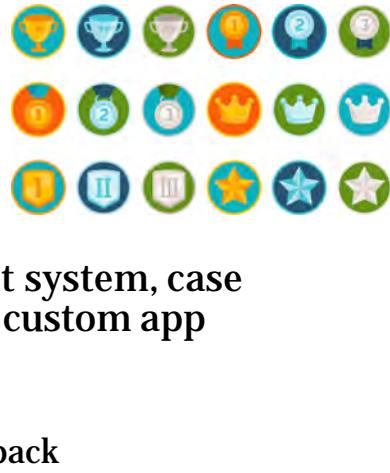


Gamification and the SOC

Gamification can be a powerful tool for keeping your SOC team motivated and engaged. Applying game-style mechanics to SOC workflows or tasks can allow your analysts to be creative and engage in a bit of light competition with their teammates. Of course, this requires a bit of understanding of game mechanics like point scoring, achievements, challenges, and the like. Also, as with any “improvement,” it *is* possible to apply gamification poorly. Not everyone likes competition or an additional element of challenge on top of already demanding work, and you run the risk of the team becoming more focused on the game than the work product. Start with the behaviors you want to encourage, such as, better analysis, better triage, a specific metrics target, or other measurable goals. Devise a strategy for how you want to incentivize the team to achieve that goal on an individual basis, and a fair scoring or achievement system for how they will be evaluated in doing so. Not all games should have monetary prizes – in fact, financial awards are usually better left as part of your incentive compensation plan or review cycle – but there should be *some* benefit for success, even if that is some time budgeted for a free lunch, virtual badge creation, or other recognition. Finally, make sure you have buy-in from the team and your management to gamify elements of SOC workflow. We’ll look at an approach to gamification in more detail on the next slide.

Gamification (contd.)

- Focus on desired behaviors
- Incentives should be ephemeral
 - Points, badges, coins, levels, etc.
- Feedback should be given around behaviors and results, not the "game"
- Can be implemented in a learning management system, case management system, issue tracking system, or custom app
- **Careful!** Gamification should *not*:
 - Involve monetary compensation
 - Serve as a basis for promotion or performance feedback
 - Add undue negative stress



Gamification and the SOC

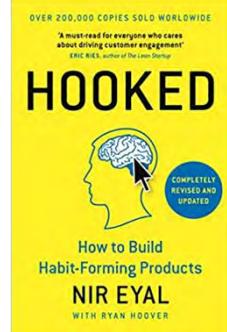
Again, the best starting point for gamifying SOC tasks or workflows is the behaviors you want to incentivize. As with SOC metrics, sticking strictly to temporal measures like time to close a case or volumetric measures like number of alerts closed may have undesirable outcomes if your team starts cutting corners to pump up their numbers. Focus on measurable results that you want to see coming out of the team. Avoid financial incentives in lieu of virtual ones like points, badges, coins, or other non-tangible benefits; tying the "game" to employee compensation is a good way to build divisions within the team and trigger negative outcomes. Give your team feedback around the behaviors you have tried to incentivize instead of the results of the game itself – remember that the point is getting the right results, not "winning the game".

There are many options for implementing a game system without much additional investment. Use your case management system or other existing tools if possible. Many learning management systems have gamification features as well if you have access to one. However, you decide to roll it out, remember our discussions of burnout and other negative impacts on SOC work. We do not want to add stress to promote competition and higher performance. Games should be fun!

Getting Analysts Hooked

"Hooked" model:

- **Trigger:** e-mail, chat message, icon, alert
- **Action:** action performed by the user for a reward
- **Variable reward:** social validation, material resource, personal gratification
- **Investment:** something the user puts back in to repeat the loop; acknowledgement, case notes, alert tuning, new detection, resolution



Getting Analysts Hooked

If you've spent a significant amount of time in this field, you may have accepted the fact that most security software (and enterprise software, for that matter) has not been optimized for effecting behavioral changes in an analyst. In many cases, the most you can hope for is that high-impact alerts will be color coded or otherwise highlight for quick prioritization. But not all the behaviors that we want in our analysts can be driven by our management approach, nor do we necessarily want behaviors to be driven by the most dominant personalities on the team or what may be easiest in any given situation.

In his book *Hooked*, Nir Eyal talks about manipulating human psychology to design habit forming products. While we may not be building apps in our SOC, we *do* want to encourage engagement and incentivize certain behaviors in our analysts. Taking a note from Eyal's writing, we can incorporate these habit-forming social features into our SOC tools and workflows wherever possible:

1. **Trigger:** visual or audible alert on possible false positive
2. **Action:** something we want our analysts to do that results in a reward; *for example, tuning an analytic to improve fidelity*
3. **Variable reward:** some form of personal gratification; *for example, a "tuning ninja" badge and/or visual change to tuning KPI*
4. **Investment:** something the analyst builds back into the SOC toolset; in this case, the analyst looks for other opportunities to tune alerts

The goal in this case is to engineer our SOC toolset around a system of rewards and reinforcement for effective behaviors like discovering incidents, improving visibility, and refining our analytics. Paying attention to how your tools incentivize the behaviors you want can help you scale your management approach and reduce reliance on key individuals to drive daily tasks. Let's talk a bit more about some of the underlying concepts here and how we might implement them in a SOC toolset.

Optimizing for Engagement

Social learning theory	Simplicity	Escalation of commitment
<ul style="list-style-type: none">• Seeing others rewarded will cause people to change their actions• Especially true for more senior role models	<ul style="list-style-type: none">• Minimize hesitation, confusion, abandonment• Bring attention to core functions tied to the outcomes we want	<ul style="list-style-type: none">• People value more highly that which they have invested in• Ensure analysts have ways of investing in toolsets

Optimizing for Engagement

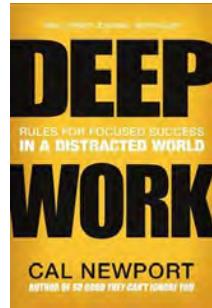
In order to optimize our toolset for the kind of engagement we want (sustained attention with the right results), we're relying on a few different social theories. The first is a system of tribal rewards referred to by psychologist Albert Bandura as "**social learning theory**". In his research, Bandura determined that people who see someone being rewarded for a certain behavior are more likely to alter their own actions accordingly. He also observed that this technique works well when people are observing the behaviors of someone slightly more experienced than they are who are likely role models. This dynamic is what drives websites like Stack Overflow, where experts share their knowledge by responding to various posts and questions from frustrated engineers. Why do these experts take time out of their day to answer these posts? Because the best responses are voted upward, resulting in points, badges, and other status increases for the respondents. Not only do the more experienced engineers get the satisfaction of knowing their helping their peers, but they can also now point to a quantifying measure of the value they bring to their field. Now think about this in the context of a SOC team, where you have some analysts who are more skilled and experienced than others. How are you highlighting opportunities for more senior analysts to share their knowledge with others and incentivizing them to do so?

The second social theory we can reference in optimizing our SOC toolset for engagement is **simplicity**. The Google home page is a great example of stark, simple design highlighting its core service offering of search. Research shows that too many choices or irrelevant options can cause hesitation, confusion, or abandonment. Now think about that SIEM console, filled to the brim with various log messages, channels, dashboard panes, and other content – does our hesitation, confusion, and abandonment problem sound familiar? Look for ways to reduce and simplify visual inputs to your analytic process, especially when it comes to alert monitoring and triage. This will increase focus and reduce negative habits like cherry picking or abandonment for want of simpler, more familiar (but potentially much less important) SOC activities.

A third social theory that we can use to optimize our tools and processes is called **escalation of commitment**. This theory essentially says that the value we perceive in a thing is directly proportional to the effort we have put into it. This is the same thinking that drives people to play video games at the expense of their own health or other obligations, and craftspeople to place a higher value on items they have built by hand versus something that can be store-bought. We can apply this theory to our SOC tools and processes by requiring our teams to build their expertise back into the tools and make as many customizations as needed to ensure alignment to SOC workflows. More effort expended on making a tool work more effectively will not only leave us with a better widget but will also have the effect of increasing analyst pride in said widget and make wide adoption more likely. Note that this isn't the same as forcing analysts to labor on a technology that is a poor fit or ineffective.

Deep Work

- Deep work: “professional activities performed in a state of distraction-free concentration that push cognitive capabilities to their limit”
- Until now, most innovations in online collaboration have focused on reduction friction
- Research done by productivity firm RescueTime indicated Slack users check comms once **every five minutes** on average!
- Balance low friction and high engagement with allowances for deep work



Deep Work

Many of our teams use a variety of different tools to communicate and collaborate, especially in this age of distributed and remote work. Looking back over the last few decades of knowledge work, we can track the evolution of communications mechanisms like e-mail and online chat. Corporate communications via email have increased exponentially over the last few decades, to the point where it has become a running joke to ask, “why this couldn’t have been an e-mail”. More recently, this capability has been replaced (or augmented) by real-time communications via Slack, Teams, and other chat platforms. Although these applications have addressed many of the things that users don’t like about e-mail, they have introduced their own set of problems around interrupting deep work and increasing interactions during the workday. Research done by productivity vendor RescueTime estimated that employees who use Slack check those channels about once every five minutes on average(!). Now consider this level of interruption in a dynamic environment like the SOC, where we are already faced with a variety of different problems evidenced by incomplete information. Trying to maintain focus while context-switching across a wide variety of different systems, tools, problems, and interruptions is an immense challenge.

Neuroscientists and psychologists teach us that our attention is fundamentally single-tasked and switching it from one target to another is detrimental to productivity. We’re simply not wired to monitor an ongoing stream of unpredictable communication while we’re trying to also finish actual work. E-mail first introduced this problem at scale in the business world, but new collaboration tools have arguably made it worse.

Deep Work (contd.)

- Reduce or eliminate “shallow work”
 - Non-cognitively demanding tasks often performed while distracted
 - Ex: Conference calls, emails, clearing known false positives, routing tickets, PowerPoint slides
- Consider daily SOC tasks – which ones could someone with no experience do with minimal training?
- Tools and processes must promote deep work by leaving room for focus, minimizing shallow work

Deep Work (contd.)

Obviously, we aren’t going to eliminate corporate chat and other productivity applications in our corporate networks, so what recourse do we as SOC leaders have to promote deep work within our teams? The first step is to try to reduce or eliminate what Deep Work calls “shallow work”: non-demanding tasks often performed at the expense of more meaningful work. Conference calls, meetings, e-mails, and repetitive tasks all fall into this category. In a recent consulting engagement, this course author concluded that the SOC team he was working with spent about 70% of their time on average on shallow work! This leaves very little time for the kind of focused attention that analysis and investigation often requires.

Second, consider daily tasks performed by your team – which could someone with no experience do with minimal training? Ask yourself if these tasks, allow considered “shallow work,” might be automated or better performed by a better team. If necessary, make your case by computing the time it takes your highly-compensated analysts to perform these tasks and assign them a dollar value. Does your management want to pay thousands of dollars a month or hundreds of dollars a month for the same outcome?

Finally, your SOC tools and processes must promote deep work by leaving room for focus and minimizing shallow work. We don’t always focus on usability in security tools, but small cycles wasted clicking through multiple screens, copying and pasting text, and other time-consuming tasks adds up and detracts from the real work of an analyst.

Engineering for Engagement Summary

- SOC tools must support our processes and our goals, not dictate them
- Enterprise software is often optimized for perceived value, not engagement and demonstrated value (solved problems, identified incidents, better outcomes)
- Gamification and other rewards/incentives reinforce training
- Optimize for deep work, reduce shallow work

Engineering for Engagement Summary

In this section, we discussed human psychological theory as it relates to habit forming technologies and promoting deep work. Remember that we must bend our SOC technology to our will and use it to support our processes, not dictate them to us. The same is true when it comes to how we use these tools to identify and respond to incidents. Customizing technology to keep our analysts engaged and demonstrating the behaviors we want through gamification and other systems, or rewards and incentives is a great way to reinforce training and ensure SOC work is aligned to our performance metrics. Stepping back from our toolset and making sure our interrupt-driven workflows aren't creating shallow work is another way to generate positive outcomes in the form of more effective problem solving and demonstrated value – incidents identified, better signal-to-noise ratio, more effective responses.

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

- Effective Execution
- Staff Retention and Burnout Mitigation
 - *Exercise 5.1 – Training and Career Development Planning*
- Metrics, Goals, and Effective Execution
- Measurement and Prioritization Issues
 - *Exercise 5.2 – Creating, Classifying, and Communicating Your Metrics*
- Continuous Improvement
- Strategic Planning and Communications
- Analytic Testing and Adversary Emulation
 - *Exercise 5.3- Purple Team Assessment*
- Automation and Analyst Engagement
- **Conclusion**
- Summary and Cyber42 – Day 5

This page intentionally left blank.

Additional References

Where to go for additional information:

1. **Crafting the Infosec Playbook** - by Jeff Bollinger, Brandon Enright, Matthew Valites
2. **Designing and Building Security Operations Centers** - David Nathans
3. **MITRE Top 10 Strategies of a World-Class Cybersecurity Operations Center** – Carson Zimmerman
4. **Blue Team Handbook Vol 1. and Vol 2.** – Don Murdoch
5. **Psychology of Intelligence Analysis** - Richards J. Heuer
6. **SANS SEC450: Blue Team Fundamentals: Security Operations and Analysis / GIAC GSOC** (*for analyst / operations training*)



Additional References

Where to go from here? As discussed throughout this book, there are numerous resources that have been published filled with best practice in multiple fields. Here are some of the key resources that have been a personal favorite throughout the years. Each of these books explores topics from this class in additional depth and are recommended reading.

If you're looking for training for SOC analysts, architects, or engineers that aligns with the mindset and approach in this course, SANS SEC450 "Blue Team Fundamentals – Security Operations and Analysis" is the partner to this course, written for those who are working in a non-management capacity. It was designed to be a course that could stand as the "standard training" for all SOC employees to create a baseline knowledge and common understanding for everyone on your team. <https://www.sans.org/cyber-security-courses/blue-team-fundamentals-security-operations-analysis/>

Course Summary

- Keeping your team engaged and empowered
 - Supports retention
 - Reduces burnout
- Knowing when, what, and how to measure core SOC functions
- Challenges in measurement and prioritization
 - Assigning numbers to qualitative measures
 - Obscuring measurable improvement by using relative values and formulas
- Strategic planning, measurement, and telling your SOC “story”
- Analyst training and career development
- Self-assessment and adversary emulation
- Improve ops through automation and engineering for engagement!

Day 5 Summary

In this fifth and final book of MGT551, we focused on our most important asset: the team itself. Keeping your team engaged and empowered is one of your most important jobs as a manager, and we covered some ways to keep your team engaged and position them for future success through training and career development. We also talked about SOC measurement: from a high-level maturity perspective using frameworks like SOC-CMM and from an operational perspective using KPIs and OKRs. Showing results is one of the most challenging parts of a job where the benefits are often transparent to end users, so identifying the right measures, the right sampling, and the right presentation is key. To this end, we also walked through self-assessment using techniques like purple teaming to ensure your technology and processes are scientifically effective. We highlighted some pitfalls in measuring security operations to which many teams fall victim and how to avoid those pitfalls, and we looked at some techniques you can use to tell the SOC narrative in a way that is visually compelling. Finally, we talked about the role of automation in the SOC, which covers far more than just orchestration and automation.

Many of the topics covered in Book 5 fall under the category of “often overlooked or done poorly because they’re hard,” but hopefully our approach to these subjects combined with the hands-on exercises has given you everything you need to get started on the right path.

As we wrap up the final day, it is our hope that you’ve taken away new knowledge on how to best plan, operate, and improve your security operations center – not an easy or simple undertaking. Our mission in creating this course was to fast forward the learning process for those who are just starting out, and help pass on best practices, tips, and tools for those who are already in operation; hopefully, we have achieved that goal.

While information security is a relatively large industry, those who manage security operations centers are a much smaller and hard to wrangle group, so please stay in touch and feel free to contact the course authors with any questions or comments related to the material. Our contact information including Twitter handles (@SecHubb and @markaorlando), LinkedIn profiles, and emails are all included in the course virtual machine’s wiki “Instructor Info” page. We would love to hear from you with any best practice you’ve found or new tools and ideas that are working for you so that we can include them in future updates to the course. Thanks for attending MGT551! – John Hubbard and Mark Orlando



Cyber42 Simulation

Day 5

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.

Lab Workbook



SANS

THE MOST TRUSTED SOURCE FOR INFORMATION SECURITY TRAINING, CERTIFICATION, AND RESEARCH | sans.org

PLEASE READ THE TERMS AND CONDITIONS OF THIS COURSEWARE LICENSE AGREEMENT ("CLA") CAREFULLY BEFORE USING ANY OF THE COURSEWARE ASSOCIATED WITH THE SANS COURSE. THIS IS A LEGAL AND ENFORCEABLE CONTRACT BETWEEN YOU (THE "USER") AND SANS INSTITUTE FOR THE COURSEWARE. YOU AGREE THAT THIS AGREEMENT IS ENFORCEABLE LIKE ANY WRITTEN NEGOTIATED AGREEMENT SIGNED BY YOU.

With the CLA, SANS Institute hereby grants User a personal, non-exclusive license to use the Courseware subject to the terms of this agreement. Courseware includes all printed materials, including course books and lab workbooks, as well as any digital or other media, virtual machines, and/or data sets distributed by SANS Institute to User for use in the SANS class associated with the Courseware. User agrees that the CLA is the complete and exclusive statement of agreement between SANS Institute and you and that this CLA supersedes any oral or written proposal, agreement or other communication relating to the subject matter of this CLA.

BY ACCEPTING THIS COURSEWARE, YOU AGREE TO BE BOUND BY THE TERMS OF THIS CLA. BY ACCEPTING THIS SOFTWARE, YOU AGREE THAT ANY BREACH OF THE TERMS OF THIS CLA MAY CAUSE IRREPARABLE HARM AND SIGNIFICANT INJURY TO SANS INSTITUTE, AND THAT SANS INSTITUTE MAY ENFORCE THESE PROVISIONS BY INJUNCTION (WITHOUT THE NECESSITY OF POSTING BOND) SPECIFIC PERFORMANCE, OR OTHER EQUITABLE RELIEF.

If you do not agree, you may return the Courseware to SANS Institute for a full refund, if applicable.

User may not copy, reproduce, re-publish, distribute, display, modify or create derivative works based upon all or any portion of the Courseware, in any medium whether printed, electronic or otherwise, for any purpose, without the express prior written consent of SANS Institute. Additionally, User may not sell, rent, lease, trade, or otherwise transfer the Courseware in any way, shape, or form without the express written consent of SANS Institute.

If any provision of this CLA is declared unenforceable in any jurisdiction, then such provision shall be deemed to be severable from this CLA and shall not affect the remainder thereof. An amendment or addendum to this CLA may accompany this Courseware.

SANS acknowledges that any and all software and/or tools, graphics, images, tables, charts or graphs presented in this Courseware are the sole property of their respective trademark/registered/copyright owners, including:

AirDrop, AirPort, AirPort Time Capsule, Apple, Apple Remote Desktop, Apple TV, App Nap, Back to My Mac, Boot Camp, Cocoa, FaceTime, FileVault, Finder, FireWire, FireWire logo, iCal, iChat, iLife, iMac, iMessage, iPad, iPad Air, iPad Mini, iPhone, iPhoto, iPod, iPod classic, iPod shuffle, iPod nano, iPod touch, iTunes, iTunes logo, iWork, Keychain, Keynote, Mac, Mac Logo, MacBook, MacBook Air, MacBook Pro, Macintosh, Mac OS, Mac Pro, Numbers, OS X, Pages, Passbook, Retina, Safari, Siri, Spaces, Spotlight, There's an app for that, Time Capsule, Time Machine, Touch ID, Xcode, Xserve, App Store, and iCloud are registered trademarks of Apple Inc.

PMP and PMBOK are registered marks of PMI.

SOF-ELK® is a registered trademark of Lewes Technology Consulting, LLC. Used with permission.

SIFT® is a registered trademark of Harbingers, LLC. Used with permission.

Governing Law: This Agreement shall be governed by the laws of the State of Maryland, USA.

Welcome to the MGT551 Electronic Workbook

E-Workbook Overview

This electronic workbook contains all lab materials for SANS MGT551: Building and Leading Security Operations Centers. Each lab is designed to address a hands-on application of concepts covered in the corresponding courseware and help students achieve the learning objectives the course and lab authors have established.

Some of the key features of this electronic workbook include the following:

- Convenient copy-to-clipboard buttons at the right side of code blocks
- Inline drop-down solutions, command lines, and results for easy validation and reference
- Integrated keyword searching across the entire site at the top of each page
- Full-workbook navigation is displayed on the left and per-page navigation is on the right of each page
- Many images can be clicked to enlarge when necessary

Updating the E-Workbook

Tip

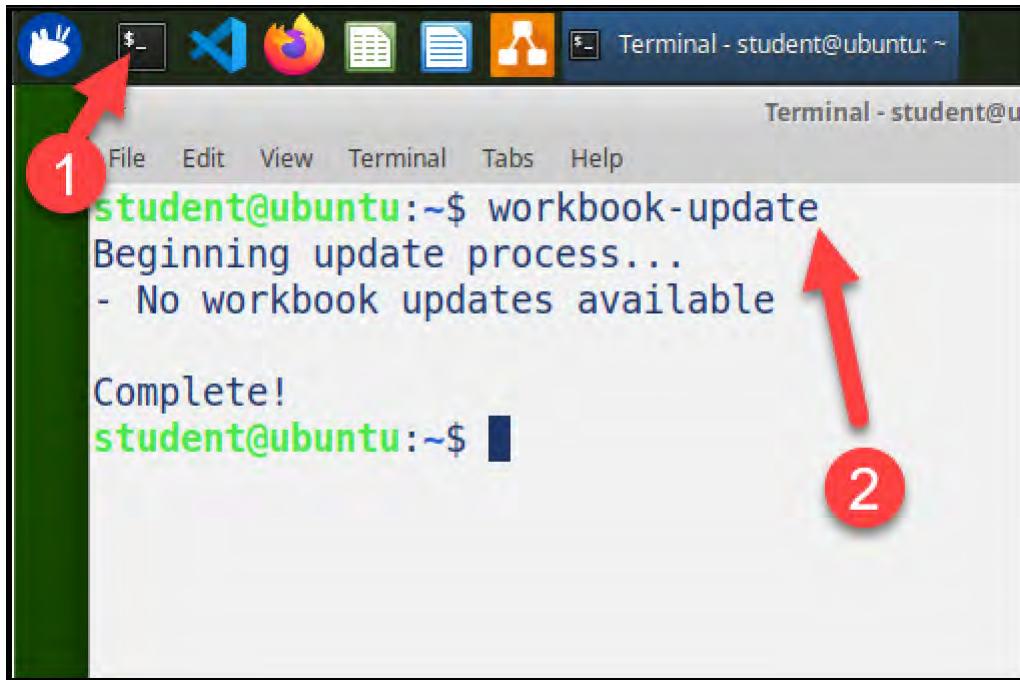
We recommend performing the update process at the start of the first day of class to ensure you have the latest content.

The electronic workbook site is stored locally in the VM so that it is always available. However, course authors may update the source content with minor fixes, such as correcting typos or clarifying explanations, or add new content such as updated bonus labs. You can pull down any available updates into the VM by running the following command in a bash window:

```
workbook-update
```

Here are specific instructions for Linux VMs:

- Open a Terminal window and run the command `workbook-update` as shown here:



The script will indicate whether there were available updates. If so, be sure to refresh any pages you are currently viewing (or restart the browser) to make sure you are seeing the latest content.

Using the E-Workbook

The MGT551 electronic workbook should be the home page for the browsers inside all virtual machines where it is maintained. Simply open a browser or click the home page button to immediately access it in the VMs.

You can also access the workbook from your host system by connecting to the IP address of your VM. Run `ip a` in Linux or in the Ubuntu bash shell in Windows to get the IP address of your VM. Next, in a browser on your host machine, connect to the URL using that IP address (i.e. `http://<%VM-IP-ADDRESS%>`). You should see this main page appear on your host. This method could be especially helpful when using multiple screens.

We hope you enjoy the MGT551 class and workbook! To get the most out of your lab time in class, we recommend following the guidance in [How to Approach the Labs](#).

Wiki Update

Before you start the course it is advised that you run update script to ensure you have the latest content updates and fixes. Internet access inside your virtual machine is required for this process.

To check for updates, open a terminal window and run the following command:

```
workbook-update
```

This command will reach out to GitHub and download the newest version of the wiki content, once complete, reload your browser page to ensure you see the new updates.

Syntax Used in This Course

The MGT551 course documentation uses consistent syntax styles with which you should become familiar. This section helps you to make sense of what the material conveys, so you can focus more on course material than styling.

Syntax Descriptions and Examples

Note

The commands listed in this section of the lab are just for reference, so you can become familiar with text styles used in the course materials. *No need to actually run them in your SIFT Workstation VMware Image!*

1. Text blocks that appear in the format shown below contain commands that you would run in the class VM. These code blocks include an icon to the far right that allows you to copy the contents of the block, suitable for pasting into the shell in your class VMs.

List the contents of the `/tmp/` directory

```
cd /tmp/  
ls -l
```

The results are shown in a slightly different format. Results will be denoted as "Expected" or "Notional". Expected Results should reflect exactly what you get from the commands shown. Notional Results are shown when some variation may be present, based on lab or classroom conditions.

Notional results

```
total 2836  
-rw----- 1 student student      0 Apr  3 17:39 config-err-S09tBf  
-rw----- 1 student student      0 Jul 21 18:39 config-err-zVMGjJ  
-rw-r--r-- 1 root      root      0 May 10 07:45 fileK8YYJh  
-rw-r--r-- 1 root      root      0 Jun 11 07:45 fileVAP3BY  
-rw-r--r-- 1 root      root      0 Jul 11 07:45 fileVeFmlj  
drwxrwxr-x 3 student student    4096 Jul  6 18:03 npm-57783-5d61223f  
drwxrwxr-x 3 student student    4096 Jul  6 18:04 npm-57819-3bc1b3dc
```

2. Direct questions are reflected in the material as shown below.

How large is the `nitroba.pcap` file, in bytes?

56,795,590

Command lines

```
cd /cases/mgt551/sample_files/  
ls -l nitroba.pcap | awk '{print $5,$9}'
```

Expected results

```
56795590 nitroba.pcap
```

What is the file's MD5 hash value?

```
d6b5df10fc572b54ceb9c543d11f10a4
```

Command lines

```
cd /cases/mgt551/sample_files/  
md5sum nitroba.pcap
```

Expected results

```
d6b5df10fc572b54ceb9c543d11f10a4 nitroba.pcap
```

Narrative answers are shown in **bold** as shown below.

What are two ways to see the contents of the `/cases/mgt551/sample_files/` directory?

The bash shell's `cd` and `ls` commands provide one way, and the Ubuntu GUI file manager interface is another.

Command lines

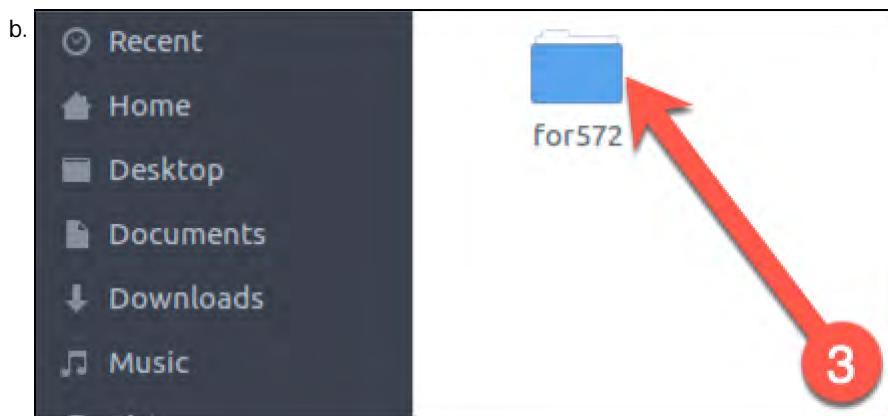
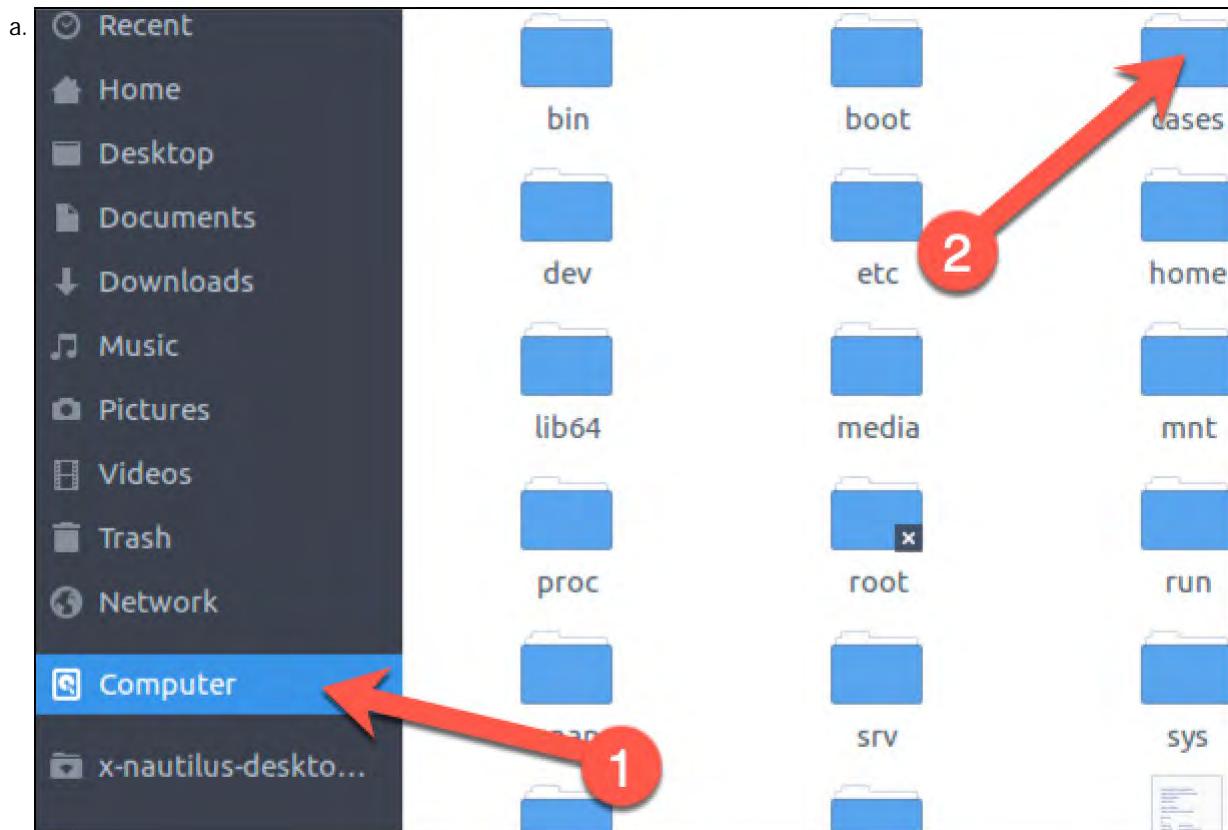
```
ls -l /cases/mgt551/sample_files/
```

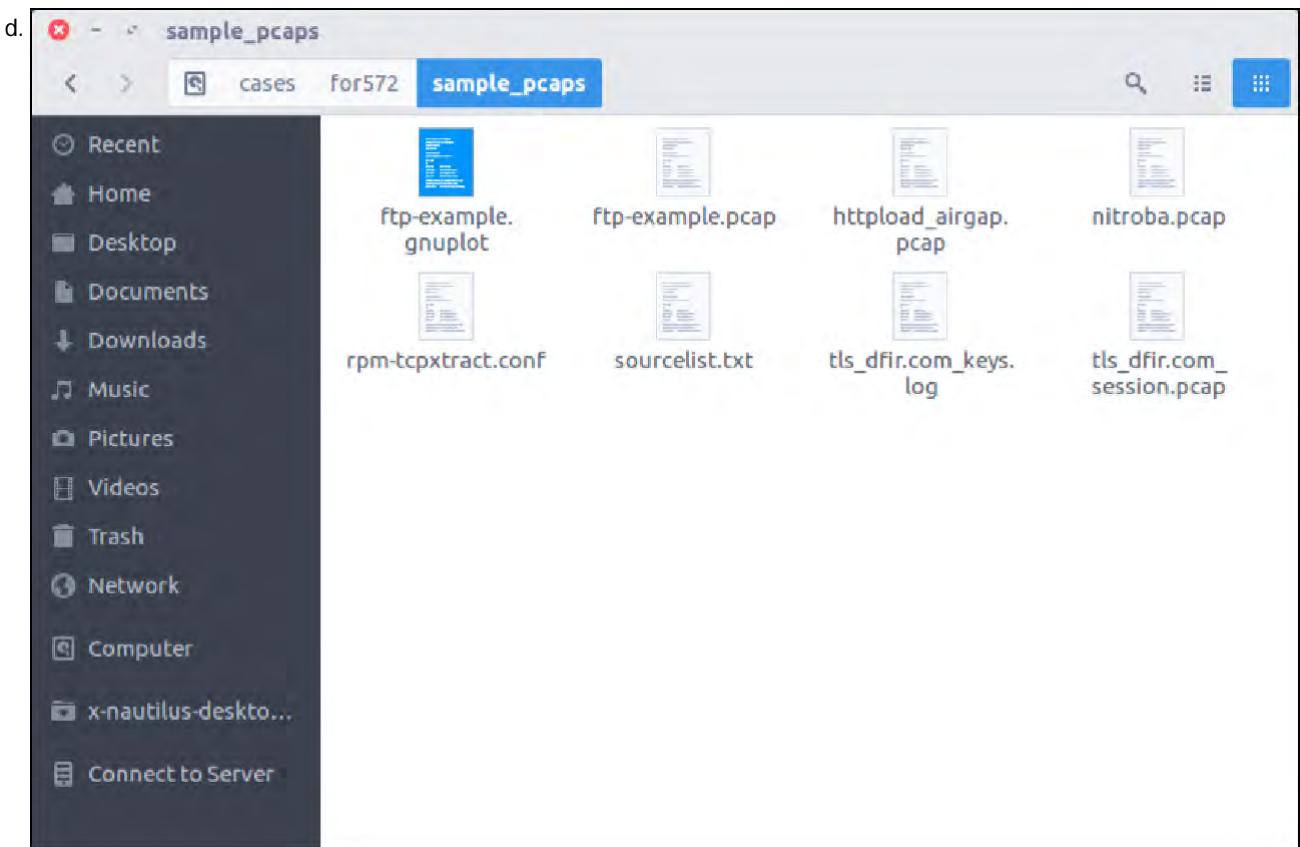
Expected results

```
total 115872  
-rw-r--r-- 1 student student 449 Aug  1 2016 ftp-example.gnuplot  
-rw-r--r-- 1 student student 36114110 Nov 22 2013 ftp-example.pcap  
-r--r--r-- 1 student student 23462957 Jan  3 2019 httpload_airgap.pcap  
-rw-r--r-- 1 student student 56795590 Jul  5 2013 nitroba.pcap  
-rw-r--r-- 1 student student 116 Aug  1 2016 rpm-tcpextract.conf  
-rw-r--r-- 1 student student 555 Jan 28 01:01 sourcelist.txt
```

```
-rw-rw-r-- 1 student student      8039 Dec  6 2018 tls_dfir.com_keys.log  
-rw-r--r-- 1 student student 2250712 Dec  6 2018 tls_dfir.com_session.pcap
```

GUI file manager



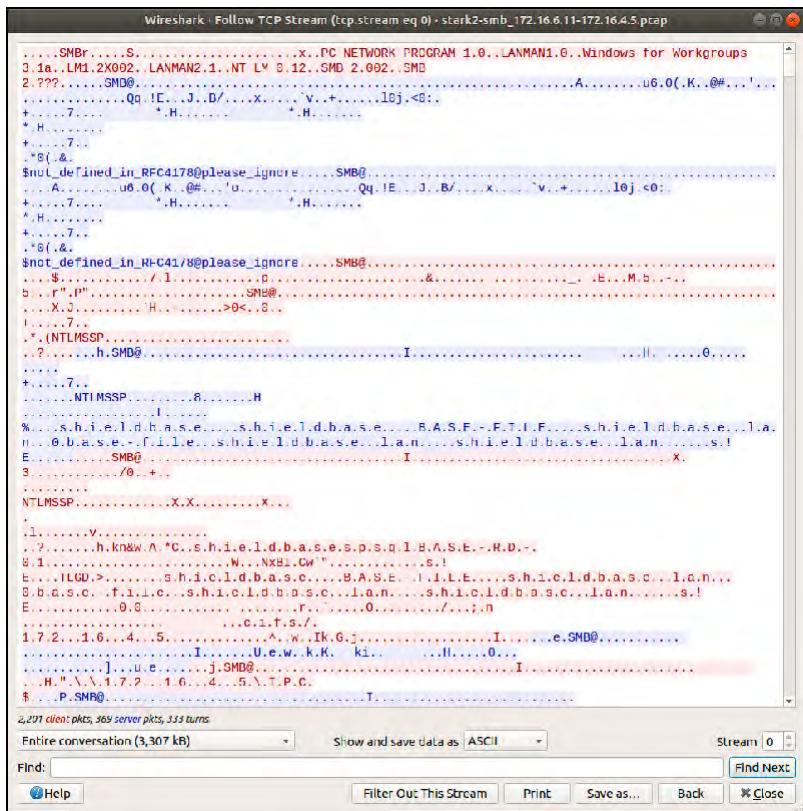


3. When referring to literal strings inline with narrative text, the strings will be in depicted in Courier New font. For example, a search string of `destination_bytes:[6000000 TO 7000000]` might be noted in the material inline, or via a call-out box as shown below:

```
destination_bytes:[6000000 TO 7000000]
```

4. It is generally unadvisable to use the `root` administrative account for normal activities. We will follow best practices and use the `sudo` utility to perform administrative actions within the VM environment wherever needed. The `student` user has full `sudo` access to provide a reasonable balance between best practices and a practical classroom-based lab environment.

5. In the electronic workbook, some images are clickable, resulting in an enlarged version. This can be helpful when examining a detailed diagram or screenshot. An example of this is below.



How to Approach the Labs

To get the most out of each lab, we will step you through the different portions of the workbook. The workbook is specifically designed to enable students from a variety of backgrounds and with different skill levels get the most out of each lab.

Exercise Objectives

This section is designed to help students understand the larger picture of what the objectives of the exercise are meant to show or teach. In some cases, we might be demonstrating an analytical technique or the specific output of a tool. We strongly recommend that students quickly look over these objectives when beginning the exercise.

Exercise Preparation

We design exercises to stand on their own. This allows students who are reviewing the exercises to jump in without having previously done the exercise. We typically outline the condition of the system, or the capabilities that must be enabled before moving into the actual exercise. Skipping over this step could mean that your system might not be ready for performing the lab.

Questions without Explanations and Questions with Step-by-Step Instructions

For most exercises, we try to get you to focus on the core concepts instead of just running blindly through a tool or exercise.

For most students doing the exercise for the first time, we recommend using the second part of the exercise that has step-by-step instructions and explanations.

Note

In the printed workbook, the step-by-step instructions and explanations are provided in a separate section following the section with the questions. In the electronic workbook, the step-by-step instructions and explanations are provided immediately following each question using a drop-down box such as this (click the box to see the solution):

Solution

Here's where an answer would go. There will be a drop-down box such as this following each individual question. The electronic workbook does not have a separate dedicated step-by-step section.

At this point, there are three ways to do the exercises.

- **Gain familiarity:** During the first time through the exercise, students should use the step-by-step questions with instructions in order to familiarize themselves with the overall topic and techniques. Remember that you are here to learn, not to fight your system or become confused. You will get more from the exercise by following along and mimicking what you see directly while reading the full (and sometimes lengthy) explanations.
- **Gain mastery:** When students are reviewing the exercise, we recommend that they use the “hybrid” approach. This approach has you start with the part of the exercise that has questions without any help or explanations, but then reference the step-by-step questions with instructions when you get stuck.
- **Achieve mastery:** Once you can complete the exercise using the step-by-step questions without instructions, you have mastered the skill. If you have already mastered the skills on exercises from the start, it is likely you have learned those skills already, or know them from previous courses. Many students take a class to obtain new skills, but the more advanced they are, the fewer the new skills they will learn each time they take a course.

Takeaways

For every exercise, the takeaway section highlights important exercise related lessons. We advise looking through takeaways to summarize the work you have just performed in the exercise.

Exercise Steps

MGT551 incorporates many hands-on course elements to enhance the learning experience and show how to apply concepts taught. We employ varied approaches to hands-on components including Linux-based local labs.

A Linux virtual machine is provided in the MGT551 media files that will need to be configured on your system. All student labs will be performed locally with minimal or no need for internet access.

The Xubuntu Linux VM (virtual machine) will be used for lab exercises in this course.

Note: Although this printed workbook contains hard-copy versions of all the labs in the course, after virtual machine setup it is highly recommended that you work through the labs using the built-in wiki provided within the virtual machine .

0. Install VMware Player/Workstation Pro/Fusion

It is assumed that you have come to class with a **VMware** virtualization product pre-installed as listed in the course requirements (other virtualization products such as Virtualbox Parallels and Hyper-V are not supported). If you have not yet installed one of the VMware virtualization software packages for your host, please download and install it now.

Free: [VMWare Player \(Windows/Linux\)](#)

[Workstation Pro \(Windows/Linux\)](#)

[Fusion \(Mac\)](#)

If you have a PC, Player can be used for free, and Workstation Pro can be used on a trial license for 30 days. Either option will work for this course, the main difference is that **Workstation Pro** allows taking snapshots of the virtual machine state. **Snapshot** functionality will not be used for any of the labs but it may be convenient for you to revert to a known-good state if something becomes broken. **If you have Workstation Pro, it is recommended that you take a snapshot of the virtual machine in the booted state the first time you get it up and running.**

1. Extract the zip file

If you have a USB drive

(Note this step is not applicable to those who downloaded .iso file)

You will receive the MGT551 media files by the first day of the course (or through download/the mail for OnDemand students). If using a USB, insert the MGT551 USB into your laptop. Browse to the USB root directory. Copy/drag the .zip file to a local directory of your choice. Copy the zip file from the USB drive to your hard drive before proceeding.

If you have a digital download

If you have the .iso image download for the course media, double click the .iso file and the image of a USB drive should mount in your operating systems file manager. Within it, you should see a .zip file, once located, proceed to the next step.

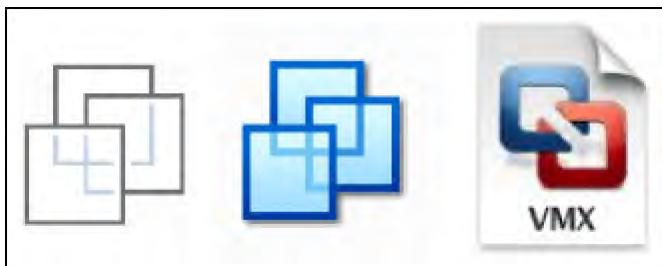
2. Decompress the zip file

Double click the .zip file to start the extraction process. The files are large so this step may take a while if you have a slower machine. Wait for the extraction to complete.

3. Import the virtual machine into VMWare

Double-click the extracted folder to open it. Then, look for the file that ends with the extension `.vmx` and double click it to open it with VMWare.

The Linux .vmx icon has three overlapping white or blue squares (shown here on the left and middle, respectively). On OS X, the icon has blue and red overlapping squares (shown on the right):

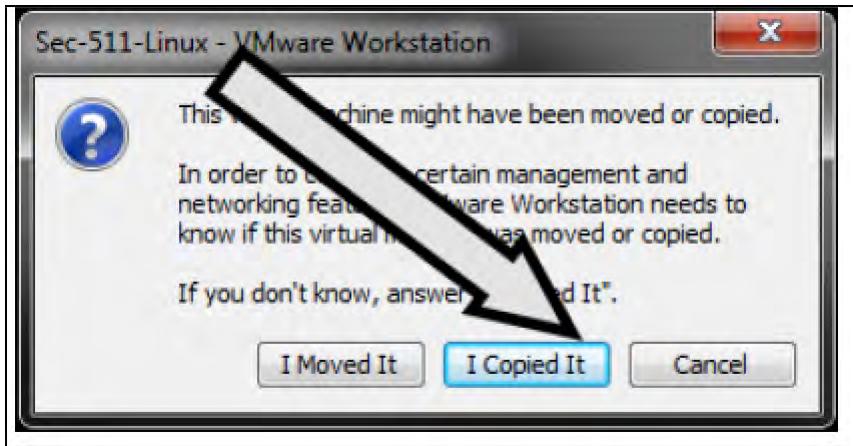


VMware should start. If VMware does not start, ensure you clicked the .vmx file. Also, ensure that VMware is properly installed.

Once VMware starts you may receive a prompt to upgrade the virtual machine to support the newest version of VMware. You can choose to if desired, but it is not necessary.

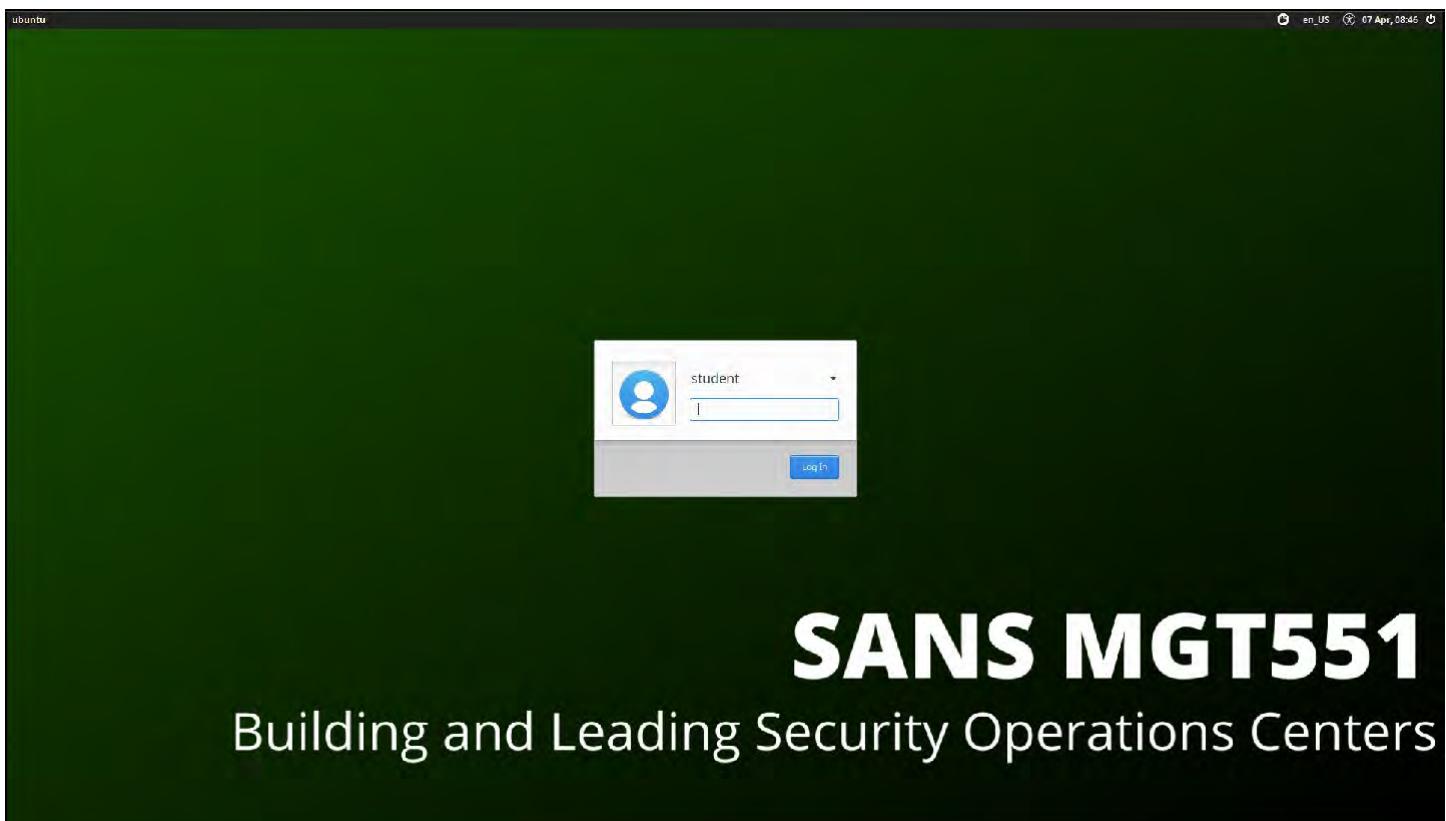
If you see the option to "Take ownership" or "Cancel", select "Take Ownership".

If you get the popup below, select "I Copied It" to move forward.



4. Login to the desktop

Depending on your version of VMware, you may need to press "Power on this virtual machine" (or it may start automatically). After the VM starts, you end up at the login prompt. Log in with the password `mgt551`. If you see a box asking for a username, the username you should enter is `student`.



Once you are logged in and see the desktop, you are ready to go!

 **Note**

If you have a high DPI screen and the resolution of the virtual machine is too small, you can adjust the resolution of Windows to correct for it, or use VMware Fusion settings to disable the "Retina" resolution in the virtual machine for Macs.

Optional: If you have a VMware version capable of taking snapshots, now is a good time to make one as a convenient restore point.

Exercise 1.1: Threat Actor Assessment

Background

A crucial early step in building or improving your SOC is aiming at the right goals. Towards that end, in this first exercise, we will seek to answer the fundamental question "What types of threat actors will our organization face?". Every team has a limited budget and a (minimum) goal of reducing the chance of material impact to their organization through cyber attacks. So it makes sense that designing your defenses to go up against known, relevant threats is the right way to optimize your defense spending. The first step of getting that done is figuring out what threats are out there that are the highest risk to you and your organization.

Objectives

- Identify threat groups relevant to your organization
- Prioritize relevant threat groups to enable a risk-based defensive planning approach
- Learn how to create threat actor profiles with relevant data the team needs to know about each group
- Organize threat actor data in a way that is easily accessible and readable
- Take the first step in building a threat intelligence-driven defensive strategy

Exercise Preparation

This exercise is completed in your MGT551 Linux VM, if you have not yet completed exercise 1.0 to set up the class virtual machine, please complete the setup before proceeding. If you have trouble with the setup, see the troubleshooting information or reach out to an instructor or SANS support.

Launch the MGT551 Linux VM and log in.

```
- LOGIN = `student`  
- PASSWORD = `mgt551`
```

Once you are at the Linux virtual machine desktop, you are ready to proceed with the exercise.

Exercise Steps

Before you get started: Ensure you have internet connectivity, feel free to do the research for this exercise in your host system browser or within the virtual machine. You should be able to copy and paste text back and forth between the virtual machine and your host.

Organizational Introspection

There are many reasons a threat group may target your organization, so in order to come up with a list of threat actors who may be interested in you, the initial task is to consider what an attacker may find worthwhile about your organization and its data. In this step, we'll try to stretch a bit beyond the obvious "proprietary data" or "customer information" to be sure you have a complete list of factors that make your organization enticing to the various threat actors.

Note

If you do not want to use your own organization for this exercise, select an example org that is very similar to your own - a competitor or other company that has the same industry and rough size as your employer, or just pick an example organization type that you can continue to use as an example as we build on the output of this exercise throughout the course. If you work for an MSSP or other organization where you have multiple customers with varied potential attackers you can either focus on threats to you as an MSSP directly, or pick an example organization to focus on that is representative of your most typical customer profile.

Threat actor groups range from highly-sophisticated advanced persistent threats, to organized crime groups, to "hacktivists", script kiddies and more. Each of these types of threat actors may have different motivations. For this step, put yourself in the shoes of an attacker from each of these groups and consider what your goals might be if performing a cyber attack against your organization. Anything from disruption of business to data theft to financial theft is on the table, and don't forget to consider groups that are not after you directly, but instead are interested in leveraging access your organization has to reach the *real* target (think supply chain attacks or privileged access given to MSSPs).

Brainstorm 5-10 reasons an attacker of various types might have to perform a cyber attack on your organization. Remember, money should be on nearly every organization's list given that is the common target of many organized crime groups and publicly available reports like the Verizon DBIR list it as by far the most common attacker motivation. While listing these items, don't just think of data, consider the specific systems, people, access, and data types your organization uses, and how they could be misused or appropriated by an attacker for his/her own benefit.

If you are working out of your workbook, here is a space to write down your ideas. If using the digital wiki or PDF files, capture your ideas in a file, OneNote, or similar, so you can easily reference them later.

1. _____
2. _____
3. _____
4. _____
5. _____
6. _____
7. _____
8. _____
9. _____
10. _____

Here's an example of the list you might come up with if you work in a pharmaceutical company:

Asset Name
Manufacturing process documents
Drug formulas
Patient Research Data
R&D project info
Cash
Employee tax info
Operating information
Compute resources

Most individuals completing this step will probably have a list that contains some mix of money, sensitive customer data, proprietary information and intellectual property such as manufacturing processes, formulas, or source code, or perhaps even high-impact items like classified government information, weapons systems, or life-sustaining industrial processes such as power generation or water treatment. This step should be rather easy if you are at all familiar with your organization does on a day to day basis.

Note

One thing to consider - if given this task, would **everyone** in the SOC be able to produce a similar list of items they are responsible for defending? If not, they may be hyper-focused on the details of the job but not "see the forest for the trees".

Threat Group Identification

Now let's turn our attention to threat groups.

Preliminary Focusing

Remembering that attackers may belong to any of the groups below and more:

- State-Sponsored
- Organized Crime Groups
- Hacktivists / Ideologically Inspired Attackers
- Terrorists
- Malicious Insiders
- Competitive Organizations

- Lone Hackers
- Script Kiddies

Which general buckets of attackers do you think might be the most relevant to your organization? If one of these groups is your primary focus, now is the time to make a note, as we're about to dive into some initial research.

At this point our example organization may come up with the following list:

Group Type
State-sponsored
Organized Crime
Hacktivists
Malicious Insiders
Script Kiddies

Now that we have an idea of why we might be interesting and what types of groups we are interesting to, let's continue to find some specifics. The following step will require a bit more effort than the first part since we will perform some open-ended external research. In this next step, we're going to perform some open-source intelligence (OSINT) gathering and try to narrow down the pool of threats to identify which ones are of primary relevance to *you*. To do identify groups of interest we will focus on threat groups that have affected organizations like yours (or the ones we have chosen for this exercise).

Remember, in order for a "threat" to exist, there must be:

- Intent
- Capability
- Opportunity

We're assuming the *opportunity* is present for now, and will later enumerate the *capabilities* of each group. For this step, we're focusing on the third factor - identifying which groups have *intent* to harm you, and we will do this by researching past attacks, assuming these will continue similar patterns into the future.

There are multiple ways we can approach gathering this data, for this exercise we will focus on two. One is to use a search engine to find news articles about recent attacks and scan them for relevance, the second is to go to pre-existing organized bodies of cyber threat intelligence about attackers and extract the information. We will walk through both methods to try to get the best info from both types of sources.

Search Engines and News Reports

The first piece of this step will be to head over to a search engine and type in terms like "cyber attack" and the type of industry you work in, then hitting the News section in the results.

For our example organization, we might find the following:

A screenshot of a Google search results page for the query "pharma cyber attack". The results are filtered to show the "News" category. The search took 0.36 seconds and found approximately 72,800 results. Three news articles are displayed:

- BleepingComputer**: Russian-Speaking Hackers Attack Pharma, Manufacturing ...
A red arrow points to the timestamp "1 day ago".
- SC Magazine UK**: State-sponsored hackers target big pharmaceuticals
A red arrow points to the timestamp "3 weeks ago".
- The Hill**: Experts report recent increase in Chinese group's cyberattacks | TheHill
A red arrow points to the timestamp "2 days ago".

Pay particular attention to the dates of the articles you find. You should place the most weight on the attacks and attack groups that have been active recently. Articles from years ago may be useful in that a group that was interested in your industry years ago is likely to still be interested, but the tactics and techniques used in the attack are likely no longer as relevant. It's highly likely the group has improved and shifted their approach significantly since then.

Try to find articles that go beyond a surface level and contain some useful intelligence about the reported incident and who was behind it. Key details might include group names, nationalities, motivations, TTPs, tools and more. **Keep these tabs open for the next step.**

Fill in or write down the names of at least three threat groups that you find through this method.

Names of threats / threat actors:

1. _____
2. _____
3. _____

Note

If you do not find threat group names but rather general attack types, feel free to alternatively write down attacks types or malware names if attribution is not available. Things such as "ransomware" or "ragnarlocker" may still be useful if you can't find group names.

Here's an example of findings you might gather from this step if pursuing info on our example company. This picture is from the article listed in the previous picture, which gives our example pharmaceutical company a key piece of info on specific threat group names that might be interested in them and why:

Russian-Speaking Hackers Attack Pharma, Manufacturing Companies in Europe

By [Ionut Ilascu](#)

 March 27, 2020

 05:42 AM

 0

Malware belonging to Russian-speaking threat actors was used in attacks in late January against at least two European companies in the pharmaceutical and manufacturing industries.

Based on the tools employed in the attacks, the suspects are likely the **Silence** and **TA505** financially-motivated groups.

Next let's move on to a different source of information - cyber threat intelligence sources.

Organized Cyber Threat Intelligence Sources

Use the resources below to continue your search. Look to combine your findings with the information from the previous step.

- Save the names and pages of any additional groups you might want to add to your list
- Bookmark or save the URLs for any useful additional info you find on your target threat groups

Tip

Remember - Cyber threat intelligence research groups may refer to threat actors by different names, be sure to look for and note any alternative names for any groups of interest. A great resource for mapping threat group names across vendors is Florian Roth's [APT Groups and Operations Spreadsheet](#)

Again, keep the tabs open and bookmark any useful information you find, as we will need it in the next step.

Resources:

- [Kaspersky Targeted Attack Logbook](#) - Search by industry
- [ThaiCERT Threat Encyclopedia](#) - Search by group name or industry
- [MITRE ATT&CK Groups page](#) - Search by group name, pivot to additional ATT&CK info easily
- [FireEye Advanced Persistent Threat Groups](#) - Search by group name

Fill in the names of three relevant threat actors or attacks you've found, based on the additional information from this step:

1. _____	Aliases: _____
2. _____	Aliases: _____
3. _____	Aliases: _____

At this point you should have some idea of the types of threats your organization might face, and ideally the names of the groups as well. This information can begin to (or continue to) inform our security prevention and detection strategy that we will build throughout the course.

Threat Group Profile Creation

In this final step, we'll take our saved tabs and fill in some structured info about our top 3 threat groups identified in the previous step and organize the data in a document that can be easily shared with your team and saved for later reference.

We are *not* looking to do a full-on threat intelligence analysis of each threat group, but rather get a high-level view that helps us prioritize and profile known relevant groups. A complete analysis is a task best left for threat intelligence specialists, and is something you either should seek to do as a result of your findings in this exercise, or ensure is rapidly updated if you already have a functioning threat intelligence vendor or internal team.

Open up the pre-made threat actor profile document in your virtual machine from the shortcut on the desktop labeled "MGT551 Threat Actor Profile.docx".



You should see the following document:

MGT551 Threat Actor Profile 1	
Basic Info	
Threat Actor Name	
Description	
Aliases	
Last Seen	
First Seen	
Motivation and Goals	
Threat Actor Type <i>Options: [activist, competitor, crime-syndicate, criminal, hacker, insider-accidental, insider-disgruntled, nation-state, sensationalist, spy, terrorist]</i>	
Sophistication <i>Options: [none, minimal, intermediate, advanced, expert, innovator, strategic]</i>	
Threat Actor Motivation <i>Options: [accidental, coercion, dominance, ideology, notoriety, organizational-gain, personal-gain, personal-satisfaction, revenge, unpredictable]</i>	

Tip

If instead you'd like to directly perform this exercise on your host PC and save your document in Word outside your virtual machine, you can either copy and paste the template from the document into your own document, or drag and drop the file itself from inside your virtual machine window to your desktop... (most, but not all versions of VMware should support this). It is located in the / home/student/labs/1.1 folder.

There are 3 pages in this document ready to fill out with the information you found as described below. Feel free to make additional pages if you have more than 3 threat groups.

Here is a list and description of the details we will be filling out for the threat groups:

- **Name** - A name used to identify this Threat Actor or Threat Actor group.
- **Description** - A description that provides more details and context about the Threat Actor, potentially including its purpose and its key characteristics.
- **Aliases** - (Optional) Alternative names used to identify this malware or malware family.
- **Last Seen** - The time that this Threat Actor was last seen.
- **First Seen** - (Optional) The time that this Threat Actor was first seen.
- **Goals** - The high-level goals of this Threat Actor, namely, what are they trying to do. For example, they may be motivated by personal gain, but their goal is to steal credit card numbers. To do this, they may execute specific Campaigns that have detailed objectives like compromising point of sale systems at a large retailer.
- **Threat Actor Type** - The type(s) of this threat actor.
 - [activist, competitor, crime-syndicate, criminal, hacker, insider-accidental, insider-disgruntled, nation-state, sensationalist, spy, terrorist]
- **Sophistication** - The skill, specific knowledge, special training, or expertise a Threat Actor must have to perform the attack.
 - [*none, minimal, intermediate, advanced, expert, innovator, strategic*]
- **Threat Actor Motivation** - The primary reason, motivation, or purpose behind this Threat Actor. The motivation is why the Threat Actor wishes to achieve the goal (what they are trying to achieve). For example, a Threat Actor with a goal to disrupt the finance sector in a country might be motivated by ideological hatred of capitalism.
 - [accidental, coercion, dominance, ideology, notoriety, organizational-gain, personal-gain, personal-satisfaction, revenge, unpredictable]
- **Observed Attack Campaigns** - Which attacks have specifically been attributed to this adversary (and links to references)

Note

This information is based on a subset of the Structured Threat Information Expression (STIX) Version 2.1 standard for defining threat groups. Additional info can be found here: <https://oasis-open.github.io/cti-documentation/stix/intro> This is just a starting point for fields that is based on a pre-existing standard, but feel free to include any additional data you find useful to your team.

Fill out the information that you have found on priority threat groups in the Word document before continuing.

Once complete, save your file somewhere that is easy to continue to access. We will need this information for future labs!

Taking It Further

If you are brand new to building a security operations team for your organization, no need to stop at just listing threat actors. Continue this exercise by listing out other important assets, threat scenarios, and more. The better you enumerate what you must defend, and the types of attacks you must defend it against, the better you will be prepared and positioned if the scenario ever does occur. There is a great GIAC Gold paper on the process for creating an organizational threat profile in much more detail than we have time for available here: <https://www.sans.org/reading-room/whitepapers/threats/creating-threat-profile-organization-35492>.

Congratulations! You now have a great start to understanding the threat groups you may deal with and the threat environment in which you must operate. In nearly all cases, being an effective SOC means not just paying attention to a single threat type (such as state-sponsored or organized crime), but using a balanced and risk-based and intelligence-driven approach to aligning the protections and services your SOC offers to the attacks that are out there and most relevant to your organization.

Note

For those in organizations that already know or have similar information, when you return to work, see how the output of this exercise compares to what your threat intelligence team has given you. You may have discovered new threat actors or attacks that may need to be added to your watch list.

Keep this file handy. We will have you refer to these threat groups throughout the rest of the course and using these group names as input in future labs. We are trying to build and optimize defenses, and tailoring your collection, detection, and more starts with basing your strategy on pre-identified relevant and high-risk threats.

Exercise Conclusion -- Key Takeaways

In this exercise, you have:

- Created a base for which you can start to focus your security efforts

- Found high-risk and relevant threat actors
- Organized discovered data in an easy to share, structured format
- Taken an important step in building a risk-informed, threat intelligence-drive defense

Exercise 1.1 is now complete!

Exercise 1.2: Attack Path Development

Background

In this exercise we'll take the next step in getting ahead of our attackers by pre-considering some of our highest risk scenarios and how they might play out. Doing this activity *before* the attack occurs allows us to stay one step ahead of attackers and further customize our detection and prevention capability, hopefully stopping the scenario from ever playing out to completion in reality. To accomplish this, in this exercise, we'll be putting on our "evil hat" and making an example attack tree - a great way to visualize and prepare your team for a highly likely event.

Objectives

- Utilize threat intelligence to prepare your security operations team
- Learn how to create an attack tree
- Stay ahead of attackers by pre-calculating and blocking their likely moves before they get a chance to try them
- Create a list of asset, user, and other priorities that will help prevent high-impact damage via cyber attack

Exercise Preparation

This exercise is completed in your MGT551 Linux VM, if you have trouble with the setup, see the troubleshooting information in the wiki, or reach out to an instructor or SANS support.

Launch the MGT551 Linux VM and log in.

```
- LOGIN = `student`  
- PASSWORD = `mgt551`
```

Once you are at the Linux virtual machine desktop, you are ready to proceed with the exercise.

Exercise Steps

Understanding the Attack Tree Process

Attack trees (sometimes called threat trees) are based on an idea that has been around in multiple forms for decades. Often used in the realm of system security and reliability engineering, the process, in a general sense, is an analytical tool that helps enumerate all of the permutations of events that may lead to a specified outcome. Attack tree-style methodology has been applied for fault assessments, enumerating potential software vulnerabilities and more, and excels at guiding a team on where to focus finite resources to prevent a high-impact event. Today, we will use an attack tree to enumerate ways an attacker may perform a high-impact cyber attack in your environment.

Note

If you'd like to see the genesis of the attack tree method, check out the paper titled "A System Security Engineering Process" by J.D. Weiss at AT&T Bell Labs on page 572-581 (pdf page 224) of [this link](#). Shockingly, the proceedings of the 14th NIST/NCSC National Computer Security Conference, held in October 1991 still exist online, and give us a historical glimpse into where the idea came from.

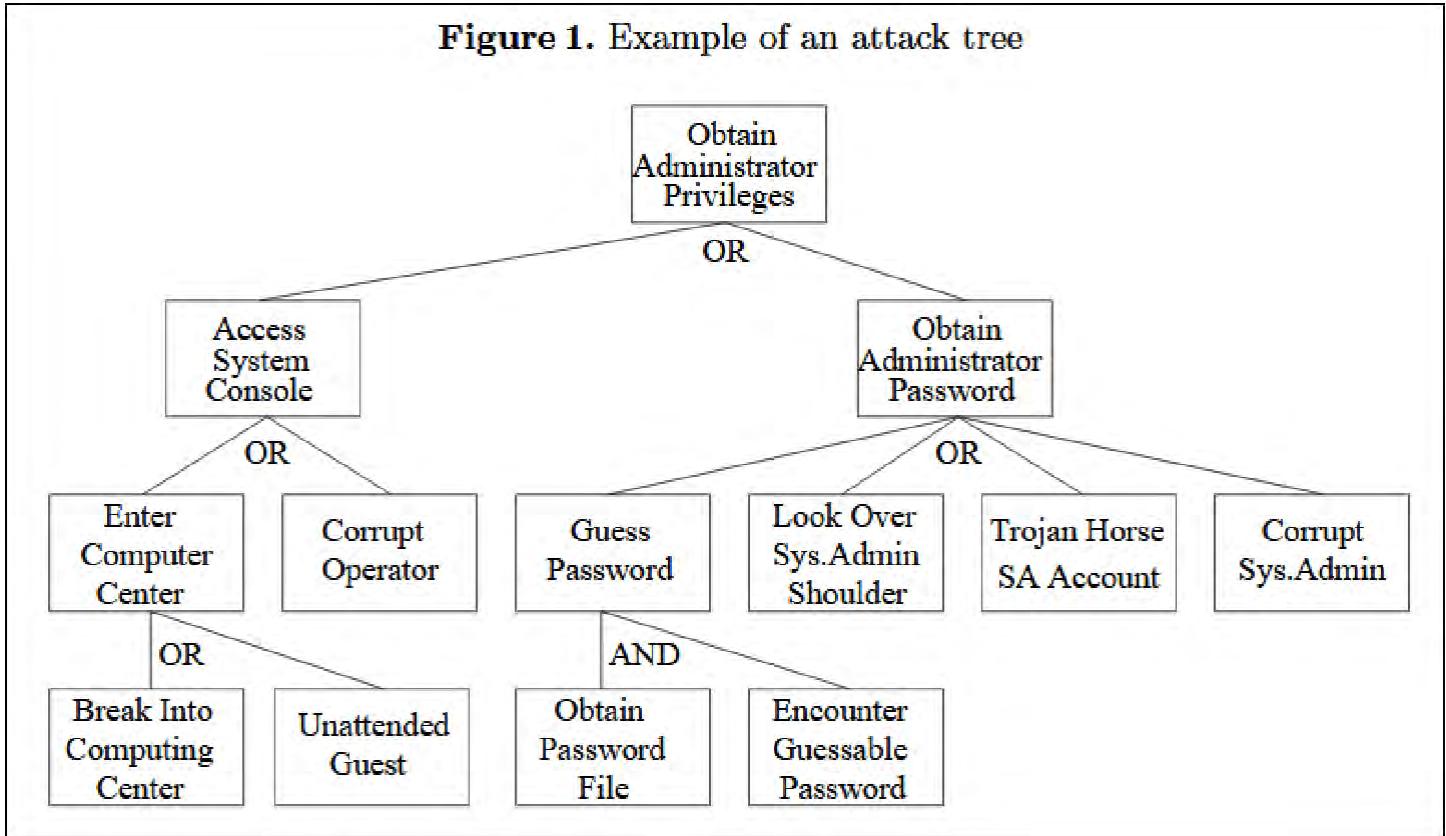
Here is how the process works:

1. Threats to your organization must first be identified - you just did this activity in exercise 1.1, so you should have a high-level idea of the potential possibilities.
2. A single threat's goal is placed at the "root" of a tree.
3. All possible avenues that could lead to that root scenario occurring are enumerated - in other words, what would happen *right before* the final step. This creates branches of potential routes of attacks an adversary might choose to use.
4. Each branch is then expanded again and again, walking backwards, until a detailed map of avenues of attack leading to that goal is enumerated.
5. After multiple branches of possibility are listed, each path can optionally be ranked with difficulty levels, probabilities, logical and/or conditions, or any other labels that help prioritize the most likely approach an attacker might take.

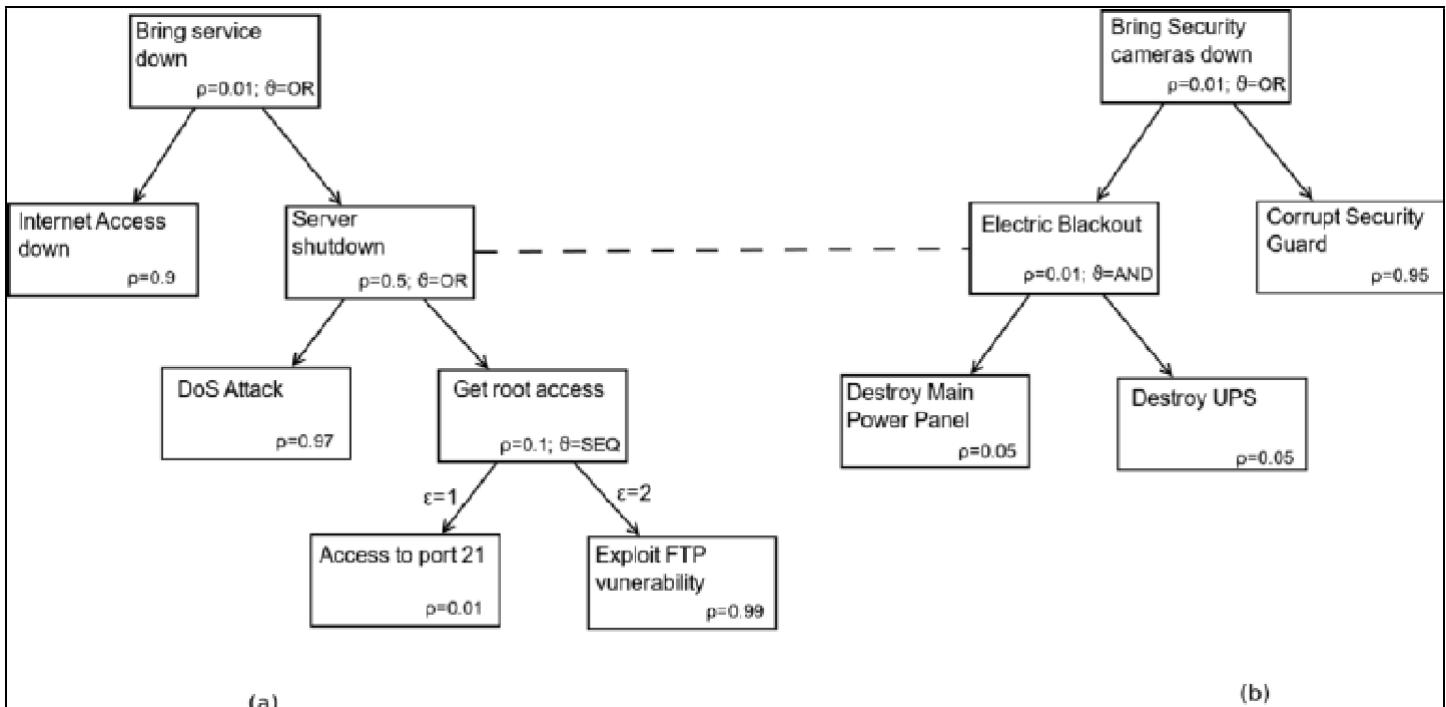
Once you have a complete attack tree, you now have a pre-planned map of how an attacker could accomplish their goal, and also a list of all the places you need to focus resources to protect. By ensuring there is some hurdle for an attacker to jump at every step of the attack tree, and assuming your attack tree is complete, this exercise logically ends in a very difficult defense-in-depth position for you, and a nightmare for the attacker - exactly what we want!

Here are a few attack trees that show what this process might look like for various scenarios. (*Note, these trees are more complex examples with additional detail beyond what will be doing today; focus simply on reading the labels and attack flow.*)

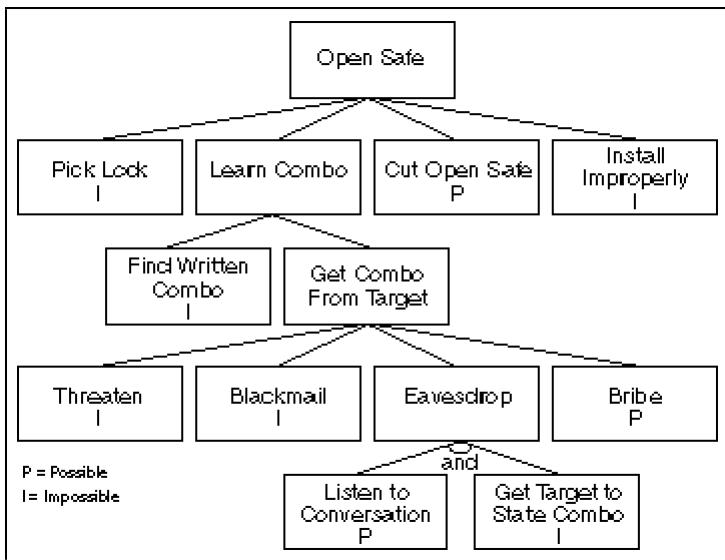
Obtaining administrator privileges[1]:



Disrupting a server or security Cameras[2]:



Robbing a bank[3]:



This should give you an idea of the level of detail we're looking for here - not extreme technical minutiae, just a well thought out set of chronological steps. Now let's make one of our own!

Brainstorming Your Scenarios

To create your own attack tree, the first step is to consider which scenario you'd like to map out. How should you choose? Consider a few top nightmare cyber-attack scenarios to your organization. This can be DDoS, production or critical infrastructure disruption, business email compromise that leads to a large monetary loss, public website defacement, loss of sensitive data, etc.

Take the time now to think of 5 scenarios you think would be on your management's "biggest fear" list for cyber events that are actually reasonably possible. Note them down here or somewhere you can keep handy.

1. _____
2. _____
3. _____
4. _____
5. _____

Since the goal will be to use this scenario through this exercise and others throughout this course to analyze your security posture, consider an important one you'd like to create an attack tree for. A good option is a scenario that is meaningful, high-risk, possible, and that you understand well enough to map out the components of at the level of detail shown in the previous examples.

Chosen Scenario: _____

With a scenario chosen time to create the attack tree!

Creating Your Attack Tree

To create your attack tree, we will use the mind mapping software in your virtual machine called "Draw.io".

Tip

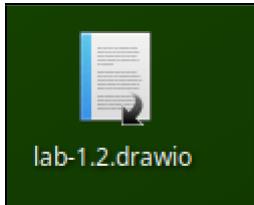
If you prefer, you can also use the online, browser-based version of draw.io available at <https://app.diagrams.net/> to perform the exercise outside your virtual machine, but you will need to start by creating a new, "Blank Diagram" style drawing. Choosing this option may mean the interface is slightly different than what is presented below since it is not under VM control. If you choose this route, you will be responsible for dragging boxes from the side bar at the start instead of starting with the pre-made template and must also save your file somewhere you won't lose it! We will need this diagram again later.

Can I Use Other Drawing Software?

Sure! But you're totally on your own. Since there is nothing in particular about Draw.io that is required for this exercise, if you have another preferred method of creating flow chart / mind map-like diagrams, feel free to use it instead. As long as you can create a series of connected boxes and label the boxes and links(optional), it will work for this lab.

Navigating Draw.io

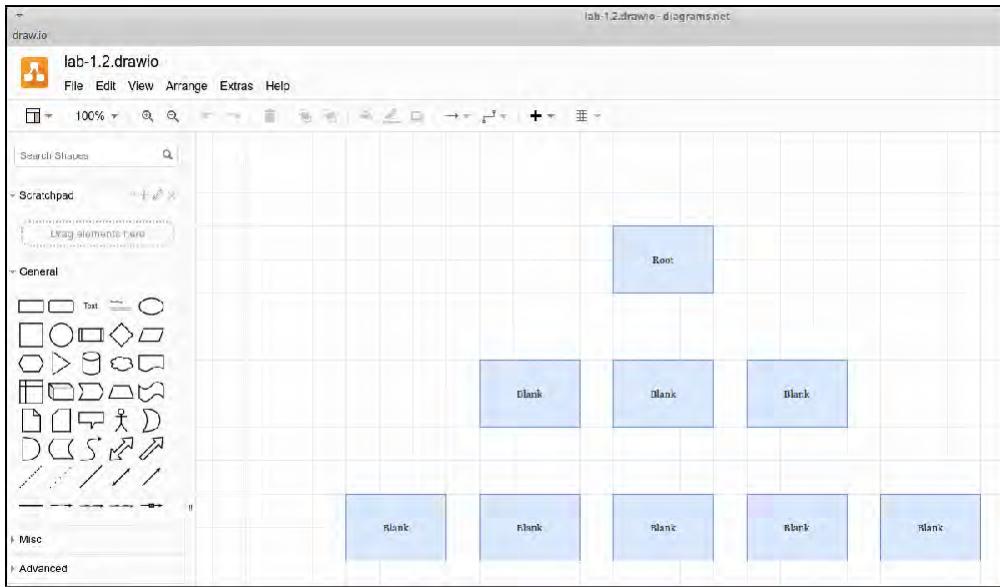
To use the desktop version of Draw.io, load the template file we will use in your virtual machine by clicking the `lab-1.2.drawio` shortcut icon on your desktop. This will load a blank template file that you can start with.



Note

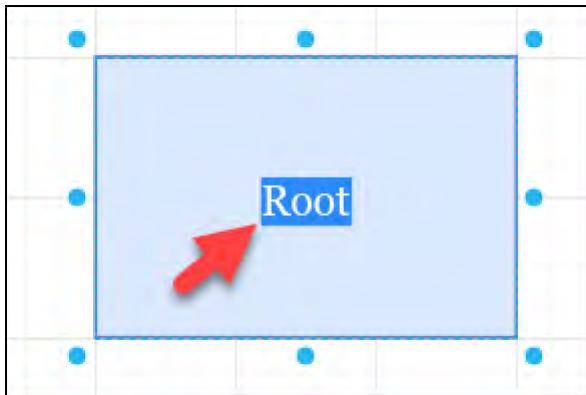
This file is stored in `/home/labs/1.2/lab-1.2.drawio`, this is also another copy of a blank template in that folder (`lab-1.2-template.drawio`) if you need one in the future to create another drawing.

You should now see Draw.io loaded with the provided template file.

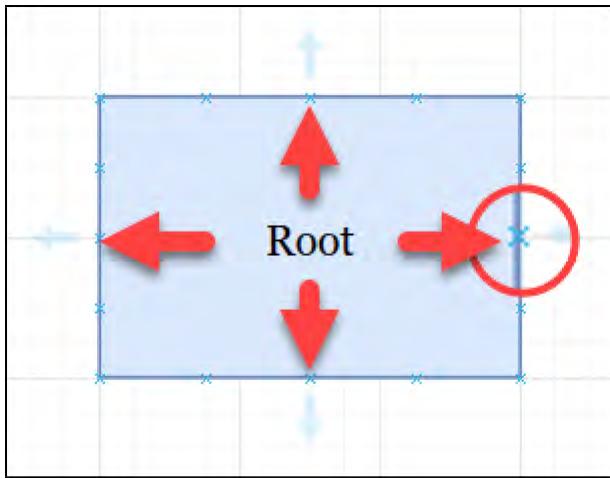


The Draw.io application has a rather intuitive interface.

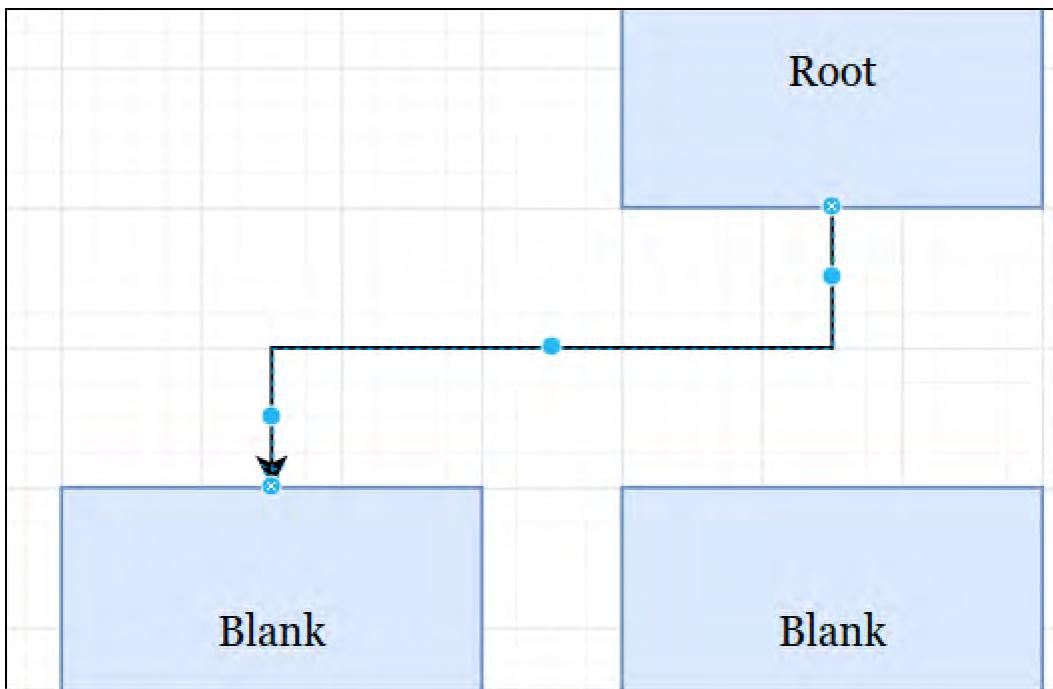
To change the text inside any of the boxes, double click on the box to highlight all the text, make your edit, then press the `Esc` key to exit edit mode.



To connect one box to another with an arrow, hover your mouse over the box on the side where you want the arrow tail to be, you will see a series of small x's show up in various locations around the border of the box as shown below.



Hover over the "x" on one of the box edges and you will see a small green circle appear, this will be the starting point for your arrow. Click and hold, then drag the arrow to another box's edge where another "x" will pop up to serve as the connection point. Let go of the mouse button to make the connection.

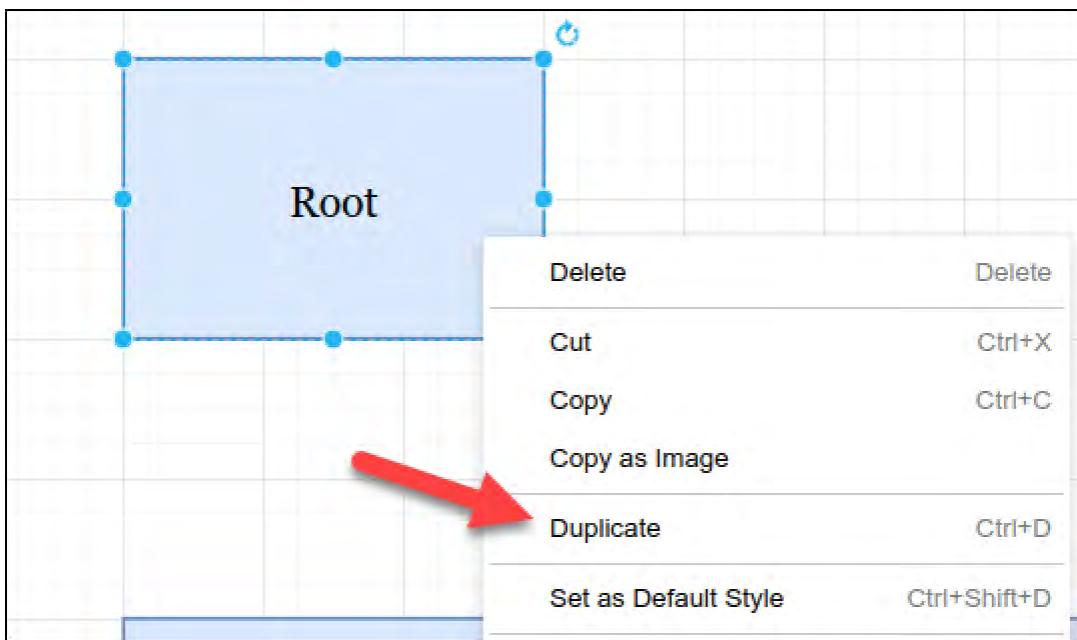


If you make a mistake, simply click the connection line so that it is shown as selected as in the picture above and hit the `Del` key or right click it and select Delete.

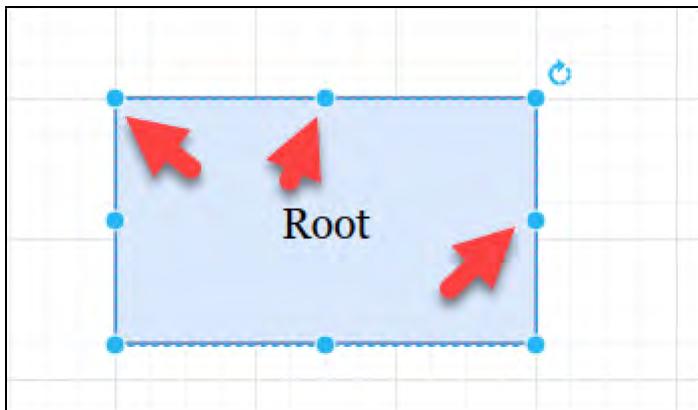
Zoom control is done at the top bar near the File menu, you can also hold `ctrl` on your keyboard while scrolling in and out with your mouse.



If you need to make another box or delete a box, simply click on an existing box and press `Ctrl + D`, or right click on it and select `Duplicate`. Feel free to create or delete all the boxes necessary to get your design done with as much detail as possible.



Resizing a box can be done by clicking on the blue dots that appear on the edges of the box when hovering over it with the cursor.



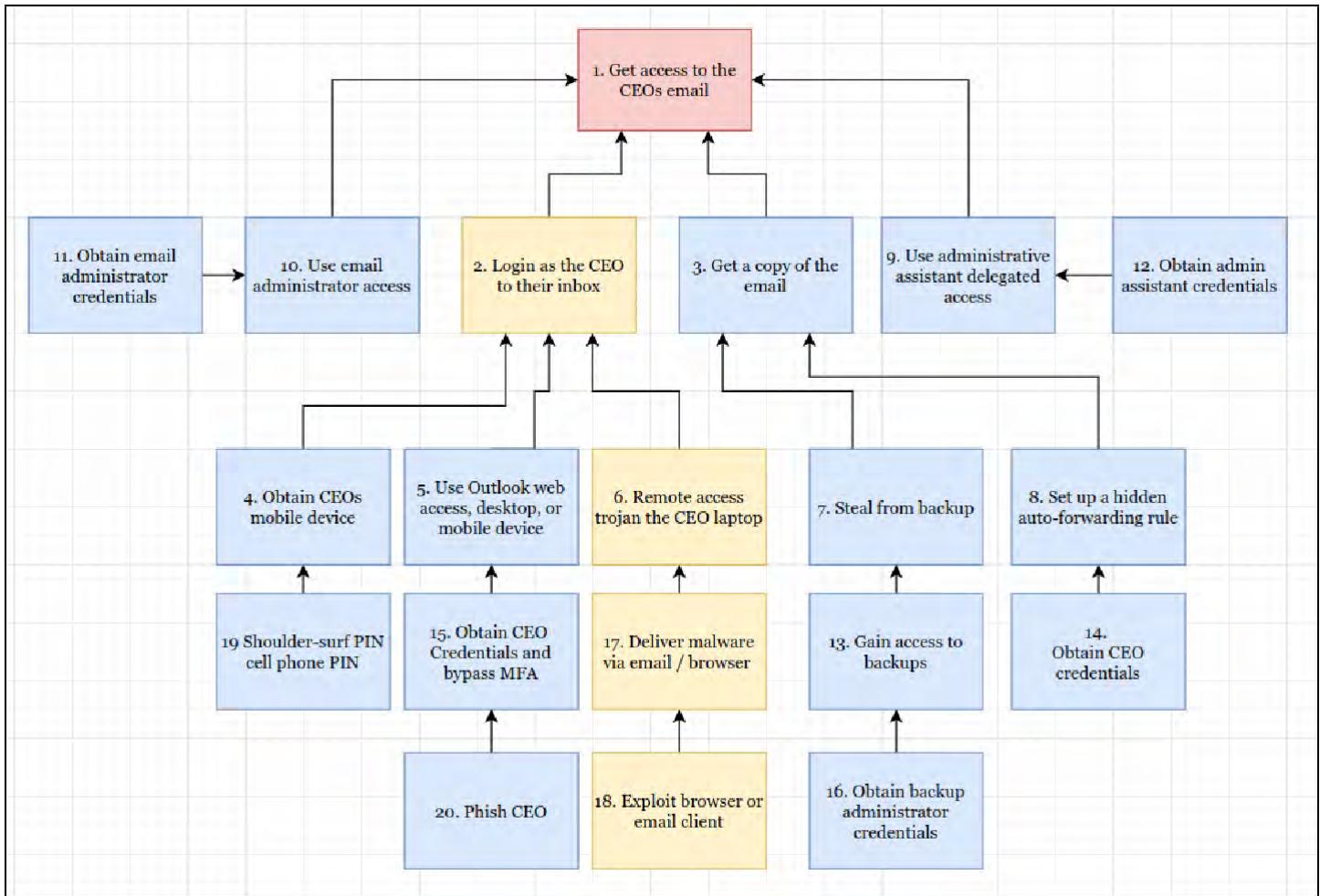
At this point, you should have the basic Draw.io Application navigation skills required to form your attack tree. If you want to get fancy, feel free to explore colors, fonts, and more through the customization sidebar that shows up on the right side of the interface when you select an arrow or box.

Fill In Your AttackTree

It's time to get to work!

1. Start by filling in your root node with your chosen attack scenario from the previous step.
2. Consider all possible paths that would lead to that scenario, layer by layer, taking one step back for each level you go down in the diagram of boxes. Don't stress too much about phrasing, the goal here is to capture as many options at each layer as possible. Fill in a numeric **identifier** in each box as you go such as "3. Steal email administrator credentials." Do not worry about the order of the numbers vs. attack tree layers. We will just be using the numbers as a simple way to reference and organize boxes in a future lab.
3. Use arrows to connect your attack tree. In Draw.io, you can easily label the arrow links themselves if desired by double clicking on them and filling in clarifying text, if necessary. You can link them pointing upwards toward the root, or flowing out of the root, whichever makes the most sense to you.

Once complete, you should have something that looks similar in complexity to below (or perhaps even more!) Here is an example diagram for an attacker gaining access to the CEO's email inbox. The red color was used to indicate the root node, and yellow was used to highlight a potential "easiest" path as assessed by the creator.



Note

This step does require some knowledge of potential attacks, if you feel you do not have enough experience in this realm to fully complete an attack tree, feel free to Google around, talk to the person next to you, a colleague, or even a penetration tester/red teamer you may know to solicit options for attack steps. Alternatively, keep your attack tree high-level and choose a scenario you do understand and fill in / edit out the infeasible paths with technical detail later.

Pause here in the instructions and complete your attack tree diagram for at least one scenario. If you finish quickly, feel free to use the template file to start again fresh and make a second tree or continue to fill out details in the one you have made. Continue on once you have one complete attack tree with 3-4 layers of detail.

Arrow Line Routing

If your arrow lines get tangled as you connect more and more boxes, you can click on the lines and adjust how they are routed by dragging the blue circles - this is called setting a "way point" for the line. If you continue to move boxes after setting a waypoint, however, the waypoints will remain, potentially making things worse. To return an arrow line to snapping back to its default path, right click on it and select `Clear Waypoints` or delete and remake the connection.

Hopefully you were able to fill out a multi-layer attack tree and have some new insight on how an attacker may view your environment. As the saying goes "[Defenders think in lists. Attackers think in graphs. As long as this is true, attackers win.](#)" What you have done is preemptively created a graph knowing with 100% certainty, an attacker will be doing the same, but you beat them to the punch, giving you a jump start on getting in their way!

Export Your Attack tree

Let's finish up by saving the attack tree to ensure we have it for later. In Draw.io go to "File" then "Save" or push `Ctrl + S`. This will (assuming you haven't changed it), save the file in `/home/labs/1.2/lab-1.2.drawio`.

Tip

If you'd like, you can also render the attack tree as an image, PDF, HTML and more by selecting `File > Export As` then selecting your preferred format.

Congratulations, you've now created a well-thought-out attack scenario complete with chronological steps that an attacker would have to progress through to complete it! Now what? You now have a list of the key places you need to monitor to ensure you are quickly alerted if an adversary were to start down the path. You also know where you can place preventative or mitigation controls to produce a strong, defense-in-depth position for your security team.

In a future exercise, we'll be using this chart and diving deeper into the specifics, but for now, we wanted to get you thinking about where your controls are now, and if they align with what you just made, or if perhaps you might be more blind than you thought to one of the "worst case scenarios"!

Taking It Further

Take this diagram back to your team and see what they think. Did you miss any obvious attack paths? Completeness is the goal, the more complete your vision the better you are prepared for an actual attack!

Exercise Conclusion -- Key Takeaways

In this exercise, you have:

- Learned the attack tree process and how it can benefit defensive planning
- Learned how to use the free draw.io program to produce an attack tree
- Created a carefully considered attack tree for a high-priority cyber attack scenario

Exercise 1.2 is now complete!

Exercise 1.3: Developing and Implementing SOC Playbooks

Background

In this exercise you will walk through the process of brainstorming, organizing, entering, and using a standard set of steps for a response playbook. For those who have never done this before, the goal is to walk through a representative shortened version of the process and see how it drives analyst actions. For those who have gone through this process before, pay particular attention to optimizing the steps of playbook selection, and how the demonstration incident response system used in this exercise (TheHive) differs in both positive and negative ways from your current solution.

Objectives

- Brainstorm commonly encountered scenarios that might be good candidates for playbook development
- Pick a scenario and develop a set of steps
- Label and group the developed steps into an organized process
- Set up the playbook in an incident management system

Exercise Preparation

This exercise is completed in your MGT551 Linux VM, if you have trouble with the setup, see the troubleshooting information in the wiki, or reach out to an instructor or SANS support.

1. Launch the MGT551 Linux VM and log in.

- LOGIN = `student`
- PASSWORD = `mgt551`

2. Before starting this exercise, you must start the required services. To do this, open a command terminal from the start bar.



Once the window is open, start the services by entering the following command at the command line:

Copy and Paste

You can mouse over the upper-right portion of the following text box and click the icon to automatically copy the commands to the clipboard! This method is highly recommended to error-proof your class experience. Be aware that using this method with multi-line commands like the one below will execute all commands as soon as you paste it to the command line.

```
cd /home/student/labs/1.3  
docker-compose up -d
```



```
cd /home/student/labs/1.3  
docker-compose up -d
```

You should see output similar to the following, the list order of container startup may vary. If you receive an error message inform your instructor or run the script from the "troubleshooting" page in the wiki.

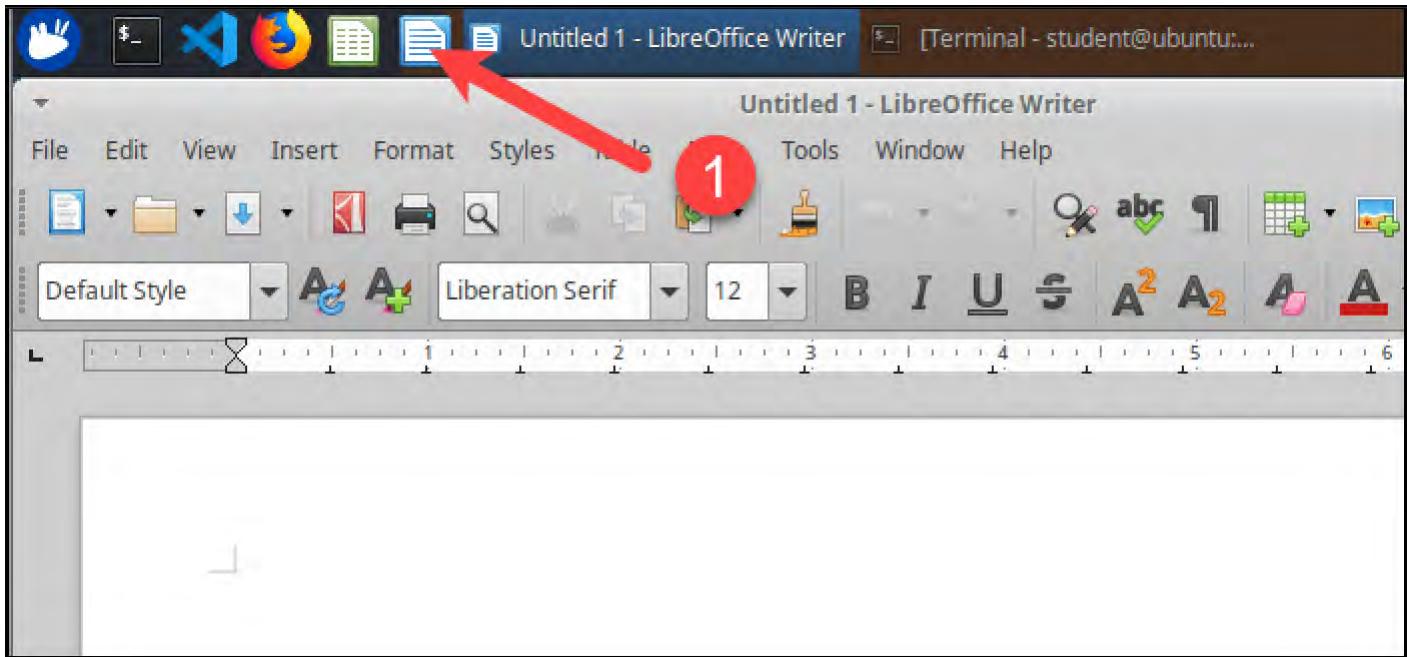
```
Creating network "13_default" with the default driver  
Creating 13_elasticsearch_1 ... done  
Creating 13_thehive_1      ... done
```

Keep the terminal open in the background, we will use it to shut these services down at the end of the lab.

Exercise Steps

Brainstorm Investigative Playbook Scenarios

Before beginning, open up the LibreOffice Writer text editor built into the Linux VM by clicking on the icon in the panel at the top of the screen:



In this first step, think of some of the most common scenarios for your SOC where analysts will be starting an investigation. This could be a common alert you see, a step in the kill chain, a MITRE ATT&CK Tactic or Technique, or any else of the sort. The goal will be to identify items from this list that can or should be converted into playbooks. If your SOC already has a set of playbooks for common scenarios, think of one that doesn't yet exist.

Take a minute or two and think of 5-10 of the scenarios your SOC most commonly experiences on a daily or weekly basis, especially those where a common set of steps for investigation or remediation is required.

(Write your list before continuing.)

If you need help, consider things like:

- Kill chain stages / MITRE ATT&CK Tactics and Techniques
- Server vs. Desktop vs. Cloud
- User types
- Locations (network subnets and physical)
- Blocked vs. detected vs. successful attack attempt
- Unidentified files/website access
- Unexpected user actions

An example of things you might think of could be the list below (feel free to use these too if you'd like). These are all scenarios that a SOC is likely to run into at some point and will be better off if a pre-determined set of response actions is created to ensure a complete investigation and response. Try to think of options you haven't yet covered before, in the next step, we'll be further defining one of them.

Examples:

- Attempted spear-phishing report
- A spam wave with a malicious attached file
- A spam wave with a malicious link
- Client-side browser exploitation attempt by a malicious website
- Service side exploitation attempt detected by HIDS / NIDS
- Successful running of an unsigned and unknown file
- Application control violation alert - user tried to run an unapproved program
- Alert from host-based antivirus of an identified virus
- Web traffic threat intelligence IOC match (domain / IP address)
- Active command and control protocol detected
- Unexpected addition of a user to an administrative group
- New auto-run item installed with suspicious path/filename
- Admin login attempt from unexpected machine/IP/location

Pick a Playbook to Write In Detail

Next, take a look at the list you've brainstormed and select a use case for further development and definition during this exercise. Pick a scenario that you know well that will be easy to work with, and ideally, one that you may have not defined in your SOC yet. (If you do not want to actually develop a new playbook for this exercise and rather follow along instead, you can simply read through the exercise and use the provided example). In this step, we'll be developing a comprehensive set of investigative options for analysts that encounter this scenario.

For an example to work with, we'll select the "**Phishing Email Wave Received**" scenario - the situation in which the SOC realizes there is a torrent of malicious email being sent to the organization that is not being automatically filtered.

Brainstorm investigative directions that could be followed to chase it down

Once you have that scenario selected it's time to get down to business and define the required actions. The goal in this task is to brainstorm all the possible investigative actions that could be taken, and eventually, which *should* be taken in response to this scenario. This can be a big task, so one easy way to start is to break it down into the different components of a complete response. One way I like to break steps down is into the following broad categories:

- Investigation - Understanding and identifying the important pieces of the attack. What are the IOCs, targeting, intended goals?
- Response - Preventing and remediation of damage. Depending on how far the attack got, this includes containment, then working to fix any issues that have already been caused

- Improvement / Lessons Learned - Taking the details of the attack and feeding it back to threat intel and the rest of the SOC so that it doesn't happen again.

Note

If you're thinking this looks very much like the incident response cycle, you're right. Using the model to break up playbooks steps into categories/response phases is exactly what the model is for.*

For our example use case (phishing wave), here are some of the items that might fall under each of these steps.

- Investigation
- Identify email targeting - Is there any pattern or reason to the user list that received the email? (Are they VIPs, admins, etc.?)
- Identify delivery details - Look at the source, subject, and contents of the email, did it pass anti-spoofing checks? (DKIM, SPF, DMARC), what was the source IP that sent it?
- What does the email try to persuade the user to do? Click a link? Open a file? Return sensitive data? If it is a file, what does that file do/install?
- Response
- Block the sender - Block the email sender by from address, source IP, subject, or any other commonality that can be found to contain the phishing wave from getting any bigger
- Remove the email from inboxes - If possible, request from the email team that the already delivered emails be removed from user inboxes so that they cannot fall victim to them.
- Block the link/attachment - If there is a link to click or file to block, put a prevention mechanism in place to stop users who do access the email from becoming infected (proxy block, DNS RPZ, firewall block to the IP, HIPS / EDR rule, etc.)
- Identify users who are already victims - Search for anyone who has visited the link using proxy/firewall / Zeek logs. Look for evidence in EDR / Sysmon data of any user that has already opened the attachment.
- Lessons Learned
- Improvements - Is there anything that can be better done in the future to prevent this situation from happening?

Ideally, our incident management system should accept each of these as either a required or optional step that must be completed to close any incident of the corresponding type, ensuring that analysts are being complete in their triage and analysis of the situation, as well as the remediation and feedback steps.

Review Steps and Edit as Necessary

At this point, you should have a list of possible investigative steps written down that you may want your analysts to take each time the chosen use case scenario is encountered. If necessary, go through your list and edit it down to the most important items and remove anything that may not be necessary or overly-detailed /specific. Remember, playbooks should be a general guide for actions to take, but not so specific that they become impossible to manage. Notice that the

items in the example above are at a high enough level that the specific procedure for completing them is not prescribed, but the goal is "block the link/attachment", "Identify users who are victims". Phrasing things this way helps give analysts the autonomy to select which method to accomplish the given task is the best for the situation. Where warranted, possibilities for methods can be given (proxy block, firewall, block, etc.), doing this helps newer analysts who might not know every option available to them while not forcing a particular method that may be inappropriate for a specific situation. In the next section, we'll be taking these steps and putting them into an example incident management system (TheHive) to illustrate how laying out playbook steps can help ensure a repeatable process across alerts of the same type.

Note

While we moved through this step quickly, when actually developing playbooks, you should take your time to develop a thorough list of all possible options and then select the best. How exactly? A suggested process for improving the quality of your playbooks is described in a 2020 research paper titled

"[Creative Choices: Developing a Theory of Divergence, Convergence, and Intuition in Security Analysts](#)" published by Chris Sanders and Stef Rand, both infosec experts with backgrounds in Psychology (excerpt below). Their research, aside from developing a very interesting theory of analyst's cognitive processes during alert investigation, suggests the following method for creating better playbooks:

Divergent and convergent thought exercises have great potential to benefit playbook development. As an example of how this may work within a given SOC, analysts can identify common investigative scenarios and create categories of investigations. From there, they would gather a series of examples (hypothetical or based on real cases) within each category. The analysts complete the same divergent and convergent exercises previously mentioned, then they collectively evaluate the utility of each investigative step and rank them accordingly. This ordinal list provides a basis for their playbook. A series of these playbooks reduces the reliance on intuition and offers high-quality, peer-reviewed investigation paths analysts can pursue based on the nature of the investigation scenario at hand. While prescriptive, these playbooks should not be so strict that they prevent analysts from manipulating and evaluating additional ideas outside the scope of the playbooks when warranted by the evidence.

Consider this another piece of required reading for SOC managers, and extra homework for this exercise. :)

Set Up a Playbook in an Incident Management System

In this step, we'll move from our playbook written down in a text editor to implementing it in a ticketing system. We'll use TheHive as representative IMS software in this step as it is an outstanding free and open-source incident management system. If you don't currently have a security-focused ticketing system, TheHive makes a great zero-cost choice. If you *do* already have an IMS, consider how TheHive works as we step through the rest of the exercise, and compare and contrast it to your own solution and look for any improvements to your current process that could be made.

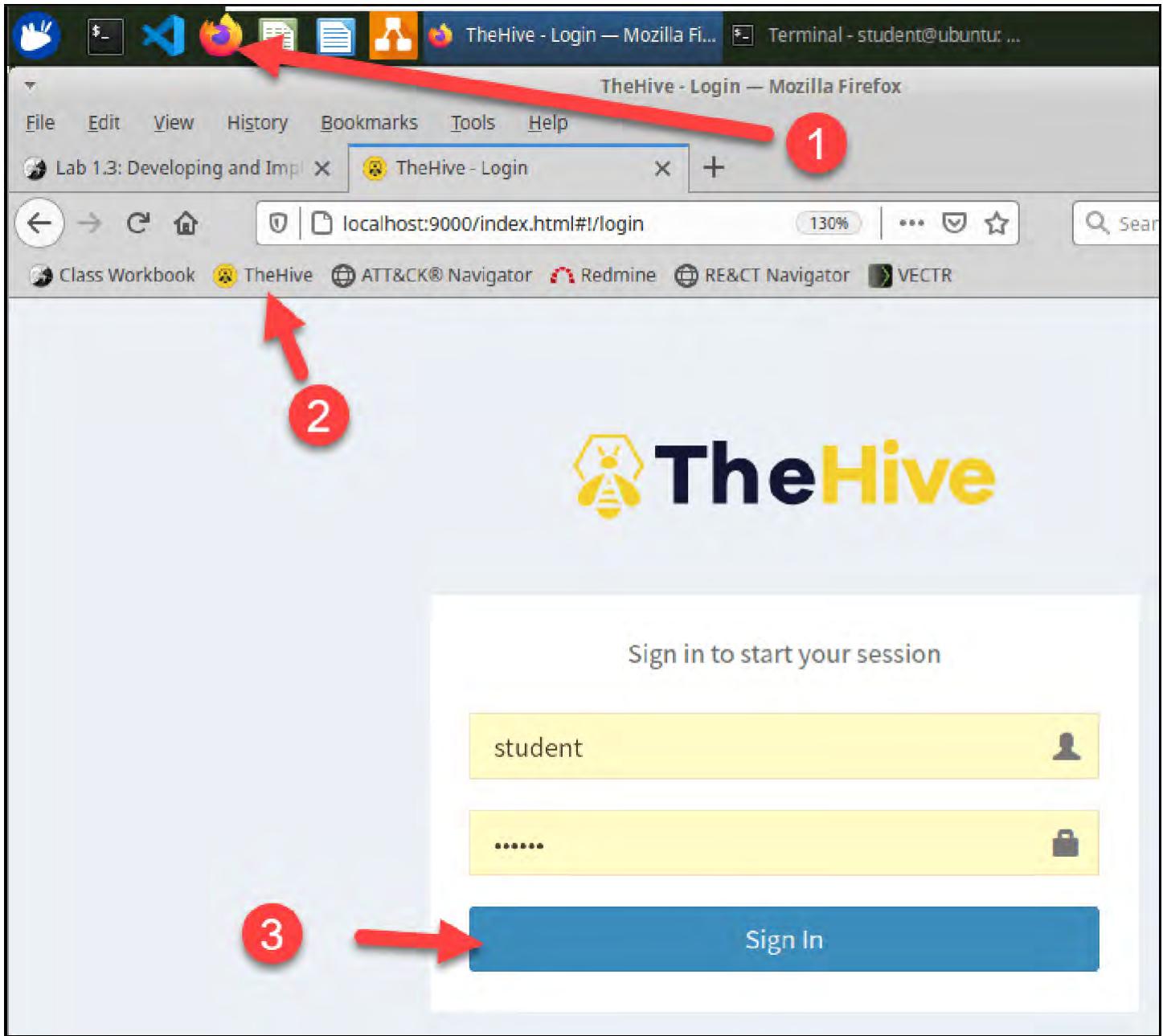
Explore Case Templates in TheHive

Note

Once you run the docker-compose command in the beginning of the exercise it may take a minute or two for TheHive's containers to be ready before the login page is functional. If you do not see the page in the following steps, wait a few moments before trying again. If the page never becomes available, run the troubleshooting script from the wiki and run the docker-compose command again, or inform your instructor.***

To get started with TheHive, open it up by opening a Firefox window in your Linux VM, then clicking on TheHive in the bookmarks toolbar or clicking [this link](#). You should see the login screen as shown below.

Enter the credentials username and password student / mgt551 to log in.



TheHive incident management system considers all accepted "alerts" as "cases". Each case can have a "case template" applied to it, which in this case is a set of "tasks" that must occur to close that case - a playbook in other words. For this step, we're going to copy and paste the steps we just wrote into a new case template in the admin interface of the hive so that it can be used for creating new cases. This is the activity you should be able to do with your own incident management software in your own SOC as it guides the workflow of analysts for each specific use case scenario.

To create a "case template" click on the Admin link in the upper right corner of TheHive's web interface:

A screenshot of the SOC247 web application interface. At the top, there is a navigation bar with a search bar, a dropdown for 'CaseId', and an 'Admin' button. To the right of the Admin button is a user profile icon labeled 'student'. A red arrow points from the 'Admin' button to the 'Admin' button in the top right corner of the main content area. Inside this content area, there is a sidebar with a 'Users' section and a list of management options: 'Case templates', 'Report templates', 'Case metrics', 'Observables', and 'Case custom fields'. A red circle with the number '1' is placed over the 'Users' section, and a red circle with the number '2' is placed over the 'Case templates' option.

You should now see the case template management page.

A screenshot of the 'Case template management' page. On the left, there is a sidebar with buttons for 'New template' (highlighted in blue) and 'Import template'. Below these are sections for 'Current templates' containing 'Virus/Malware Execution Detected' and 'Recon/Scan Activity Identified'. On the right, there is a form for creating a new template. It includes fields for 'Template name *' (set to 'Virus/Malware Execution Detected') and 'Title prefix' (set to 'MALWARE'). A red arrow points from the 'Case template management' title in the sidebar to the 'Case basic information' section on the right.

Investigate Pre-Existing Case Templates

In TheHive, some pre-staged case templates have already been entered as an example for you as shown in the photo below. Click on one of the templates such as the **Phishing Email/Wave Received** item as discussed in the previous example.

The screenshot shows the 'Case template management' section on the left and a 'Playbook steps' section on the right.

Case template management:

- + New template
- Import template

Current templates:

- Virus/Malware Execution Detected
- Recon/Scan Activity Identified
- Exploit Attempt Detected
- Phishing Email/Wave Received** (highlighted with a red box and a red arrow pointing up from the 'Playbook' label)
- Command and Control

Playbook steps:

Category	Playbook steps
Tasks (8)	<ul style="list-style-type: none">[Investigation] Identify email targeting[Investigation] Identify Delivery Details[Investigation] Identify the exploitation method[Investigation] What was the next step?[Response] Block sender by email or IP[Response] Remove email from user's inboxes[Response] Block link or file attachment[Continuous Improvement] Lessons Learned / Improvements

Note that the steps the analyst must take are organized under "Tasks" and the category I have chosen to assign (an optional feature in TheHive) the example tasks to is listed in square brackets before the task. Tasks have an order in which they are presented to the analyst that can be rearranged, although working them in this order is not forced.

In the middle section TheHive allows you to specify other details about the playbook (use case template):

- A case template name
- A prefix that will be used on all cases of that type
- Defaults for case severity, TLP, PAP, and Tags to attach
- A description of the playbook

Case basic information

Template name *	Phishing Email/Wave Received	Tasks (8)
This name should be unique		
Title prefix	PHISHING	
This is used to prefix the case name		
Severity	M	
This will be the default case severity		
TLP	TLP:AMBER	
This will be the default case TLP		
PAP	PAP:AMBER	
This will be the default case PAP		
Tags	Tags	Metrics (0)
These will be the default case tags		
Description *	This template is for alerts regarding investigation and response to single or multiple phishing emails being received.	
Custom fields (0)		

Rearrange tasks

Defaults

Click through some of the pre-made case templates and investigate the steps. If you'd like to see the description of a specific step, click the Edit but on the bar for that task:

Tasks (8)	
	[Investigation] Identify email targeting
	[Investigation] Identify Delivery Details
	[Investigation] Identify the exploitation method

This shows the setup details available for each step as well as the categorization in the "Task Group" field (which is optional). Analysts who assign each of these tasks to themselves as part of working any given case will see this description which can provide the goals and details of what to do to complete it, and will be given an interface to write notes on the completion of the task.

The screenshot shows a 'Update task' window with the following fields:

- Task title ***: Identify email targeting
- Task group ***: Investigation
- Task description**:
Consider the recipients of the email, is there any pattern or apparent targeting that gives a clue of how the spammers made their target list? (All C-suite members, accounting group, all emails start with "A", people that were in a recent breach, list of employees on linked/other site)
- Assignee**: (empty dropdown)

At the bottom are 'Cancel' and 'Update task' buttons.

Click cancel to close the Update Task window and return to the case template management page. We will now enter our own use case as a new item into TheHive.

Enter Your New Case Template

At the top left of the case template management window, click the "New Template" button to start the creation of a new case template.

The screenshot shows a software interface for managing case templates. On the left, there's a sidebar titled "Case template management" with two buttons: "New template" (highlighted by a red arrow) and "Import template". Below these are sections for "Current templates" and a specific entry for "Virus/Malware Execution Detected". The main area is titled "Case basic information" and contains fields for "Template name *" (with "Phi" typed in), "Title prefix" (with "PHI" typed in), and "Description" (with "This na" visible). There are also dropdown menus for "Severity" and "TLP".

Case template management

+ New template

Import template

Current templates

Virus/Malware Execution
Detected

Case basic information

Template name *

Title prefix

Phi

This na

Severity

TLP

You now see an empty screen where you can enter the details of the use case you've noted in LibreOffice Writer.

First, fill in the case template name and Description (required). If desired, set the optional title prefix and default severity, TLP, and PAP.

Case basic information

Template name *

Template name

This name should be unique

Title prefix

Case title prefix

This is used to prefix the case name

Severity

M

This will be the default case severity

TLP

TLP:AMBER

This will be the default case TLP

PAP

PAP:AMBER

This will be the default case PAP

Tags

Tags

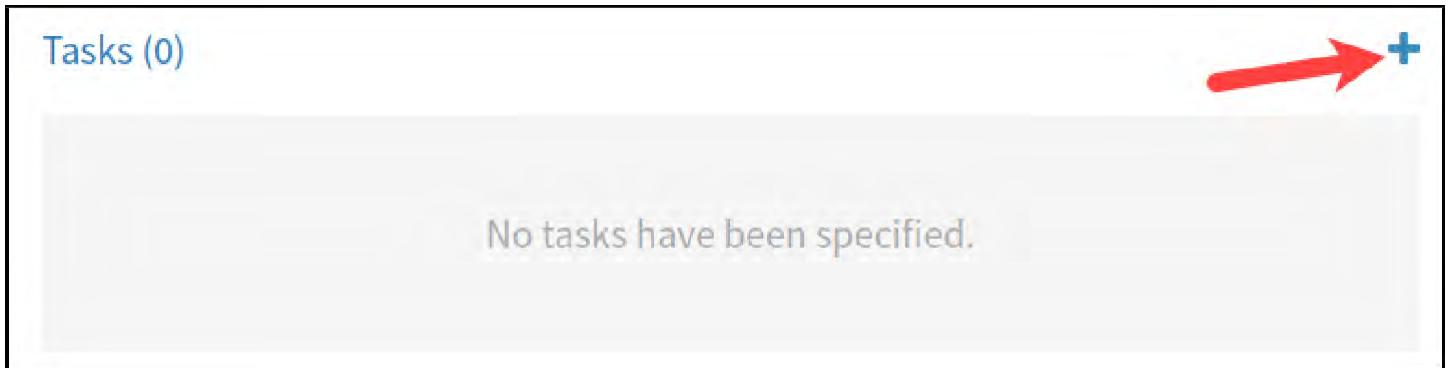
These will be the default case tags

Description *

Case description

Optional

Next, it's time to enter your playbook steps. To create a new task in the case template, click the plus button in the upper right of the screen under the Tasks heading.



You should see the "add task" screen as shown below. Type in the title of the task, a group name (if desired, "default" can also be selected if your tasks aren't categorized) and fill in a short description of the task then save it by hitting "add task".

A screenshot of the "Add task" form. The title is "Add task".

- Step 1: A red circle with the number 1 has a red arrow pointing to the "Task title *" input field, which contains "New Task Here".
- Step 2: A red circle with the number 2 has a red arrow pointing to the "Task group *" input field, which contains "task_group".
- Step 3: A red circle with the number 3 has a red arrow pointing to the "Task description" text area, which contains "A description of what to do for this task." Below it, a smaller text box says "Task's default description".
- Step 4: A red circle with the number 4 has a red arrow pointing to the "Add task" button at the bottom right of the form.

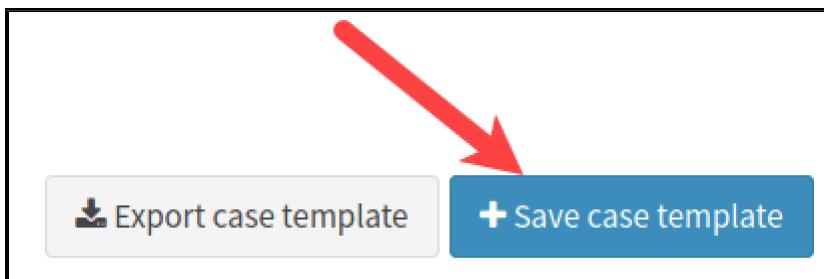
After you hit "add task" you should be taken back to the case template management screen and should see your newly created task added to the list on the right side of the screen.

Tasks (1)

[task_group] New Task Here

Edit Delete

Repeat this process by copying and pasting your playbook step details in until all steps are entered, and they are in the order you desire. Once you are done, or at any time in the process you can save your work by pressing the "Save case template" button as shown below. If the button is not active, it's because you haven't type in a required field on this page, such as the template name or description.



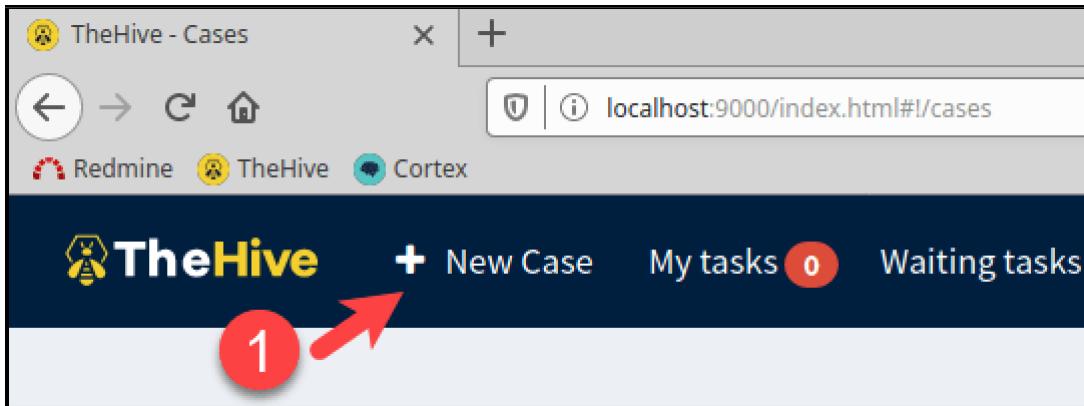
Once all of your steps are entered, press the "save case template" button - your playbook is now ready for use!

Use Your New Playbook

In this final step, you'll see how analysts will experience when a new case is created with your new playbook using TheHive IMS. This will give you an idea of how your newly created playbook should be used to guide, but not restrain analyst investigation.

Create and Walk Through A New Case**

To see the case template (playbook) we have just created, you must first create a new case. The normal workflow for TheHive is to use the "Alerts" dashboard to accept an alert as a case, but since we don't have any alerts staged we'll just make one manually. To do so, click on the "New Case" button on the top bar of the interface in TheHive.



You will be presented with the "Create new case" screen. This screen asks you to choose a case template or to pick an "empty case" if none of the pre-made playbooks cover the current situation. Select your new playbook by clicking on it or one of the other already defined cases.

A screenshot of a "Create new Case" dialog box. At the top is a blue header bar with the text "Create new Case". Below it is a large blue button labeled "Empty Case". To its right is the word "OR". Below "OR" is a section titled "Select a template" with a "Filter templates" search bar. On the right side of this section is a vertical scroll bar. The main content area lists several case templates:

- My_case_template**
For demonstration purposes.
- Virus/Malware Execution Detected**
This template is for dealing with active or recently detected viruses.
- Recon/Scan Activity Identified**
This template is for investigation of identified reconnaissance activity.
- Exploit Attempt Detected**
The template is for investigation of attempted exploitation of a host.
- Phishing Email/Wave Received** ←
This template is for alerts regarding investigation and response to single or multiple phishing emails being received.

At the bottom left is a "Cancel" button.

Once your case template is selected, the "Create a new case" window will ask you to fill in the specific case details. This includes the title as a requirement only as tasks have already been defined and defaults for description, severity, TLP, and PAP are already selected.

Fill in the Title with any name you'd like - in the example here I've chosen "Invoice phishing with macro". The Case Tasks area should show the steps from your playbook you've defined for this template showing that your playbook will be used to investigate this case.

When you're ready, hit the "Create Case" button to create your new case.

Create a new case

Case details

Title * PHISHING **"Invoice" phishing wave with macro** **Date *** 06-04-2020 14:02 **now**

Severity * L M H **TLP *** WHITE GREEN AMBER RED

Tags Tags **Description *** This template is for alerts regarding investigation and response to single or multiple phishing emails being received.

PAP * WHITE GREEN AMBER RED

Case tasks (from [Phishing Email/Wave Received] template)

- Identify email targeting
- Identify Delivery Details
- Identify the exploitation method
- What was the next step?
- Block sender by email or IP
- Remove email from user's inboxes
- Block link or file attachment
- Lessons Learned / Improvements

Case metrics (from [Phishing Email/Wave Received] template)

No metrics have been specified

1

2

Create case

Cancel *** Required field**

© 2021 John Hubbard and Mark Orlando

You should now see your newly created case and the details you selected as shown in the photo below. This is the case overview page in TheHive. If you were an analyst creating this case to work on it, the next steps would be moving to the pre-defined playbook tasks and starting to work on them individually.

To see the playbook tasks you've defined, click on the "Tasks" tab.

M Case # 3 - PHISHING "Invoice" phishing wave with macro

Created by student Mon, Apr 6th, 2020 14:03 -07:00

Details Tasks 8 Observables 0

Summary

Title	PHISHING "Invoice" phishing wave with macro
Severity	M
TLP	TLP:AMBER
PAP	PAP:AMBER
Assignee	student
Date	Mon, Apr 6th, 2020 14:03 -07:00
Tags	Not Specified

Additional information

No additional information have been specified

Description

This template is for alerts regarding investigation and response to single or multiple phishing emails being received.

In this view, as shown below, you'll see each task you've defined in your case template, the group (if used) that it is associated with, and a button to start working on that task. The idea is that analysts must start and work each task through in any order they deem appropriate, once complete, the case itself is complete, and can be closed.

To see how this works, click "Start" next to one of your tasks. You will be taken to a new tab for that task as shown below.

The screenshot shows a screenshot of the TheHive interface. At the top, there's a header bar with a yellow 'M' icon, the text 'Case # 3 - PHISHING "Invoice" phishing wave with macro', and a 'Show live stream' button. Below the header, it says 'Created by student' and 'Mon, Apr 6th, 2020 14:03 -07:00'. On the right, there are buttons for 'Close', 'Flag', 'Merge', and 'Remove'. Underneath the header, there are tabs for 'Details', 'Tasks' (with a count of 8), and 'Observables'. The 'Tasks' tab is selected. In the center, there's a table with columns: 'Group', 'Task', 'Assignee', and 'Actions'. A red box highlights the 'Add Task' button in the top left of the table area. A red arrow points from a circled '1' to the 'Start' button for the first task in the list. The table data is as follows:

Group	Task	Assignee	Actions
Investigation	Identify email targeting	Not assigned	
Investigation	Identify Delivery Details	Not assigned	
Investigation	Identify the exploitation method	Not assigned	
Investigation	What was the next step?	Not assigned	
Response	Block sender by email or IP	Not assigned	
Response	Remove email from user's inboxes	Not assigned	
Response	Block link or file attachment	Not assigned	
Continuous Improvement	Lessons Learned / Improvements	Not assigned	

The screen you are brought to is the tab for the individual task that you have clicked to start working on. Each task can have individual "task logs" added that correspond to any important information created while performing that task. Feel free to use the rich text editor to try out the task log function in TheHive.

When done exploring this screen, click on the "Close" button as shown in the picture below but beware, there are two close buttons on the screen that look identical. One close button is on the Case section of the UI, the other is under the task tab. Use the close button for the task, as highlighted in the picture to close only this specific task, we do not want to close the whole case yet.

M Case # 3 - PHISHING "Invoice" phishing wave with macro

Created by student Mon, Apr 6th, 2020 14:03 -07:00

Do NOT close case

Close Flag Merge Remove

Details Tasks 8 Observables 0 **Identify email targeting**

Basic Information

Task tab

Title	Identify email targeting	Date	Mon, Apr 6th, 2020 14:07 -07:00
Group	Investigation	Duration	Started a minute ago
Assignee	student	Status	InProgress

Description

Consider the recipients of the email, is there any pattern or apparent targeting that gives a clue of how the spammers made their target list? (All C-suite members, accounting group, all emails start with "A", people that were in a recent breach, list of employees on linked/other site)

Add notes if desired

+ Add new task log Sort by: Newest first 10 per page

You will be brought back to the task list page where the task you had open will now show a green checkmark and a timestamp showing when it was closed, and the duration it took to complete.

Group	Task	Date	Assignee	Actions
✓ ✓ Investigation	Identify email targeting Closed after 4 minutes	Mon, Apr 6th, 2020 14:07 -07:00	student	<input checked="" type="checkbox"/> Reopen
✓ Investigation	Identify Identiy Delivery Details		Not assigned	▶ Start
✓ Investigation	Identify the exploitation method		Not assigned	▶ Start
✓ Investigation	What was the next step?		Not assigned	▶ Start
✓ Response	Block sender by email or IP		Not assigned	▶ Start
✓ Response	Remove email from user's inboxes		Not assigned	▶ Start
✓ Response	Block link or file attachment		Not assigned	▶ Start
✓ Continuous Improvement	Lessons Learned / Improvements		Not assigned	▶ Start

On this screen, the workflow for an analyst would then be walk through each task and complete them all. Once all tasks are complete, the case could then be closed.

You've now seen how TheHive uses pre-defined playbook steps to steer analyst investigative action without forcing the completion, giving them the flexibility to complete each task as appropriate.

Close Your Case

To finish out our example case, click through your tasks and close each one so that all are complete (alternatively, you can leave them open and click through a warning box in the next step.)

You should now see a "completed" case with all tasks finished as shown in the picture below.

The screenshot shows the TheHive interface for a case titled "Case # 3 - PHISHING "Invoice" phishing wave with macro". The top navigation bar includes "Show live stream", "Created by student" (Mon, Apr 6th, 2020 14:03 -07:00), and buttons for "Close", "Flag", "Merge", and "Remove". Below the header, there are tabs for "Details", "Tasks" (with a count of 8), and "Observables" (with a count of 0). A red arrow points to the "Close" button, which is highlighted with a red box and has a red number "1" above it. The main content area displays a table of tasks:

Group	Task	Date	Assignee	Actions
Investigation	Identify email targeting Closed after 4 minutes	Mon, Apr 6th, 2020 14:07 -07:00	student	<input checked="" type="button"/> Reopen
Investigation	Identify Delivery Details Closed after a few seconds	Mon, Apr 6th, 2020 14:11 -07:00	student	<input checked="" type="button"/> Reopen
Investigation	Identify the exploitation method Closed after a few seconds	Tue, Apr 7th, 2020 4:06 -07:00	student	<input checked="" type="button"/> Reopen
Investigation	What was the next step? Closed after a few seconds	Tue, Apr 7th, 2020 4:09 -07:00	student	<input checked="" type="button"/> Reopen
Response	Block sender by email or IP Closed after a few seconds	Tue, Apr 7th, 2020 4:09 -07:00	student	<input checked="" type="button"/> Reopen
Response	Remove email from user's inboxes Closed after a few seconds	Tue, Apr 7th, 2020 4:09 -07:00	student	<input checked="" type="button"/> Reopen
Response	Block link or file attachment Closed after a few seconds	Tue, Apr 7th, 2020 4:09 -07:00	student	<input checked="" type="button"/> Reopen
Continuous Improvement	Lessons Learned / Improvements Closed after a few seconds	Tue, Apr 7th, 2020 4:09 -07:00	student	<input checked="" type="button"/> Reopen

Click the "Close" button.

You will be presented with the "Close Case" window as shown below.

Close Case #3

! You are about to close Case #3. Are you sure you want to continue ?

Incident				
Status *	True Positive	False Positive	Indeterminate	
	Other	<input type="checkbox"/> Investigation clearly demonstrates that there is something malicious (scam, phishing, malspam, malware, cybersquatting...)		
Impact *	Yes	No	<input type="checkbox"/> Security measures blocked the attack or infection	
Summary *	<p>This is where the case summary goes. Fill out bullet point details on the action and impact of the case to make reviewing the case later fast and efficient.</p>			

Cancel *** Required field** **Close case**

Fill in an example status (true/false positive), select an Impact, and fill in some text for the Summary. As shown, the summary of a case should give those who visit it later a quick overview of what occurred and the impact it had.

Click "Close case" once complete to close your case. You have now worked through the whole playbook and task completion workflow inTheHive!

Exercise Conclusion – Key Takeaways

In this exercise, you have:

- Brainstormed a list of scenarios that are common enough to warrant playbook development
- Selected a playbook for development and implementation
- Considered all the actions that are appropriate to consider taking when working through the scenario described by your playbook

- Been given a research-driven reference for high-quality playbook investigation step development
- Implemented your playbook in an incident management system
- Stepped through a workflow similar to what your own incident management system should provide for analysts using your playbook

To shut down the services used for this exercise go back to your terminal window (or open a new one) and enter the commands below:

```
cd /home/student/labs/1.3
docker-compose down
```

You should see a response similar to the following, if you do not, please alert your instructor:

```
$ docker-compose down
Stopping 13_thehive_1 ... done
Stopping 13_elasticsearch_1 ... done
Removing 13_thehive_1 ... done
Removing 13_elasticsearch_1 ... done
Removing network 13_default
```

Exercise 1.3 is now complete!

Exercise 2.1: Attack Path and Data Source Assessment

Background

The next step in building our threat-informed defense is converting our attacker intelligence and assumed attacks (attack trees) into a plan to detect progression toward those goals, and that is what we will do in this lab. This exercise will continue to develop our security monitoring strategy leveraging the threat intelligence we found in exercise 1.1, and the attack tree you built in exercise 1.2. We will take the next step and analyze the attack tree steps in more detail, grouping and categorizing the items into themes and techniques. From these groupings, you may start to see tactics in common among multiple branches, highlighting detection capabilities that need to be prioritized for your team. You will also analyze the data required to spot these attacks to find any critical visibility gaps your team may have that relate to those worst-case cyber attack scenarios.

Objectives

- Categorize attacker actions common to your high-impact cyber attack scenarios
- Create a priority list of detection capabilities for your SOC
- Learn how to assess your monitoring capabilities for high-risk cyber events
- Assess your host and network data collection strategy
- Evaluate your security visibility in a threat-intelligence driven manner

Exercise Preparation

This exercise is completed in your MGT551 Linux VM, if you have trouble with the setup, see the troubleshooting information in the wiki, or reach out to an instructor or SANS support.

Launch the MGT551 Linux VM and log in.

```
- LOGIN = `student`  
- PASSWORD = `mgt551`
```

Recall your attack tree diagram for yesterday's lab. If you did your exercise using the virtual machine local version of Draw.io, you can double click the `lab-2.1.draw.io` shortcut on your desktop to open the file in the Draw.io desktop application.

Exercise Steps

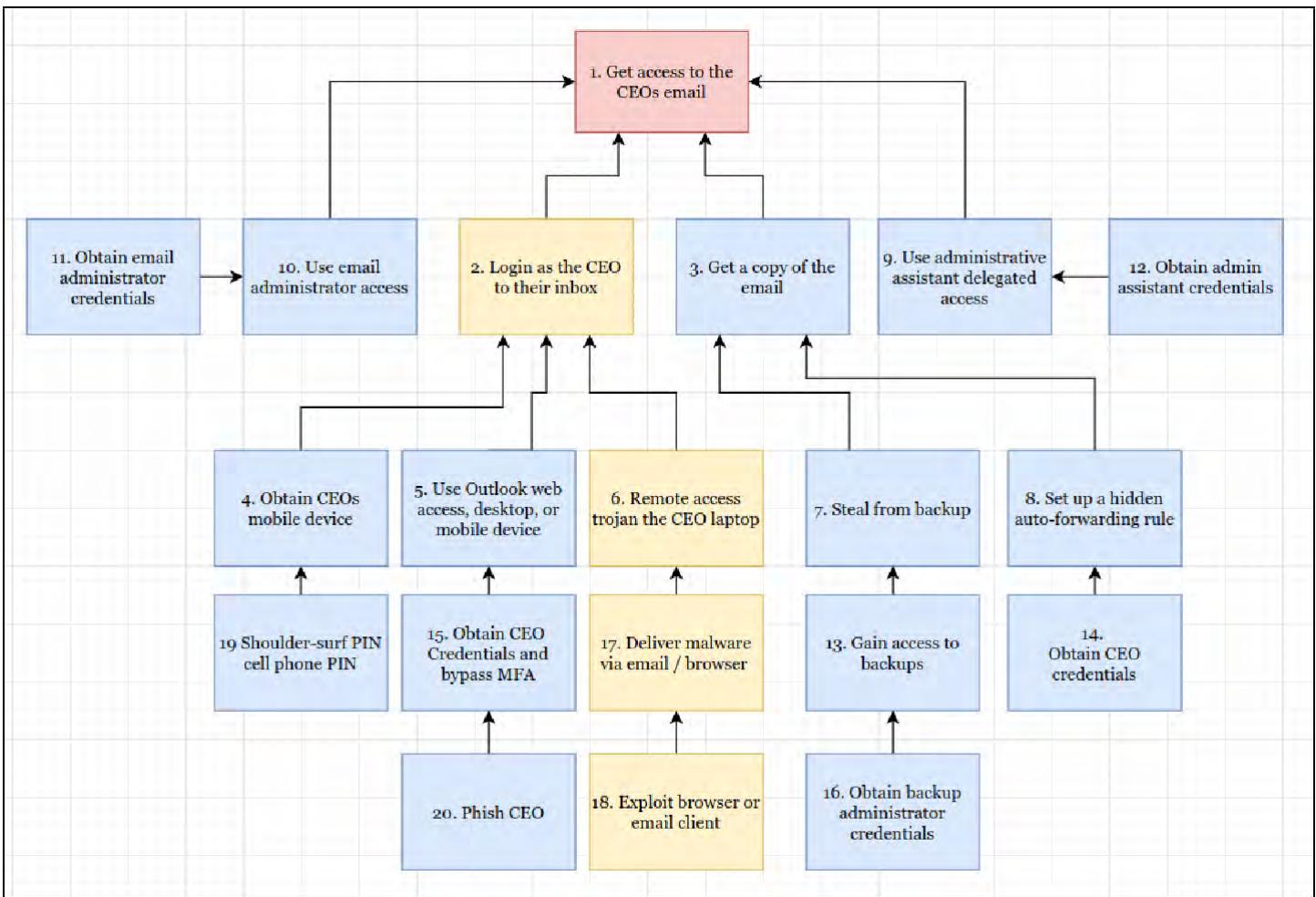
Before you get started: It may be useful to bring up the most recent [MITRE ATT&CK Matrix](#) in a browser for this lab. We will be looking at attack tactics and techniques, and the ATT&CK matrix lists important data sources on each tactic page. You may find this useful if you aren't familiar with the technical aspects of detecting certain attacks.

Organize Attack Steps

Organization and Motivation

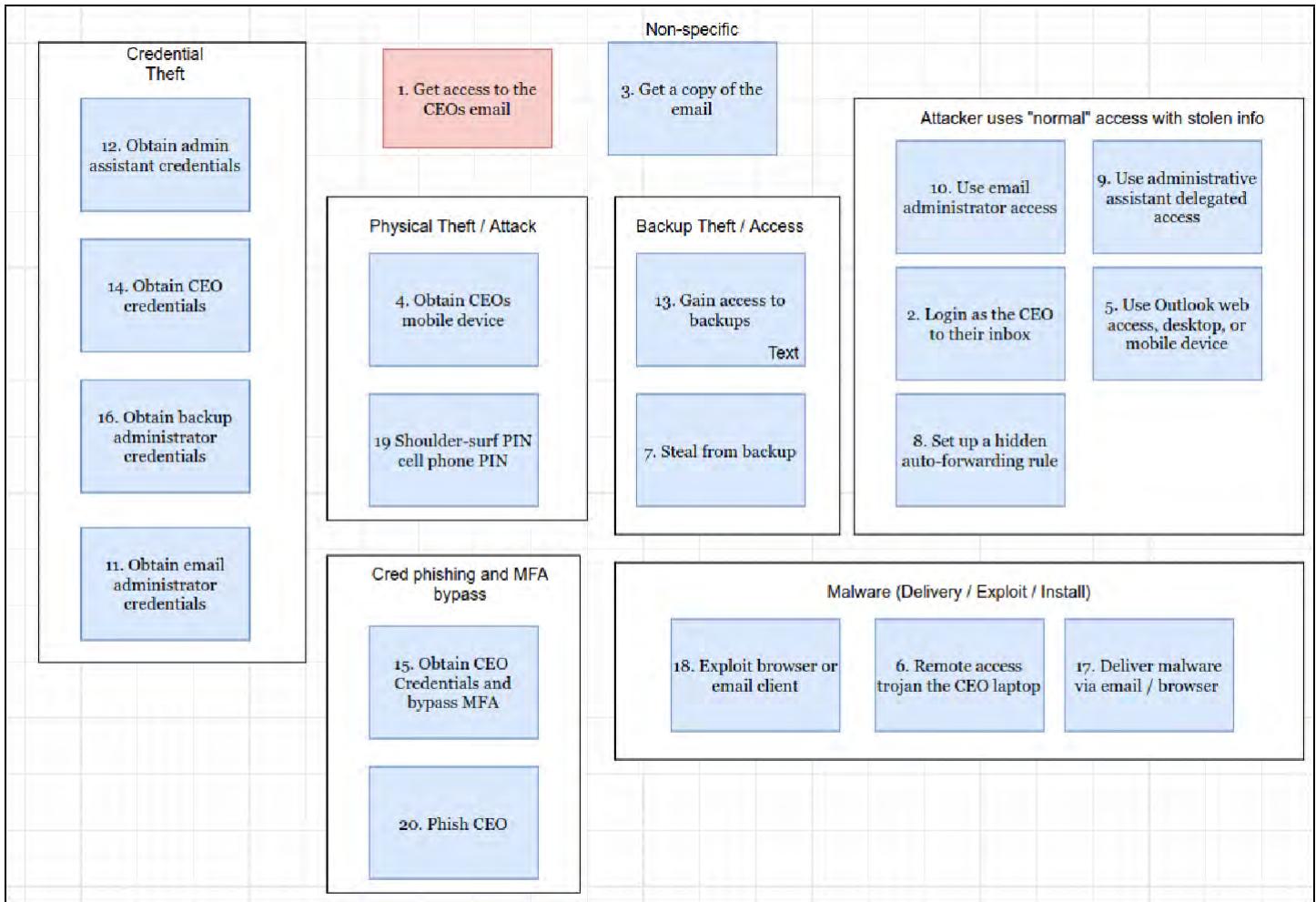
In this first step, we'll deconstruct the attack tree diagram you previously made, and group the items within it not based on attack chains but based on common themes such as kill chain steps, attack techniques, or other similar characteristics. You may have noticed in the example attack tree presented in exercise 1.2 (shown again below) that there are multiple boxes that involve credential theft with the only difference being the account involved. This is a quite common occurrence when creating attack trees; when using them for understanding mitigation and visibility, grouping similar items together first enables us to be more efficient in our analysis.

As an example, here is the given attack tree from exercise 1.2 for a scenario of stealing the CEOs email:



Looking closely at the image, you'll see 11, 12, 14, and 16 all involve obtaining credentials of someone in the organization. If we're trying to understand if we could catch these attack chains, the method as to how you would log and detect admin credentials being abused would likely be very similar to finding backup administrator credentials being used. If you can catch it for one account, you can probably catch the others as well. These are the types of similarities we're going to be looking for.

Here's an example of the output to expect after doing this - If you were to take the example attack tree file created in exercise 1.2, delete all of the arrow connections and rearrange the boxes into groups (and label the groups by dragging additional boxes and text to put them in), you might end with something like this:



Organizing Your Own Diagram

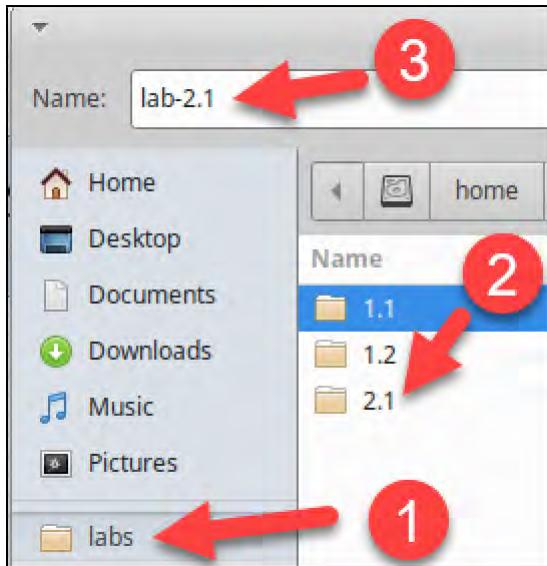
In this step, the goal is to take *your* attack tree, and group similar items together, looking for common themes tied together by kill chain step, logging or detection capabilities, or anything else that helps you understand if that step is covered or not within your own environment.

Tip

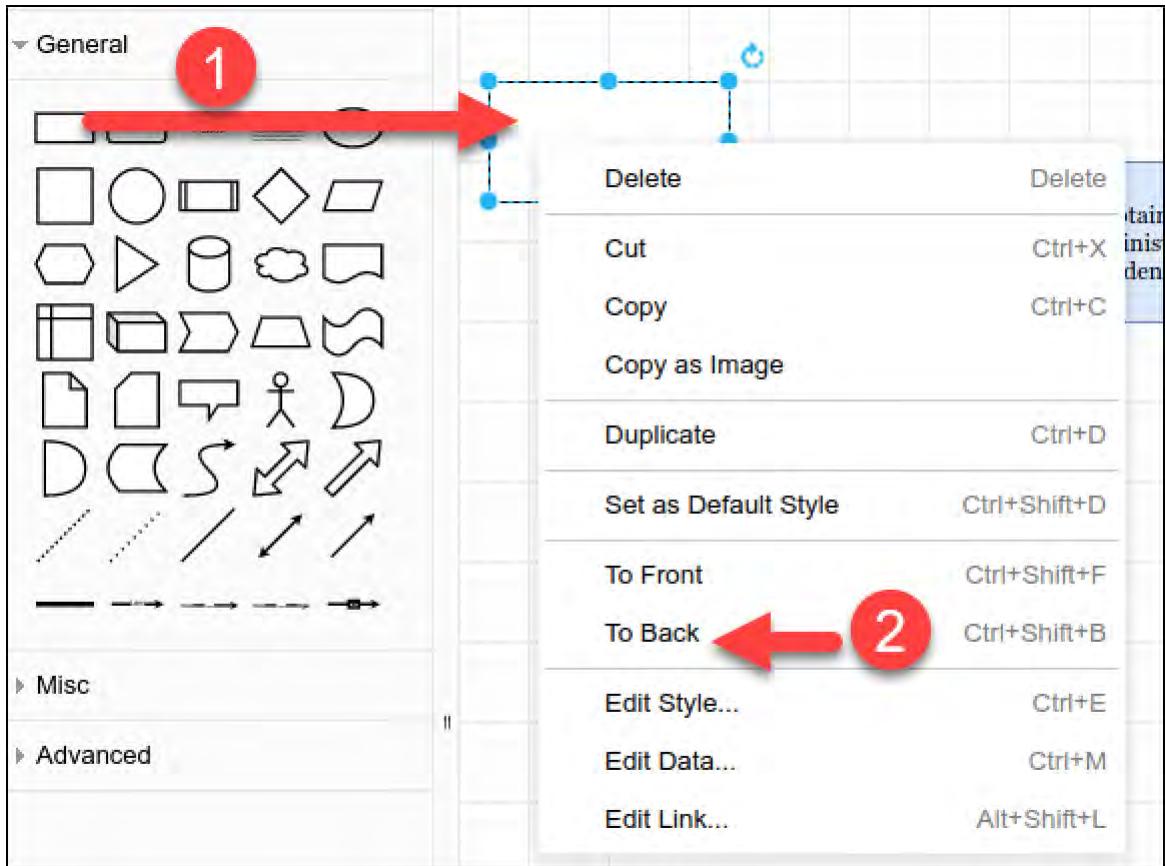
If you created your diagram on your own software - use whatever features are available in that software to group your attack tree steps together and make a label or theme for each group.

If you created your diagram using Draw.io (either in the VM or on the website), proceed by doing the following:

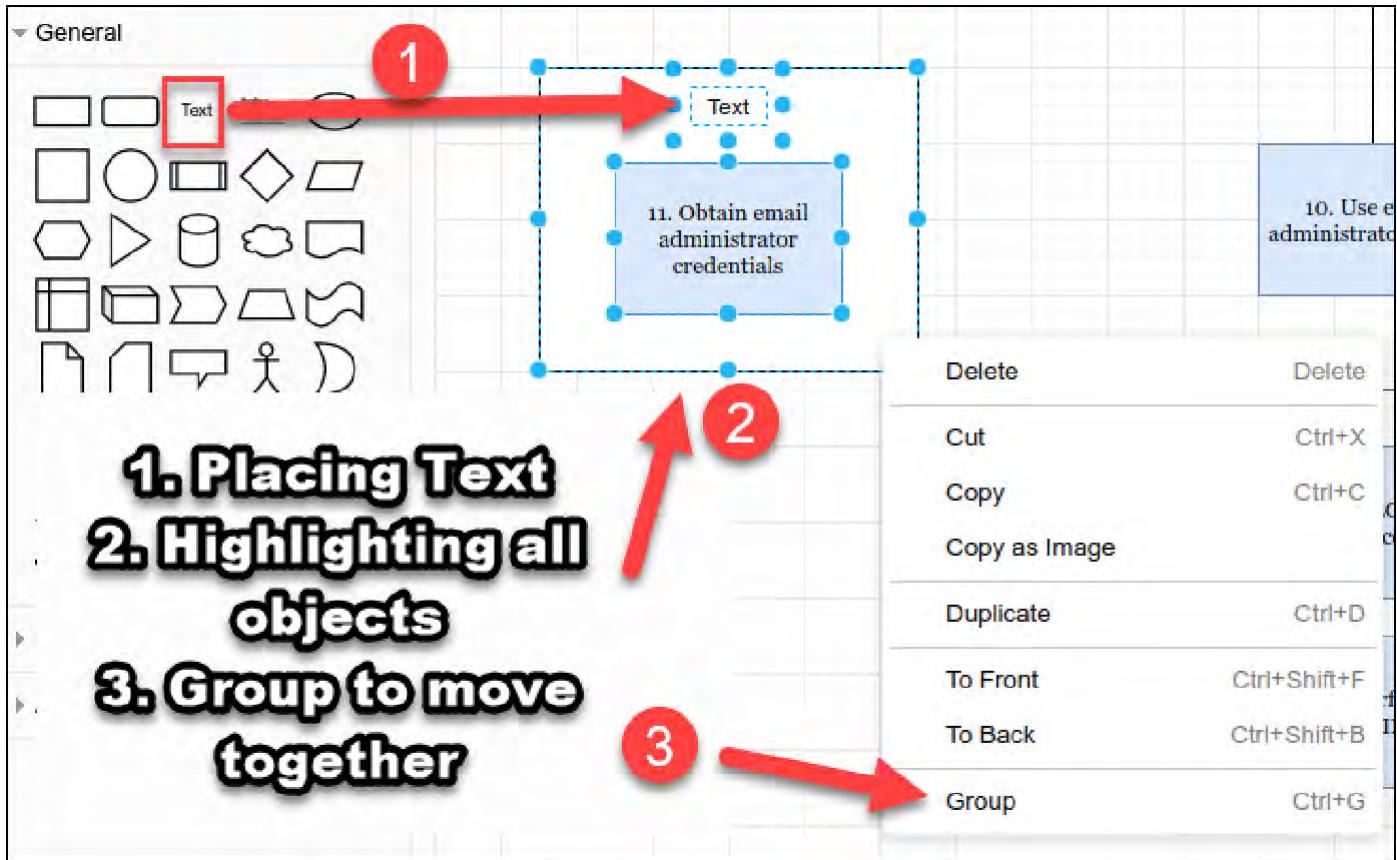
- Open up your draw.io diagram from exercise 1.2 (using the link on the desktop, assuming you saved the diagram to the same filename `lab-1.2.drawio`)
- Go to File > Save As and navigate to `/home/student/labs/2.1` and save a copy of the diagram as `lab2.1.drawio`, this ensures you're working with a copy and won't be destroying your output from exercise 1.2



- Delete all the arrows on the diagram so that you can move the boxes freely around the screen
- Sort and group boxes by any themes that you find, if you are unsure of how to group them, don't worry, there may not be a perfect answer, just do your best or ask an instructor for guidance
- If you'd like to collect all the boxes inside another box like the previous screenshot, click and drag a rectangle from the General pane in the left sidebar and drop it on the work area. Then, select "To Back" so that it falls behind your previously created boxes.



- Once you've resized the box, add a label by dragging the "Text" object off the left side of the screen and placing it inside the box as well, along with any attack tree steps that belong inside. - If you want to make moving multiple items together easy, highlight all the objects at once by clicking and dragging a box completely around everything you'd like to select, right-click on the objects, then select Group.



It may take you a little while to adjust boxes and find what the themes are and where they fit best. You'll know you have it close enough when you can fairly easily put a label on the box that makes sense with the grouped items.

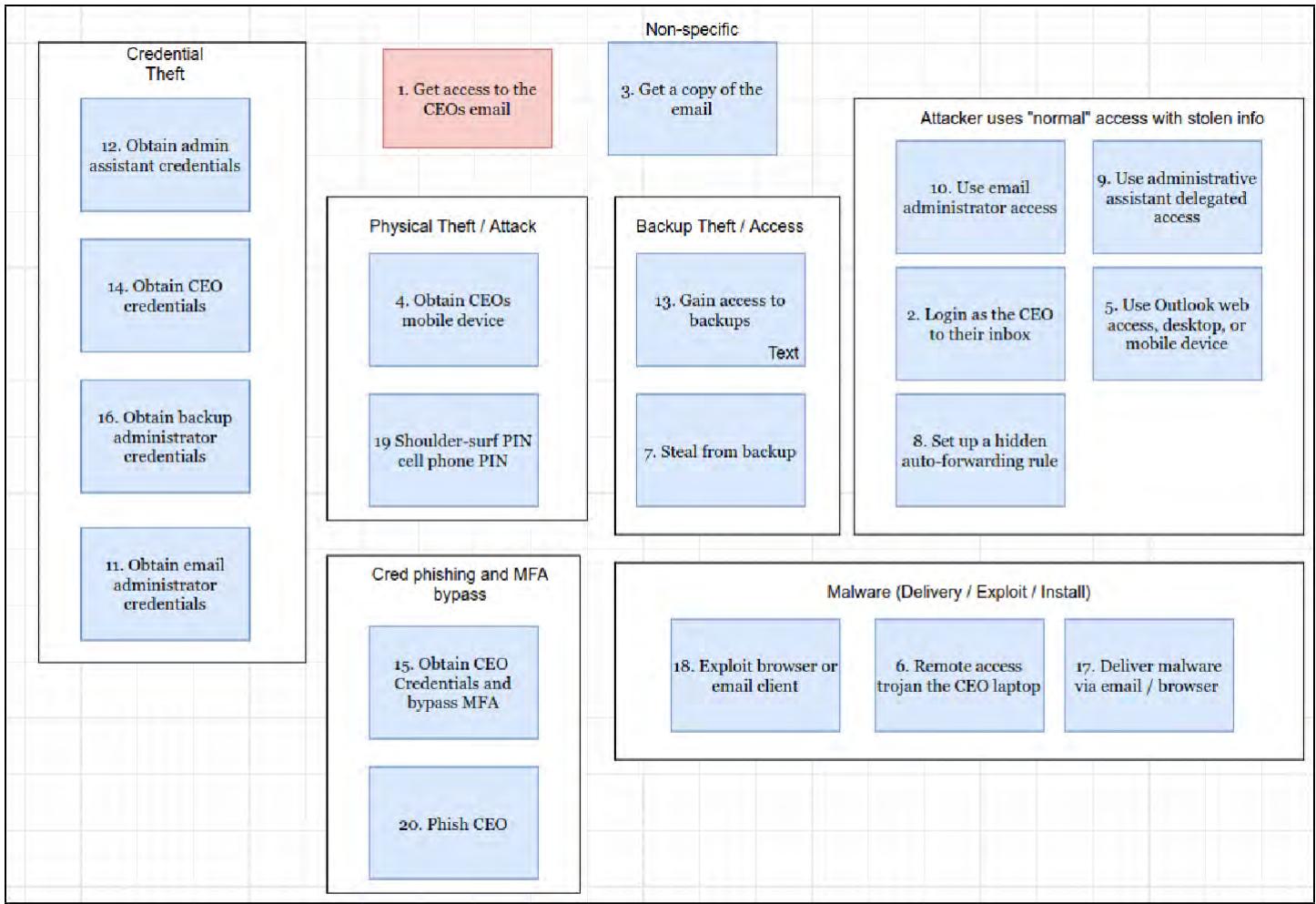
Go ahead and take a moment now to organize your attack tree in this manner, then proceed once you have groups and labels for each.

Mitigation and Data Source Assessment

In this section we'll take the groups we've created and ask ourselves several important questions. This will help us assess whether such a step is possible in our environment, how we might detect it, and what data sources specifically make it visible.

Converting the Diagram to **Mitigations**, **Data sources**, and **More**

Take a look again at the example diagram after grouping...



Here are some questions that came to the author's mind based on this scenario:

- "What process would detect or prevent someone from gaining access to our backups, or stealing information from them?" or "How would we detect that?" (Boxes 7 and 13)
- "How could we ensure the CEOs phone is secure from physical attacks combined with shoulder surfing?", "Do we have biometrics turned on to minimize use of PINs?" etc. (Boxes 4, 19)
- "If someone gained illegitimate access to the email administrator account and tried to use it to access employee inboxes, how could we detect that?"
- "How would we know if / when a malicious auto-forwarding rule is set up for an employee's inbox?" (Box 8)
- "How confident are we that malware emailed to a VIP email account would be blocked or detected?" (Box 17)
- "What if the CEOs laptop started running a new, unsigned program or script, would we see it?" (Boxes 6, 17, and 18)

As you look at your diagram, you should be asking similar questions. Capturing these thoughts can give you a great high-level set of questions to ask your SOC team about, and also a great set of scenarios to purple team with. We can take it further, however. For each section, you can also get more specific about prevention and detection capabilities, and data

sources for visibility of that action. For the data in the example diagram, a structured list like the one below can more explicitly list out methods and capabilities to spot these attacks.

An example table for the credential theft related group:

Category	Detail
Category /Label	Attacker uses legitimate access with stolen credentials
Box Numbers	2,5,8,9,10
Prevention Capabilities	Passwordless Azure AD login with biometric FIDO2 USB keys, Credential Guard enabled on Windows workstations, Legacy Authentication Disabled, Privileged access workstations deployed for highly privileged accounts
Detection Capabilities	Azure AD Identity Protection Enabled with sign-on risk policy, SIEM rules for suspicious logins based on IP
Other Mitigations	User awareness training for phishing attacks
Data Sources	Windows Security Logs from endpoints, servers, cloud, and domain controllers (authentication success/failure events), Kerberos service logs from domain controllers, Azure AD authentication logs, Azure AD Identity Protection event logs

This list more formally captures exactly what is being done to ideally prevent these types of attacks, detect them if they do happen, and specific data sources that power those detections.

Assessing Your Own Mitigations and Data Sources:

We'll now fill in this information for your own diagram. On the virtual machine desktop there is a shortcut icon titled **MGT551 Mitigations and Data Source Assessment.docx**. Either double click it to open inside your virtual machine using LibreOffice Write, or click and drag the file out of your virtual machine (should be supported on most versions of VMware) and use Word on your host PC to fill in the corresponding details.

This can be a fairly time-intensive exercise so don't worry if you don't have time to fill in everything perfectly and completely. Do strive to at least get some initial entries for each item or highlight places where you've realized there are important gaps in your defense. You can always complete this further with your team once you get back to work. If you need additional detail on how any tactics or techniques might work, this is where referencing the MITRE ATT&CK framework can come in handy, as most techniques and sub-techniques have specific log sources or data types listed.

Email Collection: Remote Email Collection

Other sub-techniques of Email Collection (3)

Adversaries may target an Exchange server or Office 365 to collect sensitive information. Adversaries may leverage a user's credentials and interact directly with the Exchange server to acquire information from within a network. Adversaries may also access externally facing Exchange services or Office 365 to access email using credentials or access tokens. Tools such as [MailSniper](#) can be used to automate searches for specific keywords.

ID: T1114.002

Sub-technique of: [T1114](#)

Tactic: Collection

Platforms: Office 365, Windows

Data Sources: Authentication logs, Email gateway, Mail server, Office 365 trace logs

Version: 1.0

Created: 19 February 2020

Last Modified: 19 February 2020

[Version Permalink](#)



Open up the worksheet now and fill in the details for your attack tree category groupings. As you look through each box, ask yourself:

- What would we prevent this item from occurring?
- If this did happen, do we have any active detection and alerting capabilities?
- What data source would show evidence of this activity?

You've done it! You should now have a filled sheet with categorized items from your attack tree, questions to ask your security team, and some assurance that you can spot individual steps from a high-risk cyber attack scenario.

Taking It Further

If you'd like to go even deeper with this exercise, take the items on this list and set up a breach and attack simulation tool to automate emulating these steps. A continuous verification of these top-priority items means that you can rest easy, assured that an adversary taking steps towards your nightmare scenario will indeed be caught!

Exercise Conclusion -- Key Takeaways

In this exercise, you have:

- Further broken down your attack trees, analyzing the information from a defensive angle

- Categorized prevention and detection capabilities for high-priority scenarios
- Verified (or exposed) sources of data collection for each set of items
- Produced a list of questions, scenarios, and items to purple team, and add to continuous security validation tools

This is a great step forward in either figuring out how to improve your SOC or verifying that it's operating and protecting your organization in a risk-aligned manner!

Exercise 2.1 is now complete!

Exercise 2.2: Prioritizing and Visualizing Attack Techniques and Security Controls

Background

To take the next step in building a threat-informed defense, we need to resolve the names of the threat actors we previously found into some lower level operational threat intelligence - which tactics and techniques these groups are known to use. While in the past, this would have potentially required an extreme amount of time, effort, and difficult to obtain knowledge, or a subscription to a threat intelligence vendor, as you will see in this exercise, the MITRE ATT&CK framework, and ATT&CK Navigator application now makes this task incredibly easy. Through this exercise we will map multiple threat groups to tactics and techniques that group uses, see where our threat groups overlap, and even start to get an understanding how the mitigations we have in place will cover (or not cover) those tactics and techniques.

Objectives

- Learn to use MITRE's ATT&CK Navigator
- Create layers with threat intelligence to prioritize threat group techniques
- Use Navigator to assess technique coverage based on your security controls
- Understand how to align threat capabilities and mitigations to identify gaps in coverage
- Export Navigator layers for use in other security tools

Exercise Preparation

This exercise is completed in your MGT551 Linux VM, if you have trouble with the setup, see the [troubleshooting](#) information in the wiki, or reach out to an instructor or SANS support.

1. Launch the MGT551 Linux VM and log in.

- LOGIN = `student`
- PASSWORD = `mgt551`

2. Start up ATT&CK Navigator

Before starting this exercise, you must start the required services. To do this, open a command terminal from the start bar.



Once the terminal window is open, start the services by copying and pasting the following command on the command line:

```
cd /home/student/labs/2.2  
docker-compose up -d
```

You should see output similar to the following. If you receive an error message inform your instructor or run the script from the "troubleshooting" page in the wiki.

```
Creating network "22_default" with the default driver  
Creating navigator ... done
```

Keep the terminal open, we will use it to shut these services down at the end of the exercise.

Exercise Questions

Open ATT&CK Navigator

In the first step of this exercise, we'll learn how to use ATT&CK Navigator - the visualization tool for ATT&CK Tactics and Techniques built by MITRE to help you understand your protection gaps and data source needs.

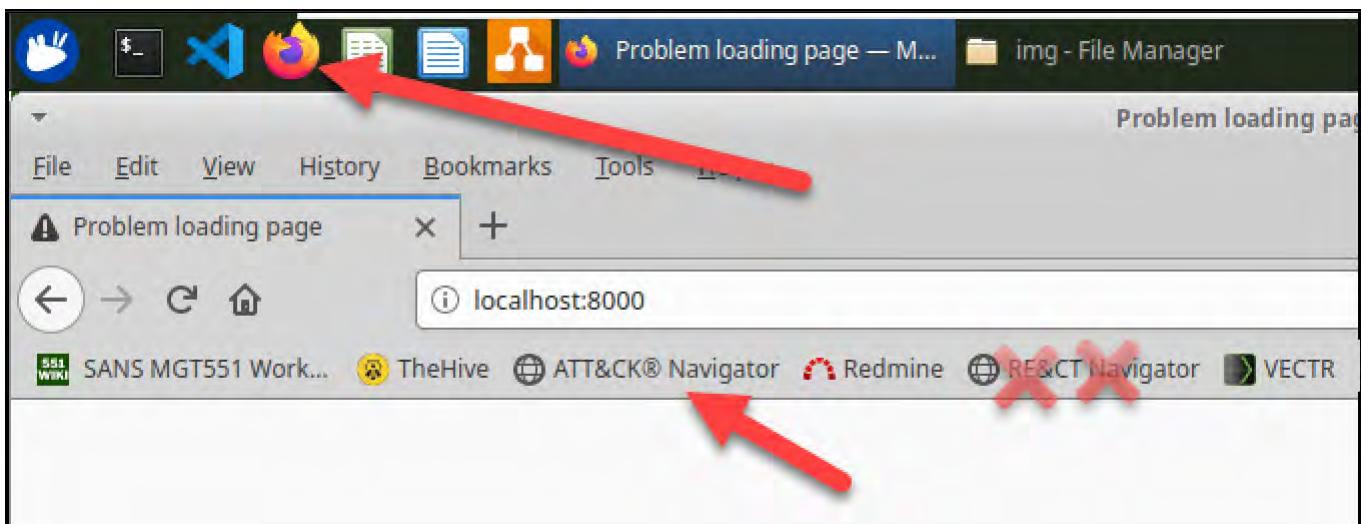
Tip

The ATT&CK Navigator software is a web-based application that is both hosted locally on your machine using the docker command above, as well as on the MITRE site: <https://mitre-attack.github.io/attack-navigator/> You can use either version you'd like for this exercise, but be aware that if you use the live version there may be slight data and interface differences from the screenshots you see here due to continual development by the MITRE team. Those who are ok with screenshot differences and want the newest information, or want to do this exercise directly on their host machine in a non virtual-machine browser can use the website version.

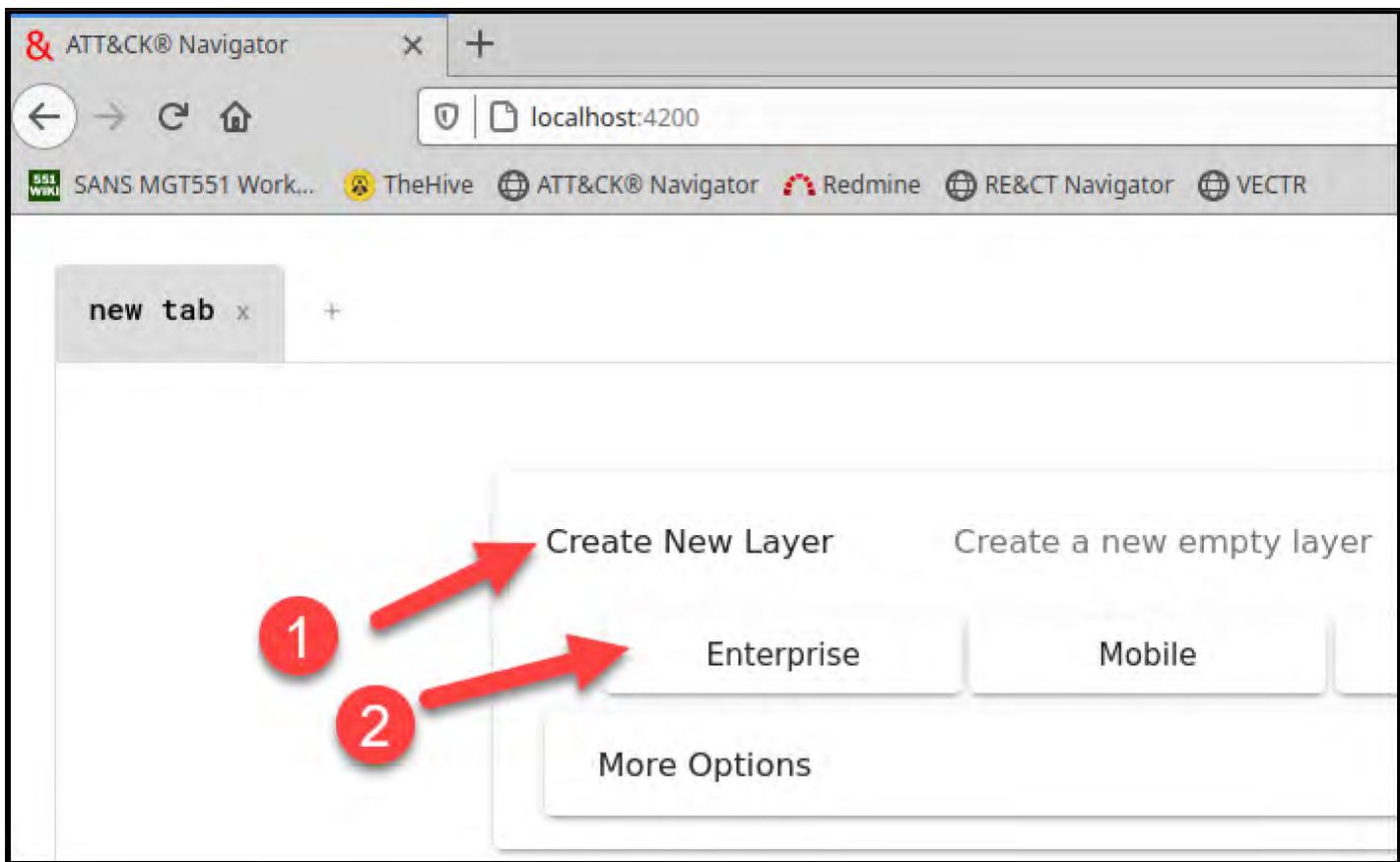
Note

Once you run the docker-compose command above it may take a minute or two for the ATT&CK Navigator container to be ready before it is functional. If you do not see the page in the following steps, wait a few moments before trying again. If the page never becomes available, run the troubleshooting script from the wiki and run the docker-compose command again, or inform your instructor.

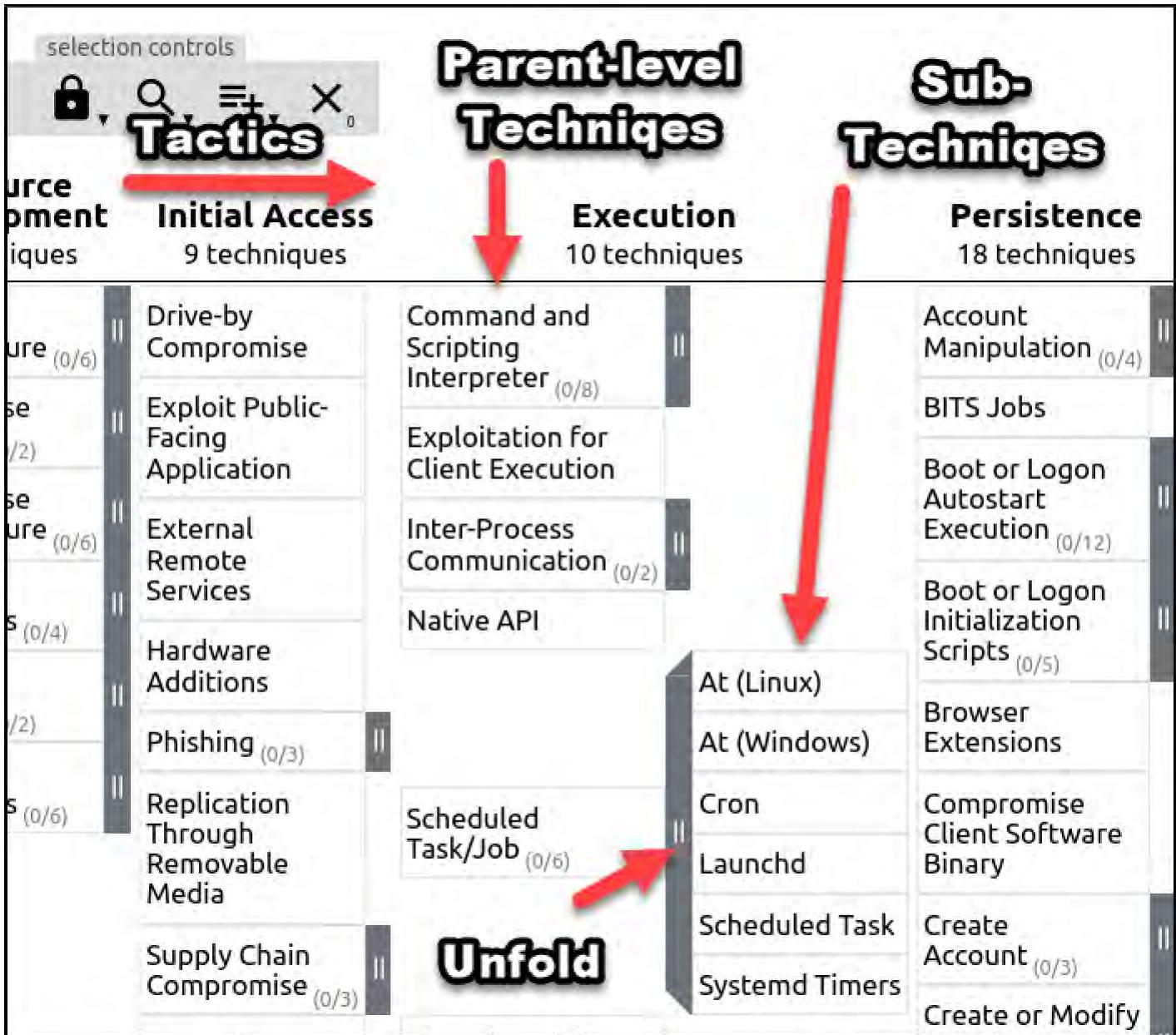
To load the ATT&CK Navigator application first open up Firefox in the virtual machine by using the link on the bar, then clicking the ATT&CK Navigator link in the bookmark bar.



Next, click "Create New Layer", then select "Enterprise".



You should now see the MITRE ATT&CK Matrix with Tactics across the top bar such as "Initial Access", "Execution", etc. and techniques down each column. One key thing that is different about the ATT&CK matrix in the *new* version, however, is the inclusion of **sub-techniques**. To view sub-techniques, find any technique that has a dark gray bar with two lines on the side and click on the bar, you will see that technique expand to display the related sub-techniques, as shown below.



Click around and take a look at the new layout and technique set for the ATT&CK Matrix, don't worry about the functions in the selection and layer controls bar yet, we will be explaining and using those shortly. Once you are done investigating the new layout, move to the next step.

Select Relevant Platforms for Display

To begin you first need to select the set of platforms that will apply to your environment. To do this, click on the filters button under the "layer controls" section as shown in the photo below and click the box for each "platform" applicable to your environment. Select Windows, Linux, and macOS as applicable, as well as any cloud environments you use and/or the SaaS services in general. As you click them the techniques that relate to those environments will be added or removed from the matrix. Once you are done, click the multi-select box again to make the drop-down go away.

Note

If you are following along without putting in data for your actual organization, leave the options at default, or at any other setting you'd like to use.

The screenshot shows the MITRE ATT&CK Navigator software interface. The main view displays the 'Execution' section, which contains 10 techniques. To the right, there is a sidebar titled 'Privilege Escalation' with 12 techniques. A large red arrow labeled '1' points to the filter icons at the top of the Execution section. Another red arrow labeled '2' points to the 'platforms' dropdown menu, which lists various operating systems and cloud environments. A third red arrow labeled '3' points to the list of execution techniques, including 'Command and Scripting Interpreter', 'Exploitation for Client Execution', 'Inter-Process Communication', 'Native API', 'Scheduled Task/Job', 'Shared Modules', and 'Software Deployment Tools'.

You have now limited the techniques shown to only those items that are relevant to your organization and we can begin adding information to the matrix.

Create Layers for Each Attack Group

There are multiple use cases for the Navigator software explained in a MITRE blog post by Katie Nickels here: <https://medium.com/mitre-attack/getting-started-with-attack-cti-4eb205be4b2f>. According to MITRE, those use cases include:

- Threat Intelligence
- Detection and Analytics

- Adversary Emulation and Red Teaming
- Assessment and Engineering

In addition, MITRE has a great "Getting Started" here if you'd like more information after this exercise: <https://attack.mitre.org/resources/getting-started/>

We'll be using the Navigator software as referenced in the blog post with a goal of knowing how our adversaries act and using that information to improve our SOC decision-making. This is referred to as "Level 1" usage of ATT&CK in the blog post and is the level applicable and achievable by any organization, which is why it makes a great starting point.

One of the most useful things about the Navigator software is that it takes the threat intelligence MITRE has gathered about different threat groups and the techniques they use and makes it available directly within the application. Remember exercise1.1 from yesterday where we did an initial identification of known threat actors that appear to be important to our organization? Navigator is the software that allows us to take that information and turn it into actionable detection requirements, which is exactly what we'll do in this exercise.

To start, we'll take the threat groups we identified in the previous exercise and identify the specific techniques they are known to use. Navigator makes this simple, and it can be done either via the built-in information or filled out manually. Here's how it works.

Look back to your exercise 1.1 information and find the names of the threat groups you identified as being relevant to your organization. (If you're using our examples, the pharmaceutical company we imagined in exercise 1.1 discovered threat groups named *APT41*, *TA505*, and *Silence* were all potential threats.)

In this step, we'll create a tab in the Navigator interface for the techniques known to be used by each individual threat group. To do this, in Navigator under the "selection controls" portion of the toolbar, find the multi-select button and click it. Then, under the "threat groups" heading, locate the name of *one* of the threat groups you've identified and click the "Select" button next to it as shown below. (If you cannot find your group of interest built in, you could manually extract the tactics and techniques and select them manually, we'll describe the method for this later. For now, we are focusing on how to use the tool, so if you cannot find your own threat groups, follow along with the examples.) Once you've clicked "select" on a single threat group, click the multi-select box again to make the drop-down go away.

selection controls

layer controls

threat groups

		view	select	deselect
APT39		view	select	deselect
APT41		view	select	deselect
Axiom		view	select	deselect
BlackOasis		view	select	deselect

software

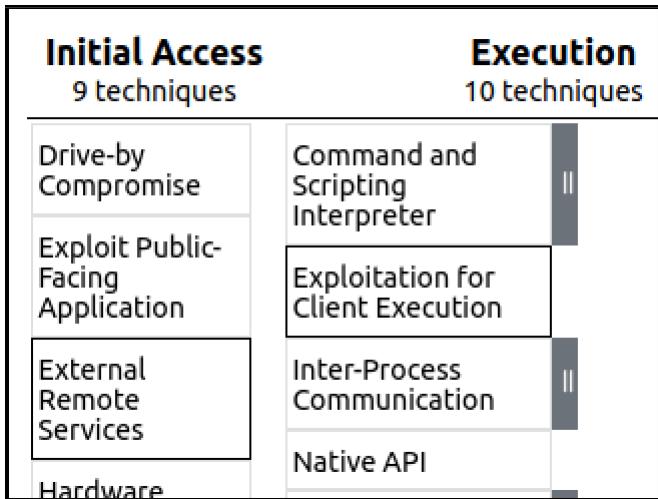
	view	select	deselect
3PARA RAT	view	select	deselect
4H RAT	view	select	deselect
adbupd	view	select	deselect
ADVSTORESHELL	view	select	deselect
Agent Tesla	view	select	deselect

mitigations

	view	select	deselect
Account Use Policies	view	select	deselect
Active Directory Configuration	view	select	deselect
Antivirus/Antimalware	view	select	deselect
Application Developer Guidance	view	select	deselect
Application Isolation	view	select	deselect

Remote Access

When you click "select" on a threat group name, Navigator will multi-select all applicable techniques in the matrix. The selected items will show up with a border around them as shown below (unless there are no known techniques associated with that threatgroup.)



Next, move to the "technique controls" toolbar area and select the "scoring" button as shown in the photo below. In the box that pops up, we can now give a numeric score that will be attached to all techniques selected in this "layer" of Navigator. Enter a number like "10" in the box, then click the scoring icon again to make the box go away. You should see that as soon as the number is entered, all techniques selected by our threat group multi-select immediately become highlighted.

technique controls

Initial Access
9 techniques

Execution
10 techniques

Persistence
18 techniques

Privileged Escalation
12 techniques

Abuse
17 techniques

Score: 10

Technique	Status
Drive-by Compromise	Normal
Exploit Public-Facing Application	Selected
External Remote Services	Selected
Hardware Additions	Normal
Phishing ...	Normal
Command and Scripting Interpreter	Selected
Exploitation for Client Execution	Selected
Scheduled Task/Job	Selected
Inter-Process Communication	Normal
Native API	Normal
BITS Jobs	Selected
Account Manipulation	Normal
Boot or Logon Autostart Execution	Normal
Boot or Logon Initialization Scripts	Normal
Browser Extensions	Normal
Abuse Elevation Control Mechanism	Normal
Access Token Manipulation	Normal
Boot or Logon Autostart Execution	Normal
Boot or Logon Initialization Scripts	Normal
Deobfuscate/Decode Files or Information	Normal
Direct Volume Access	Normal
Domain Policy	Normal

You now have a visualization of all the techniques MITRE has identified to be associated with your selected threat group. Let's label this tab with the name of the threat group you selected. To do this, click on the tab name in the Navigator interface and type the threat group name - "APT41" in the case of the example.

Click and change name

Initial Access
9 techniques

Execution
10 techniques

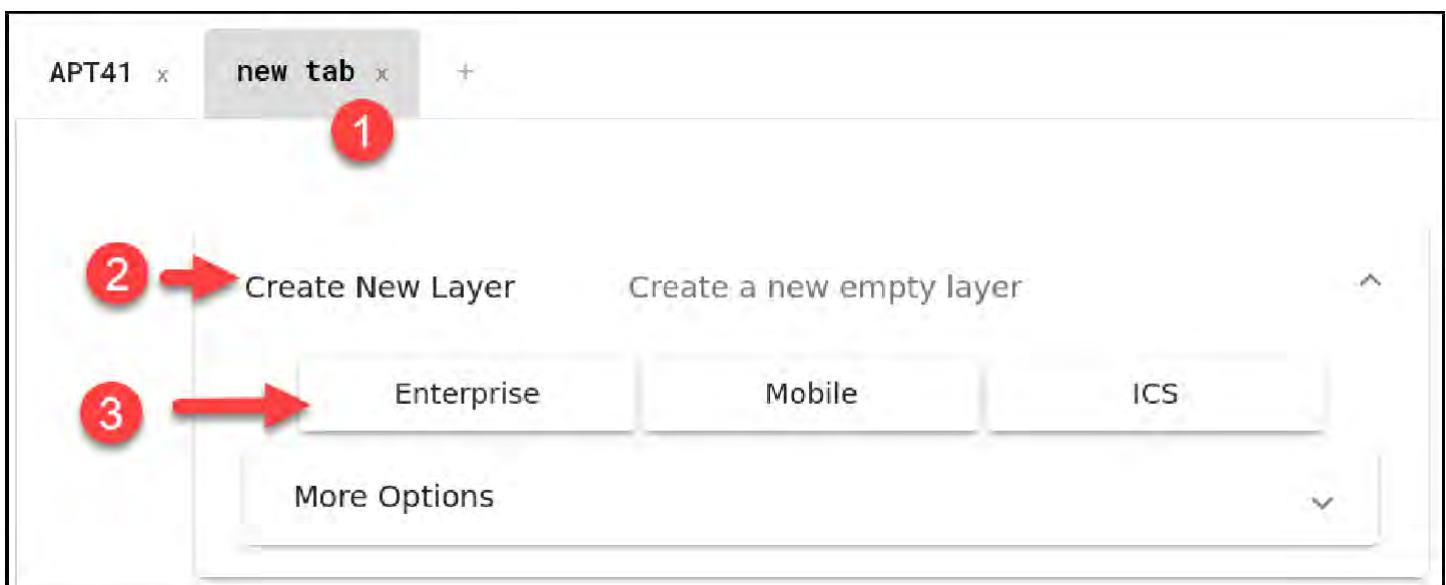
Drive-by Compromise Command and Scripting

We will be repeating this process for each threat group you have identified, creating a new Navigator layer tab for each. The ultimate goal here will be to create multiple layers all corresponding to different threat groups and technique mitigation methods and combine them to create insight for defenders.

To create a new tab, click on the "+" icon next to the tab you just renamed to bring up the new layer interface:



You should now see the "new tab" at the top. Click the "Create New Layer", then select "Enterprise" to bring up a blank additional layer in Navigator.



In your new layer, repeat the same process we just did for the first threat group. If using the example, this would include:

1. Finding the TA505 Threat Group in the multi-select box and pressing "select".
2. Adding a score to the multi-selected items, causing them to highlight.
3. Naming the layer after the selected threat group.
4. Creating a 3rd layer if needed and repeating the process for any additional groups.

The screenshot shows the MITRE ATT&CK Navigator interface. At the top, there are three tabs labeled APT41, TA505, and Silence. The TA505 tab is active. Red numbers 1 through 4 are overlaid on the interface. Number 1 points to the 'selection controls' button (a plus sign inside a square) in the top right. Number 2 points to the 'technique controls' button (a square with a minus sign) in the top right. Number 3 points to the TA505 tab. Number 4 points to the 'Execution' tab, which is highlighted with a red box. The main table below shows various tactics and techniques for different threat groups, with the 'Command and Scripting Interpreter' technique highlighted in red in the Execution section of the TA505 tab.

Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement
9 techniques	10 techniques	17 techniques	12 techniques	32 techniques	13 techniques	21 techniques	9 techniques
Drive-by Compromise	Command and Scripting Interpreter	Account Manipulation	Abuse Elevation Control Mechanism	Abuse Elevation Control Mechanism	Brute Force	Account Discovery	Exploitation of Remote Services
Exploit Public-Facing Application	Exploitation for Client Execution	BITS Jobs	Access Token Manipulation	Access Token Manipulation	Credentials from Password Stores	Application Window Discovery	Internal Spearphishing
External Remote Services	Inter-Process Communication	Boot or Logon Autostart Execution	Boot or Logon Autostart Execution	BITS Jobs	Exploitation for Credential Access	Browser Bookmark Discovery	Lateral Tool Transfer
Hardware Additions	Native API	Boot or Logon Initialization Scripts	Boot or Logon Initialization Scripts	Deobfuscate/Decode Files or Information	Forced Authentication	Domain Trust Discovery	Remote Service Session Hijacking
	Scheduled Task/Job			Direct Volume Access	Input Capture		
				Execution Guardrails			

Repeat this process for every threat group in your assessment from exercise 1.1. If using the example data, you should now have a Navigator interface with 3 tabs, one for APT41, one for TA505, and one for Silence.

The screenshot shows the MITRE ATT&CK Navigator interface with three tabs at the top: APT41, TA505, and Silence. Red numbers 1, 2, and 3 are overlaid on the tabs. Number 1 is over the APT41 tab, number 2 is over the TA505 tab, and number 3 is over the Silence tab. Below the tabs is a toolbar with various icons. The main table below shows the initial access, execution, persistence, privilege escalation, and defense evasion techniques for each threat group. The 'Command and Scripting Interpreter' technique is highlighted in red in the Execution section of the TA505 tab.

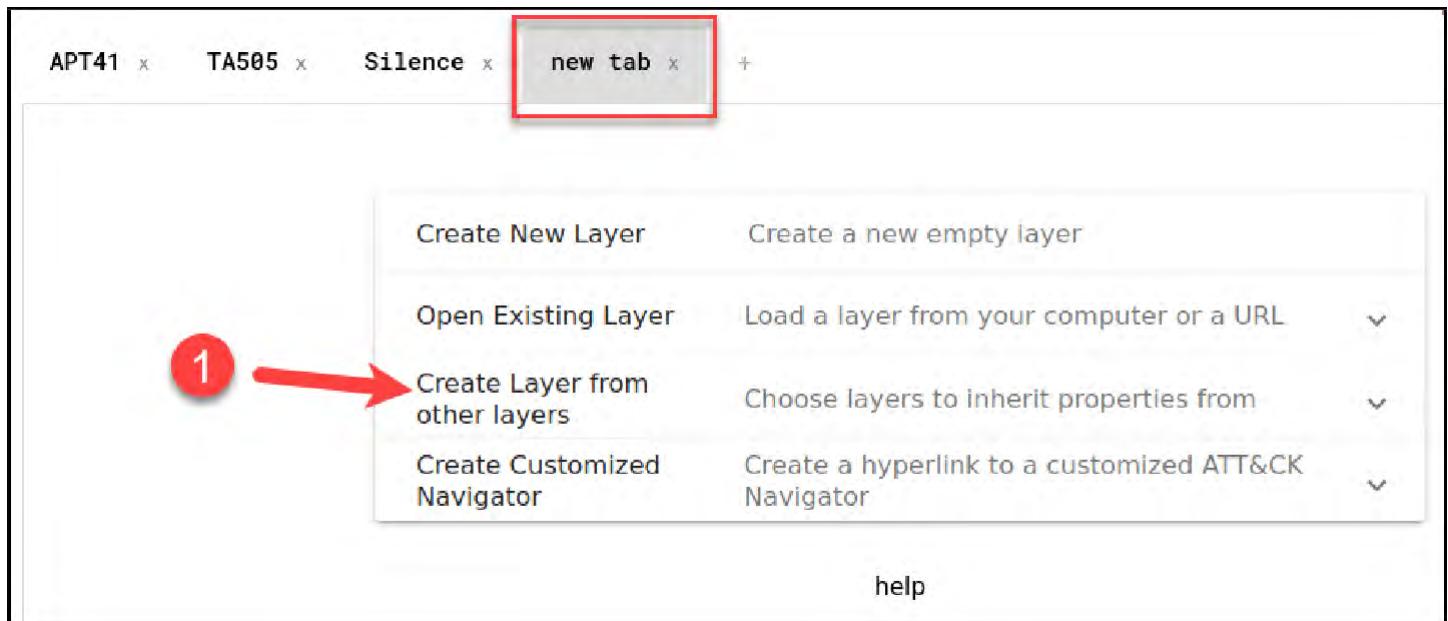
Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion
9 techniques	10 techniques	17 techniques	12 techniques	32 techniques
Drive-by Compromise	Command and Scripting Interpreter	Account Manipulation	Abuse Elevation Control Mechanism	Abuse Elevation Control Mechanism
Exploit		BITS Jobs	Access Token Manipulation	Access Token Manipulation

You now have mapped individual threat groups to a visual representation of their tactics and techniques.

Create a Combined Threat Group Layer

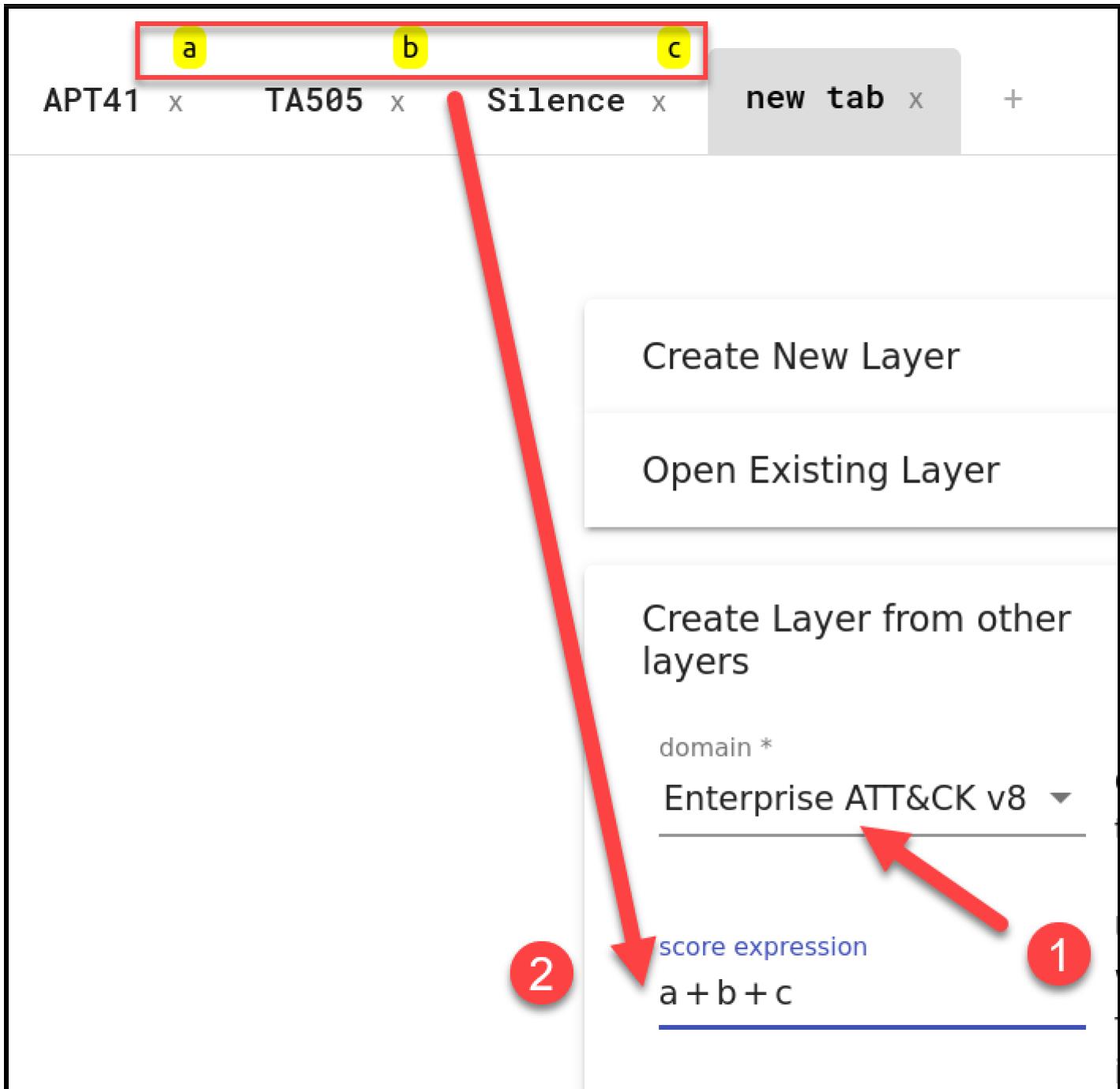
In this step, we'll take the three separate layers and combine them into one new, all-encompassing, combined threat groups layer. Once complete, we'll be able to see not only data from all 3 at once but also where there is an overlap where multiple groups use the same techniques. This is important because if we have 3 threat groups that we expect will attack us, and there is one specific technique they all use, it's important to prioritize detection and mitigation of that technique above the others.

To create a combined layer, click the "+" sign to add another layer to Navigator, but this time, select "Create Layer from other layers".



When selected, you see that each threat group tab is now labeled with a letter, and the interface provides a drop-down with options for creating the new layer. Since we used scoring to label the techniques, in the "score expression" box, we'll tell Navigator to create a new layer by adding the scores of each individual technique from the threat group layers we've created.

As the picture below shows, first select "Enterprise" as the domain, then, if you have three threat group layers - a, b, and c, type the score expression `a+b+c`. Ensure you add letters for all layers you have added and then press the "Create" button at the bottom of the screen.



You should now see an additional tab with all items combined, for the example data this looks like the matrix below.

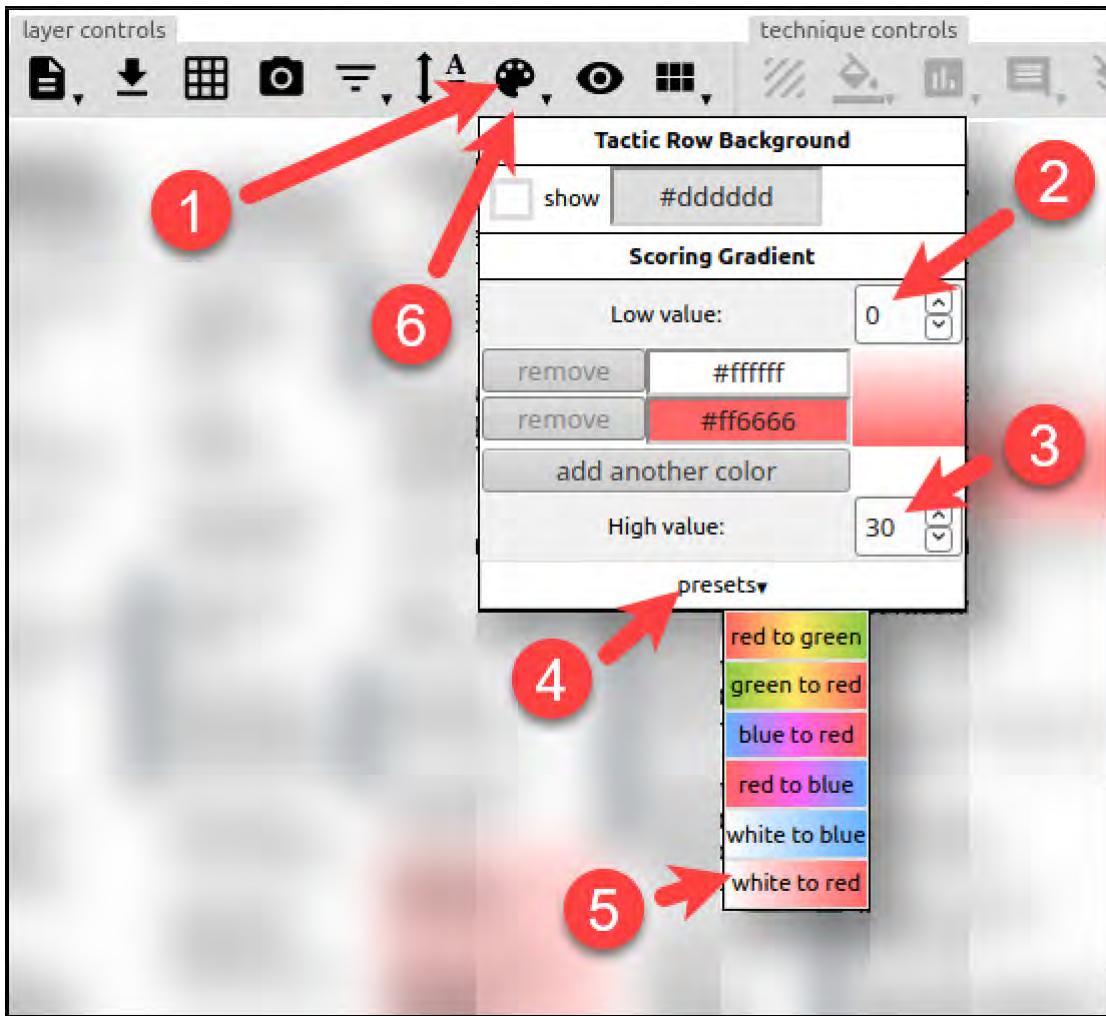
Initial Access 9 techniques	Execution 10 techniques	Persistence 18 techniques	Privilege Escalation 12 techniques
Drive-by Compromise	Command and Scripting Interpreter (5/8)	Account Manipulation (0/4)	Abuse Elevation Control Mechanism (0/4)
Exploit Public-Facing Application	Exploitation for Client Execution	BITS Jobs	Access Token Manipulation (0/5)
External Remote Services	Inter-Process Communication (1/2)	Boot or Logon Autostart Execution (1/12)	Boot or Logon Autostart Execution (1/12)
Hardware Additions	Native API	Boot or Logon Initialization Scripts (0/5)	Boot or Logon Initialization Scripts (0/5)
Phishing (2/3)	Scheduled Task/Job (1/6)	Browser Extensions	Create or Modify System Process (1/4)
Replication Through Removable Media	Shared Modules	Compromise Client Software Binary	Domain Policy Modification (0/2)
Supply Chain Compromise (1/3)	Software Deployment Tools	Create Account (1/3)	Event Triggered Execution (1/15)
Trusted Relationship	System Services (1/2)	Create or Modify System Process (1/4)	Exploitation for Privilege Escalation
Valid Accounts (1/4)	User Execution (2/2)	Event Triggered Execution (1/15)	Hijack Execution Flow (1/11)
	Windows Management Instrumentation	External Remote Services	Process Injection (1/11)
		Hijack Execution Flow	

There are now different colors that relate to the various combined score of each layer. By default, the lower scores are tinted red while the higher scores are green. For what we'd like to know, (which techniques are most important to us) this is a bit unintuitive, so it can be easily changed. We can also change the display order such that the highest scored items are on the top of each column to make it easier to see what matters.

To change colors, click on the "color setup" in the layer controls section of the toolbar. In this section, we must change two things.

1. The range of possible values. As shown in the photo, Navigator defaulted to coloring based on the range of 10 to 20. Since I have 3 layers, each with a possible score of 10 the max is 30 and the minimum is 0, so we must change the Low and High value to reflect that. Change the low value to 0 and the high value to the maximum score possible based on the number of threat groups you have added.

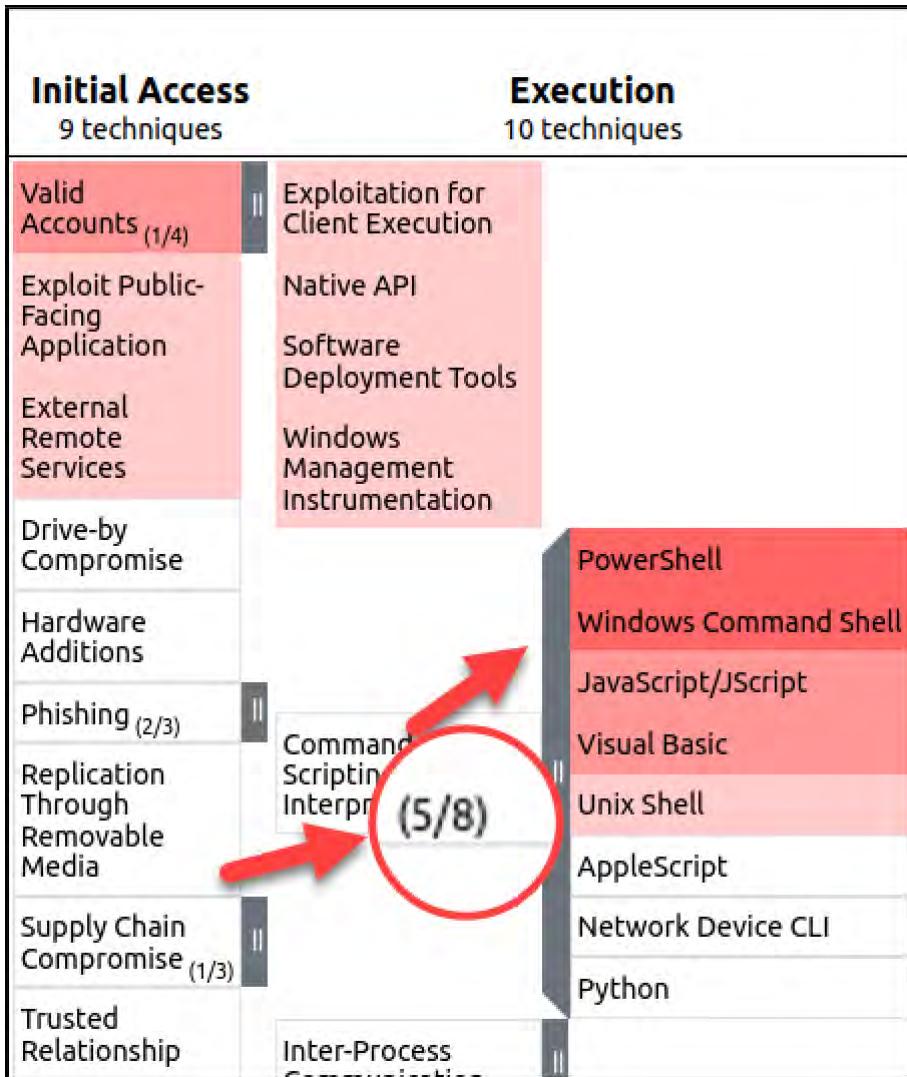
2. The color scheme. It makes more sense to see white for a score of zero, and darker red for a higher score moving up from there. To accomplish this, click the "presets" button on the bottom of the drop-down and select "white to red". This will implement the scheme of 0 for white and red for the highest score.



Afterward, when using the example data, the matrix should look like this (yours may look different of course, depending on the threat group combination you've used.)

Note

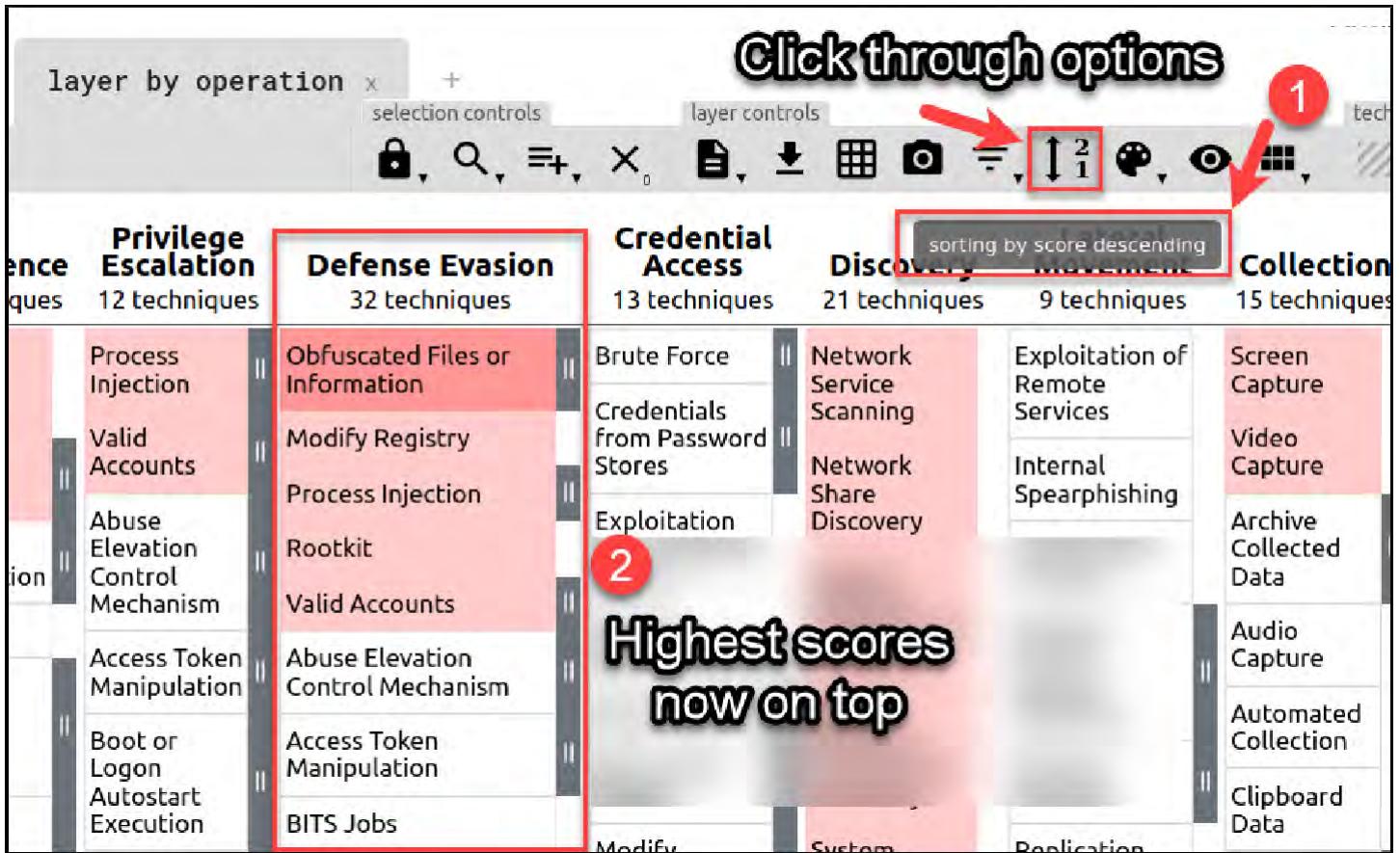
Some entries may have an uncolored parent level technique with scored sub-techniques - look for a number in the lower right of the box to identify these techniques. You can also click the "Expand sub-techniques" button in the toolbar of Navigator (next to the eye shaped icon) to unfold and display all sub-techniques at once.



When mousing over each item, you should be able to verify that the darker colors represent higher scored techniques. (Note that if your threat groups have no overlap in techniques it is possible that all highlighted items could have the same, low score.)

Initial Access	Execution	Persistence	Privilege Escalation
9 techniques	10 techniques	17 techniques	12 techniques
Drive-by Compromise	Command and Scripting Interpreter	Command and Scripting Interpreter (T1059) Score: 20	Abuse
Exploit Public-Facing Application	Exploitation for Client Execution	Boot or Logon Autostart Execution	Access Token Manipulation
External Remote Services	Inter-Process Communication	Boot or Logon Autostart Execution	Boot or Logon Autostart Execution
Hardware	Native API	Initialization	File System Manipulation

To reorganize the order and bring the highest scored items to the top of each column, click on the sorting button under layer controls. After several clicks, you will reach the "sorting by score descending" option which will leave you with a matrix similar to the following.

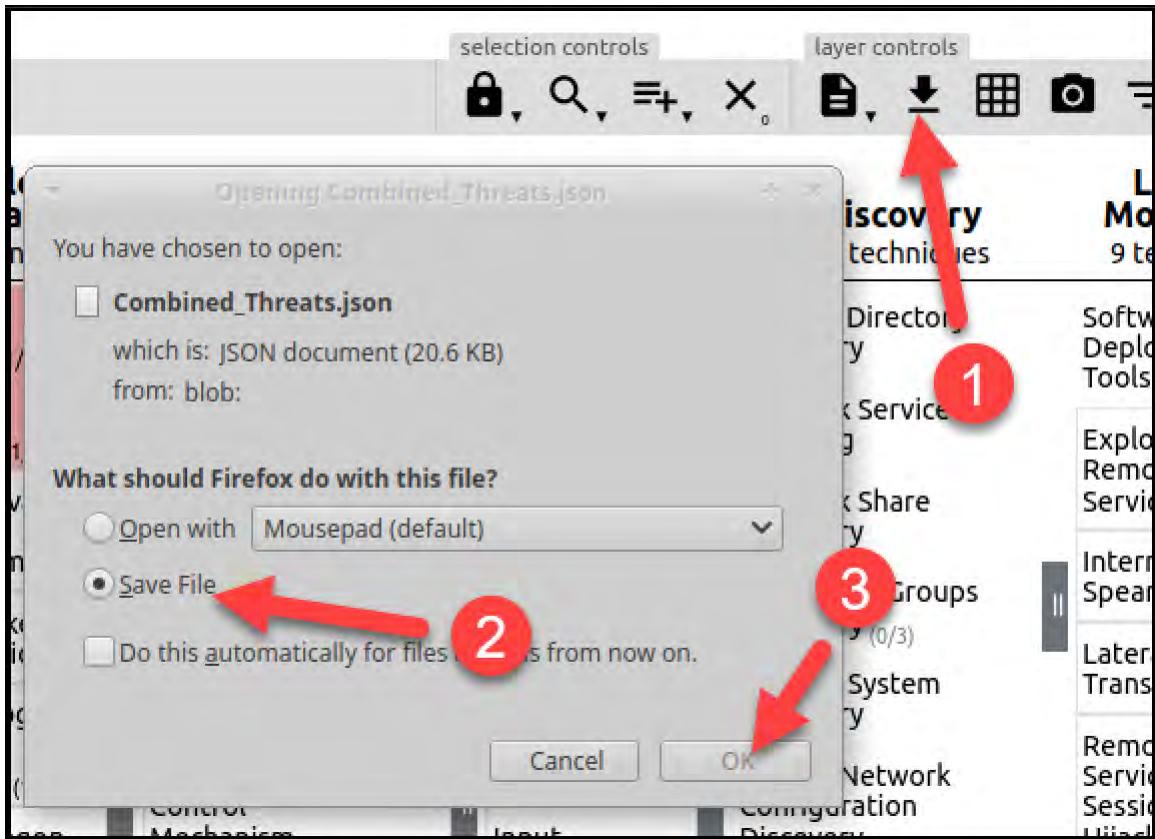


As a final step, rename the tab to something like "Combined Threats".

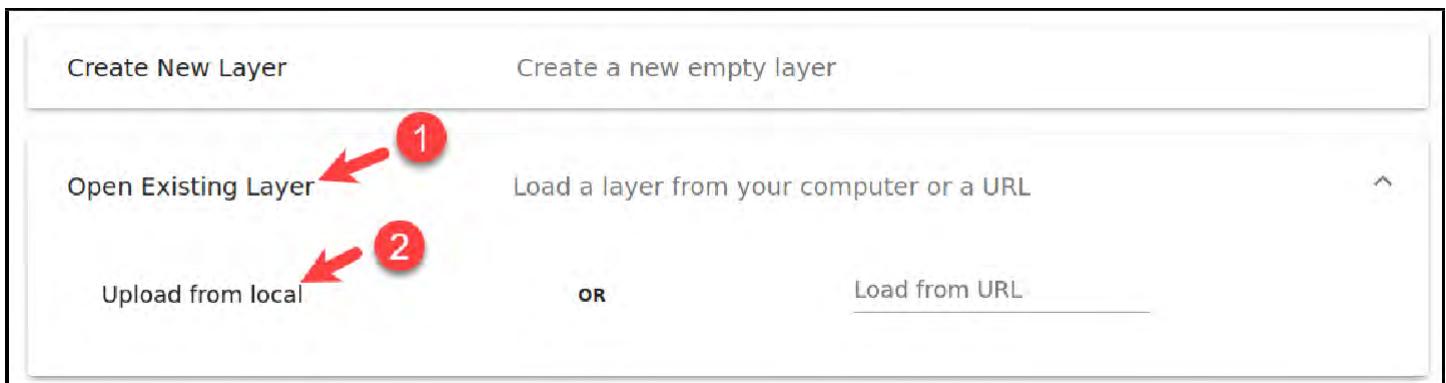


You have now taken all identified high-risk threat groups, visualized the techniques they used, built a custom visualization of where they overlap and brought the most important items to the front! Now that you have this info, you should use it to prioritize defensive controls and detection mechanisms as these techniques represent the attack techniques (in theory, assuming the intelligence is correct) your team will most likely encounter.

At this point, if you'd like to save your work, you can use the "Download layer as JSON" functionality to save the state of the Navigator layer to your hard drive for quick recall later or for loading on another version (or the online version) of ATT&CK Navigator.



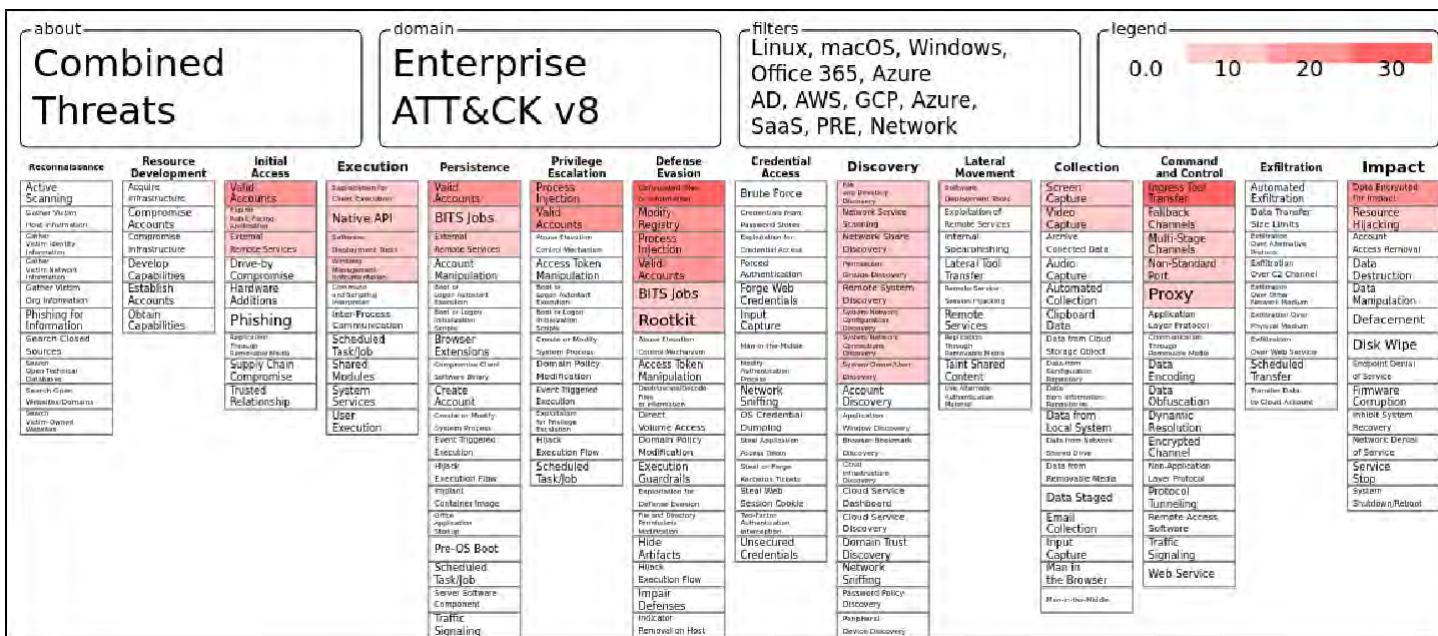
This layer can now be easily loaded on any instance of Navigator by clicking the "+" to create a new layer then selecting "Open Existing Layer", then selecting the file using the "Upload from local" option.



Additionally, the interface has a button for exporting the document to an Excel spreadsheet, an export of the example data has been placed in the `~/labs/2.2/Combined_Threats.xlsx` file and is shown below.

	A	B	C	D
1	Initial Access	Execution	Persistence	Privilege Escalation
2	External Remote Services	Command and Scripting Interpreter	External Remote Services	Process Injection
3	Valid Accounts	Exploitation for Client Execution	Valid Accounts	Valid Accounts
4	Drive-by Compromise	Native API	Account Manipulation	Abuse Elevation Control Mechanism
5	Exploit Public-Facing Application	System Services	BITS Jobs	Access Token Manipulation
6	Hardware Additions	Windows Management Instrumentation	Boot or Logon Autostart Execution	Boot or Logon Autostart Execution
7	Phishing	Inter-Process Communication	Boot or Logon Initialization Scripts	Boot or Logon Initialization Scripts
8	Replication Through Removable Media	Scheduled Task/Job	Browser Extensions	Create or Modify System Process
9	Supply Chain Compromise	Shared Modules	Compromise Client Software Binary	Event Triggered Execution
10	Trusted Relationship	Software Deployment Tools	Create Account	Exploitation for Privilege Escalation
11		User Execution	Create or Modify System Process	Group Policy Modification
12			Event Triggered Execution	Hijack Execution Flow
13			Hijack Execution Flow	Scheduled Task/Job

You can even export the current map to an SVG graphic for printing:



Identify and Enter Mitigations

While previous to April 2020 this exercise would have to stop at this point unless we were willing to do a lot more manual work, in version 3.0 and above (the Navigator version with sub-techniques), MITRE has also built-in tracking of *mitigation* effects into the multi-select toolbar! Mitigations are the controls you can implement within your network to defend against these techniques, and while in previous versions of ATT&CK, they were listed individually in text for analysts to read, in this version they are a new object type, complete with their own "Mxxxx" naming scheme and association to which techniques they can prevent (see link for complete mitigations list and details). In this next section, we'll follow a similar workflow to the first step, but instead, look to visualize our own defenses.

Create a MitigationsLayer

To start, create yet another new layer by hitting the plus sign near the tabs at the top of the screen then selecting "Enterprise", then "Create new layer".



First, rename the tab "Mitigations" so we know what we're dealing with, then press the multi-select button again and go down to the "mitigations" heading. Under the mitigations list, you will be selecting **all** mitigations you'd like to evaluate. At this point, you can select **all** the mitigations your company is currently using (which could take a while), or, to save time, select 5 or so as an example to save some time.

A good example choice might be the following simple mitigations that are present in many environments:

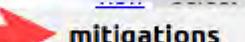
- Antivirus
- Audit
- Execution Prevention
- Exploit Protection
- Network Segmentation

Combined Threats x Mitigations  1

selection controls layer controls

Technique	Privilege Escalation	Defense Evasion	Credential Access
	12 techniques	32 techniques	13 techniques
Abuse Elevation Control Mechanism	Abuse Elevation Control Mechanism	Brute Force	Access Discretionary
Access Token Manipulation	Access Token Manipulation	Credentials from Password Stores	Application Wi-Fi Disconnection
Boot or Logon Autostart Execution	BITS Jobs	Exploitation for Credential Access	Browser-Based Exploit Delivery
Boot or Logon Initialization Scripts	Deobfuscate/Decode Files or Information	Forced Authentication	Do Not Disturb
Create or Modify System Process	Direct Volume Access	Input Capture	File and Directory Discretionary
Event Triggered Execution	Execution Guardrails	Man-in-the-Middle	Network Session Selection
Exploitation for Privilege Escalation	Exploitation for Defense Evasion	Modify Authentication Process	Network Share Selection
Group Policy Modification	File and Directory Permissions Modification	Network Sniffing	Account Use Policies
	Group Policy Modification	OS Credential Dumping	Active Directory Configuration
	Hide Artifacts	Steal or Forge Kerberos Tickets	Antivirus/Antimalware
	Hijack Execution Flow	Steal Web Session Cookie	Application Developer Guidance
	Impair Defenses		Application Isolation
	Indicator Removal on Host		Peripheral

Select all mitigations at once

mitigations  3

 4

 5

As you select mitigations, you will again see techniques begin to highlight. Every time you click select on a new mitigation it adds that set to what was already selected. The result is, in the end, after you select every mitigation, every technique covered by *all* of those mitigations will be selected in the matrix. This is the goal.

Once you have all applicable techniques selected by clicking the mitigations, go again to the scoring button and use a score of 10 for all selected boxes. This will cause all selected techniques to highlight as before.

As before, you should see a bunch of selected boxes (black borders) that are now highlighted.

Initial Access 9 techniques	Execution 10 techniques	Persistence 17 techniques	Privilege Escalation 12 techniques	Defense Evasion 32 techniques
Drive-by Compromise	Command and Scripting Interpreter	Account Manipulation	Abuse Elevation Control Mechanism	Abuse Elevation Control Mechanism
Exploit Public-Facing Application	Exploitation for Client Execution	BITS Jobs	Access Token Manipulation	Access Token Manipulation
External Remote Services	Inter-Process Communication	Boot or Logon Autostart Execution	Access Token Manipulation	BITS Jobs
Hardware Additions	Native API	Boot or Logon Initialization Scripts	Boot or Logon Autostart Execution	Deobfuscate/Decode Files or Information
Phishing	Scheduled Task/Job	Shared Modules	Boot or Logon Initialization Scripts	Direct Volume Access
Replication Through Removable Media	Software Deployment Tools	Browser Extensions	Browser Extensions	Execution Guardrails
Supply	System Services	Compromise Client Software Binary	Create or Modify System Process	Exploitation for Defense Evasion
				File and Directory Permissions Modification
				Group Policy

Here is where the process is different: to do what we want to do in the next step, there is one key action that must be performed at this moment. After you have entered the score of 10 for selected items, hover your mouse over a *non-selected* technique (a white one), right-click, and press "invert selection".



You should now see that the black border selection of boxes has switched from all highlighted techniques to the white techniques.

White boxes selected

Privilege Escalation 12 techniques					Defense Evasion 32 techniques
Drive-by Compromise	Command and Scripting Interpreter	Account Manipulation	Abuse Elevation Control Mechanism	Abuse Elevation Control Mechanism	
Exploit Public-Facing Application	Exploitation for Client Execution	BITS Jobs	Access Token Manipulation	Access Token Manipulation	
External Remote Services	Inter-Process Communication	Boot or Logon Autostart Execution	BITS Jobs	Deobfuscate/Decode Files or Information	
Hardware Additions	Native API	Boot or Logon Initialization Scripts	Direct Volume Access	Execution Guardrails	
Phishing	Scheduled Task/Job	Browser Extensions	Boot or Logon Initialization Scripts	Exploitation for Defense Evasion	
Replication Through Removable Media	Shared Modules	Compromise Client Software Binary	Create or Modify System Process	File and Directory Permissions Modification	
Supply	Software Deployment Tools			Group Policy	
	System Services				

Now the key piece - go to the scoring box again and score *these boxes* with a negative ten.

Discovery 21 techniques		Lateral Movement 9 techniques	Collection 15 techniques	Infiltration 10 techniques
Account Discovery	Exploitation of Remote Services	Archive Collected Data	Application Layer Protocol	Automated Exfiltration
Application Window Discovery	Internal Spearphishing	Audio Capture	Communication Through Removable Media	Data Transfer Size Limits

The colors won't snap to a perfect intuitive meaning, and you can change them at this point if you'd like to have the mitigations layer to view separately (a good setting would be -10 to 10 range, with red to green preset, making green things mitigated). Save the separate Mitigations layer again if desired.

Let's take a second to explain what's going on here though. We now have a layer with active mitigations scored as 10, and a technique with a lack of mitigations scored as -10. This is what we need for the next step, which will be using layer math to make a new layer, but this time instead of addition we will use multiplication. **The negative value trick is crucial for making sense of the output**, without it, it would be impossible to differentiate the unique conditions.

Combine Threat and Mitigation Layer

Now that we have our Mitigations layer and our Combined Threat layer, the ultimate goal of this exercise is to combine both and answer the question "Which techniques are used by most of our adversaries, and which ones are mitigated by one of our controls?". By combining the separate Combined Threats Navigator layer with the Mitigations layer in a clever way, we can now answer that question.

To make a single comprehensive layer that covers both our Mitigations and Combined Threats click the "+" to add another layer and pick "Create Layer from other layers". Again, the highlighted letters will appear above the tabs that already exist. Find the letter that represents your Combined Threats layer and the letter from your Mitigations layer and put them in the score expression and **multiply** them instead of adding as shown in the photo below, then press "Create".

Silence c Combined Threats d Mitigations e new tab + 1

Create New Layer Create a new empty layer

Open Existing Layer Load a layer from your

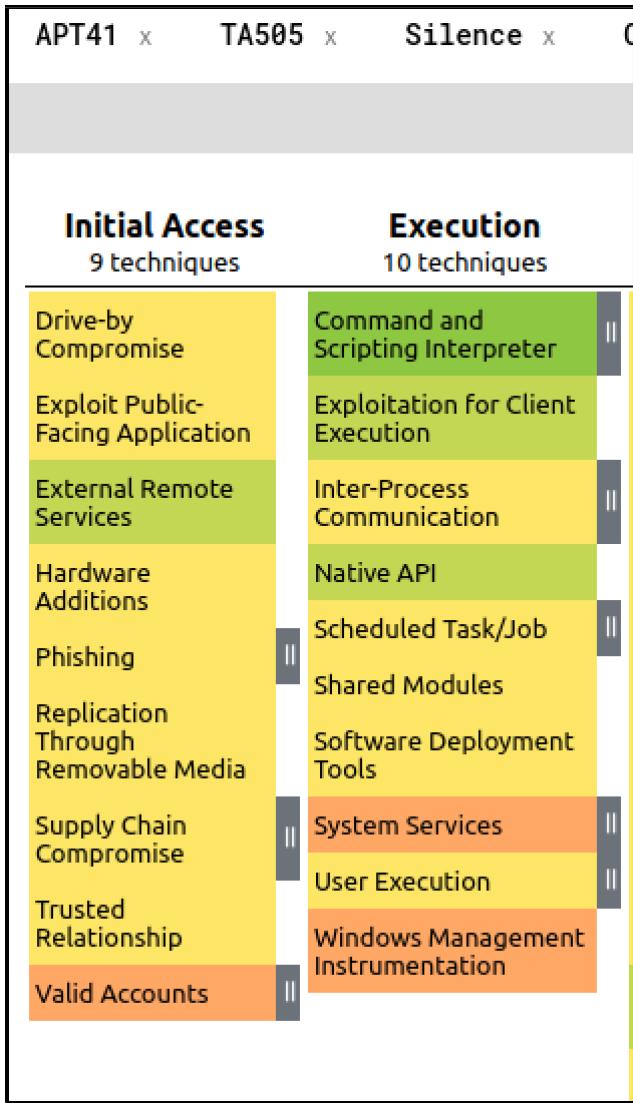
2 Create Layer from other layers Choose layers to inherit from

3 domain * Choose the domain and version for the layer that can be merged.

4 Enterprise ATT&CK v8 Use constants (numbers) and layer variables to set the initial value of scores in the new layer. Leave blank to initialize scores to 0.

score expression **Multiply**

You will then be taken to the new layer that shows all the information with the multiplied threat and mitigation score. You will likely see red, yellow, and green boxes where darker green boxes are the most positive scores and dark red are the most negative.



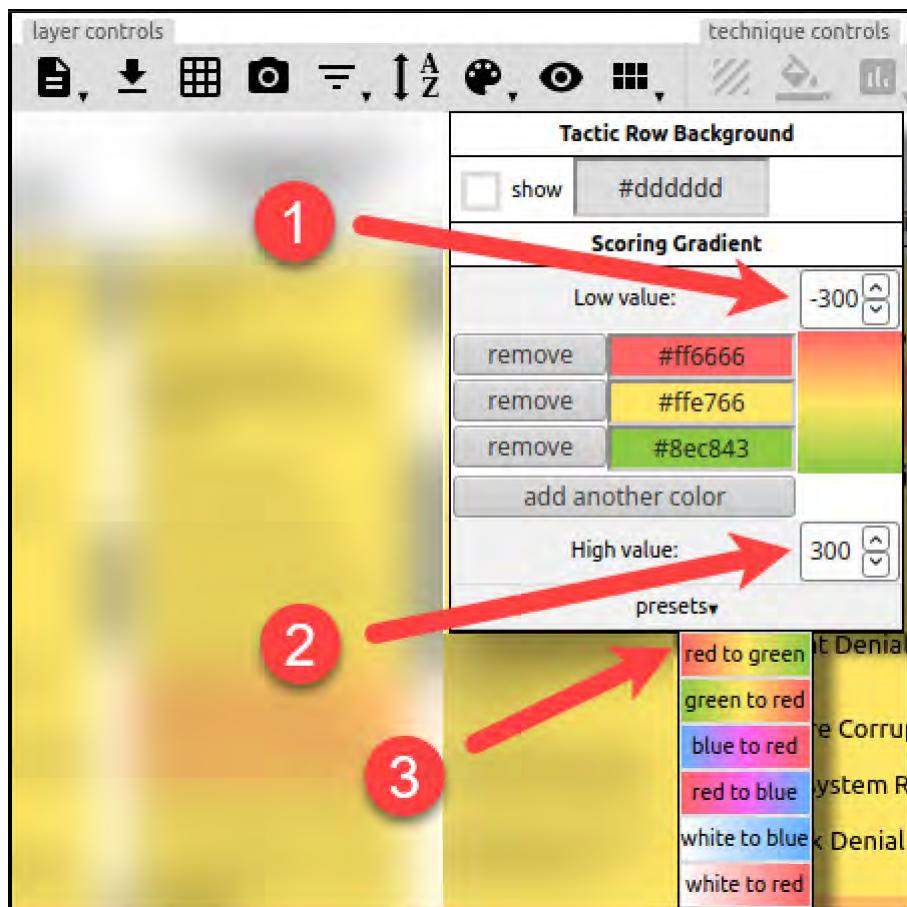
Name this tab something like "Threats + Mitigations":



Now it's time for some quick math to interpret what you see. If as a score you used 10 for each of your threat group layers and 10/-10 for your techniques that are mitigated and not mitigated, all scores should now be multiples of 100, either

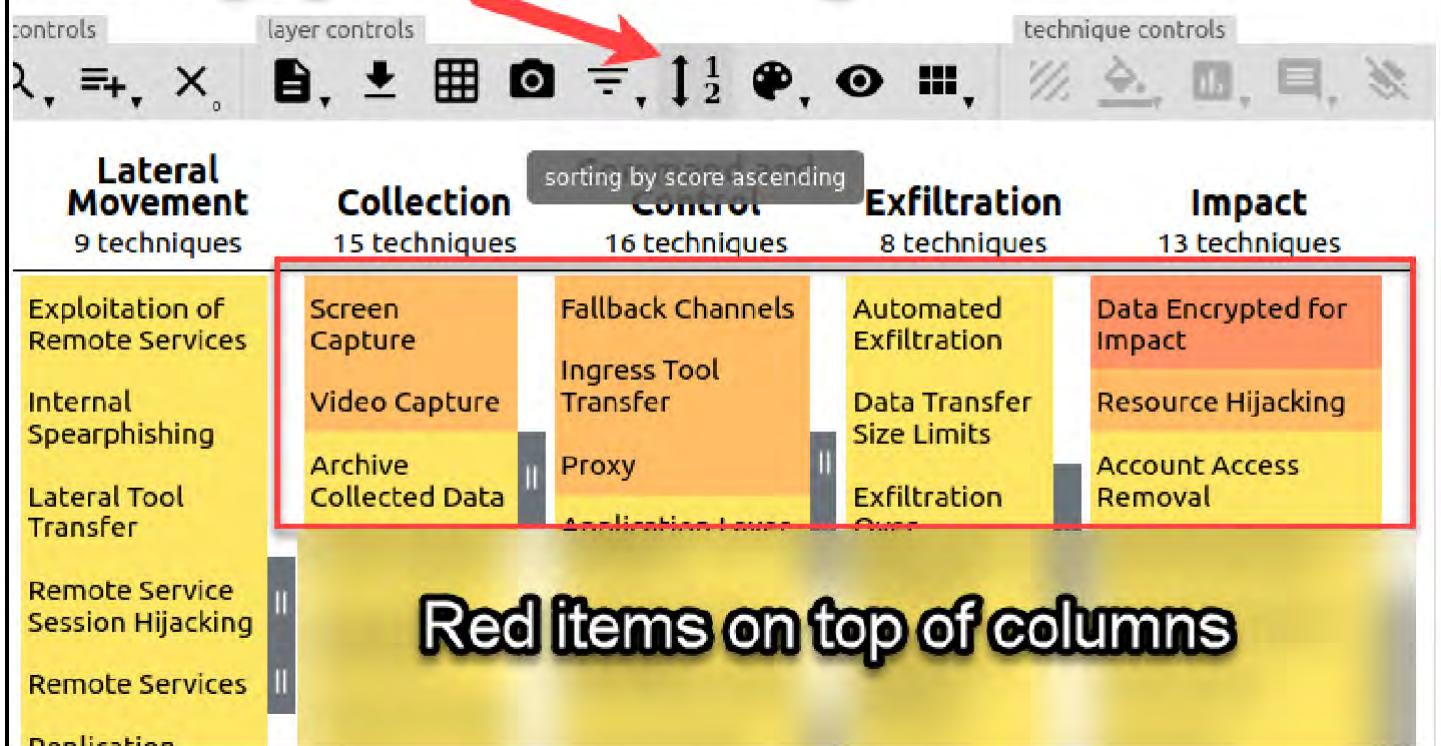
negative or positive. The maximum and minimum possible scores if you had 3 separate threat groups, for example, would be -300 (10 pts x 3 groups x -10 mitigation score) and 300 (10 pts x 3 groups x 10 mitigation score). These maximums could be hit only if one technique applies to all 3 threat groups, anything else will fall somewhere in the middle.

Before evaluating what you see, ensure your color scheme is set correctly and that the numbers represent the full range of possible values. Click the "color setup" button on the layers controls toolbar and enter in the lowest and highest possible values and ensure the color scheme is set to use red for the most negative numbers and green for the most positive.



Finally, organize your matrix again by selecting the sort button until you reach "Sort by score ascending", which should place all the darkest red (most important, unmitigated techniques) on top of each column.

Sorting by score ascending



You should now see your final matrix that combines both threat intelligence for multiple important threat groups as well as information on your mitigations!

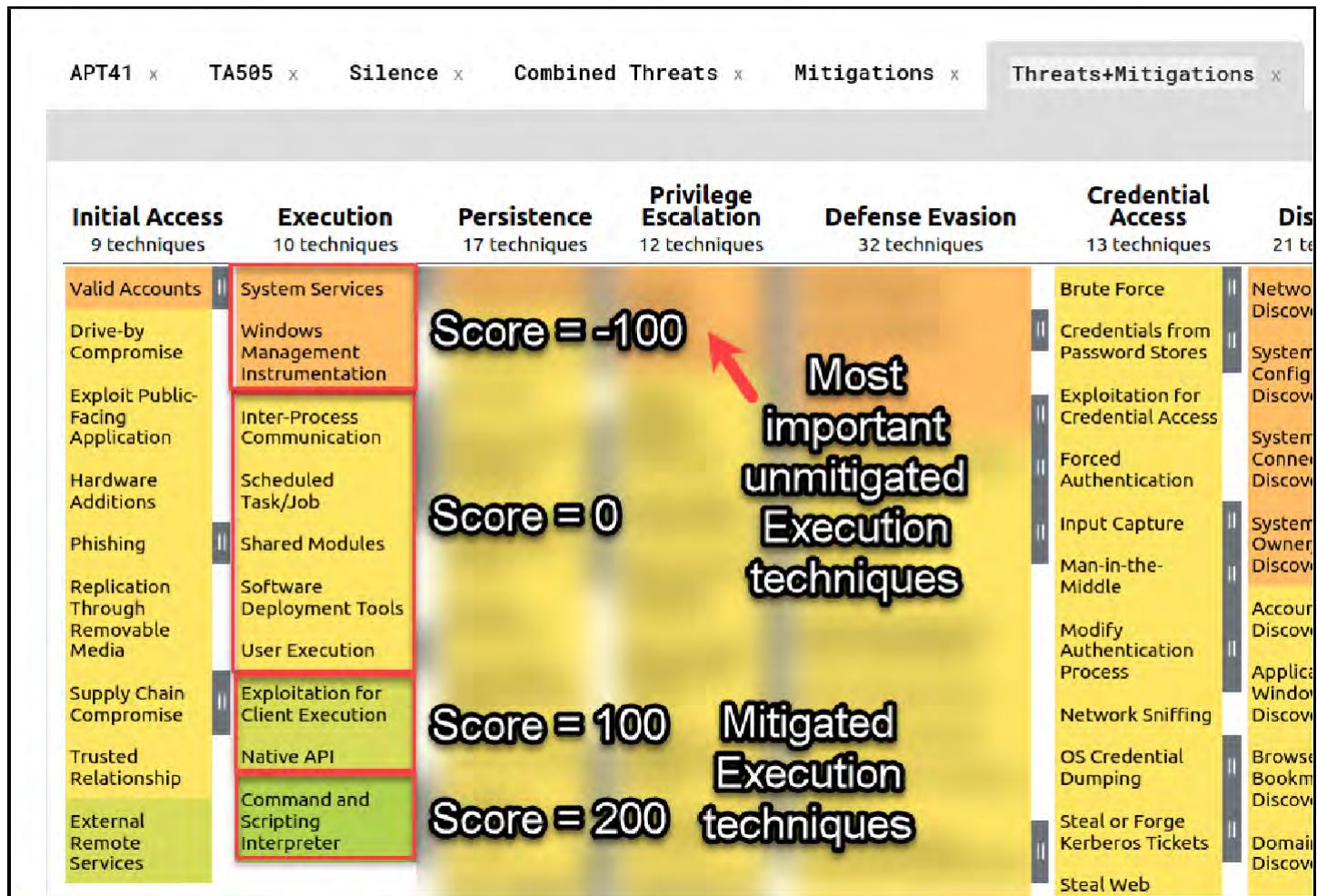
Note

Your final output may or may not match the screenshots here depending on the exact mitigations and threat groups you selected.

Here is how to interpret all the possible scores that will result in this layer:

- **Max negative score** = This technique applies to *all* entered threat groups, and *is not* mitigated. **This is the most important blind spot you have and should be colored RED.**
- **Other negative score** = *Some* threat groups use this technique, and it *is not* mitigated.
- **Zero** = No threat groups entered used this technique, so the mitigation 10/-10 was multiplied by zero. This is not something threat intelligence says you need to focus on.
- **Other positive score** = *Some* threat groups use this technique and it *is* mitigated.
- **Maximum positive score** = This technique applies to *all* entered threat groups but *is* mitigated by one or more controls, meaning whatever control mitigates this is **one of your most important mitigations**. These should be colored GREEN to designate they are covered.

In short, green is good, red is bad, as we've used on other layers. Here's what we would see if you used the examples throughout this exercise. It tells us that as a SOC, for execution, the most important items we need to work to mitigate are the items that scored "in the red", in this case, it was "System Services" and "Windows Management Instrumentation". We can now take this information and investigate what we can do to improve our coverage in this area, knowing that doing so will be the best possible use of our time given that we know the most relevant adversaries are likely to use these techniques when attacking us!



Congratulations! You've now made a map of how your known high-risk threats and mitigations intersect, and how important each one is, based on MITRE's threat intelligence built into ATT&CK Navigator. Save this layer as a JSON Navigator layer, Excel file, or SVG as desired.

Taking It Further

In this exercise, we focused on matching mitigations with detection methods. In the future, MITRE will be taking this tool even further as they have promised to revisit and further organize the "Data Sources" listings that are already stored in ATT&CK and standardize them in the same way they recently added Mitigations.

ID: T1189

Tactic: Initial Access

Platform: Windows, Linux, macOS, SaaS

Permissions Required: User

Data Sources: Packet capture, Network device logs, Process use of network, Web proxy, Network intrusion detection system, SSL/TLS inspection

Contributors: Jeff Sakowicz, Microsoft Identity Developer Platform Services (IDPM Services); Saisha Agrawal, Microsoft Threat Intelligent Center (MSTIC)

Version: 1.1

Created: 18 April 2018

Last Modified: 11 October 2019

That means soon we will likely not only be able to show mitigations, but hopefully will be able to multi-select sources of host and network data the SOC collects and also create layers that show visibility into detecting these tactics as well. Watch the MITRE blog and Navigator GitHub closely as this is one of the near-term items on the road map.

What else can be done with this application? There are plenty of paths forward!

- Use the MITRE [Threat Report ATT&CK Mapping \(TRAM\)](#) tool for automatic technique extraction from threat intelligence reports
- Have your threat intelligence team (if you have one) or analysts keep a consistently up to date version of each threat group layer
- Map detection capability into a new layer and combine with threat group knowledge
- Further refine by adding *fidelity* of detection using different scores
- Use for red team testing and tracking (more on this later in the course)
- Mapping incident response sightings of each tactic back to each tactic to improve focus with real data from *your* environment.

Now that you understand the main functionality of the tool, you can use Navigator to chart out any data that can be helpful with creating an offense-informed defense.

Exercise Conclusion – Key Takeaways

In this exercise, you have:

- Mapped out attack techniques used by multiple previously identified threat groups
- Combined all threat group attack techniques into a single visualization that prioritizes the most important attack techniques
- Created a visualization of mitigation controls you have in place
- Created a master visualization overlaying mitigations with techniques used by your high-risk threat groups, giving you a prioritized list of items to address

To shut down the services used for this exercise go back to your terminal window (or open a new one) and enter the commands below:

```
cd /home/student/labs/2.2
docker-compose down
```

You should see the following response, if you do not, please alert your instructor or run the script in the troubleshooting page in the wiki:

```
Stopping navigator ... done
Removing navigator ... done
Removing network 22_default
```

Exercise 2.2 is now complete!

Exercise 2.3 - Writing Priority Intelligence Requirements

Background

In previous labs, we answered the following key questions about our adversaries and our defenses:

1. Who are they and what do they want? ([Exercise 1.1](#))
2. How will they try to get those things? ([Exercise 1.2](#))
3. Can we observe the paths they are likely to take to achieve their objectives? ([Exercise 2.1](#))
4. What specific techniques might attackers use to move through these paths? ([Exercise 2.2](#))

This a great foundation for our detection and response efforts, but we also know that the answers to these questions will change over time. In this exercise, we will write Priority Intelligence Requirements (PIRs) to ask additional questions and stay up to date on the tactics, techniques, and procedures of those who would target us.

Recall from the lecture that quality cyber threat intelligence starts with quality requirements. The lack of clear requirements is a common breakdown between SOC teams and intelligence capabilities, whether those capabilities are third party vendors, internal groups, or individuals within the SOC itself. Often the main reason for this is many SOC analysts are not used to writing intelligence requirements.

Strong requirements are:

- *Singular*: focused on one question only
- *Atomic*: specific to a particular fact or event
- *Decision-centric*: should support a decision or specific improvement
- *Timely*: within the timeframe that the intelligence will be useful and actionable

In a defensive context, requirements can cover anything from attacker tools and TTPs to major events or corporate activities.

Objectives

- Capture key questions based on your adversary research and defensive planning
- Consider leadership concerns in CEO's Intelligence Requirements (CIRs)
- Develop Priority Intelligence Requirements (PIRs) to keep your detection strategy up to date
- Validate, apply and provide feedback on the intelligence your team receives

Exercise Preparation

This exercise is completed in your MGT551 Linux VM, if you have trouble with the setup, see the troubleshooting information in the wiki, or reach out to an instructor or SANS support.

Launch the MGT551 Linux VM and log in.

```
- LOGIN = `student`  
- PASSWORD = `mgt551`
```

Once you are at the Linux virtual machine desktop, you are ready to proceed with the exercise.

Exercise Steps

Defensive Planning and Key Questions

Based on the planning work done in previous labs, we now have at least a high-level awareness of the types of adversaries, if not the specific threat actors, that might target our organization. We have also identified the most common TTPs seen from those threat actors based on open source intelligence. But there are limits to what this point-in-time intelligence can do for us.

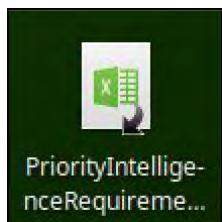
Brainstorm questions

Review the Attacks and Mitigations layer you created in ATT&CK Navigator in [Exercise 2.2](#). Consider the following questions:

1. Is there anything we have missed that should be here?
2. Is there anything we aren't aware of that might change what is here?
3. What might this list look like next week or next month?

As an operations leader, these questions probably occupy you a fair bit. Translating them into intelligence requirements can help you leverage the resources at your disposal to get some specific, and actionable, answers.

Load the template file we will use in your virtual machine by clicking the `PriorityIntelligenceRequirements.xlsx` shortcut icon on your desktop. This will load a blank template file that you can use for this exercise:



Note

This file is stored in `/home/labs/2.3/`.

Enter any questions you might have based your own output from Exercise 2.2 in the top row. We will revisit and refine these questions in the next step, and provide example questions if you have trouble coming up with any.

Info

CIRs, or **CEO's intelligence requirements***, is a play on a military acronym that describes high-level, strategic security requirements for the organization. These are usually specific to the organization, not the adversary, and are unlikely to change from one month (or even one year) to the next. They are also usually open-ended; for example, "what is the probability that we will experience a major security incident in the next x years?", or "will we be able to recover from a security incident without losing capability or business value?". Since these are strategic, long-term intelligence requirements, we will not re-write them every time we have tactical questions we need answered. However, before we write Priority Intelligence Requirements, we should consider our CIRs to ensure they are aligned and that the threat intelligence we are asking for will help us answer the bigger questions in the long term.

Write Intelligence Requirements

Now it is time to refine the high-level questions from the previous step and express them as intelligence requirements. In the PIR spreadsheet, you will see various items listed in the left-hand column. We can use these as "dimensions" to classify and focus our questions:

- **Tools:** tools attackers are likely to use and artifacts they may leave behind
- **TTPs:** attacker tactics, techniques, and procedures as they can be observed in our environment
- **Actors:** threat actors likely to target our organization and key assets
- **Events:** internal or external events that may shape the threat landscape for our organization; for example, increasing our exposure or drawing new threat actors
- **Publications:** reporting, either internal or external, that shines a light on additional activity or causes an adversary to change their behaviors
- **Assets:** business and/or IT assets that may be targeted by our adversaries
- **Goals:** what goals are we trying to achieve by getting the answers to these questions?

	A	B
1	Questions	<i>Question 1</i>
2	Tools	Example: What tools are utilized in the TTPs we have identified?
3	TTPs	Example: Are there TTPs utilized by these actors that we have missed?
4	Actors	Example: Are there threat actors not listed here that have been observed targeting our sector or key assets?
5	Events	
6	Publications	
7	Assets	
8	Goals	

Tip

The goals portion of your PIR matrix is where you can tie your requirements to CIRs.

Looking at the questions we've come up with from the perspective of these dimensions should help us refine our questions a bit further. For example, consider this question:

Are there any new and emerging tools that these actors might use in the future?

This is not a terrible question as far as intelligence requirements go, but it falls a bit short of our atomic, decision-centric, and timely requirements. We can refine it by adding some additional parameters:

What new and emerging open source tools have been utilized by threat actor x during the time they have targeted our sector?

And we can add a follow-on question:

What artifacts do those tools create in the target environment?

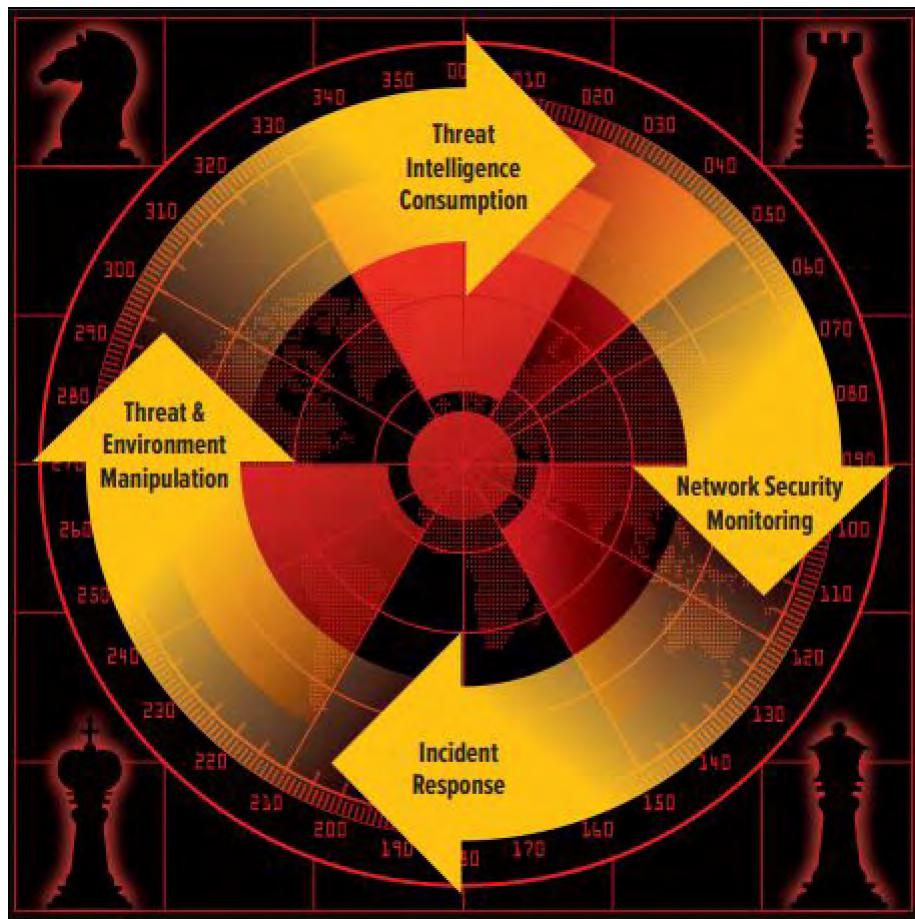
If you have come up with your own questions based on Exercise 2.2, walk through the same thought process to refine them and fill out requirements for each dimension. Now that we have broken down our questions into specific requirements and refined them as needed, we can share our PIRs and await the response, which hopefully comes in the form of actionable intelligence!

Tip

Priority intelligence requirements should be accessible by the security operations team and the intelligence team and updated regularly. PIRs are not a point-in-time exercise, but a set of guiding questions that drive intelligence collection activities over time.

Review and Apply Intelligence

Let's jump forward and assume that you have shared your PIRs and received some intelligence in return. This intelligence may have been gathered from a variety of different sources, including your own internal telemetry. Recall the operational frameworks we discussed in the lecture, like the Active Cyber Defense Cycle and F3EAD; use these frameworks to apply your intelligence in asset identification, security monitoring, and incident response contexts.



Note

The ease of application of your threat intelligence is not a topic that should be overlooked. Discuss the method of sharing as well as the format of threat intelligence with your providers to ensure your team receives the appropriate context and, where applicable, ability to apply indicators directly to automated detections.

Provide Feedback

Remember, the intelligence cycle does not end when intelligence is applied. Review your PIRs given the new information you have received plus any resulting detections or incident response, and adjust or further refine them as needed. Closing this loop will ensure that the SOC team continues to receive high quality, actionable intelligence. If the intelligence need not meet the requirements you provided, let your intelligence team know and discuss ways they can improve. If you are tracking incidents by detection mechanism, you should be able to trace these metrics back to the intelligence products behind that use case.

Exercise Conclusion – Key Takeaways

In this exercise, we have:

- Asked questions about our adversaries and their capabilities based on what we know from our initial defensive planning and activities observed within our environment
- Broken down those questions into priority intelligence requirements
- Considered format and application of the responses to our requirements
- Provided feedback on intelligence products we have received

Exercise 2.3 is now complete!

Exercise 3.1: Capacity Planning

Background

Capacity planning is an incredibly important part of being a manager - but in a situation with so many variables, where do you start? When faced with this task, many managers become incredibly stressed due to the perceived inability to produce any realistic feeling estimates, and the consequences of being wrong can be expensive. How can we approach this task with more analytical rigor than simply guessing? That's the question this exercise will answer. Through clearly defining how we expect our analyst time to be spent, as well as using historical data combined with some lesser-known estimation tricks, we can produce a well reasoned answer to this seemingly incredibly complex question.

Objectives

- Consider the goals for time usage you have for your analysts
- Learn how to estimate workload based on both historical data and quantitative methods for estimation
- Develop an estimate for the workload that you will be presented with in terms of alert count and time to complete those alerts on average
- Compare your ideal goals for analyst time usage to a well-developed guess on actual time required to get that work done

Exercise Preparation

This exercise is completed in your MGT551 Linux VM, if you have trouble with the setup, see the troubleshooting information in the wiki, or reach out to an instructor or SANS support.

Launch the MGT551 Linux VM and log in.

```
- LOGIN = `student`  
- PASSWORD = `mgt551`
```

Once you are at the Linux virtual machine desktop, you are ready to proceed with the exercise.

Exercise Steps

Capacity planning is a complex task that is hard to get precisely correct due to the varying internal environment, external threat conditions, and changing levels of automation that occur throughout time. However, getting a sense of the work capacity of each person and the workload that will be presented to us can help us approach a reasonable answer that can be tweaked over time. Throughout this exercise, we'll use multiple estimation and numerical methods to produce the best

possible answer we can with the data that is available. Even if we can't get to a completely precise number, there are many ways we can put bounds on the problem, giving us a sense of the constraints and expectations we'll need to work within, and if nothing else, we'll look to define those problem space boundaries through this exercise.

Estimating Available Capacity

In this first step of the exercise, we will take the first step in attempting to figure out how big our team will need to be by estimating how much time each person has for different types of work, and what our goals are for their time usage.

"What are your analysts doing all day?" While this may initially sound like a fairly straightforward question, breaking items down into specific activities helps us more accurately understand what the average employee's day actually consists of. For the average SOC during the average week, some activities may include:

- Overhead and more
- Meetings
- Training
- HR paperwork
- Vacation
- Sick days
- Coffee /bathroom breaks, etc.
- Engineering / Project Work (one time - making existing services better style work)
- Fixing issues
- Automation Development
- Process Improvement
- New tool /software /security appliance deployment and tuning
- Operational Activity (Threat Detection and Response)
- Threat Hunting
- Detection Engineering
- Alert response
- Incident response

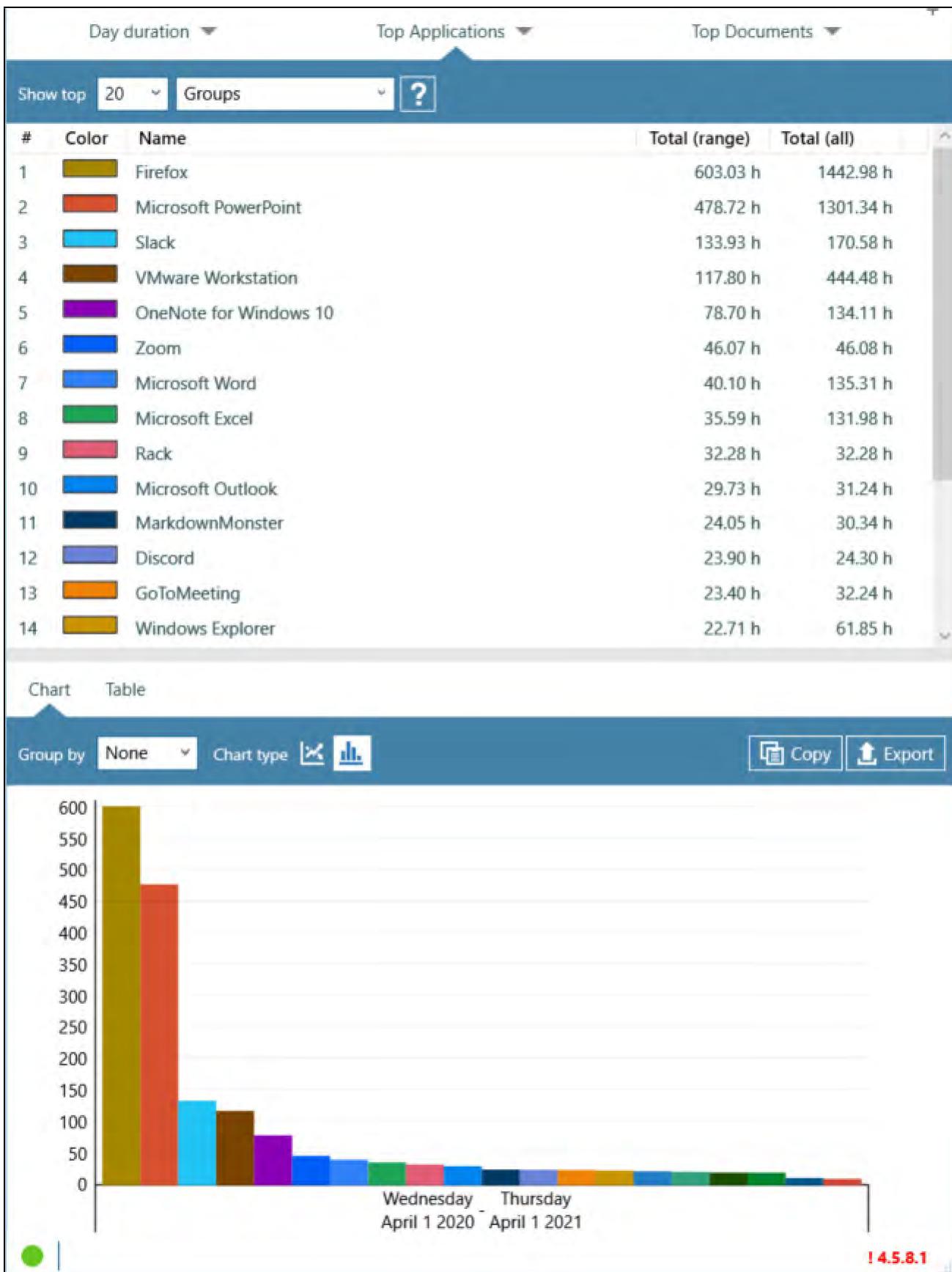
Given the diversity of tasks, this raises two questions:

- How much time *should* each person be spending in each category, in an ideal world?
- How much time are you *actually* spending on each task?

While the second question might take a little bit of tracking and observation, or at least a look back at people's calendars, the first question we can jump into a spreadsheet and easily get a ballpark estimate.

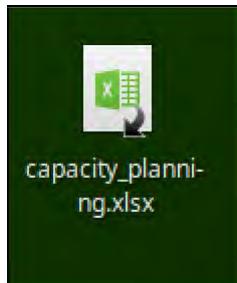
Tip

If you want to track what you (or any willing participant) does with their time in granular detail, check out tools like [ManicTime](#) or [RescueTime](#), which are agents that will run in the background on your computer and give you a highly detailed breakdown of exactly how your time was spent. These automatic time tracking agents will report computer usage by time, application, and even which files and names of the active window /website you had open. The results can be highly interesting. Curious how a SANS Author spends their time? See the pic below courtesy of his ManicTime agent.



Let's jump into a spreadsheet and take a first crack at how much time our analysts actually have available to respond to alerts that require manual intervention on a daily basis.

To begin, open the file `/home/student/labs/3.1/capacity_planning.xlsx` from the shortcut on your VM desktop.



Tip

Alternatively, if you'd like to, you can click and drag it out of your virtual machine to your host machine and open it in Microsoft Excel. This may provide a much more familiar experience.

Once the document is open, select the first tab "Time Available". You should see a table as shown below.

Time Expected per Week	Goal / Expected	Lower Bound (5%)	Upper Bound (95%)
Overhead and more			
Meetings	4	2	8
Training	0	0	0
Vacation	0	0	0
Sick days	0	0	0
Coffee, chatting, bathroom breaks, etc.	0	0	0
Total	4	2	8
Engineering / Project Work			
Fixing broken tools / software	0	0	0
Automation Development	2	0	2
Process Improvement	2	0	2
New tool / software / security appliance deployment and tuning	0	0	0
Total	4	0	4
Operational Activity			
Threat Hunting	0	0	0
Detection Engineering	0	0	0
Incident Response	0	0	0
Total	0	0	0
Total Time Accounted For	8	2	12
Hours Left for alert response per week	32	38	28
	Goal / Expected	Best Case	Worst Case

In this file, we have a breakdown of the items you might expect someone to spend their time on throughout a week and a place to enter an estimate in hours for that activity. Our goals in this step are to:

1. Understand where we expect/want time to be spent
2. Extract a realistic amount of time that each analyst might be able to dedicate to alerts

Follow the instructions included on the sheet to fill out the estimates at the desired or known level of detail. The factor we are trying to determine by subtracting the other pieces of our analyst's day is the time available to spend on alert triage and investigation, which will be shown on the bottom row of the table once the numbers are entered.

Here's an example of a filled out sheet you can use if you are just following along and want to fill in the details later. It is also stored in the lab folder at /home/student/labs/3.1/capacity_planning_example.xlsx

Time Expected per Week	Goal / Expected	Lower Bound (5%)	Upper Bound (95%)	Description / Notes
Overhead and more				
Meetings	Hours	Hours	Hours	All non directly security-related tasks Consider all repeating team meetings + likely one-off meetings per week
Training	5	2	8	
Vacation	0	0	1	
Sick days	0	0	0	Not considered in estimate since most weeks are 0
Coffee, chatting, bathroom breaks, etc.	0	0	0	Not considered in estimate since most weeks are 0
Total	9	2.5	15	
Engineering / Project Work				"One time" improvements, "making existing services better" work
Fixing broken tools / software	Hours	Hours	Hours	
Automation Development	0	0	1	
Process Improvement	2	0	2	Desired time set aside for automation dev each week
New tool / software / security appliance deployment and tuning	2	0	2	Desired time set aside for improvements each week
Total	4	0	5	NA - not in scope for analysts
Operational Activity				Ongoing security-related work that has no "end"
Threat Hunting	Hours	Hours	Hours	
Detection Engineering	4	4	4	Expected 4 hours of threat hunting per week
Incident Response	0	0	0	NA - not in scope for analysts
Total	4	4	4	Not considered in this estimate
Total Time Accounted For	17	6.5	24	(Assumes 40 hour work week)
Hours Left for alert response per week	23	33.5	16	Are any of these negative?
	Goal / Expected	Best Case	Worst Case	

This example shows that analysts may actually have roughly 23 hours of their 40 hour work week actually available on average with a lower estimate of 33.5 hours per week, and a worst case scenario of 16 hours per week.

With this information in hand, we can move on to the next step.

Estimating Alert Count Workload

Now that we have an idea of how much time might be *available* per person, let's look at it from a different angle - how much time we will *need* to address the alerts we expect to create. To estimate the workload we must estimate **alert count** and the **average time required for those alerts**. This is where things may seem extremely difficult, but using some clever numerical estimation methods combined with historical data, it is likely you can get a better result than you might expect.

When it comes to capacity planning, it's helpful to know what happens on the average day, but we also need to consider the extremes. Therefore, the questions we set out to answer in this step are the following:

1. What is the **average** number of alerts we expect to see a day
2. What is the **range** of alert counts that can be reasonably expected on most (90%) of days?

Initially, there are two ways to approach this problem - using historical data, and making a rational estimation based on the facts you do have available. You can use either depending on your situation, or both. If you're an established SOC, you may know this number right off the bat and can go with the historical data method below. If you are brand new and have no historical data, you may want to skip down to the "Using Estimation" section below.

Using Historical Data For Average Count

If you have operated your SOC for a while, hopefully you have some idea of the number of alerts you analysts deal with in a manual way each day. That's fantastic, write that number down because you already have that answer solved, but remember it's only part of the problem. Estimation of Alerts per Day Requiring Human Interaction: _____

In order to do capacity planning, we may not only be wondering if we're going to be covered on the *average* day, but rather what things might look like on high or low volume days as well. In order to do this, you'll need to look at the range of values you might expect to see.

Since you already have your value, you can either skip down to the "Alert Count Range" section, or continue on reading the "Using Estimation" section next in case you'd like to refine or double check your estimate. The "Rule of 5" mentioned below is a highly useful rule that you may be able to use elsewhere, and a tip you don't want to miss.

Using Estimation

If you don't have a good sense of the average number of alerts your SOC might receive because your SOC is brand new, we have to approach this problem in a different way. You might initially think solving this problem in a meaningful way is impossible, but you likely know more, and need less data than you might think. Using some of the methods described by the FIAR institute and author Doug Hubbard of "How to Measure Anything" we can apply concepts like the "Rule of 5", and calibrated estimation techniques to significantly narrow down the possibilities. As Doug Hubbard says "There is literally nothing we will ever need to measure (estimate) where our only bounds are negative infinity to positive infinity." In other words, you may not get a perfectly precise and accurate answer, but can certainly do much better than saying "I have no idea".

One way to approach this problem is the "Rule of 5", if you have at least 5 days of data you can pull from, you may be able to significantly bound the problem space.

Here's the rule:

"There is a 93.75% chance that the median of a population is between the smallest and largest values in any random sample of five from that population." (How to measure anything in cybersecurity risk, Hubbard & Seiersen - Wiley - 2016)

Applied in this scenario, if you can draw on at least 5 random days in the past and count how many alerts you needed to address on those days, you can be nearly sure that the true average of alerts you can expect to have to work is between the highest and lowest values of those samples.

Why does this work? Probabilistically speaking, every day is like a random drawing where the number of alerts produced that day will be centered around the true average (whatever that happens to be). That means every day there is a 50% chance the number of alerts you saw that day was either above or below the true average. If you draw 5 randomly

sampled days, what are the chances you'd draw samples that were *all* above the average or *all* below the average? The same chance of you flipping a coin 5 times and having it come up heads or tails 5 times - not that likely. The same math applies here and tells us, with only 5 days data, we can assume the true average is somewhere in between.

For example, lets say in the 5 randomly selected past days you experienced, 5, 8, 14, 11, and 9 alerts that needed to be manually investigated. That means your real average has a 93.75% chance of being bewteen 5 and 14 - not that hard, right? Has your SOC existed for at least 5 days? If so, you just went from "I can't guess at all", to having the number down to a range of 9 with a 93% confidence, that's pretty awesome! If you have *more* data than 5 days, of course you can extend the math behind this rule and gain even more confidence. It may not bring to a single number, but at least you have a potential range for the average.

If you now have a number in hand, you can move on to the next "Alert Count Range" step and either pick the middle of this range as the **average**, or play "worst case scenario" and pick the top number (14 from our example). If this method did not help you come to an **answer**, continue on.

If you still don't have a number because you literally have no history at all to go on, or work with different clients in multiple SOCs, let's bound the problem as best we can using [claibrated estimation](#).

The steps are as follows:

1. **Start with the absurd** - First think of what numbers you know are absolutely not correct. Could you create 0 alerts? 1M alerts? How far can you bring the upper and lower bounds in before they no longer seem in the realm of completely impossible? Start here to get to boundaries that are no longer completely implausible. As an exampole, perhaps you think that you'd have no less than 2, and no higher than 100.
2. **Eliminate highly unlikely values** - Consider if there are any values that statistically just don't feel right. If you have a gut feel that 10 alerts per day might be the number, what kind of day would it take for 100 alerts to occur that need to be investigated, can you think of a scenario like that? (Seeing the ranges in the next step may help you get a grasp on this)
3. **Reference what you know to narrow the range** - Have you had any days worth of data at all? Do you know how many use cases or signatures you have active? Are there any [industry studies you can find on alerts per day per analyst](#) that may help add information to your estimate? (this link indicates that 93% of SOCs saw 50 or less per person per day). This is the step in which to consider these bits of information.
4. **Use the equivalent bet method to gauge your confidence** - Finally, with the bounds you have now settled on, consider if you are more or less than 90% confident with this test: Would you rather spin a wheel with a 10% chance of winning \$1000 if the needle lands on the 1 of 10 possible spaces marked "WIN!"? Or would you rather have a magical oracle tell you the true value of your "average daily alerts", and be given the money if the correct number was within your range? If you are not ambivalent about which scenario to choose, you are not yet at a 90% confidence in your guess. Adjust the numbers until these feel like the **same** chance of winning to you.

Do you have a reasonable 90% confidence interval estimate for alert count at this point? Pick the highest of the range if you want to be ultra conservative in your time estimates, or somewhere in the middle of the range if you want to go with a more normal scenario, and let's continue.

Alert Count Range

Now that you've come up with a working average, the hard part is mostly over. It's coming up with that estimate that gives people the most struggle, from here the rest of the math flows rather easily.

Given a reasonable 90% CI average - which we're assuming you now have as a result of historical data or a best guess estimate, go to your spreadsheet and go to the next tab named "Event Count Estimation". On this page near the top, there is a box to fill in with your assumed alert count average.

Fill in that number now with your chosen average alerts per day from the previous step. We'll show the estimates for an assumed 15 alerts per day in these screenshots.

Note

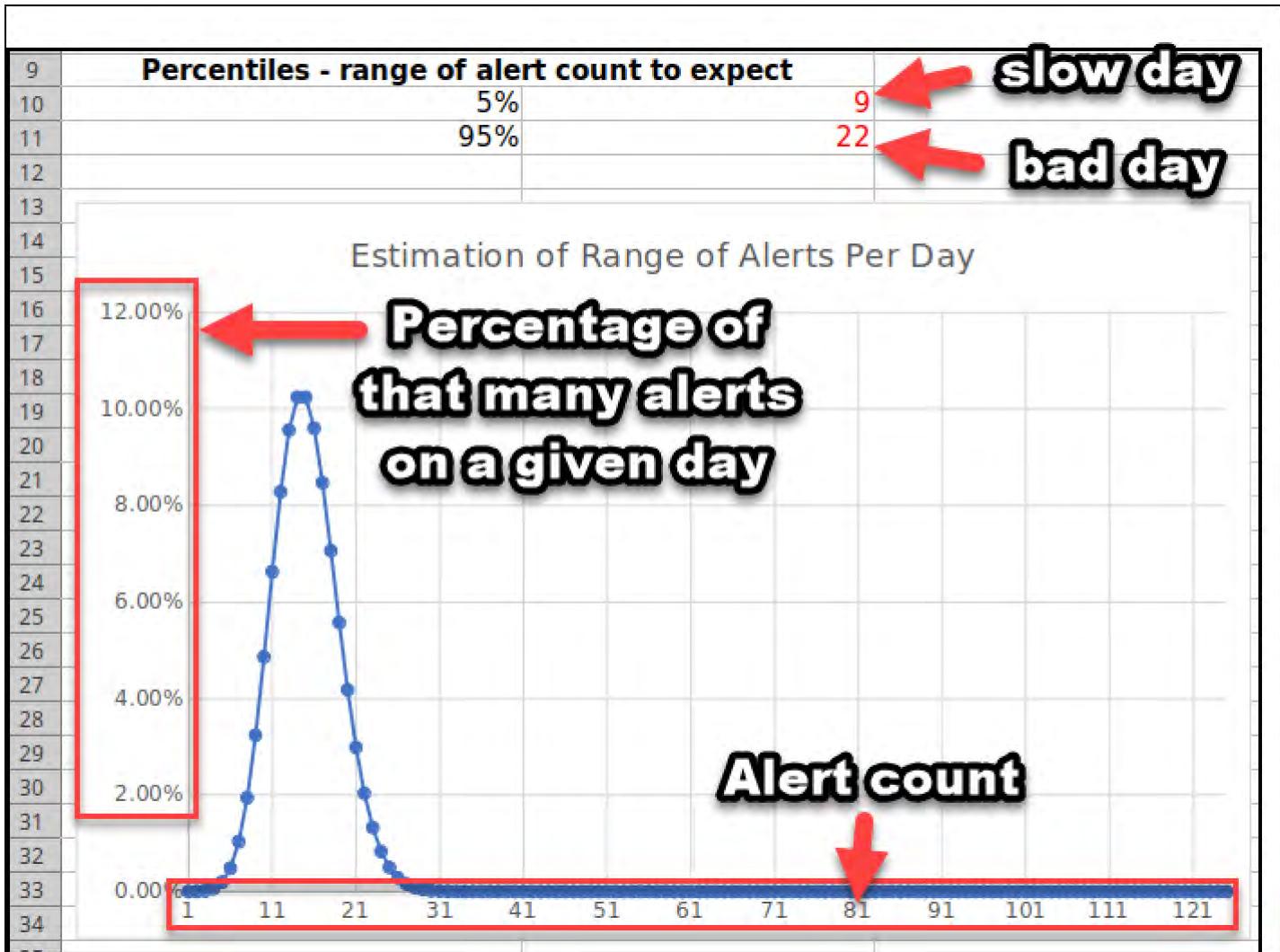
If you are using the numbers from the example (15 alerts), your results will match the screenshots below. When putting in your own numbers, the results will be different than what is shown. These numbers are pre-entered in the `capacity_planning_example.xlsx` version of the file, the other file has blanks for all variables for you to fill out.

3	
4	How many alerts that require manual attention do you typically see per day?
5	15 Average alerts per day

Note

Due to limitations of this calculator you should **not** enter a number higher than 100 in this box or the calculations will become inaccurate. If you do actually investigate more than 100 alerts per day, then calculate the number for half a day or less until you are under 100 then multiply the results by 2 to get to the number for a full day.

As you fill in that number, you should see the numbers and graph below recalculate. If you filled in 15 as shown, here are the results you would receive.



As mentioned in the slides, the [Poisson distribution](#) is a discrete probability distribution that can be used to produce these estimates. The graph you see on the slide is the probability density function for a randomly occurring phenomenon (alerts arriving at the SOC) within a set period of time (one day), given a known average rate of occurrence (from the previous step). In our case, plugging in our previously derived average number of events per day can give us a chart that shows what percentage of days we should expect to see the number of alerts on the x-axis. (Remember, this assumes the alerts are truly random, independent events, which they are not, but it's a much closer model than guessing).

You should also see a new adjusted number for the *range* of alerts you might expect to see per day, assuming your alerts are driven by independent random events that can be estimated using a Poisson distribution, as discussed in class. For instance, in this chart, if the assumed average alert count per day is 15, then in 90% of days, we can expect a range of 9 to 22 alerts to appear, with only 10% of days being more or less than that. (These numbers are derived through a Monte Carlo simulation happening on the right side of the spreadsheet using 1000 simulated days at your chosen average.)

Think about this for second, we just moved from a single average number to a range of alerts we can expect on 90% of days. This is *much* more useful information for capacity planning than an average alone as it numerically describes our

workload on both easy and harder days, and also draws a limit on what we might reasonably expect if we look at the very peak of the graph or the Monte Carlo chart results. For 15 events average per day, there is almost no chance we will have higher than 32 alerts that require human attention in one day. We now can say not only do we expect 15 alerts on average, but that the range will vary from 9 to 22 on 90% of days, with a peak of 32 and a minimum of 3-4 per day - very useful information, hooray for math!! With this info in hand, on to the final step.

Estimating Average Time Required Per Alert

The final step for capacity planning is converting our event count into an expected time. The math is relatively straightforward - again how long does the average event take, and what is the range of possibilities? If we can come up with reasonable estimates, we can again apply a Monte Carlo analysis to give us the range of time we can expect we'll need on most days, and at the maximum and minimum, to cover all the alerts.

For this estimation, we will use a different distribution to model the time required to work our alerts. As we know, many alerts can be triaged and investigated quickly, but there are also some that take longer, and even fewer that may take a very long time. One model that fits this description is the log-normal distribution. While there are other options (normal distribution, beta distribution, and more) in the author's opinion, the log-normal distribution captures that most alerts will gather around a certain average time to complete, with a few rare outliers (see Doug Hubbard and Richard Seiersen's [book](#) for a description of other distributions and their uses).

Our approach to simulate how long *your* alerts will take to work through on any given day is therefore fairly straight forward. We will run many simulated days utilizing your expected range of alerts that could appear. For each alert on each simulated day, we will then have Excel randomly select a time it might take to complete that alert from a log normal distribution with the appropriate boundaries (90% CI) set. After running this simulation many times over and summing up the total time taken to complete all alerts, we will have our range of answers - the minimum, maximum, and average expected time to complete alerts! Lets jump into it.

Go back to your spreadsheet and click to the 3rd tab labeled "Time Required Estimation".

In this sheet you have 3 numbers to fill out highlighted in yellow boxes

1. Your estimation for the what the 5th percentile is in terms of speed for analysts needing to manually triage and investigate an alert (probably in the single-digit minutes)
2. Your estimation for the 95th percentile of cases - how long might some of the longest alerts take (not the extreme outliers though, remember this is 95th percentile.)
3. The average alert count from the previous slide, you can also use the weekly count (if it is less than ~100).

Note

Due to limitations of this calculator you should **not** enter a number higher than 100 in this box or the calculations will become inaccurate. If you do actually investigate more than 100 alerts per day, then calculate the number for half a day or less until you are under 100 then multiply the results by 2 (or as necessary) to get to the number for a full day.

As soon as you enter data, you should see the numbers recalculate adjusting to your input. If you'd like to re-run the simulation multiple times, press the F9 key.

Here is an example, carrying through from the previous 15 alerts per day average, and assuming our 5th and 95th percentile times are 5 and 45 minutes respectively.

Log-Normal Distribution Numbers							
Lower Bound (5%)	5 Minutes						
Upper Bound (95%)	45 Minutes						
Average COUNT of alerts per day or week (from previous tab)							
15 Alerts							
(Do NOT go higher than 100, scale down then multiply results if necessary)							
Results:							
<table border="1"><thead><tr><th>Average</th><th>4.9 Hours</th></tr></thead><tbody><tr><td>5%</td><td>2.7 Hours</td></tr><tr><td>95%</td><td>7.7 Hours</td></tr></tbody></table>		Average	4.9 Hours	5%	2.7 Hours	95%	7.7 Hours
Average	4.9 Hours						
5%	2.7 Hours						
95%	7.7 Hours						

There you have it! What then, are our results for a SOC with 15 alerts on average (modeled with a Poisson distribution) and the time taken to address those alerts ranging from 5-45 minutes, skewing towards the shorter end (in a log-normal distribution)? Give the 250 trials we ran with our Monte Carlo simulation, the average amount of time taken to address all alerts was about 5 hours, with a 5th percentile easy day taking about 2.7 hours and a more high-volume day taking roughly 7.7 hours!

Converting to weekly numbers

Number	Calculation	Weekly
Average	4.9 hrs/day * 7	34.3 hrs.
5 th percentile	2.7 hrs/day * 7	18.9 hrs.
95 th percentile	7.7 hrs/day * 7	53.9 hrs.

Now Let's analyze this information to see how it compares with our available time we calculated in the first step of this exercise.

Comparing Results

You now should have the estimated amount of time it will take to triage and investigate alerts manually on the average day/week, as well as a range of expected values on 90% of days. To wrap this up, we need to go back our first step of the exercise and compare the output of our time estimation with the output of our estimated time available.

In our example numbers we calculated that:

- Our analysts will have somewhere between 16 and 33.5 hours per week available for alerts, with an expected average of 23 hours per week
- Our SOC will have 34.3 hours of alert workload to deal with per week on average, with a 5th and 95th percentile of 18.9 hrs and 53.9 hrs respectively

How many analysts might we need to plan to staff then?

- Average week = 34.3 hrs. / 23 hrs. per person = 1.5 people
- Bad week = 53.9 hrs / 23 hrs per person = 2.3
- Worst case = 53.9 hrs / 16 hrs per person = 3.4 people

And there you have it! Most weeks, assuming these numbers, we'll need 2 people on higher-volume weeks where things are working normally we'll need 3, and if alerts are high volume AND analysts have a lot of demands on their time, you would need 4 to cover everything.

With these numbers you can now choose your approach. Either choose to staff for worst case to ensure you always have enough people, staff for average and be more efficient (and perhaps supplement with outsources help from an MSSP), or anywhere in between. The choice is yours!

While capacity planning may have seemed like a nearly impossible task at the start of this exercise, hopefully we've shown you that with some basic assumptions about the distributions of the underlying data, the ability to draw on historical data, and some estimation tricks, deriving these numbers may have not been as elusive as you first thought!

Taking it Further

Of course this is only a starting point (and you may have not had the exact data available to work with during this lab). As you start/continue to operate your SOC, collect the data related to these questions and see if it matches the predictions and refine the process. Ask your analysts to look at their calendars for the past few weeks to get an estimate of how much time they think they had. You can use your incident management system to pull metrics on alert count and manually triaged and investigated alerts to calculate the real 5% and 95% timings. Any additional info you have will improve your ability to estimate. While predictions will never be 100% accurate, the goal here is to realize that it is very

possible to eliminate a lot of the uncertainty we might otherwise face and zero in on an answer that is "close enough" to answer our staffing and capacity planning questions.

Exercise Conclusion -- Key Takeaways

In this exercise, you have:

- Calculated a detailed assumption and expectation for how much time each person is spending on various activities, and how much time that leaves them to address the varying volume of alerts each day
- Learned how to produce a reasonable estimation for alert count per day, as well as the ranges you might expect to experience day to day in the best and worst case scenarios
- Learned how to use the log-normal distribution to estimate the time required to address alerts in your SOC
- Improved your estimation ability using the "Rule of 5" and the Calibrated Estimation process
- Used Monte Carlo analysis to combine estimates for alert count and time required for alert triage and investigation to simulate workload for your SOC, and how that determines staffing levels necessary to cover it

Exercise 3.1 is now complete!

Exercise 3.2: Structuring, Documenting, and Organizing Use Cases

Background

In this exercise you'll see an example of how to create a structured use case database. While this exercise will leverage Redmine for use case database storage, the same principles can be applied to the solution you use (or will use) in your own SOC. The key takeaways will be to focus on the fields and data items to track, and to enable metrics collection of the entered data.

Objectives

- Develop some example use cases and categorize them 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 SOC data organization

Exercise Preparation

This exercise is completed in your MGT551 Linux VM

1. Launch the **MGT551 Linux VM** and log in.

- LOGIN = `student`
- PASSWORD = `mgt551`

2. Start the Redmine Service

Before starting this exercise, you must start the required services. To do this, open a command terminal from the start bar.



Once the window is open, start the services by entering the following command at the command line:

```
cd /home/student/labs/3.2  
docker-compose up -d
```

You should see output similar to the following, the list order of container startup may vary. If you receive an error message inform your instructor or run the script from the "troubleshooting" page in the wiki.

```
$ docker-compose up -d
Creating network "32_redmine-network" with the default driver
Creating redmine ... done
Creating db      ... done
```

Keep the terminal open in the background, we will use it to shut these services down at the end of the lab.

Exercise Steps

Redmine Orientation

For this exercise we will be using a free and open-source project management tool called [Redmine](#). Redmine is a mature, supported software tool that has been around for years and is used by large organizations all over the world for information capture and storage. While Redmine was primarily designed with software development project management in mind, many teams find it is easy to repurpose for tracking other types of projects as well. With some intentional setup, Redmine can work great for tracking much of the information SOC analysts, engineers, and admins need daily. With a basic understanding of Redmine's layout, object types, and features, it can be used for important tasks such as:

- Act as a use case database to document the details of alerts types analysts may need to triage
- Store critical SOC information in organized documents
- Store commonly accessed files (think policies, procedures, and more)
- Provide wiki-based common knowledge capture
- Track SOC tool/process bugs and improvements that need addressing
- Provide structure file storage in version-controlled repositories
- Facilitate analyst communication through Forums and News posts

Here is a breakdown of the terminology you need to know to operate in Redmine:

Redmine project tracking is broken into Projects

- PROJECTS have a NAME and MEMBERS (users assigned to it)
- MEMBERS belong to one or more GROUPS
- GROUPS of users have one or more ROLES
- ROLES have a set of PERMISSIONS assigned to them
- PROJECTS have one or more TRACKERS used to track different ISSUE types

- TRACKERS have:

- NAMES - Describe what ISSUES type they track (Use Cases, SOC Issues, Improvement Ideas, etc.)

- ISSUES - Unique, tracked items with statuses, notes, etc.

- ISSUES have:

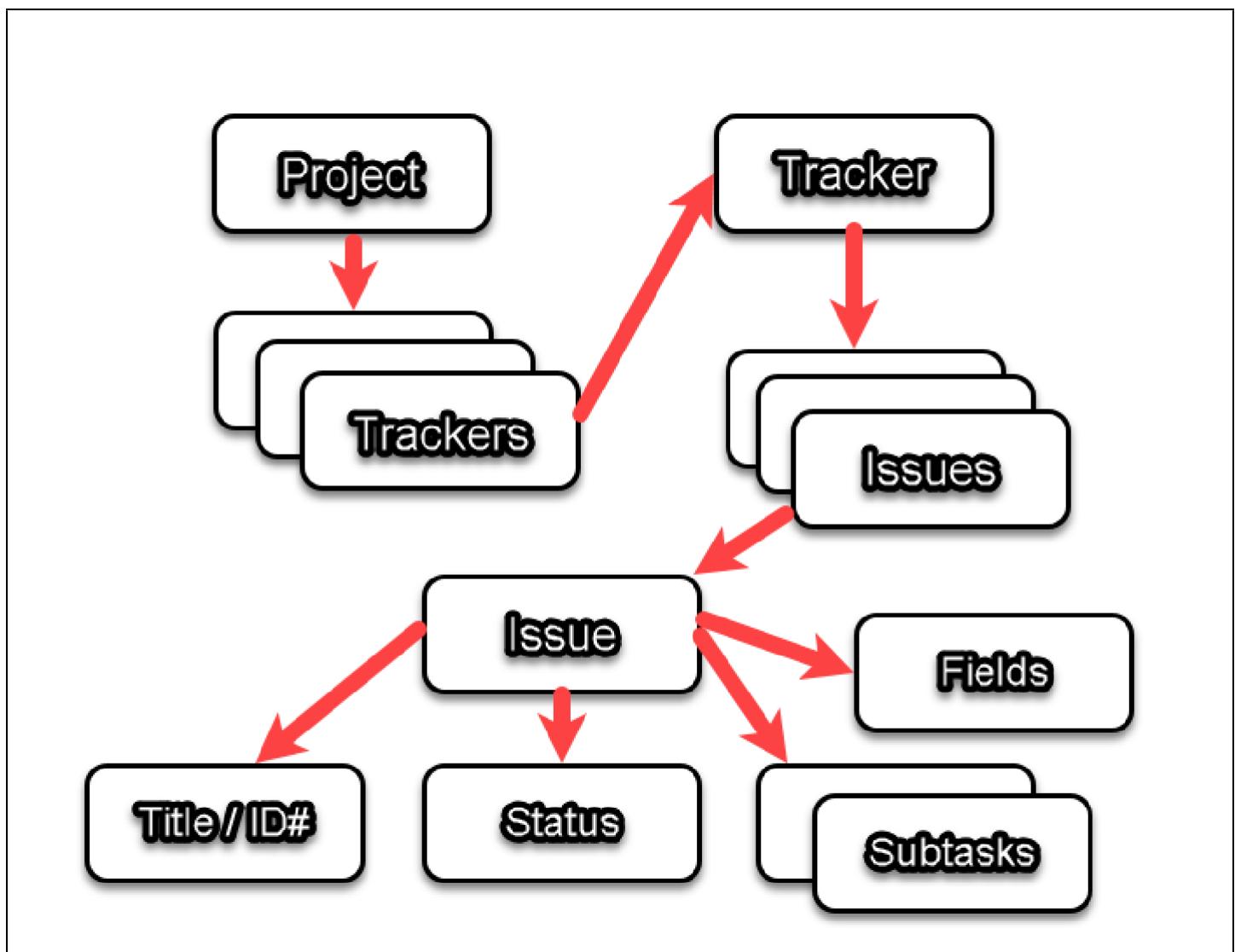
- TITLES - A "Subject" and unique identification numbers

- STATUSES - Transitioned via defined WORKFLOWS

- STANDARD FIELDS - Built-in items associated with all issue types

- CUSTOM FIELDS - Custom fields defined by the administrator

- SUBTASKS - If desired, issues can be nested



In this exercise, we will see how to use a project and issue-based tracking system (whether you decide to use Redmine or not) to track, organize, and make critical SOC information available. Having an organized system of documents and data that is easily accessible makes SOC much more effective and makes life much easier for analysts.

Note

This exercise will begin with Redmine already set up in a possible SOC-oriented configuration. To learn how to set this up on your own from a blank installation, see the appendix at the end of this lab.

Initial Use Case Definition

As a first step towards building our collection and detection capability, think of either a use case you have implemented in your organization already, one that you would like to implement, or use the example provided below. The use case could be anything from detecting unexpected lateral movement attempts, to brute force logins, password spraying, new autorun items, unauthorized data access attempts, and more. If using your workbook, fill in the title below.

My Use Case

For your use case, consider the following attributes. Pencil them in if desired in the table below in your workbook - if using the virtual wiki, no need to write these down now, you will be typing them in the next step.

Note

If you do not have a use case in mind, use something simple like "Brute force login attempts" (the goal of the exercise is to consider which dimensions of the use case you'd like to track, not the complexity of use case itself).

Attribute	Value
Objective	_____
Priority	_____
Primary Data Source	_____
Coverage Required (within your desktop/server estate)	_____
Analytic Pseudo-logic	_____
Potential False Positives	_____
MITRE ATT&CK Tactic Name(s)	_____
MITRE ATT&CK ID	_____
Kill Chain Stage	_____
Any URLs to Reference in relation	_____

For the example use case, we might have answers like the following:

Attribute	Value
Objective	To identify attackers trying to guess their way into a server
Priority	High
Primary Data Source	Authentication logs
Coverage Required (within your desktop/server estate)	Servers, Desktops
Analytic Pseudo-logic	If login failures > 10 within 1 minute, alert
Potential False Positives	Broken service accounts/vulnscanners
MITRE ATT&CK Tactic Name(s)	Credential Access, LateralMovement
MITRE ATT&CK ID	T1110
Kill Chain Stage	Exploit
Any URLs to Reference in relation	https://attack.mitre.org/techniques/T1110/

Note that this is by no means an exhaustive list of what could be defined, it is just a sample of the type of things you might want to track.

Using Redmine to Document and Track Use Cases

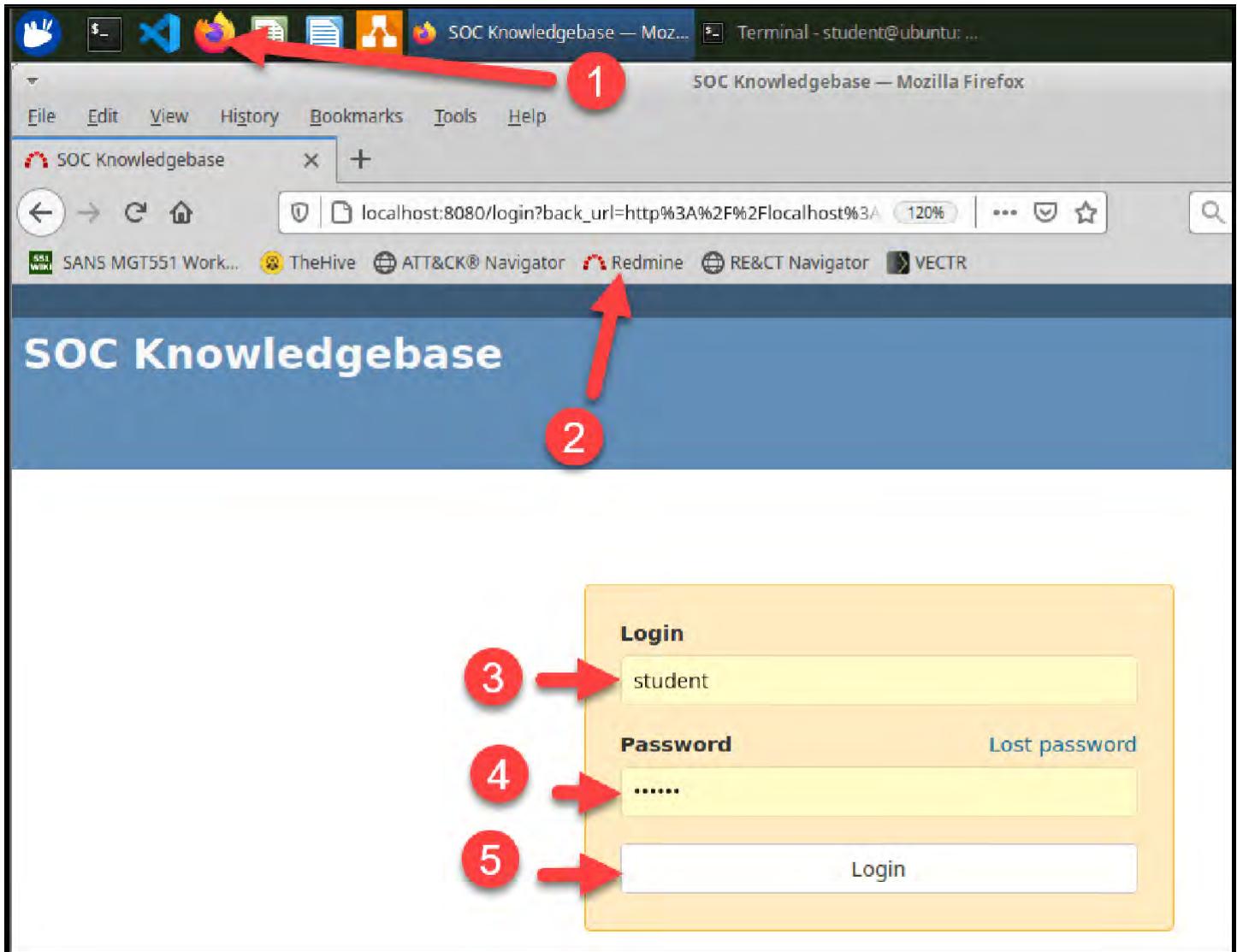
In this step we will take the use case you have just noted (or the example) and enter it into Redmine, using it as a Use Case database. Entering the details for all of our high-level use cases (or in more detail, as will be discussed in a moment) allows us to stay organized and know what we are trying to detect and why. Use case databases with organized custom fields can act as a great reference point for new analysts as well as help managers understand analytic coverage in relation to frameworks such as MITRE ATT&CK.

Entering Your Use Case Into the Database

Note

Once you run the docker-compose command in the beginning of this exercise it may take a minute or two for the Redmine containers to be ready before the page is functional. If you do not see the page in the following steps, wait a few moments before trying again. If the page never becomes available, run the troubleshooting script from the wiki and run the docker-compose command again, or inform your instructor.

To get into Redmine, **first**, open up a Firefox browser by clicking the icon in the virtual machine menu bar and then clicking the Redmine icon in the bookmark toolbar. When you see the login screen login with the username and password student/mgt551 or use the student username credentials saved in the Firefox built-in password manager:



After logging click, click on the "Projects" link at the top of the page. This will bring you to the page pictured in the image below, you should only see the "SOC" project listed. Click on it to enter the SOC project overview:

The screenshot shows a web browser window with the title "Projects - SOC Knowledgebase". The address bar displays "localhost:8080/projects". The top navigation bar includes links for Home, My page, Projects (which is highlighted with a red box and has a red arrow pointing to it), Help, SANS MGT551 Work..., TheHive, ATT&CK® Navigator, Redmine, RE&CT Navigator, and VECTR. Below the navigation bar, the page title "SOC Knowledgebase" is displayed. A blue header bar contains four tabs: Projects, Activity, Issues, and News. The "Projects" tab is selected. A red circle with the number "1" is placed over the "Projects" tab. A red arrow points from this circle to the "Projects" tab in the top navigation bar. In the main content area, there is a section titled "Projects" with a "Filters" dropdown menu. Under "Filters", there is a "Status" field set to "active". Below the filters, there are "Apply", "Clear", and "Save" buttons. A red circle with the number "2" is placed over the "SOC" button in a callout box. A red arrow points from this circle to the "SOC" button in the callout box. The callout box contains the text: "For organization and storing Security Operations Center team information."

Once you've clicked to open the SOC project you can see each of the listed trackers under the "Issue tracking" box and the count of issues in the open/closed state listed in each. You will also see a member list for the project and any news that has been entered recently.

Home My page Projects Help Logged

SOC

Search:

+ Overview Activity Issues News Documents Wiki Files Settings

Overview

For organization and storing Security Operations Center team information.

Issue tracking

	open	closed	Total
Use Cases	0	0	0
SOC Fixes	0	0	0
SOC Improvements	0	0	0

[View all issues](#) | [Summary](#)

Latest news

New Sensors Deployed
Added by Redmine Admin about 3 hours ago

[View all news](#)

Members

Analyst: MGT551 Student

To create a new tracked item from our use case - an "issue" in the terminology of Redmine, click on the "Issues" tab, then click on the "New issue" button as shown below.

Home My page Projects Help Logged in as student My account Sign out

SOC

Search: SOC

+ Overview Activity **Issues** News Documents Wiki Files Settings

1

2

Issues

Filters: Status open Add filter

Apply

No data to display

Also available in: Atom | CSV | PDF

In the following screen, ensure "Use Cases" is the tracker selected.

- If you are using one of your organizations current use cases, fill in the boxes on the screen with the use case information you previously noted.
- If you are stepping through this exercise using the "Brute Force Login Attempt" example use case, copy and paste the following info below in the corresponding boxes to match the picture.

Example Use Case Info:

Tracker: Use Cases

Subject: Brute Force Login Attempt

Description:

This use case is to track analytics that detect attempts to guess a password for a given account by attempting to guess it many different times.

Status: In Development

Priority: Normal

Assignee: MGT551 Student

Objective:

Attackers either internal or external to the environment may try to gain access to a given system by guessing an account's password using a list of default passwords. This will manifest as many rapid login failures and should be investigated when it occurs.

Author: Your name

Primary Data Source: Authentication Logs

Coverage: choose any

(Pseudo) Logic:

If there is a failed login attempt for the same username more than 10 times within 1 minute - fire an alert.

Known False Positives:

- Broken scripts or serviceaccounts
- Vulnerability Scanners

Reference: <https://attack.mitre.org/techniques/T1110/>

MITRE ATT&CK ID: T1110

Compliance Requirement: Yes

Kill Chain Stage: Check Exploit

MITRE ATT&CK Tactic: Check Credential Access and Lateral Movement

Your screen should now match the picture below if using the example, or have your own use case data filled in.

SOC

Search: SOC

+ Overview Activity Issues News Documents Wiki Files Settings

New issue

Tracker * Use Cases 

Subject * Brute Force Login Attempt

Description

This use case is to track analytics that detect attempts to guess a password for a given account by attempting to guess it many different times.

Status *	In Development	Parent task	
Priority *	Normal	Start date	04 / 01 / 2020 
Assignee	MGT551 Student	Reference	https://attack.mitre.org/techniques/T11
Objective	Attackers either internal or external to the environment may try to gain access to a given system by guessing an account's password using a list of default passwords. This will manifest as many rapid login failures and should be investigated when it occurs.	MITRE ATT&CK ID	T1110
Author	John	Compliance	Yes 
Primary Data Source	Authentication Logs	Requirement?	
Coverage	Desktops Servers Cloud Appliances	Kill Chain Stage	<input type="checkbox"/> Recon <input type="checkbox"/> Weaponization <input type="checkbox"/> Delivery <input checked="" type="checkbox"/> Exploit <input type="checkbox"/> Install
(Psuedo) Logic	If there is a failed login attempt for the same username more than 10 times within 1 minute - fire an alert.	MITRE ATT&CK Tactic	<input checked="" type="checkbox"/> Credential Access <input type="checkbox"/> Discovery <input checked="" type="checkbox"/> Lateral Movement <input type="checkbox"/> Collection <input type="checkbox"/> Command and Control
Known False Positives	- Broken scripts or service accounts - Vulnerability Scanners		

Once all the information is filled in, press the "Create" button to save it.

Known False Positives

- Broken scripts or service accounts
- Vulnerability Scanners

Files

Browse...

No files selected.

Create

Create and add another



You should now see the screen for the Issue presented with all of your entered data as shown below. (Note: The ID number assigned to the case may be different.)

Use Cases #1

Brute Force Login Attempt

Added by MGT551 Student 1 minute ago.

Status:	In Development	Start date:	04/01/2020
Priority:	Normal	Reference:	https://attack.mitre.org/techniques/T...
Assignee:	MGT551 Student	MITRE ATT&CK ID:	T1110
Objective:	Attackers either internal or external to the environment may try to gain access to a given system by guessing an account's password using a list of default passwords. This will manifest as many rapid login failures and should be investigated when it occurs.	Compliance	Yes
Author:	John	Requirement?:	
Primary Data Source:	Authentication Logs	Kill Chain Stage:	Exploit
Coverage:	Desktops, Servers	MITRE ATT&CK Tactic:	Credential Access, Lateral Movement
(Psuedo) Logic:	If there is a failed login attempt for the same username more than 10 times within 1 minute - fire an alert.		
Known False Positives:	- Broken scripts or service accounts - Vulnerability Scanners		

Description

This use case is to track analytics that detect attempts to guess a password for a given account by attempting to guess it many different times.

Using and Updating Use Cases in the Database

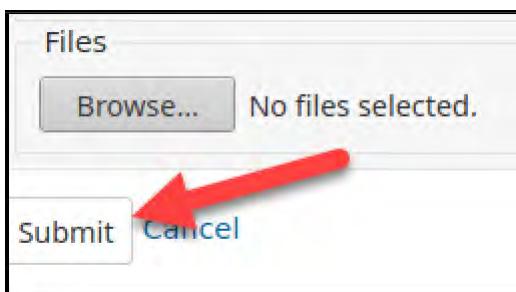
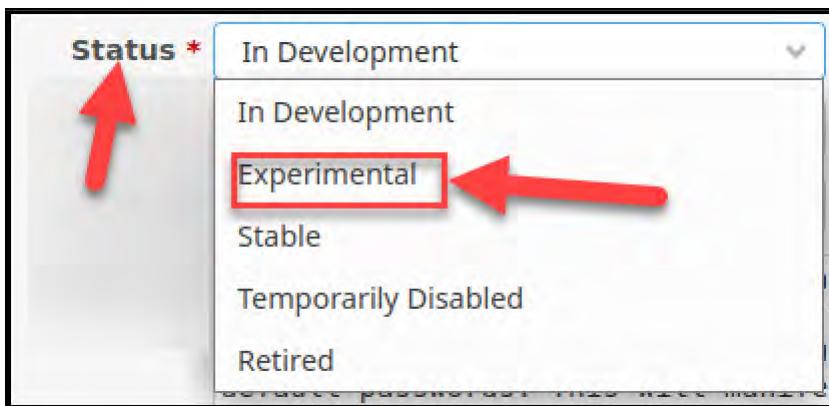
Once your use case is initially documented, there are several actions that commonly occur as the use case matures that Redmine can help facilitate and organize.

USE CASE LIFE CYCLE RECORDING AND MORE

Since use cases have a natural life cycle that can be defined in Redmine using "issue statuses" that moved through customized workflow stages, Redmine can track your use case from idea to testing to becoming a stable use case (and use any terminology you'd like for each step). As your analysts or content engineers develop and improve the use case, the status can be changed by clicking the "Edit" button while viewing the use case "issue".



Change the status to the "Experimental" stage and press Submit at the bottom of the page as shown below - this will simulate what it's like to move a rule through life cycle stages and how the modification is recorded:



Upon making this change, the time, date, and modifier are recorded. This change is now listed under the "History" tab in the issue as shown below.

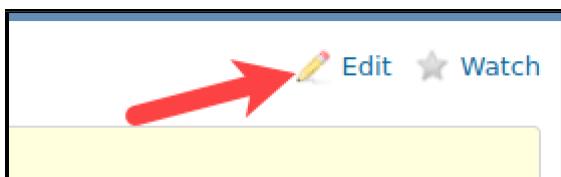


As time goes on and use cases continue to evolve, all changes to use case life cycles will be recorded like this in the Redmine history (as well as any other detail change). If there is any doubt as to when a use case was changed, the record of when, why and who made the change will be recorded in the use case database. This gives a very useful history for each of your key detection capabilities - a crucial resource to have when doing incident response or if something starts to misfire with false positives. These tracking capabilities are not just for property changes however, as you'll see in the next section, Redmine also can facilitate discussion and comments of each individual use case.

DISCUSSION AND MODIFICATION

A frequent occurrence in the SOC is that after implementation, analysts start to notice something wrong with a use case's analytic. Perhaps it is triggering on a false positive situation and needs to be modified. If analysts aren't responsible for the fix themselves, they need a dependable way to pass that information on to those who are. Redmine's Notes feature can provide this function, which creates a "wall" like discussion attached to a given issue.

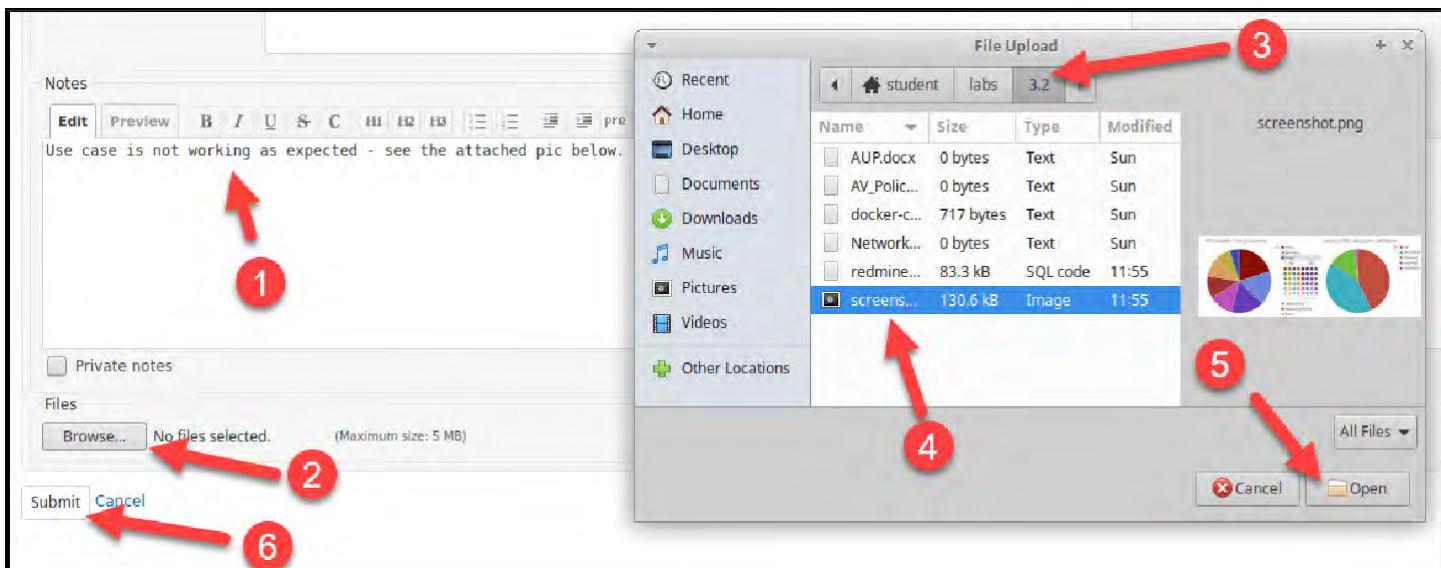
To create a note, first, click on the edit button on your newly created issue:



In the following screen, scroll to the bottom and locate the "Notes" panel. Here you can type any information you'd like to attach to the issue, and even include screenshots by attaching them as a file and referencing them in your comments between exclamation points. Here's an example, type the following in the "Notes" box:

1 Use case is not working as expected - see the attached pic below.

Then, press the "Browse" button and navigate to the /home/student/labs/3.2 folder and select the example picture `screenshot.png` and press open. You should see the file attached to the notes, press Submit to finish your note.



On the Issue overview screen, at the bottom in the Notes tab, you should now see your newly entered note (and screenshot if attached). Here's an example of how the previously shown note would show up:

History Notes Property changes

Updated by MGT551 Student less than a minute ago

- File screenshot.png added

The note contains a pie chart titled "AWS (security) - Non-US Countries" and a screenshot of a dashboard titled "Security (AWS) - Data Source Distribution".

AWS (security) - Non-US Countries

Country	Percentage
China	~30%
Germany	~15%
Japan	~15%
Netherlands	~10%
Republic of Korea	~10%
Brazil	~10%

Security (AWS) - Data Source Distribution

Data Source	Percentage
elb	~45%
VPC_FLOW	~35%
S3Access	~10%
cloudtrail	~5%
cloudfront	~5%

How do the people responsible for making changes get alerted to these notes? The "Watch" function in Redmine can be used to alert interested parties any time there is a change made to one of the use cases that may require their action:

Use Cases #1

Brute Force Login Attempt

Added by MGT551 Student about 3 hours ago. Updated 5 minutes ago.

[Edit](#) [Watch](#)

A red arrow points to the "Watch" button in the top right corner of the note card.

Also, all logged in users can view Activity across all issues by looking at the "Activity" tab in the main SOC project page, which will show updates as shown below.

Activity

From 03/03/2021 to 04/01/2021

Today

03:59 PM Use Cases #1: Brute Force Login Attempt

Use case is not working as expected - see the attached pic below.

MGT551 Student

01:21 PM Use Cases #1 (Experimental): Brute Force Login Attempt

MGT551 Student

01:16 PM Use Cases #1 (Experimental): Brute Force Login Attempt

This use case is to track analytics that detect attempts to guess a password for a given account by attempting to gue...

MGT551 Student

Use Case Organization - Optional Parent/Child Nesting

Redmine can be used in several ways to track use cases, and how you do it is up to you and the process you use in your SOC - there isn't a single "correct" method. One option is to make every single use case an individual issue and just list them all. One potential problem with this, however, is that it can be slightly difficult to manage. There may be apparent overlap between multiple analytics, leaving you unsure whether you should create another entry or not.

To solve this problem the second option is to create additional issues as "subtasks" under a higher-level, more generic "parent" use case. Tracking items this way means you can enter more specific implementation details for the multiple ways and locations you may implement the given use case. Then, when going to create metrics, you can either query all of the parent high-level items, all of the specific child items, or both, depending on what question you're trying to answer.

Here's an example before you do it on your own: Using the "Brute Force Login Attempt" use case we just created - there are several specific methods and places a team implementing this use case may need to place analytics to cover the detection of this attack technique. Attackers could attempt this on Windows servers, Linux servers, web applications, network appliances, and more. If your SIEM doesn't normalize these events into an "authentication" event that can be monitored with a single analytic, breaking it down into subtasks might make sense. If you wanted to track the status of the detection of brute force attacks through each of these locations in a more granular way, subtasks could be created for each under the parent "Brute force login attempt" use case - one subtask for Windows servers, one for Linux servers, etc.

The photo below shows what doing this might look like (do not fill this in yet):

Use Cases #1

Brute Force Login Attempts

Added by Redmine Admin about 1 hour ago. Updated 5 minutes ago.

Status:	New	References:	https://attack.mitre.org/techniques/T...
Priority:	Normal	MITRE ATT&CK ID:	T1110
Assignee:	MGT Student	Kill Chain Stage:	Exploit
Primary Data Source:	Authentication Logs	Compliance Related?:	Yes
Objective:	To identify if attackers are attempting to brute force an account password		
Author:	John		
Known False Positives:	Broken service accounts, people who can't type their password		

Subtasks

Use Cases #2: Windows Brute Force Login	In Progress	MGT Student
Use Cases #3: Linux Brute Force Login	New	MGT Student

Related issues

In this photo, we see the parent level use case "Brute Force Login Attempts" with the custom fields filled in, but also have "child" issues that each would individually list the details for the implementation of the higher-level parent use case - one for Linux, and one for Windows.

If you clicked into the issue for the Windows login use case, you might find the following:

Use Cases #2

Use Cases #1: Brute Force Login Attempts

Windows Brute Force Login

Added by Redmine Admin about 1 hour ago. Updated less than a minute ago.

Status:	In Progress	References:	https://www.ult...
Priority:	Normal	MITRE ATT&CK ID:	T1110
Assignee:	MGT Student	Kill Chain Stage:	Exploit
Primary Data Source:	Windows Logs	Compliance Related?:	Yes
Objective:	Identify brute force login attempts to Windows operating systems		
Author:	Analyst2		
Known False Positives:			

Description

Tracking attempts to brute force login to windows using event ID 4625 from the Windows Security log channel.

Subtasks

Each of these child analytics would use a different primary data source and may have different coverage or life cycle stages. Tracking use cases like this enables an organization a finer level of detail.

Other examples of where this might be useful are other generalized attack tactics like lateral movement (you could even literally use the tactics or techniques within the MITRE ATT&CK matrix). You could create a high level "Detection of lateral movement of attackers in the network" parent use case and fill it with subtasks that designate different methods for catching that lateral movement. Perhaps you have separate rules by protocol - SSH, SMB, PowerShell Remoting, VNC, RDP, etc. You could also break it up by network segment, asset type, or any other method that made sense.

ADD ADDITIONAL USE CASES OR SUBTASKS

If you'd like to see how this works, **enter a few more use cases at this point**, either as parent level items or as sub-tasks to the use case you've already entered. In the next step we'll review how to use the API to pull the information out of all the entered issues so that when the database is full of all of your use case information, you can use the structured nature of the data to query and understand its contents.

To add a subtask to a use case, click on the "Add" button next to it in the issue:



You will be presented with the same screen as before when creating your initial issue. The only difference is that once you hit submit, the new item will be shown inside the parent level issue as a subtask item.

A screenshot of a 'New issue' creation form. The form includes fields for Tracker (set to 'Use Cases'), Subject ('Web Application Brute Force Login'), Description (with a rich text editor toolbar), and Status ('New'). At the bottom right of the form, there is a 'Parent task' field containing the value '1'. A large red arrow points from the text above to this 'Parent task' field, highlighting the process of attaching a subtask to a parent issue.

Note the inclusion of the ID of the parent task in the box - this is how you attach the subtask to the previously created parent. Use cases that have already been created can be modified to become child items through this method as well.

Once submitted, your task will now appear in the parent level ticket as a subtask:

Subtasks	Add
Use Cases #2: Web Application Brute Force Login	New 04/02/2020 

If you click into the subtask issue, the parent is listed at the top of the page near the issue title, telling you that it is a subtask of another item:



Use Cases #2 ←

Use Cases #1: Brute Force Login Attempt ←

Web Application Brute Force Login

Added by MGT551 Student 1 minute ago.

This flexibility of parent/child relationships between "issues" in your use case database gives you lots of needed flexibility to implement a more detailed organization system if desired and leads to the ability to create much more flexible and useful metrics.

Use Your Use Cases Database to See Metrics About Your Use Cases

The wonderful thing about having your use cases organized in a system like this is that once you have the data entered into issues, you should be able to use your organization software's API to pull information out about the data. Being able to programmatically query the use case database allows you to extract all of the issues and their corresponding custom fields to take stock of what you are and are not monitoring for, as well as integrate it with other systems in the SOC. Use case information, combined with fields like the MITRE ATT&CK Technique number can be fed into tools like ATT&CK Navigator to visualize your analytic coverage against your framework of choice, and quantitatively assess the SOC's use case coverage.

Some built-in charting ability is present in Redmine, you can access it from the main project issues page by clicking on the three dots in the corner then selecting Summary.

A screenshot of the SOC Issues page. At the top right, there is a search bar labeled "Search:" with "SOC" typed into it, and a dropdown menu showing "SOC". A red circle with the number "1" is placed over the dropdown menu. Below the search bar, there is a "New issue" button with a green plus sign icon, followed by three options: "Summary", "Import", and "Settings". A red arrow points from the "New issue" button towards the "Import" option. Another red circle with the number "2" is placed over the "Import" option. On the left side, there is a sidebar with "Filters" expanded, showing "Status" and "Tracker" with checkboxes checked. Below the filters are "Options", "Apply", and "Clear" buttons. At the bottom of the sidebar, there are buttons for "Assignee" and "Updated".

This will show the option to graph issues based on *Standard Fields*.

A screenshot of the SOC Reports page. At the top, there is a navigation bar with "Overview", "Activity", "Issues" (which is selected and highlighted in orange), "News", "Documents", "Wiki", "Files", and "Settings". Below the navigation bar, there is a section titled "Reports" with two tables. The first table is for "Tracker" and the second is for "Priority". Red arrows point from the text "Tracker" and "Priority" to their respective filter icons. The "Tracker" table shows data for "Use Cases" (3 open, 0 closed, Total 3) and "Feature" (0 open, 0 closed, Total 0). The "Priority" table shows data for "Immediate" (0 open, 0 closed, Total 0), "Urgent" (0 open, 0 closed, Total 0), "High" (0 open, 0 closed, Total 0), "Normal" (3 open, 0 closed, Total 3), and "Low" (0 open, 0 closed, Total 0). To the right of the tables, there are sections for "Version" (No data to display) and "Category" (No data to display).

Unfortunately, the ability to chart issues based on the contents of custom fields is not available as a built-in feature. Therefore, pulling the full information from each case with our custom field information included will need to be done through a different method. Fortunately, it is easy to do this through the built-in [Redmine HTTP REST API](#).

The simplest way to pull all information is through a simple HTTP request. Issuing a simple GET request to Redmine like the one below will return the following JSON formatted information about every issue in the specified project or tracker:

```
$ curl -u student:mgt551 localhost:8080/issues.json | jq
{
  "issues": [
    {
      "id": 3,
      "project": {
        "id": 1,
        "name": "SOC"
      },
      "tracker": {
        "id": 1,
        "name": "Use Cases"
      },
      "status": {
        "id": 1,
        "name": "New"
      },
      "priority": {
        "id": 2,
        "name": "Normal"
      },
      "author": {
        "id": 1,
        "name": "Redmine Admin"
      },
      "assigned_to": {
        "id": 6,
        "name": "MGT Student"
      },
      "parent": {
        "id": 1
      },
      "subject": "Linux Brute Force Login",
      "description": "",
      "custom_fields": [
        {
          "id": 1,
          "name": "Primary Data Source",
          "value": null
        },
        {
          "id": 2,
          "name": "Objective",
          "value": null
        }
      ]
    }
  ]
}
```

```

{
  "id" 3
  "name" "Author"
  "value" null
},
{
  "id" 4
  "name" "Known False Positives"
  "value" null
},
{
  "id" 5
  "name" "References"
  "value" null
},
{
  "id" 6
  "name" "MITRE ATT&CK ID"
  "value" null
},
{
  "id" 7
  "name" "Kill Chain Stage"
  "value" null
},
{
  "id" 8
  "name" "Compliance Related?"
  "value" null
}
],
{
  "created_on" "2020-03-31T11:41:18Z"
  "updated_on" "2020-03-31T11:43:29Z"
  "closed_on" null
}
...<s ip>...

```

If you'd like to try this for yourself, enter the following command in a terminal. The -u specifies the username and password to be used for Redmine access. Piping the output to jq just formats the output for easier reading.

```
curl -u student:mgt551 localhost:8080/issues.json | jq
```

Using a script to pull this information and drop it into an analysis or metrics generation tool of choice makes it easy to programmatically sample our use case database and gives an easy snapshot of SOC analytic coverage. Nearly any business intelligence tool that can create visualizations (such as Microsoft PowerBI, Tableau, etc.) could ingest the JSON file and create a dashboard of the custom field data in short order.

(Optional) Explore Redmine

If you have time, feel free to click around in the rest of Redmine and see what other help it may be able to offer you in SOC organization. If you'd like to log in as the administrative user and see all of the settings and options, log out of the student account and log back in using the credentials admin/mgt551mgt551. As Administrator, you can use the Administration tab to see all of the setup and other items that went into forming Redmine to use it as we did above.

If you like what you see and would like to implement this, the directions for setting Redmine up as you've seen here are included in the appendix at the end of this exercise, as well as a docker-compose file that will let you have a functioning version of Redmine up and running with only seconds of effort!

Exercise Conclusion -- Key Takeaways

In this exercise, you have:

- Created and documented a new use case for your SOC
- Entered that use case into a structured use case database
- Used features to track use case life cycle stage and facilitate discussion
- Seen how the free Redmine project management software can be used to organize SOC data and files

To shut down the services used for this exercise go back to your terminal window (or open a new one) and enter the commands below:

```
cd /home/student/labs/3.2
docker-compose down
```

You should see a response similar to the following, if you do not, please alert your instructor:

```
$ docker-compose down
Stopping redmine ... done
Stopping db      ... done
Removing redmine ... done
Removing db      ... done
Removing network 32_redmine-network
```

Exercise 3.2 is now complete!

Appendix: Setting Up Redmine for SOC Use

Here is a brief walk through of how to set up Redmine for use in a SOC (and as was used in this lab) from an empty state:

 Note

A mysqldump database dump with all of the following already done is stored in `/home/student/labs/3.2/redmine_SOC_setup.sql`. See [this page](#) for backup/restore instructions.

1. Install or start Redmine and bring up the webpage (see docker instructions below).
2. Login with default administrator name (admin/admin if using the docker setup contained below).
3. Click the "Administration" tab and select "Load the default configuration" to load defaults that we can start with - this populates much of the information we need, but we will need to rename some items so that it makes sense in a SOC context.
4. PROJECTS - Click PROJECTS and create a new project called SOC. Uncheck the Public box or else everyone with Redmine web access will be able to see your information stored inside it.
5. Select the modules within the Project you want to be available - for a SOC, this is most likely to be the following:
 - Issue Tracking - For the Use Case database and more
 - Wiki - For a SOC knowledgebase
 - Files - For storing common files
 - Documents - For easily storing policies, procedures, and more
 - Repository - For common code required for use among all analysts
 - News - To make announcements, if desired
 - Forums - If you have a large SOC and want to enable forum-style discussions
 - Others:
 - Calendar - While Calendar sounds useful, it only tracks dates related to created Issues, it's not a general use calendar. There are plugins to change this behavior to act more like a general calendar that may be useful.
 - Time Tracking - Time Tracking is for use in software development teams and will likely not be needed unless you want to track time for other types of issue effort.
 - Gantt Charts - Likely not useful for our Redmine use case.
6. Once done, press "Create".
7. ROLES AND PERMISSIONS - Click on the ROLES AND PERMISSIONS section, select "Developer" and change the name to "Analyst". Press Save.
8. GROUPS - Go to GROUPS, select "New Group" and create a group called "Analysts", and any other groups you would like to create.
9. USERS - Go to USERS and create new users and set their passwords.
10. Once done, click back to GROUPS, select the "Analysts" group, then the Users tab, then click "New user" and checkbox all new users and press "Add" to add them to the "Analysts" group.

11. Select the Projects tab, click "Add Projects" then checkbox the SOC project as well as the Analyst role on the bottom, then hit Add. You are now ready to use your users, groups, and roles within the SOC project.

12. TRACKERS - Once the project is created and users are added go back to the Administration tab and select TRACKERS. Trackers are used for anything you want to record a separate set of "Issues" for - Use Case Database, SOC improvements, Infrastructure issues, etc. Rename the default configuration trackers to whatever you'd like. An example could be "Use Cases", "SOC Improvement Ideas", and "SOC Fixes". While renaming the trackers, STANDARD FIELDS that are not applicable can be removed. For example, when renaming the tracker for Use Cases, it is likely you would want to uncheck Target version, Due date, Estimated Time, and % Done, since these don't make much sense to track for use cases.

13. CUSTOM FIELDS - Create CUSTOM FIELDS for the Use Case Database Tracking - select the Custom Fields section of the Administration tab, then select "new custom field", select the type of "Issues" and click next.

On the New custom field screen, fill in the following information at least:

14. Name - If you want to record use cases in Redmine, here are some custom field names and the data types you should select as you create them.

15. Format - The format of that custom field - how it will appear on your issue page. (list, link, text, long text box, etc.)

16. Trackers - Select the Use Cases tracker or any tracker this custom field should be available for

17. Projects - Select the SOC project or any project this custom field should be available for

18. Required - If you want to require the field to be filled out, check this box

In addition to the items above, select the ability to choose multiple values if necessary, or the default value a field should have.

Press "Create and add another" to make bulk custom field entry quick and easy.

Some ideas for Custom Fields for Use Case Databases you might want to track:

19. Objective (Why is this of interest?)

20. Author (if not using assignee field to track)

21. Lifecycle Stage (if not using issue status to track)

22. Primary Data Source - What is the initial source of the data used to detect this? (List - Firewall, AV, Windows Logs, EDR, IDS, etc., any type of log source you have)

Some items you might want to include in this list: - Switch - Router - NetFlow - Firewall - WAF - CASB - Amazon CloudTrail - Azure Event Hub - Network Anti-Virus - Host Anti-Virus - EMET / Exploit Guard - PowerShell Logs - Authentication Logs - Windows System Logs - Linux System Logs - EDR - UBA/UEBA - Malware Detonation - NIPS - HIPS - Secondary Data Source - Additional data sources (could also use a multi-select list) - Coverage - (Multi-select list - Desktops, Servers, Cloud, etc.) - (Pseudo)Logic - In plain language, how does this detection work? Example: "Looks for 10 failed login attempts in under 1 minute" - Playbook / Analysis Steps - May also be attached as a text file since formatting of large custom fields isn't great - Category / Type - What type of activity is this identifying? Hacking? Malware? Insider Threat? - Priority - How important is this item when the alert fires? (assuming it is correct) - Known

False Positives - List of reasons this might fire incorrectly - References - Link to additional information - MITRE ATT&CK Technique ID - MITRE ATT&CK Tactic Name - Kill Chain Stage - Compliance Requirements Related?

23. ENUMERATIONS - Click on ENUMERATIONS

24. Document Categories - Delete the "user documentation" document category - unless you would like to keep it. Make any others you wish to have available for document storage.

25. Issue Priorities - Adjust this as desired

26. Activities - Modify names if you want to use time tracking features, otherwise leave alone or delete.

27. ISSUE STATUSES - Click on ISSUE STATUSES. In this section, you have a chance to enumerate all statuses that will be available across all trackers. In the Use Case tracker, you will probably want different statuses compared to the "SOC Fixes" tracker for example. List *all* of them here. You will set up which statuses are available to each tracker in the next step. You will probably want to create something similar to the following:

28. For Use Case trackers

- New
- In Development
- Experimental
- Stable
- Temporarily Disabled (issue closedstate)
- Retired (issue closed state)

29. For SOC fixes /improvements

- New
- In Progress
- Testing
- Complete
- Rejected

After this, you will notice the message saying some statuses are not used in workflows. We will now specify the workflow for each status. 11. WORKFLOW - Click on WORKFLOW in the Administration section. In this section, you must select which issue statuses are available per tracker and per role. The easiest way to manage this is to allow all roles (manager, analyst, reporter) to change between any issue status to any other. Here is what that setup would look like:

The screenshot shows the Redmine Workflow configuration interface. At the top, there are tabs for 'Status transitions' and 'Fields permissions'. Below that, a section titled 'Select a role and a tracker to edit the workflow:' includes dropdowns for 'Role' (set to 'all') and 'Tracker' (set to 'Use Cases'). To the right of these is an 'Edit' button and a checked checkbox labeled 'only display statuses that are used by this tracker'. The main area is a grid titled 'New statuses allowed' with columns for 'New', 'In Progress', 'Testing', 'Complete', 'Rejected', 'In Development', 'Experimental', 'Stable', 'Temporarily Disabled', and 'Retired'. The rows represent current issue statuses: 'New Issue', 'New', 'In Progress', 'Testing', 'Complete', 'Rejected', 'In Development', 'Experimental', 'Stable', 'Temporarily Disabled', and 'Retired'. A large red box highlights the 'In Development' row, and a red arrow points from the 'Role' dropdown towards it.

To create the workflow for the SOC Fixes and SOC Improvements tracker, a similar setup would be used, but the box of green would surround the issue statuses that relate to those trackers.

Note: If you do not see your newly created issue statuses, ensure to uncheck the "Only display statuses that are used by this tracker" box at the top of the screen.

Once all workflows have been designed, hit Save.

30. SETTINGS - In the settings tab:

31. Click General and change the Application Title to "SOC Knowledgebase" or whatever name you desire.
32. Click API and enable the REST web service.
33. Click the Authentication tab and select "Yes" for "Authentication Required" ensuring no one who doesn't have a login can see Redmine's contents.
34. Click through the other tabs and set up configuration as desired. In the Display tab, you can apply any Redmine visual themes you have downloaded and staged in the correct folder to change the appearance of Redmine.

You are now ready to start using Redmine for SOC organization!

Docker Configuration

To quickly and easily start Redmine using Docker, put the following text into a file called docker-compose.yml:

```
version 3.1

services

  redmine
    image  redmine:4.1-alpine
```

```

container_name  redmine
networks
  redmine-network
restart  unless-stopped
ports
  8080:3000
environment
  REDMINE_DB_MYSQL  db
  REDMINE_DB_PASSWORD  example
  REDMINE_SECRET_KEY_BASE  supersecretkey
#volumes:
# - /path/to/save/redmine/files:/usr/src/redmine/files

db
image  mysql:5.7
container_name  db
networks
  redmine-network
restart  unless-stopped
#Comment the ports section out if you want don't host connectivity to the database
ports
  3306:3306
environment
  MYSQL_ROOT_PASSWORD  example
  MYSQL_DATABASE  redmine
#volumes:
# - /path/to/save/database:/var/lib/mysql

networks
  redmine-network

```

Assuming docker and docker-compose have been successfully installed, to start Redmine, simple type `docker-compose up -d` in the folder where the docker-compose.yml file is stored. After a minute, go to localhost:8080 and use the name and password admin/admin to log in and begin configuration. Note that configuration will not be saved with this file unless you uncomment the "volumes" section and map the folder to a path on the host.

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 Preparation

This exercise is completed in your MGT551 Linux VM, if you have trouble with the setup, see the troubleshooting information in the wiki, or reach out to an instructor or SANS support.

Launch the MGT551 Linux VM and log in.

```
- LOGIN = `student`  
- PASSWORD = `mgt551`
```

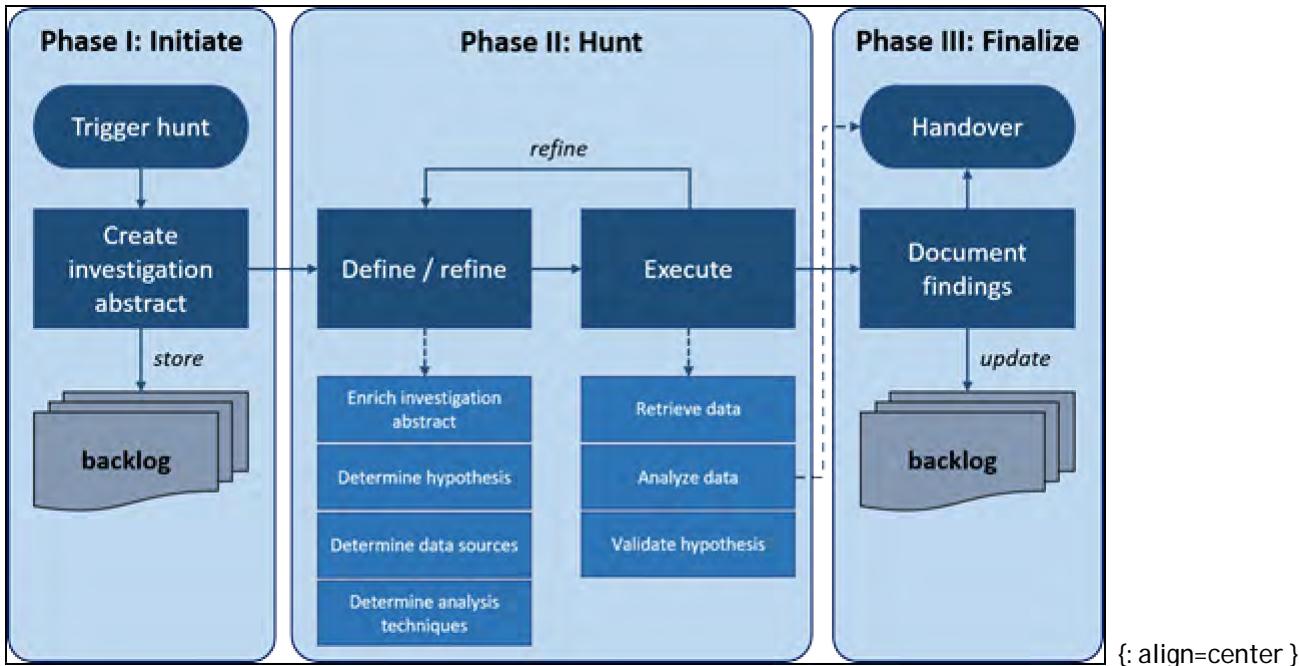
Once you are at the Linux virtual machine desktop, you are ready to proceed with the exercise.

Exercise Steps

Introduction

Threat hunting is most effective when there are clear goals and measurable results for each hunt. In this exercise, we will plan a threat hunt using the TaHiTI methodology developed by Rob Van Os and other members of the FI-ISAC. TaHiTI stands for **T**argeted **H**unting **i**ntegrating **T**hreat **I**ntelligence. As shown in the figure below, TaHiTI is a three-phased process:

1. **Initialize:** Generate ideas and document them in an abstract which is stored in a hunting backlog.
2. **Hunt:** Refine the abstract, enrich with threat intelligence, and execute the hunt.
3. **Finalize:** Document findings and submit to peer groups or other teams. These findings may include incidents, new intelligence, or engineering requirements.



{: align=center }

To execute this process, we will be using the **MaGMA for threat hunting** tool, also created by the FI-ISAC team. Results from threat hunting efforts can be tracked in this tool to gain insights into the results and overall performance of your threat hunts.

Hunting Triggers

There are several potential triggers for a threat hunt, including (but not limited to):

- New intelligence about an exploitable vulnerability, attack campaign, or threat actor
- Incomplete detection use cases
- Historical incidents
- Red team assessments
- Crown jewel analysis
- Results from previous hunts

Hunting is an iterative process, so it is helpful to think about these triggers being part of a cycle that includes threat hunting and in which threat hunting feeds back into some of these tasks. For example, a hunt that successfully uncovers malicious activity should become a detection use case to avoid manual analysis the next time the activity occurs.

In this exercise, we are going to build upon the attack tree we built in Exercise 2.1. Let us assume that we do not yet have any automated detections for these new attack methods we have identified. Based on our research and the resulting intelligence about likely methods of attack, we will conduct a threat hunt to look for evidence that these attacks have occurred.

Select a TTP from the attack tree you built in Exercise 2.1

If you have not yet completed that Exercise or wish to use another example, use Schedule Tasks/Jobs using the Windows at command as a means of persistence.

Referring to the MITRE ATT&CK Matrix, here is a description of that technique:

Adversaries may abuse task scheduling functionality to facilitate initial or recurring execution of malicious code. Utilities exist within all major operating systems to schedule programs or scripts to be executed at a specified date and time. In Windows environments, an adversary may use the at command to execute programs at system startup or on a scheduled basis for persistence.

Since we are already using ATT&CK as a reference, we can also refer to the detections for this sub-technique:

```
* Event ID 106 on Windows 7, Server 2008 R2 - Scheduled task registered  
* Event ID 140 on Windows 7, Server 2008 R2 / 4702 on Windows 10, Server 2016 - Scheduled task updated  
* Event ID 141 on Windows 7, Server 2008 R2 / 4699 on Windows 10, Server 2016 - Scheduled task deleted  
* Event ID 4698 on Windows 10, Server 2016 - Scheduled task created  
* Event ID 4700 on Windows 10, Server 2016 - Scheduled task enabled  
* Event ID 4701 on Windows 10, Server 2016 - Scheduled task disabled
```

Open the MaGMA for Threat Hunting tool

Load the template file we will use in your virtual machine by navigating to `/home/labs/3.3 and opening the Magma-for-Threat-Hunting.xlsx`` file located there. This will load a blank template file that you can use for this exercise.

Populate initial details

Enter the hunt identifier, description, and other details on tab L3 of the MaGMA spreadsheet in the white cells. Note that several cells have been filled out with example text for hunt identifiers, subjects, hypothesis, and reference links to get you started.

Hunt identifier	Date Completed	Subject	Hunting hypothesis
AO-EXF-01	okt/18	Data exfiltration through covert channel	Cyber criminals are using covert channels based on DNS to exfiltrate data from the organization
AO-EXF-02	nov/18	Automated data exfiltration with PowerShell	Cyber criminals are leveraging PowerShell to automatically exfiltrate sensitive data from the organization.
AO-LAT-01	nov/18	Lateral movement by account takeover	Cyber criminals are using existing user accounts for lateral movement

Now that we know what we are looking for and have a starting point from which to hunt, it is time to create our abstract.

Note

When entering a new hunt into the L3 tab, start with the hunt identifier which follows the syntax <\$kill chain identifier> - <\$attack type> - <\$hunt number>. For example, AO-PER, in which AO is Actions on Objectives (pulled from L1) and PER is Persistence (pulled from L2). Using the at command example, we'll use the hunt identifier AO-PER-02, since there is already a persistence mechanism hunt entry in our list.

Refine Your Investigative Abstract

The investigation abstract is a high-level description of the investigation you will perform. It is meant to be expanded and refined as the hunt progresses, but at the outset will contain the following information at a minimum:

- Current date
- Initial hypothesis of an incident that is occurring or has already occurred
- Trigger that has prompted the threat hunt
- Hunt priority

Document your hunt in MaGMA

Select the “Investigation Template” tab in the MaGMA tool. Fill out the fields based on the attack you selected from Exercise 1.2, or from our [at](#) example.

Note

The most important part of this initial investigative abstract is the hypothesis. The best hypotheses for a hunt are those that are specific, testable, limited in scope, and consistent with known facts. Hypotheses should also be reasonable in the sense that they should be provable or disprovable in a relatively short amount of time. If you plan to hunt for something that will require telemetry your team does not even have yet, you may want to shelve that abstract in lieu of something more pressing (or achievable).

Remember that this abstract is meant to be a living artifact that you will update and refine as the hunt progresses.

Hunting Investigation Template	
General Information	
Date	<Date>
Created by	<Hunter initials>
Last execution date	<Date>
Hypothesis & trigger	
(Initial) Hypothesis	<Initial hypothesis to be refined later>
Hypothesis status	<Initial / Refined>
Trigger	<What triggered the creation of this abstract?>
Reference	<Reference to the trigger>
Priority	<Priority level of the abstract>
Threat Intelligence	
MITRE Reference	<Reference to attack techniques from MITRE ATT&CK>
Possible actors	<Any actors that use these techniques>
Possible motivations	<Possible motivations>
Other TTPs	<Other TTPs associated with this actor group>
Active campaign?	<Is there an active campaign in which these techniques are used?>
Actor capability	<High / Medium / Low>
Classification & Resources	
Classification & Resources	<Step in the cyber kill chain>
Estimated resources	<Rough estimation of time and resources required>

Once the abstract has been created, it should be added to a hunt backlog: a list of hunts waiting to be executed. Much like a development backlog, this list must be actively managed to ensure hunts are correctly prioritized and reflect the evolving TTPs of our adversaries. The platform you use to store the hunt backlog does not need to be a complex, purpose-built system. Much like a knowledge base, simple collaboration tools can be more than enough to host your backlog. The most important features of whatever platform you use are availability to the hunt team and ability to capture all the relevant information.

Your team now has what it needs to search for this malicious activity. The early stages of the hunt will be broader and less targeted but should get more specific as the investigation progresses. This narrowing of focus can be achieved by adding additional context to your abstract using one or more of the following:

- **MITRE references:** References to additional techniques and sub-techniques can be added to the abstract.
- **New intelligence:** Think back to the threat actor profiles we developed in Exercise 1.1. Threat actors associated with the technique(s) we are hunting for likely have multiple motives and capabilities. We can use this information to identify other techniques that may be used in the same attack as the technique we are looking for.

- **Kill chain classification:** You can also update your abstract with additional classifications your team may use to track various kinds of activity, such as cyber kill chain phase.
- **Additional resources required:** We will cover hunting metrics at the end of the exercise, but at this point it can be helpful to estimate the time, data sources, team members, interaction with other groups, external technical resources, and other resources required to complete the hunt.
- **A more specific hypothesis:** The hypothesis, like the rest of the abstract, can and should be refined as the hunt progresses.

Update Tab L3

As your team works through the hunt and refines the Investigation Template, they will likely add new data sources, techniques, and references to their approach. These should be captured in tab L3 as new line items:

Scope	Main data source	Main analysis technique	Comments
All networks	DNS logging	Statistical analysis	DNS TXT records are of particular interest https://www.icann.org/news/blog/what-is-a-dns-covert-channel https://github.com/Arno0x/DNSExfiltrator
All networks	Web server logging	Querying	Looking for suspicious PUT requests and user agents containing PowerShell. https://sqrrl.com/hunting-misbehaving-powershells-examining-network-patterns/
All users	Authentication logging	Querying	Look for signs of account compromise by identifying accounts that are being used on multiple systems at a time https://sqrrl.com/threat-hunting-lateral-movement-identifying-pivot-points/
Web servers	Web server logging	Querying	Look for successful HTTP POST, outdated or aberrant user agents and basic authentication https://sqrrl.com/3-threat-hunting-starting-points-web-shells-edition/
All assets	End-point forensic information	Statistical analysis	Use the Damerau-Levenshtein distance algorithm http://detect-respond.blogspot.com/2016/11/hunting-for-malware-critical-process.html#/2016/11/hunting-for-malware-critical-process.html

Note

Tab L3 has several examples of different data sources and analysis techniques; having your team add these as they go along will leave a valuable audit trail for team members who come in after them or want to re-create a similar hunt in the future.

Refer to the threat actor profiles you created in Exercise 1.1 and MITRE ATT&CK to identify additional information you can add to the abstract. Fill out the remaining fields in the Hunting Investigation Template with additional information you have gathered. Using the *at* example for our hunt, here is what our refined abstract may look like:

Tip

Regularly review your hunt backlog to make sure your approach to selecting hunts remains effective. Compare abstracts in the backlog to your threat model and incident response activities to ensure your priorities are aligned to the rest of the security team and the priorities of your organization.

Conduct the Hunt

Now it is time to dig into the data and execute the hunt. There is rarely only one way to analyze the data at your disposal, but the general approach should be consistent:

Pivot through datasources

In the refinement process described above, we identified data source requirements to prove or disprove our hypothesis. Now is the time to retrieve that data and pivot to other sources as needed. Do not forget to revisit your abstract as you pull in other data sources to add them to your list of requirements.

Analyze raw data and develop a timeline of events

Once you have collected the necessary data, it must be analyzed using whatever technique is most appropriate. These techniques may include:

- Simple queries
- Stack counting
- Request/response ratio
- Statistical analysis
- Clustering
- Grouping

Normally, whichever analysis techniques you use will ultimately result in a sequence of events: x happened, followed by y and then z. Ordering the raw data in this manner will help you construct the timeline events and separate *subjects* from *objects*, i.e. the infected internal host (subject) accessed the secondary host (object).

Tip

In the MaGMA tool, you selected the analysis technique from a drop down menu on tab L3. In the *References* tab, you can customize and expand this list to suit your team.

Info

In his post “Four Common Threat Hunting Techniques with Simple Hunts,” Ely Kahn describes some common analysis techniques with specific use cases: <https://www.linkedin.com/pulse/four-common-threat-hunting-techniques-sample-hunts-ely-kahn/>

Draw conclusions

At the conclusion of the hunt, one of the following conditions should be true:

1. Your hypothesis has been proven due to evidence of an attack (or attack attempt).

- Your hypothesis has been disproven due to lack of evidence. This is a bit more difficult to achieve since the lack of data may not necessarily indicate the absence of a threat.
- The hunt is inconclusive based on insufficient evidence to prove or disprove the initial hypothesis.

Note

Regardless of the outcome, the threat hunt should still generate value in the form of blind spots you have uncovered, other things you have learned about your environment, or detections that require tuning.

In our final step, we will document these findings and make sure they are shared to the benefit of the team.

Document Findings

You may be familiar with the famous saying, "If a threat hunt fails in the SOC and no one learns from it, did the hunt actually occur?" (Ok, maybe that isn't the actual saying but it is a valid question). At the conclusion of the hunt, your team must process the results - positive or negative - and document their findings. This documentation should include recommendations for new or updated detections, additional logging, required enrichment data, and other improvements to the detection function. The hunter should also include lessons learned that may help teammates plan and execute more effective hunts in the future.

Add findings to the MaGMA tool

In the MaGMA tool, note the fields for data entry on L3 based on time spent, incidents found, recommendations, and other results. Feel free to enter arbitrary data here and see how the metrics update automatically on tab L1:

Time spent (hours)	Dwell time (hours)	# incidents found	# use cases updated	# security recommendations	# vulnerabilities found
20	80	1	2	1	1
20	160	2	3	4	2
80	800	1	5	2	1
40	0	0	2	0	3

The L1 tab contains a baseline set of useful threat hunting metrics, including:

- Dwell time of any findings (gathered during the incident response process)
- Total incidents discovered via threat hunting
- Detection use cases generated as a result of hunting
- New intelligence gathered
- Vulnerabilities or misconfigurations discovered

Exercise Conclusion – Key Takeaways

In this exercise, you have:

- Devised an abstract for a specific threat in your environment
- Used the MaGMA threat hunting framework to document and enrich your hunting abstract
- Identified necessary data sources and analysis techniques to support your hunting abstract
- Documented and shared hunt findings

Threat hunting is a powerful and necessary element of your SOC's detection capabilities, but it can be a major time and resource sink if it is not managed and tracked appropriately. Using this kind of approach to define specific goals, measure progress towards those goals, and capture other valuable outputs of the hunting process is a great way to avoid the many pitfalls of threat hunting and maximize the benefits.

Exercise 3.3 is now complete!

Exercise 4.1 - Investigation Quality Review

Background

Every SOC should be concerned with the question "How do we know we're doing a good enough job?" One way to answer this question is to develop a **defined**, acceptable work standard, and measure your analysts' investigations against it on a continuous basis. By periodically taking samples of your work and measuring them against your quality standard, you can continuously verify that you are getting the work quality you expect. If you aren't meeting expected standards, you want to identify that condition as quickly as possible and take fast corrective action to fix it. In this exercise, we will focus on developing a process for you to use to check your analyst investigation quality and solve these problems. Be aware that while we focus on investigations for this exercise, the process given here can be adapted towards measuring many other stages of alert triage and incident response as well.

Objectives

- Understand how to measure analysis quality
- Develop a repeatable process for identifying errors
- Understand the severity of those errors, and measure them against your **defined** quality standard
- Understand how to make sure the investigation review process is representative of the larger group of work done in your SOC
- Learn how to quickly identify and address errors
- Understand which errors can and cannot be eliminated within a given process

Exercise Preparation

This exercise is completed in your MGT551 Linux VM, if you have trouble with the setup, see the troubleshooting information in the wiki, or reach out to an instructor or SANS support.

Launch the MGT551 Linux VM and log in.

```
- LOGIN = `student`  
- PASSWORD = `mgt551`
```

Once you are at the Linux virtual machine desktop, you are ready to proceed with the exercise.

Exercise Steps

In this exercise, our goal is to build a process and assessment that can help us, as managers, measure that our analysts are performing alert triage investigations at the quality level we desire and continue to do so over the long term. We can decompose this goal into multiple questions that must be answered along the way such as:

1. How many samples do I need?
2. What should we look for?
3. How do we assess severity of problems and categorize findings?
4. Which findings can and should be fixed?
5. When has the process become stabilized and "under control"? / How many problems of which nature would indicate a "larger problem"?

To approach these answers, we will, again, take inspiration and guidance from the well-established process and conventions from other industries that run into these same issues, and apply their methods to information security where possible. This ensures that there is some precedent for using these methods and that they are grounded in a proven reality and history of success.

How Many Samples Do You Need?

If you had 50 alerts that had been investigated in the last month, and you want to get a sense of the level of quality of those investigations. How many would you want to assess to get a feeling that are *all* likely to be of acceptable quality? 5? 10? 25? Initially, this may seem like a subjective question, some of you might even answer "all of them!"

Unfortunately, we do not have time for 100% inspection in the SOC, so what to do? Fortunately, there is well-established process behind answering this question that is used in quality control for manufacturing. While you could reasonably argue that what we do in the SOC is slightly different than high-volume product manufacturing, the principles of defect sampling in "finished products" (investigations in the case of this exercise) remain solid. When combined with our own quality goals, using established manufacturing process here as inspiration can bring us in the right direction, and help objectively define how many samples is appropriate to check to meet our own standards.

The [ISO 2859-1:1999](#) standard lays out guidance and terminology how to build a sampling plan. While a read through of the whole document can be highly informative, for this exercise, we're focused on answering the "how many samples" question. The answer given in the standard comes in the form of what is often called "ANSI" or "AQL" tables.

To answer the question, we first must find what is called the "Sample Size Code". Assuming you have some idea of how often you want to run this process - weekly or monthly at least. On the chart below (sourced from [ansiaction.com](#)), the sample size code is found by looking for the lot size down the left side column (how many investigations happened during that unit of time) and moving horizontally across the row to find the letter.

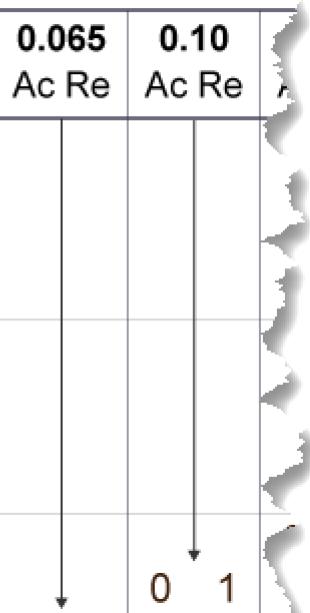
SAMPLE SIZE CODE LETTERS

Lot Size	General Inspection Levels			Special Inspection Levels			
	I	II	III	S1	S2	S3	S4
2 to 8	A	A	B	A	A	A	A
9 to 15	A	B	C	A	A	A	A
16 to 25	B	C	D	A	A	B	B
26 to 50	C	D	E	A	B	B	C
51 to 90	C	E	F	B	B	C	C
91 to 150	D	F	G	B	B	C	D
151 to 280	E	G	H	B	C	D	E
281 to 500	F	H	J	B	C	D	E
501 to 1,200	G	J	K	C	C	E	F
1,201 to 3,200	H	K	L	C	D	E	G
3,201 to 10,000	J	L	M	C	D	F	G
10,001 to 35,000	K	M	N	C	D	F	H
35,001 to 150,000	L	N	P	D	E	G	J
150,001 to 500,000	M	P	Q	D	E	G	J
500,001 and over	N	Q	R	D	E	H	K

You'll notice that there are multiple options for different inspection levels, each which give a different letter as a result. For our process, **General Inspection I-III** are the options you should be focusing on ("special inspection" is for destructive inspection testing and other use cases that generally don't align to our use.) From within General Inspection I, II, and III, the choice is made by deciding how much risk you want to take and how much budget and time you have to do inspection. **Class I will result in the lowest number of inspections, and is the most cost efficient, Class III will result in the most inspections occurring at the highest cost and time required.** The choice is yours, but one recommended approach is to start with higher levels if this is your first time running this process and move down towards I as you begin to gather data and trust that things are generally working well. (For reference, in product manufacturing, class II is typically used)

Once you have your sample size code, you take it to a second table to find the sample size. A clip of the single sampling plans table is shown below, and the number of items to inspect is shown in the 2nd column, based on the sample letter code you found in the first table.

Sample Size Code Letter	Sample Size	0.065	0.10	
		Ac Re	Ac Re	
A	2			
B	3			
C	5			
D	8			
E	13			
F	20			
G	32			
H	50			
J	80			
K	125			



The Sample Size column is highlighted with a red box. The row for Sample Code Letter K has a red box around its value '125'. Below the table, there is a vertical scale with two arrows pointing downwards, one labeled '0' and one labeled '1'.

We'll save the explanation of the rest of the table for a future step. At this point, you now have a starting place to gauge how many samples you should take from your investigations to gain a better feeling about the quality of the overall group.

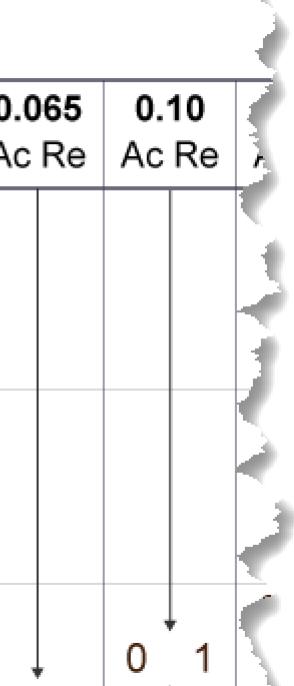
Example

Let's say you're a SOC looking to implement quality control testing. With 100 investigations performed in the last month, you choose General Inspection level II to start, based on the time you have available. This brings you to sample code letter "F".

Lot Size	General Inspection Levels		
	I	II	III
2 to 8	A	A	B
9 to 15	A	B	C
16 to 25	B	C	D
26 to 50	C	D	E
51 to 90	C	E	F
91 to 150	D	F	G
151 to 280	E	G	H
281 to 500	F	H	J
501 to 1,200	G	J	K

Looking up F in the sample size chart, you then find 20 is the suggested sample size:

Sample Size Code Letter	Sample Size	0.065		0.10	
		Ac Re	Ac Re	Ac Re	Ac Re
A	2				
B	3				
C	5				
D	8				
E	13				
F	20				
G	32				
H	50				
J	80				
K	125				



Now You Try It

Try it for your own SOC: Roughly, how many investigations have you performed in the last week or month? Find the 3 sample codes that are suggested.

Lot Size	General Inspection Levels		
	I	II	III
2 to 8	A	A	B
9 to 15	A	B	C
16 to 25	B	C	D
26 to 50	C	D	E
51 to 90	C	E	F
91 to 150	D	F	G
151 to 280	E	G	H
281 to 500	F	H	J
501 to 1,200	G	J	K

Write your results down in the space below or in a notepad:

Sample Codes: _____

Now, look them up in the sample size table - how many inspections does it suggest you perform?

Sample Size Code Letter	Sample Size
A	2
B	3
C	5
D	8
E	13
F	20
G	32
H	50
J	80
K	125

Write your results down in the space below or in a notepad:

- Sample Code Letter 1: _____
- Sample Code Letter 2: _____
- Sample Code Letter 3: _____

Do these numbers seem large? Remember it doesn't have to be done all at once. In a SOC with 100 investigations per month, using code F suggests 20 samples. 20 samples in 30 days is 2 reviews every 3 days. If you can make the review process simple or even partially automated, this may equate to only a few minutes per day on average, and sample size can be reduced as confidence is built. Alternatively, you can fall back to the "Special Inspection Level" recommendations as well, which will result in even smaller sample sizes. Once you know how long each review takes, it's easy to multiply it out and compare it to how much time you (or whoever is doing the reviews) is able and willing to put into quality control.

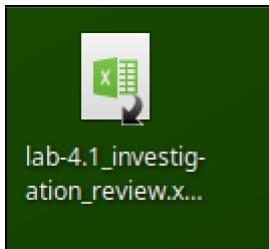
At this point, we have now answered one key question - "How many samples do we need? In the next step, we'll discuss what to look for and how to categorize those findings.

What To Look For

This is step where build the heart of the investigation quality review, which is, defining exactly what we're going to look for as an indicator of a quality investigation.

As a SOC manager, you probably have a somewhat intuitive sense for what an analyst should and should not be doing in a complete and well-performed investigation. Our goal here is to enumerate those items into a standard checklist that can be used for reviews.

For this step, open up the spreadsheet through the icon located on the desktop labeled `lab-4.1_investigation_review.xlsx`.



Tip

As with previous labs, if you'd like to click and drag this file out of the file manager in your Linux desktop and move it to your host PC to use Excel, feel free to do that as well. (You cannot drag the icon since it is a shortcut, you will need to navigate in the File Manager application to `/home/student/labs/4.1`)

You should now see the spreadsheet below. For now, we'll focus on the question and category columns, the defect type column will be explained in a moment.

Investigation Review Worksheet						
#	Category	Questions	Defect Count			Comment
			Critical	Major	Minor	
1	Investigative	Were all observables entered into the incident management system / TIP with adequate detail to describe them?				
2	Investigative	Was the playbook process followed correctly?				
3	Investigative	Was relevant evidence captured and preserved in the related playbook step notes, such that someone could go back and validate what occurred / follow the analysis?				
4	Investigative	Was evidence properly collected and considered? Were any obvious data sources ignored or missed that might have changed or aided the investigation?				
5	Investigative	Were any unwarranted assumptions made?				
6	Investigative	Did the analyst show any signs of confirmation bias? Was there an attempt to assess multiple possible hypothesis and is there evidence of attempts to *disprove* the conclusions?				

For this section, we'll start to customize this sheet to your process by brainstorming items you'd like to check when reviewing a closed investigation. You may find it helpful to break these items into categories such as "investigative questions", "response actions" and more to help decompose the types of categories of items you want to review. Here are some example questions based off a review list used by the author in their previous SOC management role:

- Investigative questions:
- Were all observables entered into the incident management system / TIP with adequate detail to describe them?
- Was the playbook process followed correctly?

- Was relevant evidence captured and preserved in the related playbook step notes, such that someone could go back and validate what occurred / follow the analysis?
- Was evidence properly collected and considered? Were any obvious data sources ignored or missed that might have changed or aided the investigation?
- Were any unwarranted assumptions made?
- Did the analyst show any signs of confirmation bias? Was there an attempt to assess multiple possible hypotheses and evidence of attempts to *disprove* the conclusions?
- Response Actions:
- Containment: Was the incident adequately contained in a timely manner? Were both network and endpoint controls utilized in the proper way for that containment?
- Containment: Were accounts locked and passwords reset for all affected accounts in a timely manner?
- Eradication: Did the affected asset (if applicable) have the infection removed in a satisfactory way?
- Lessons Learned: Were improvements to prevent this activity noted and, where applicable, suggested for development?
- Documentation:
- Were all cyber kill chain stages or MITRE ATT&CK framework tactics and techniques used in this intrusion identified and enumerated in the notes?
- Was there an attempt to predict what would have happened in further stages of the attack if it had progressed?
- Was there an attempt to link this attack to any previous attacks in order to spot any potential patterns?
- Was forensic evidence properly preserved in a timely and forensically sound manner, consistent with our procedures?
- Were all appropriate categorizations and classifications made about the investigation at the time of closing the case?

These sample questions are already entered in the spreadsheet, feel free to use, modify, or remove them as desired.

Next, take a moment to think about what else constitutes a good quality investigation on your team and add your own content to the sheet. As some additional inspiration, think of additional questions relating to facets of the investigation such as:

- **Timeliness** - How long did it take before containment and response actions were taken? Was it within acceptable limits?
- **Quality of Analysis** - Was the analysis conclusion (seemingly) correct? Was it thorough? Was all evidence gathered and considered? Were any poor or key assumptions made that may have swayed the analysis?
- **Documentation** - Were all categorizations of the case properly entered? Were observables captured in your incident management system / threat intel platform with enough context to connect this event with any future attacks?
- **Improvements** - Were lessons learned captured (if applicable)? Was root cause identified for the intrusion and any suggestions made and fed back to the proper channels that could prevent this from happening the next time?

Add any additional assessment items you think are important for your own SOC before proceeding. The categories and numbering is just there for convenience, feel free to manipulate the spreadsheet as necessary.

Now that you hopefully have at least an initial list of what defines a quality investigation to you, the next question is, while inspecting our chosen number of samples, how will we grade them as acceptable or not?

Categorizing Findings

Now let's imagine you're reviewing an investigation and find 3 errors, is that enough to consider that investigation a problem? Or were they inconsequential issues? How do we notate this? Now that we have an idea of *what* we're looking for as our standard of quality, consider how you will rank your findings in terms of importance and severity.

Not all errors are equally important, and the level of acceptable issues may be different for each team. At a minimum we need a ranking system that can account for this. In the spreadsheet there is a column labeled "Defect Type", which is the scoring mechanism aligned with the AQL /quality control process. In this system, which you can either use as is, or modify to suit your needs, there are 3 levels of defects, Critical, Major, and Minor. Here's how you might consider using those levels for investigation review:

- **Critical Defects** - Unacceptable problems, disregard for process, missed steps and other major issues that might have tainted the investigation results, or led to incorrect and/or damaging conclusions
- **Major Defects** - Non-catastrophic, but easy to identify and important failures that would not be acceptable if you were doing this analysis as a service for a customer (MSSPs take note here)
- **Minor Defects** - Issues that are not ideal to find, but probably did not have any detrimental effect on the investigation or outcome, a customer receiving this analysis might not notice or care about these issues but are still something that can be improved

As you move through each question in your list, the idea is to count any issues you identify in the labeled "Defect Count" columns and list a comment for the nature of each problem. This will help you measure the the total count of "defects" within any given analysis on a 3 levels of fidelity basis. This spreadsheet has a built-in sum formula on the bottom that will count errors from each defect type column as you go.

Move on to the next step where we will discuss the question of "how many problems is too many?"

How Many Problems Is "Too Many"?

How do we determine how many defects are "too many" to be acceptable? In manufacturing, this is referred to as **Acceptance Quality Limits**. For manufactured consumer goods, limits such as **0% critical defects, 2.5% major defects, and 4.0% minor defects** are common. In your SOC, you may adjust this bar to whatever you or your customers (if you are an MSSP) agree to, but this can serve as a starting point if you want to anchor quality limits to a commonly used quantity.

Since each sample size of alerts may be different over time, the ANSI tables have a built-in way for determining, based on your standard levels for acceptable quality, how many of each type of defect is allowed before a lot needs to be rejected. We too can use this same process once we have our acceptance quality limits. Here is how that process works - once you decide on a percentage of allowable defects per type, that number (which has standard defect percentages pre-calculated in the ANSI table) can be immediately reviewed for the count of issues that would cause an "accept" or "reject" (unacceptable) limit.

As you can see in the image below, the pre-calculated percentages are displayed across the top row with an Accept number "Ac" and a Reject number "Re" sub-item underneath it. To use the chart, you first find the percentage that is acceptable for the type of defect you're counting. Then, drop down the column to match the sample code letter you previously found and will find the box the maximum number of errors that is acceptable, and the number of errors that would cause that log to be rejected. (Notice there is no 0% column, because that math is easy, all problems = reject.)

		Percent Acceptable Quality Levels (Normal Inspection)											
Sample Size Code Letter	Sample Size	0.065	0.10	0.15	0.25	0.40	0.65	1.0	1.5	2.5	4.0	6.5	
		Ac Re	Ac Re	Ac Re	Ac Re	Ac Re	Ac Re	Ac Re	Ac Re	Ac Re	Ac Re	Ac Re	
A	2										0	1	
B	3									0	1		
C	5								0	1			
D	8							0	1			1 2	
E	13						0	1			1 2	2 3	
F	20					0	1			1 2	2 3	3 4	
G	32				0	1			1 2	2 3	3 4	5 6	
H	50			0	1			1 2	2 3	3 4	5 6	7 8	
J	80		0	1			1 2	2 3	3 4	5 6	7 8	10 11	
K	125	0	1			1 2	2 3	3 4	5 6	7 8	10 11	14 15	
L	200	0	1		1 2	2 3	3 4	5 6	7 8	10 11	14 15	21 22	
M	315		1 2	2 3	3 4	5 6	7 8	10 11	14 15	21 22			
N	500		1 2	2 3	3 4	5 6	7 8	10 11	14 15	21 22			
P	800	1 2	2 3	3 4	5 6	7 8	10 11	14 15	21 22				
Q	1250	2 3	3 4	5 6	7 8	10 11	14 15	21 22					
R	2000	3 4	5 6	7 8	10 11	14 15	21 22						

In this image, the 1% limit is highlighted, dropping down the column shows that for sample code letter E, a single error would be unacceptable (0 defect = accept, 1 defect = reject). The arrows below this row and column show that these numbers do not change until you get down to sample code H, where a single defect would become acceptable.

Continuing on with our example from above - A SOC doing 100 reviews a month using sample letter code F using the 0%, 2.5%, and 4% acceptance quality limits mentioned above would then derive the following guidance from this table:

SINGLE SAMPLING PLANS FOR NORMAL INSPECTION

Sample Size Code Letter	Sample Size	Acceptable Quality Levels (Normal Inspection)										
		0.065	0.10	0.15	0.25	0.40	0.65	1.0	1.5	2.5	4.0	6.5
		Ac Re	Ac Re	Ac Re	Ac Re	Ac Re	Ac Re	Ac Re	Ac Re	Ac Re	Ac Re	
A	2										0 1	
B	3									0 1		
C	5									0 1		
D	8									0 1		
E	13									0 1		
F	20									1 2	2 3	
G	32									1 2	3 4	
H	50									1 2	5 6	
J	80									1 2	7 8	
K	125									1 2	10 11	
L	200	0 1								1 2	14 15	
M	315									1 2	21 22	
N	500									1 2		
P	800	1 2	2 3	3 4	5 6	7 8	10 11	14 15	21 22			
Q	1250	2 3	3 4	5 6	7 8	10 11	14 15	21 22				
R	2000	3 4	5 6	7 8	10 11	14 15	21 22					

- Acceptable Critical Issues = 0
- Acceptable Major Issues = 2
- Acceptable Minor Issues = 3

You may find that the error tolerance using this system is extremely low - 2 major defects and 3 minor defects across 20 alerts? Do you think you can or currently would meet that standard? If yes, great! Perhaps you can look to become even more stringent as time progresses. If you no, are you or your customers ok with that? As long as you are hitting whatever goals you have defined as acceptable, that is the level of quality you're hoping to find over the course of these measurements. The big picture here is to define your levels of "defects" that are acceptable to your organization or your customers, and continuously measure yourself to ensure you are hitting those goals.

What are the limits you initially think you might use for percentages? This can be a hard question to decide off the top of your head without data to see where you currently stand. For the sake of working through this process, pick 3 notional numbers for Critical, Major, and Minor defects and use the chart to find the number of defects in the lot of assessments you do for that period that would be acceptable.

If this many issues, consider evaluation a "fail"

Note

Remember that while the spreadsheet we just created counts errors for a *single* assessment, remember that the results of *all* these spreadsheets for a given time period would need to go into another system designed to total *all* defects of each type for *all* assessments in that month/week. (20 assessments from our previous example) This table is *not* a calculation of errors allowed per investigation, it is errors allowed for all of them together.

Sample Code Letter: _____

Results:

Defect Level	Acceptance Quality Limit (%)	Allowable Defects
Acceptable Critical Issues		
Acceptable Major Issues		
Acceptable Minor Issues		

At this point you may be thinking "Shouldn't we always strive for an error rate of 0%?" While in an ideal world the answer to that is yes, the cost-prohibitive nature of doing so may not be worth it. Everyone has a line of what is "good enough" and surpassing that standard is good in the eyes of your stakeholders and customers, but it also technically wasting time and money to meet a standard that isn't needed, keep this in mind as you do your assessments and set your SLOs on what is the right limit for you.

Follow-Up - Assigning Causes and Additional Reading

At this point we have now planned and answered the main 3 questions associated with investigation review:

1. Appropriate sample size
2. What to assess
3. How to categorize the findings

Your investigation review process would then consist of setting your target goals for sample sizes and acceptance quality limits, deciding on the factors to measure that determine a high-quality investigation, and optimizing the process of assessment and data aggregation across the multiple assessments with each time period.

In this last section, there are two additional concepts and some reference material that it would be a missed opportunity not to mention as a suggested follow on to this lab. Those things are:

- Assessing process improvement opportunities that result from assessment findings
- How to tell when your investigation (or any process) is "in control" vs. "out of control"

When assessing defects, one highly useful concept for categorizing them is labeling each as "Non-Assignable" (or sometimes called "common") causes vs. "Assignable" (sometimes called "special") causes.

- **Non-Assignable / Common** causes are the variations in a process that arise naturally as part of the nature of the process itself. The [American Society for Quality \(ASQ\)](#) defines them as "Causes of variation that are inherent in a process over time. They affect every outcome of the process and everyone working in the process." They tend to produce a statistically stable distribution (bell curve) of variation on the output of whatever they effect.
- **Assignable / Special** ASQ defines as "A name for the source of variation in a process that is not due to chance and therefore can be identified and eliminated." These causes are often intermittent, and preventable types of variation in a process.

The book "[Statistical Process Control for Managers, Second Edition](#)" has an outstanding and immediately intuitive explanation for these two terms:

Imagine you are trying to optimize the process of driving to work such that you will always arrive on time. Traffic lights are an inherent part of the process of driving to work that no matter what you do, will always be a problem. If you tracked the effect they had on your drive time, it would have a stable average over time with total weight time have a bell curve-like distribution effect on your commute (rarely would you hit ALL red lights or hit ALL green lights). Getting through those traffic lights cannot be optimized as it's random chance whether you hit green or red lights, it's just a natural part of the driving to work process. Traffic lights, therefore, are **non-assignable / common cause** source of variation and are not something to worry about needing to fix.

Let's say, however, that one day your car breaks down because you forgot to perform required maintenance, or you run out of gas, and because of the delay you end up late to work. This is a one-off, intermittent, and preventable issue, and therefore would be considered a **special / assignable cause** source of variation (you didn't do what you should have and could have done to prevent them). Those errors are the types we're looking to flush out of our processes as they are controllable and preventable.

Understanding these concepts and terminology brings us to the guidance for this step. Look for ways to highlight whether any defects you find are common cause or special cause sources of variation. In our quest to always produce the highest quality alert triage and investigations, some errors will be fixable, and some will be part of the process, and we want to flush out all of the problems that are possible to correct. Our processes will be optimized and "in control" when we have addressed all of the **fixable** issues.

This brings us to the key takeaway concept from the above-mentioned Statistical Process Control book as it relates to defects and sources of variation:

- A process is considered "in control" when the only defects in the system are those from non-assignable causes or in ASQ's terms "A process in which the statistical measure being evaluated is in a state of statistical control; in other words, the variations among the observed sampling results can be attributed to a constant system of chance causes."
- A process is considered "out of control" when there are assignable cause variations found within the sample set or as ASQ says "A process in which the statistical measure being evaluated is not in a state of statistical control. In

other words, the variations among the observed sampling results cannot be attributed to a constant system of chance causes."

A process should therefore be considered "in-control" when all of the assignable cause errors are eliminated. What special cause errors might translate into for a SOC are clearly preventable issues like skipping playbook steps, poorly followed process, broken automation, missing observables in your incident management system or threat intelligence platform, or other errors that the process should have prevented, but somehow were looked over. Dialing in quality and controlling the controllable will largely revolve around mistake proofing and guidance for analysts down a repeatable path of investigations steps that will ensure high-quality results over the long term.

For those who want to take these concepts even further, the previously mentioned book "Statistical Process Control for Managers" has some great ideas to follow up this exercise with. While very much not an information security-related book, it breaks down the concepts used by those doing quality control in the manufacturing space in a very understandable way. The ideas for sampling, measurement, and evaluation of "in control vs. out of control" processes in the book have obvious parallels to any process where repeated similar tasks are done, and high-quality across time is required. Those who have had their interest piqued by this exercise are encouraged to pick up a copy and look into how control charts may be used to look at the results of your investigation reviews over time and take quick action to correct issues when any of your processes are found to be "out of control" preventing the issues from becoming any larger than they need to be.

Optional: Test Drive Your New Assessment Checklist

Want to test out your quality checklist? We have prepared a pre-staged alert in TheHive you can use for exactly that! If you don't have time or want to perform this step later, skip down to the Exercise Conclusion.

Open a terminal window and type the command below:

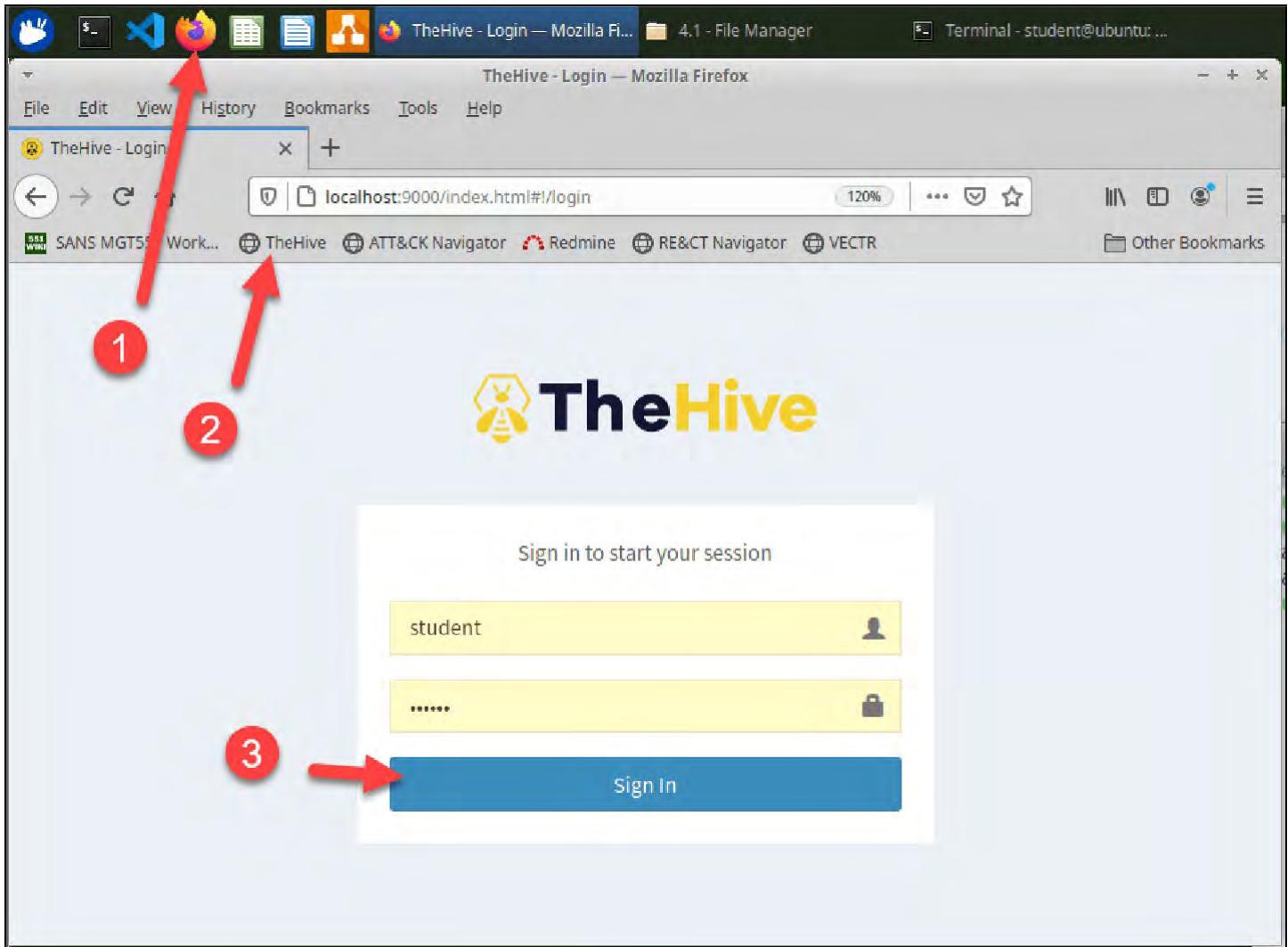
```
cd /home/student/labs/4.1  
docker-compose up -d
```

You should see the following output, if you do not and receive any type of error, run the troubleshooting script from the wiki or alert your instructor / SANS support for help.

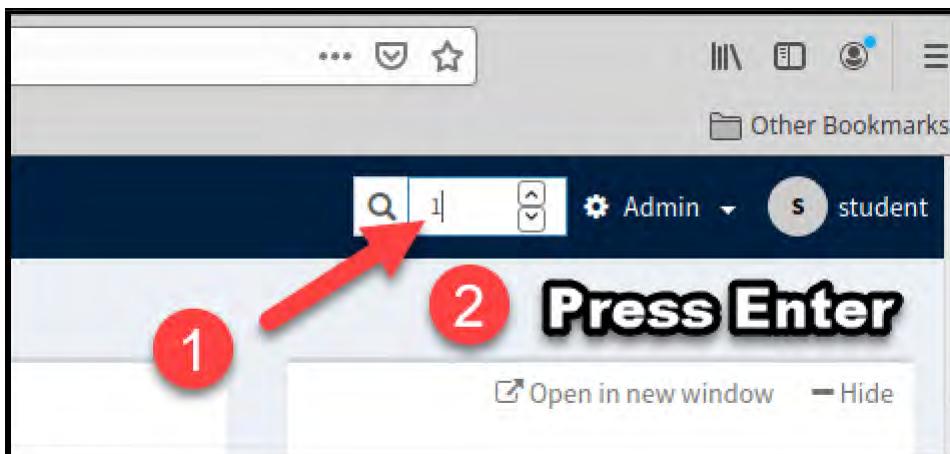
```
Starting 41_elasticsearch_1 ... done  
Starting 41_thehive_1      ... done
```

After a few moments (it may take a few minutes for TheHive and Elasticsearch to initialize), as in previous exercises, you should be able to access TheHive by opening Firefox and clicking on the bookmark in the bookmark toolbar.

Log in to the hive with the saved credentials ("student" / "mgt551").



Once you're into the main interface, pull up the example case with alert and closure notes pre-entered by searching for Case "1" and hitting enter in the top right search box.



You should now see the following completed investigation case summary screen:

The screenshot shows a completed investigation case summary screen. At the top, it displays the case title: "Case # 1 - [cred. exposure] Login attempt from unexpected geolocation". Below this, it shows the case was created by a student on Mon, Apr 5th, 2021 17:44 -04:00, and it was closed at 04/05/21 18:08 as a True Positive with No Impact.

Below the title, there are several tabs: "Details", "Tasks" (with a count of 6), and "Observables" (with a count of 7). Red arrows point from the text "Click to browse tasks and observables" to the "Tasks" and "Observables" tabs. A large red arrow points down from the text "Scroll for summary info" towards the bottom of the screen.

The main content area contains the following information:

Summary	
Title	[cred. exposure] Login attempt from unexpected geolocation
Severity	M
TLP	TLP:AMBER
PAP	PAP:AMBER
Assignee	student
Date	Mon, Apr 5th, 2021 17:43 -04:00
Tags	Not Specified
Close date	Mon, Apr 5th, 2021 18:08 -04:00
Additional information	Met

This alert has been pre-populated with simulated incident information including observables, a case summary, and detailed notes for each of playbook steps ("tasks" tab in TheHive). Look at the "tasks" and detailed notes left by the analyst by clicking on the "tasks" tab, then selecting an individual task.

The screenshot shows a digital investigation platform. At the top, there are three tabs: "Details", "Tasks" (which has a red circle with the number 6), and "Observables" (which has a red circle with the number 7). Below the tabs are two buttons: "+ Add Task" and "Show Groups". A red arrow points from a circled '1' to the "Tasks" tab. Another red arrow points from a circled '2' to a task entry in the list.

Group	Task	Date	Assignee
Investigation	Where are the credentials being used? Closed after a minute	Mon, Apr 5th, 2021 17:46 -04:00	student

Once the task is open, read the "Task Log" section at the bottom.

The screenshot shows a "Task logs and investigation detail" section. On the left, there's a "Task logs" button with a red border. A red arrow points from this button to a detailed log entry on the right. The log entry includes a timestamp, user information, and a descriptive text box.

Description

External, internal, which service?

Task logs

+ Add new task log Sort by: Newest first

10 per page

student

Mon, Apr 5th, 2021 17:46 -04:00

Credentials for [mike@sec450.com](#) have only been used to log in to the external sec450.com portal website from IP address 5[.]120[.]35[.]147 on 2019-04-21 at 12:53:26.
They account was locked and password reset within minutes of the attempt, so further attempts to use the name and password will not succeed.

Feel free to look around at what the analyst wrote and judge the investigation, and quality and completeness of the data entered based on your spreadsheet criteria.

Once you are complete with using TheHive, close firefox and go back to your terminal window (or open a new one). To stop the docker container services, copy and pasting the following commands on the command line.

```
cd /home/student/labs/4.1  
docker-compose down
```

You should see the following output, indicating the containers have been stopped and removed.

Exercise Conclusion -- Key Takeaways

In this exercise, we have:

- Learned how to create investigation quality review process
- Used ANSI tables to find the appropriate sample size for a population of items that need to be assessed
- Defined the aspects of an investigation that will assess if it was done well or contains errors or process missteps
- Created a spreadsheet for easy tracking of assessment outcomes
- Learned how using the critical / major / minor defect ranking system can help you understand if you are meeting defined quality standards
- Learned about different types of process variation sources, and how you can tell whether your processes are in control or not
- Given additional resources for statistical process control to follow up with additional useful process improvement ideas

Exercise 4.1 is now complete!

Exercise 4.2 - Planning Responses with RE&CT

Background

We have spent some time talking about the various tools and methods your team might use to respond to and contain intrusions. The fact is that these can be as dynamic and varied as your environment, and it can be difficult to wrap your brain around the many choices and capabilities that may be available to you (or to which you aspire). Fortunately, there is a framework for that!

The RE&CT Framework is a project designed for curating incident response techniques, based on the MITRE ATT&CK framework. Like ATT&CK, RE&CT is laid out as a matrix with the columns representing response stages and the cells representing response actions. While the project is still in the early stages of development, it provides a great example for how your team might catalog and arrange response actions across the incident response cycle to compliment the threat and detection tracking we have already covered.

Objectives

- Identify potential response actions based on threat model and detection planning
- Review response visualizations in RE&CT Navigator
- Develop incident response playbooks for high-impact incidents
- Automatically generate response documentation

Exercise Preparation

This lab is completed in your MGT551 Linux VM

1. Launch the MGT551 Linux VM and log in.

LOGIN = `student` PASSWORD = `mgt551`

2. Start up the Lab 4.2 container.

Before starting this exercise, you must start the required services. To do this, open a command terminal from the start bar.



Once the terminal window is open, start the services by copying and pasting the following command on the command line:

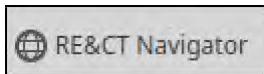
```
cd /home/student/labs/4.2  
docker-compose up -d
```

Keep the terminal open, we will use it to shut these services down at the end of the lab.

Exercise Steps

Open RE&CT Navigator

You can open RE&CT Navigator by opening Firefox in your exercise VM and clicking the link in the bookmark toolbar:



!!! note: RE&CT Navigator may take a few minutes to start after you run the docker command. If the link does not pull up the Navigator web app immediately when you click it, wait a few minutes and try again before troubleshooting.

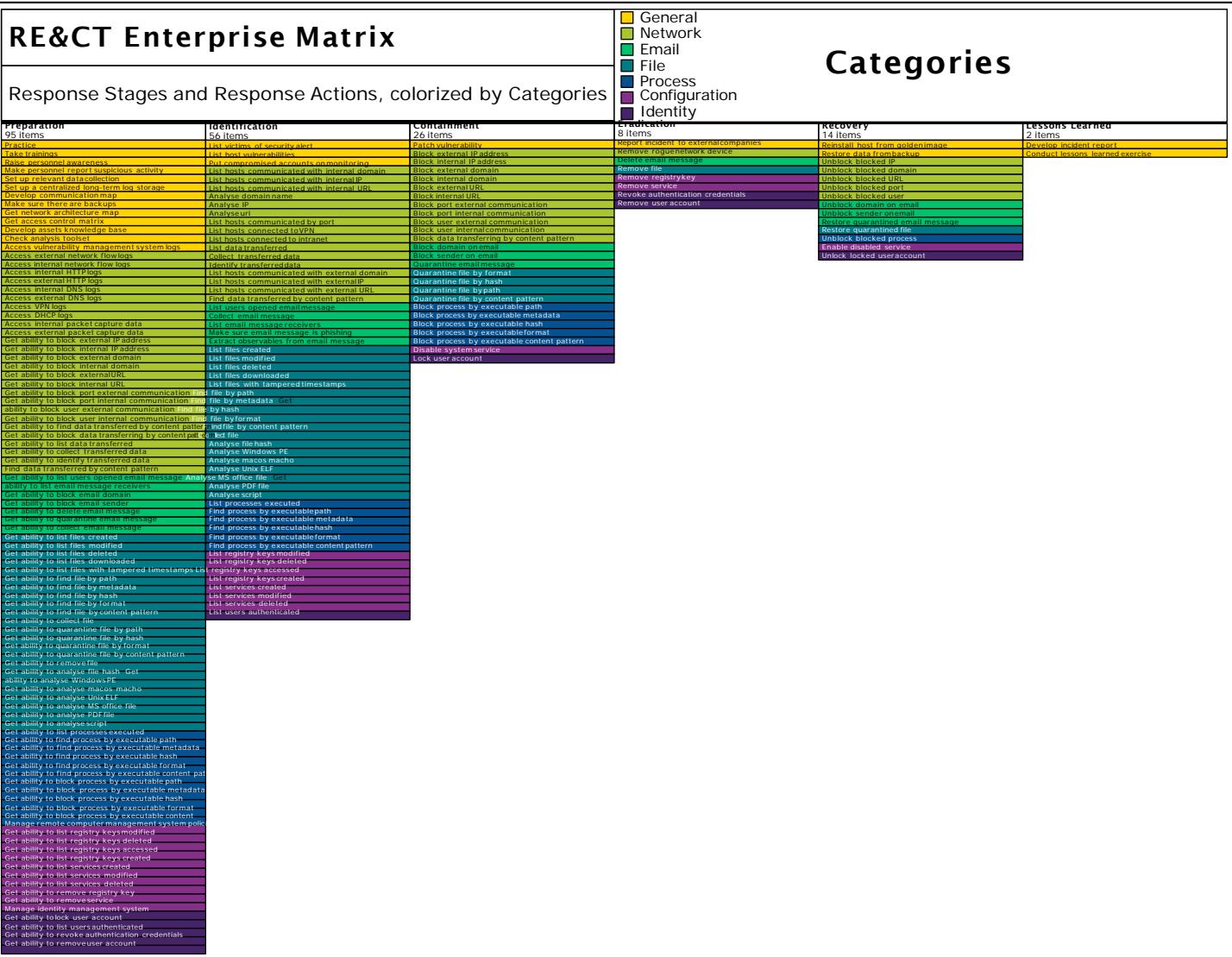
Review RE&CT

In Exercise 2.2, we used ATT&CK Navigator to highlight TTPs that are most relevant to us based on our research and intelligence. You'll notice RE&CT Navigator is very similar to ATT&CK Navigator (it's basically a fork of the application with the values changed). However, it isn't quite as mature or complete, so we will not be using the RE&CT version in this lab. Even so, we wanted to show it to you as an example of how you might visualize response actions in each phase of the incident response process. Let's quickly walk through the interface.

Instead of Tactics across the top of each column, we now have phases of the incident response cycle. You can also see various response actions listed below each header. Again, RE&CT is still in its infancy and there are some items in this matrix that are incomplete or missing. But there is enough structure here to use the matrix as a planning tool for mapping out incident response playbooks and identifying gaps in your response capabilities.

RE&CT Enterprise Matrix

Response Stages and Response Actions, colorized by Categories



Each response action in RE&CT is mapped to a specific response stage in the "PICERL" model: prepare, identify, contain, eradicate, recover, lessons learned. Response actions contain a numeric ID, the second digit of which denotes the category it belongs to:

- 0: General
- 1. Network
- 2. Email
- 3. File
- 4. Process
- 5. Configuration
- 6. Identity

Each of these categories are represented as a different color in the RE&CT Navigator. This allows you to see at a glance which stage and category each response belongs to based on its ID in the RE&CT framework and its color in RE&CT Navigator. For example, RA2202: Collect an email message is related to Stage 2 (Identification) and Category 2 (Email). Here's where you can find the response ID in each entry:

Title	Access internal HTTP logs
ID	RA1103
Description	Make sure you have access to internal communication HTTP logs
Author	your name/nickname/twitter
Creation Date	YYYY/MM/DD
Category	Network
Stage	RS0001: Preparation
References	<ul style="list-style-type: none"> • https://example.com
Requirements	<ul style="list-style-type: none"> • DN_zeek_http_log

Next, we'll use RE&CT to plan and document a set of response actions. Open a new tab in Firefox and navigate to the RE&CT project page:

<https://atc-project.github.io/atc-react/>

Scroll down a bit and you'll see the text/hyperlinked version of the RE&CT framework. Clicking on a response action will take you to a detailed entry like the one above.

Preparation	Identification	Containment	Eradication	Recovery	Lessons Learned
Practice	List victims of security alert*	Patch vulnerability*	Report incident to external companies	Reinstall host from golden image*	Develop incident report
Take trainings	List host vulnerabilities*	Block external IP address	Remove rogue network device*	Restore data from backup*	Conduct lessons learned exercise
Raise personnel awareness	Put compromised accounts on monitoring	Block internal IP address	Delete email message	Unblock blocked IP	
Make personnel report suspicious activity	List hosts communicated with internal domain*	Block external domain	Remove file*	Unblock blocked domain	
Set up relevant data collection*	List hosts communicated with external domain*	Block internal domain	Remove registry	Unblock blocked domain	

Leave this tab open; we'll use this page in the next step.

Pick an Incident

The first step in planning a response is to identify the attack scenario to which you'll be responding. For this exercise, let's pick a scenario published by the Center for Internet Security (CIS):

Malware Infection

SCENARIO: An employee within your organization used the company's digital camera for business purposes. In the course of doing so, they took a scenic photograph that they then loaded onto their personal computer by inserting the SD card. The SD card was infected with malware while connected to the employee's personal computer. When re-inserted into a company machine, it infected the organization's system with the same malware.

You may remember from the lecture that CIS also published the Critical Security Controls. Helpfully, they have also include references for applicable controls in this scenario:

- CIS Control 8: Malware Defenses
- CIS Control 9: Limitation and Control of Network Ports, Protocols, and Services
- CIS Control 12: Boundary Defense

This will give us a head start on our response planning and guide our approach. Head back over to the RE&CT framework web page and let's take a look at the specific actions we might take in this scenario.

Identify Response Actions

Since this scenario begins in the *Identification* stage, we don't need to include Preparation actions in our response playbook (though our Lessons Learned actions may reference additional Preparations for the next incident). Read the list of available actions in the Identification stage.

Identification

List victims of security alert*

List host vulnerabilities*

Put compromised accounts on monitoring

List hosts communicated with internal domain*

List hosts communicated with internal IP*

Remember that this is a framework, not an exhaustive catalog - it can be modified or expanded based on available actions within your environment. Considering this is a malware infection, we're probably most concerned with changes to the host and any suspicious network communications post-infection.

For this example, let's select the following actions:

```
List hosts communicated with internal IP  
List files created  
List files deleted  
List processes executed  
List registry keys created  
List services created
```

!!! Note: These actions, and others in the RE&CT Matrix, have asterisks next to them indicating that the detailed entries have yet to be completed by the RE&CT Project Team.

Repeat this process for the Containment and Eradication stages:

```
Containment:  
Quarantine file by hash  
Block process by executable hash  
  
Eradication:  
Remove file  
Remove service
```

Recovery actions will depend on what we find in the investigation and the results of our containment and eradication efforts, so we'll come back to that step.

There are only two actions currently documented in the RE&CT framework for *Lessons Learned*: "Develop incident report" and "Conduct lessons learned exercise". We probably won't need a lessons learned exercise for this kind of scenario, so let's just stick with this for now:

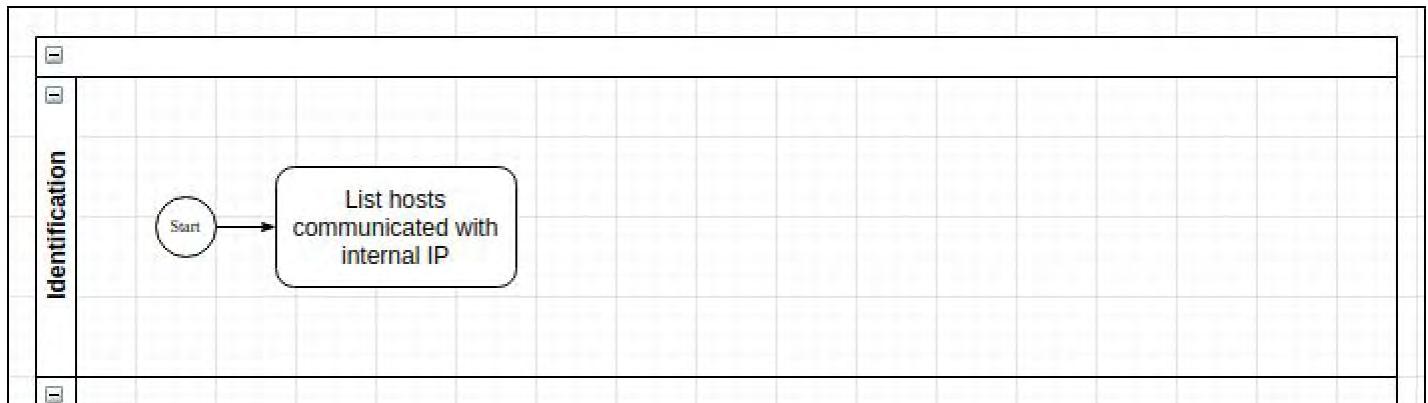
```
Lessons Learned:  
Develop incident report
```

Assemble Playbook

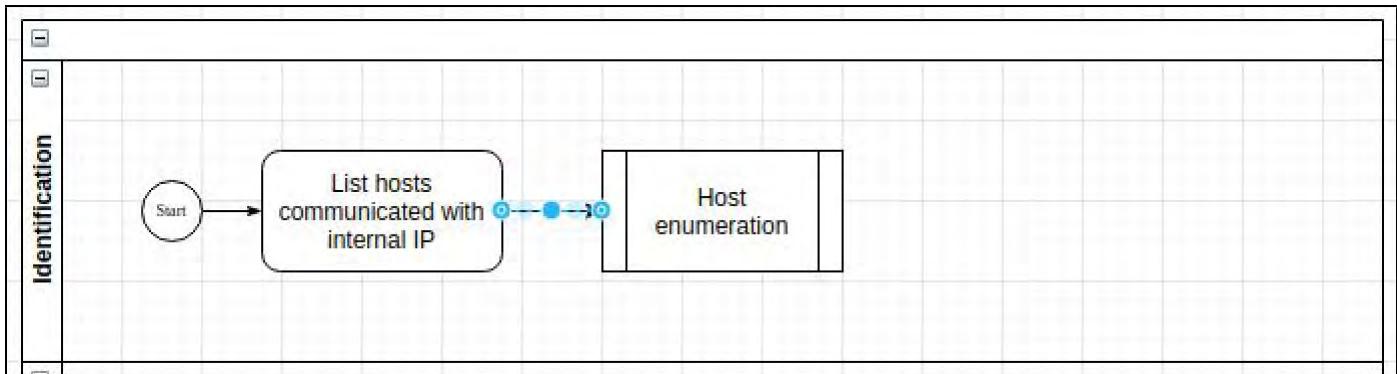
In an incident scenario, simply handing our team a list of response actions won't be very useful, so next we'll assemble our response steps into a playbook where we can arrange them in order and add decision points. We'll use another swimlane diagram in Draw.io to create a visual playbook for our malware response scenario.

```
Double-click on the Playbook-template.drawio file located in  
/home/student/Desktop/labs/4.2/
```

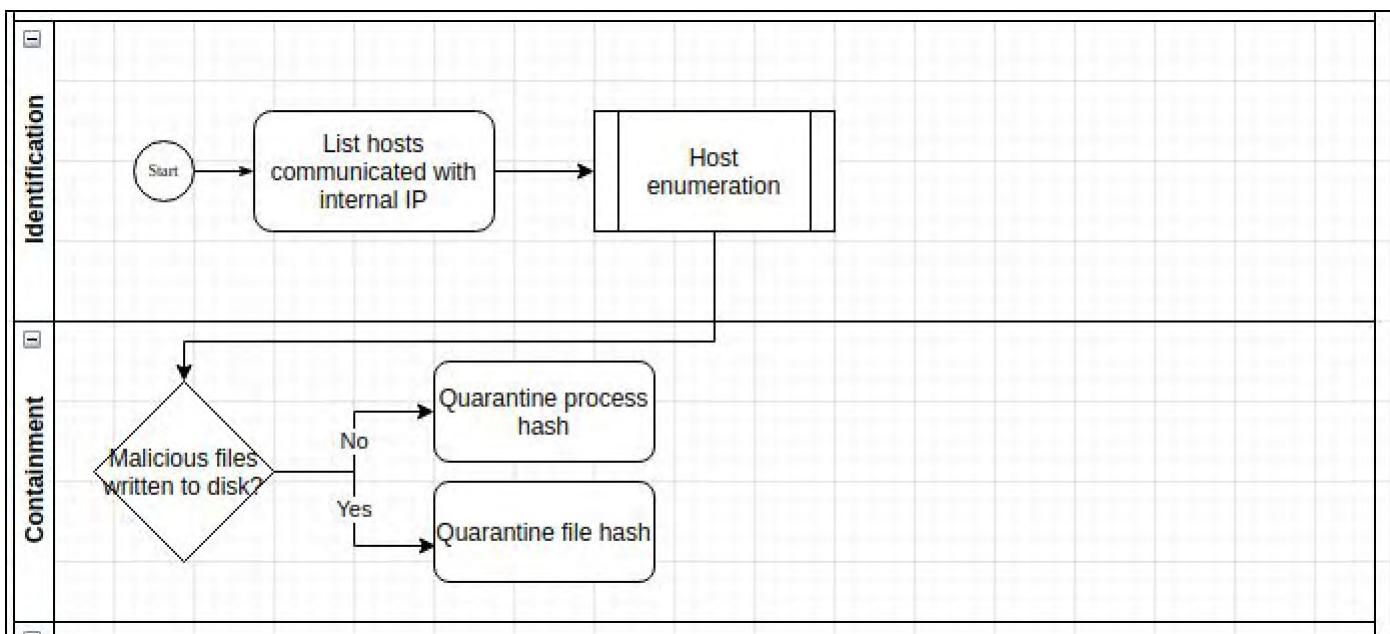
If our playbook is too long or detailed, it won't be useful in a real crisis scenario. So the goal is to assemble these steps in such a way that we don't miss anything, but we aren't stifling the responder's ability to think critically and think for themselves. Use the rounded square shape to build a flow diagram for the Identification phase using the actions we selected:



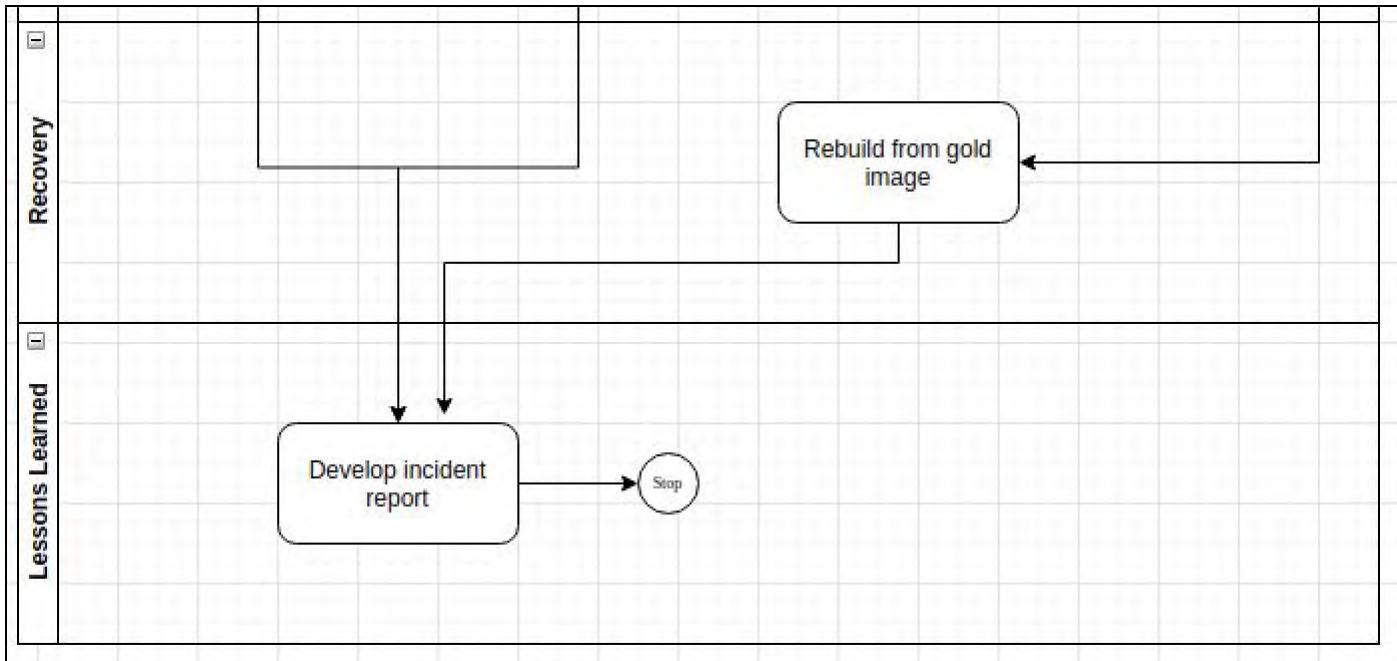
Since many of the steps in this phase involve gathering information from the host and the order and timing of those steps don't matter, we can consolidate them into a process called "Host enumeration":



Add steps for the Containment and Eradication stages based on the actions we selected. Note that some of the actions in these phases depend on the output of the previous actions - for example, our method of quarantine (isolating the file or process) depends on the malicious artifacts we find during Identification. So we can add those steps using the triangle "decision" symbol:



Once we have a sense of our containment and eradication approach, we can identify recovery efforts. "Rebuild from gold image" is a recovery action listed in RE&CT, but it may not be one we want to take in all cases. In this example, we may want to rebuild only if we have identified persistence mechanisms on the infected machine. From here, we can proceed to the Lessons Learned action we selected and close out the play:



!!! Note: We have included a completed example of the flowchart for this scenario, `Malware-playbook-example.drawio`, in the Lab 4.2 Files folder for your reference.

Operationalize Response Steps in a Case Template

RE&CT is designed to be portable and extensible, in that you can build response playbooks using each action and, if you are using The Hive, import response actions directly into case templates:

The screenshot shows the 'Case template management' interface. On the left, there's a sidebar with 'New template' and 'Import template' buttons, and a list of 'Current templates' containing 'RP_0001_phishing_email'. The main area is titled 'Case basic information' and includes fields for 'Template name' (set to 'RP_0001_phishing_email'), 'Title prefix' (set to 'Case title prefix'), 'Severity' (set to 'INFO'), 'TLP' (set to 'TLP:AMBER'), 'PAP' (set to 'PAP:WHITE'), 'Tags' (containing 'attack.initial_access', 'attack.t1193', 'attack.t1192', and 'phishing'), and a 'Description' field (set to 'Response playbook for Phishing Email case'). To the right, a large list of 'RE&CT Actions' is displayed, titled 'Tasks (33)'. The list includes various identification tasks such as 'RA_0001_identification_get_original_email', 'RA_0002_identification_extract_observables_from_email', and 'RA_0003_identification_make_sure_email_is_a_phishing'. Each task has an 'Edit' and 'Delete' button next to it. Red arrows point from the 'Tasks (33)' title towards the list of actions.

There are also pre-built Confluence pages for inclusion of RE&CT response actions in a team knowledge base:

Response Stages



Created by Daniil Yugoslavskiy
Last updated May 24, 2020

ID	Name	Description
RS0001	Preparation	Get prepared for a security incident.
RS0002	Identification	Gather information about a threat that has triggered a security incident, its TTPs, and affected assets.
RS0003	Containment	Prevent a threat from achieving its objectives and/or spreading around an environment.
RS0004	Eradication	Remove a threat from an environment.
RS0005	Recovery	Recover from the incident and return all the assets back to normal operation.
RS0006	Lessons Learned	Discover how to improve the Incident Response process and implement the improvements.

Like Be the first to like this

Each response action is defined in a YAML file that RE&CT converts into the required formats for each of these other artifacts. A response *playbook* is a set of response actions to be executed in response to a specific threat with optional mapping to MITRE's ATT&CK framework. Like the individual actions, playbooks are YAML files that can be converted into markup files (for web pages) and imported into Confluence as wiki entries or the Hive as case templates.

While we aren't going to create custom RE&CT YAML files for case templates in this exercise, you'll recall that we created our own case templates in The Hive in Exercise 1.3. We could operationalize the steps in our new playbook by creating a new template or adding these steps to an existing malware infection template that already exists. Whatever case management system your SOC team uses, it's important to understand that a streamlined set of steps *built in* to your toolset will save your analysts a significant amount of time.

Exercise Conclusion -- Key Takeaways

In this exercise, you have:

- Familiarized yourself with the RE&CT framework and its visualization application based on MITRE ATT&CK Navigator
- Selected response actions for a specific incident scenario
- Organized those response actions into a playbook and documented it visually using a swimlane diagram
- Saw how we might operationalize our response playbooks using Case Templates in the Hive

To shut down the services used for this exercise go back to your terminal window (or open a new one) and enter the commands below:

```
cd /home/student/labs/4.2
docker-compose down
```

Lab 4.2 is now complete!

Exercise 4.3 - Designing Tabletop Exercises

Background

Testing your incident response processes can save you lots of time and frustration when a real incident occurs. In this exercise, we will design a tabletop exercise based on our SOC planning efforts. The level of formality and planning a tabletop requires is highly dependent upon your goals, organizational maturity, and culture. However, as in many security disciplines, good documentation is key to building upon your efforts in the future and learning from the exercise.

Objectives

- Walk through pre-planning considerations
- Identify tabletop objectives
- Identify key participants and stakeholders
- Select a scenario and design the exercise

Exercise Preparation

This exercise is completed in your MGT551 Linux VM, if you have trouble with the setup, see the troubleshooting information in the wiki, or reach out to an instructor or SANS support.

Launch the **MGT551 Linux VM** and log in.

```
- LOGIN = `student`  
- PASSWORD = `mgt551`
```

Once you are at the Linux virtual machine desktop, you are ready to proceed with the exercise.

Exercise Steps

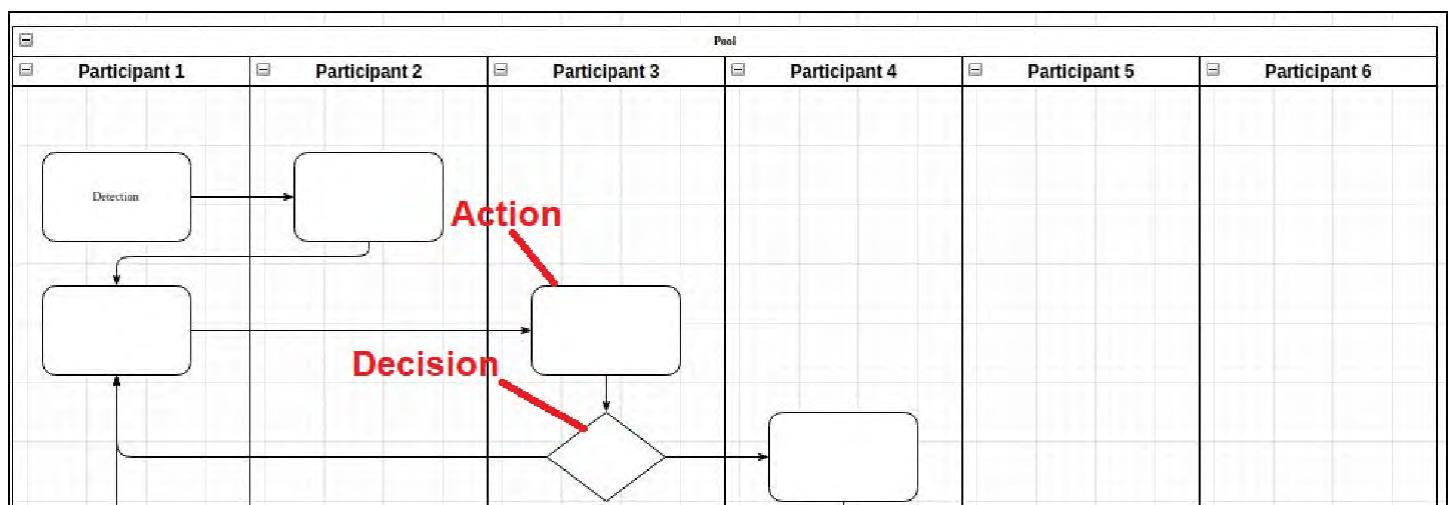
Open the Draw.io Template

In this exercise, we are going to use the Draw.io tool we used in Exercise 1.2. Log into your exercise VM and double-click on the Exercise 4.3 Files folder in your labs directory, or navigate to `/home/student/Desktop/labs/4.3/`.

Open the `TableTop-Swimlane-Template` file to open a draw.io swimlane diagram template:



We'll use this swimlane diagram to visually represent the flow of our scenario and identify the responsible groups or individuals. You will notice that the template has swimlanes for six participants and a variety of different actions and choices; we can modify these as needed based on the scenario and participants we select.



Identify Your Objectives

If you have not planned many tabletop scenarios, you may be tempted to select the scenario first. But remember that this exercise is a learning opportunity for your team. It is also a chance to review your plans to ensure that they are realistic, effective, and well-coordinated. During this planning phase, set aside the scenario and consider the **specific objectives** you have for your team or your organization in conducting this exercise. These goals probably include one or more of the following:

- generate new ideas for detection and response
- identify resource or organizational constraints
- examine plans, processes, and procedures for relevance and gaps
- clarify roles and responsibilities before, during, and after an incident
- enhance training through practice

Tip

Understanding what you are trying to achieve will help you figure out who you need to involve, what kinds of scenarios you will consider, and how you will conduct the tabletop. For example, if the goal is to test technical investigation and response processes, you'll probably want a highly technical scenario and only involve participants who perform various technical IR tasks. If your goal is to identify resource and organizational constraints, your scenario should involve activity that will require larger organizational participation - perhaps something that rises to the level of regulatory or legal considerations such as a data breach.

Remember, objectives should be SMART: specific, measurable, achievable, relevant, and time-bound.

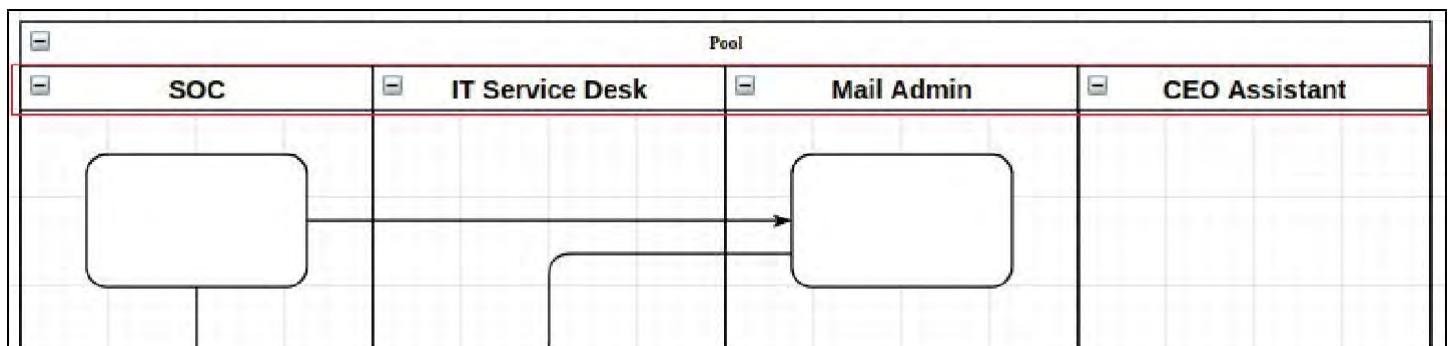
Select a Scenario

Refer to the attack tree you created in Exercise 2.1 for a list of possible attack scenarios you anticipate occurring within your environment. These are great candidates for tabletop exercises as they are both realistic and relevant to your organization - you want your team to know how to respond should these threats materialize in your network. If you would rather choose a more generic scenario, let's use one of the examples from Exercise 2.1:

While browsing the Internet, the CEO's browser was exploited by a malicious advertisement. The delivery phase of the attack loaded a remote access trojan on the CEO's laptop, which gave the attacker access to the CEO's machine. The attacker used this access to log into the CEO's e-mail.

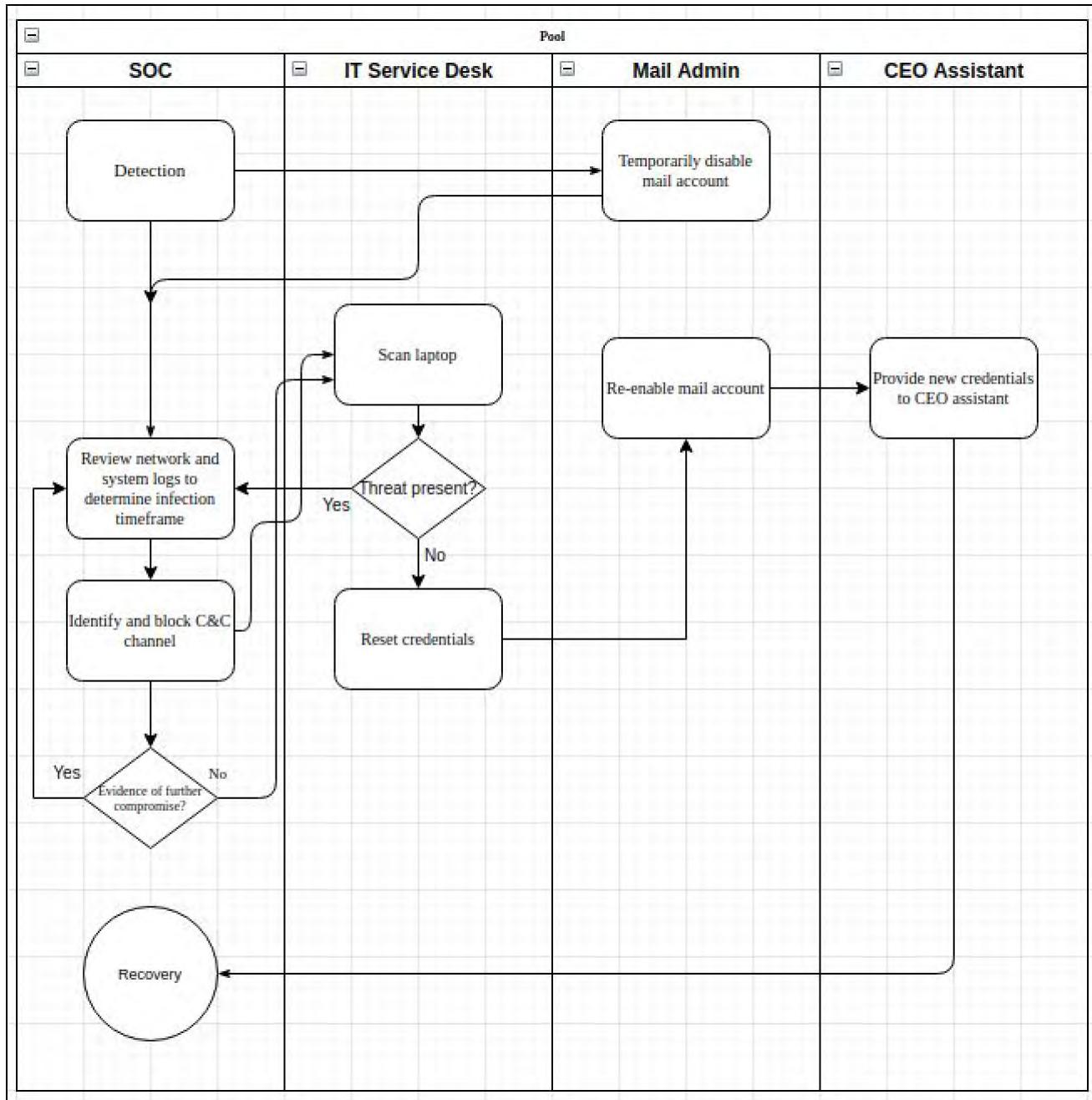
Select Participants

The next step is to select the key participants based on the scenario you chose. These might be individuals if the tabletop is happening on a team scale, or groups/functions if you're conducting the exercise at the division or organizational level. Think about the scenario you chose and enter the key participants from your organization in the boxes across the top of each swimlane. If you are using the example scenario from above, let's include the SOC, the corporate e-mail administrator, the CEO's administrative assistant, and the IT Service Desk:



Design the Exercise

Next, we will fill out the actions and decision points for each participant/group as appropriate. We have included a completed example based on our CEO laptop infection scenario for your reference, which is located in the Exercise 4.3 Files folder (filename "TableTop-Example.drawio"):



Tip

Writing a tabletop exercise with multiple decision points is not unlike writing a "choose-your-own-adventure" story. You want there to be enough choices and potential paths that the scenario is dynamic and realistic, but not so many choices that planning becomes unruly and the outcome becomes less clear. On the other hand, simplifying the scenario by removing decision points can make the exercise less engaging. Start by focusing on the ideal narrative for the scenario from start to finish. Then, as you add additional "branches" in the form of injects and decision points, ensure that all paths eventually lead back to your main narrative. This way, you can continue to control the outcome or conclusion of the scenario while giving participants the sense that they are driving these simulated events.

Now we have the basic workflow we would expect to see if this scenario had occurred for real. Of course we have simplified the process in this example, and there may be steps that would look different in your organization; ideally, this process would reflect the approved incident response playbook and inter-group operating procedures in your environment. In a live exercise, depending upon your goals, you would either let each group decide what to do next and compare it with this process flow, or prompt each group with the next step, discuss how they would execute and what results they would expect, and walk the group through to the next step.

If your goals include preparing the team to deal with the unexpected, you can also add injects to your scenario. This is your opportunity to get creative and see how well the team reacts when the scenario deviates from the narrative you described at the beginning of the exercise.

Tip

Injects are an opportunity for you as the facilitator to be creative, but they should still align to the goals of the exercise. For example, if the tabletop includes the operations team only, avoid injects that may require other participants or do not support your training goals for the team.

Following our laptop (and email) compromise example, we might add the following inject:

```
Shortly after blocking the C&C channel and resetting the CEO's credentials, one of your business partners reaches out to report some strange e-mails from the CEO around the time of the compromise requesting some accounting changes. It appears that the attackers used their access to attempt wire fraud.
```

How might your process flow change or expand based on this new information? What participants might need to jump back into the process to manage this new risk?

Consider the scenario you selected from exercise 1.2, or one that might be relevant for your team. What are some injects that you might add to increase the training value of the tabletop based on related events or findings you are likely to encounter?

Exercise Conclusion – Key Takeaways

In this exercise, we have:

- Asked questions about our adversaries and their capabilities based on what we know from our initial defensive planning and activities observed within our environment
- Broken down those questions into priority intelligence requirements
- Considered format and application of the responses to our requirements
- Provided feedback on intelligence products we have received

Supplemental Resources

While we did not use it in this exercise, we also have provided a planning template which you can find in the `/labs/4.3` directory on your exercise VM (called `tableTop.xlsx`). It was derived in part from a template published by the (National Association of Regulatory Utility Commissioners and guidance from DHS Cybersecurity and Infrastructure Agency (CISA). The document contains two tabs: one with a notional schedule and task list for planning a tabletop exercise ("TTX Planning"), and one for describing your scenario ("TTX Scenario"). Feel free to use this template to document your own tabletop planning efforts.

Exercise 4.3 is now complete!

Exercise 5.1 - Training and Career Development Planning

Background

Training is an integral part of empowering your SOC team and positioning individual team members for near-term and long-term success. Too often, training “plans” are no more than lists of classes or ad hoc recommendations specific to each job role, or worse, generic technical security training that may not even be particularly relevant. Training should not be handled casually but instead developed specifically to meet the needs of the SOC and the goals of each individual analyst. There are many considerations in developing a robust, effective, and relevant plan. In this exercise, we will baseline technical knowledge and training in your SOC, identify areas requiring training and professional development, and outline training and career development plans that foster individual empowerment and continuous team improvement.

Objectives

- Use the SOC-CMM model to measure knowledge management and training in your SOC
- Inventory technical skills and knowledge within your team
- Identify training objectives to fill gaps
- Develop a training plan for SOC staff
- Measure progress towards learning objectives

Exercise Preparation

This exercise is completed in your MGT551 Linux VM, if you have trouble with the setup, see the troubleshooting information in the wiki, or reach out to an instructor or SANS support.

Launch the MGT551 Linux VM and log in.

```
- LOGIN = `student`  
- PASSWORD = `mgt551`
```

Once you are at the Linux virtual machine desktop, you are ready to proceed with the exercise.

Exercise Steps

Files for this exercise are located in the `Lab 5.1` folder on your lab VM desktop:



Assess Knowledge Management and Team Training

Recall from the lecture that the SOC-CMM is a capability maturity model developed for SOC teams by Rob Van Os. One of the five domains in SOC-CMM is People, and we can use this model to baseline our knowledge management and team training capabilities.

Click on the `soc-cmm.xlsx` file in the Lab 5.1 Files folder:



Complete Profile Tab

Click the **home** icon, then click **1. Profile** under the General domain:

A screenshot of a Microsoft Excel spreadsheet titled "soc-cmm.xlsx". The spreadsheet has a light blue header row with the text "Click on any section name to proceed directly to that part of the assessment". Below this is a table with two columns: "Domain" and "Section".

Domain	Section
Introduction	1. Introduction 2. Usage
General	1. Profile (This cell is circled in red) 2. Scope

Fill out the details for your SOC and set your target maturity scores. Remember from the lecture on Day 1 that a maturity score of 5 is probably not ideal due to the rigidity it would require. We will not be using the Capability scoring in this exercise since those scores are specific to the technical domains, so you will not need to fill out target Capability scores.

Assess People Domain

Click the home icon again. Then click on 1. Employees next to People:

People	1. Employees 2. Roles and Hierarchy 3. People Management 4. Knowledge Management 5. Training and Education
--------	-------------------------------------------------------------------------------------------------------------------------------

Use the drop down menus and text boxes to answer each question about your SOC team. Note that when you select an answer from a drop down menu, the associated guidance is automatically populated in the adjacent cell. You can use this guidance to gauge your answers.

Do the same for the Roles and Hierarchy, People Management, Knowledge Management, and Training and Education sections. You can either select each menu item or use the arrow icons  to move through the list.

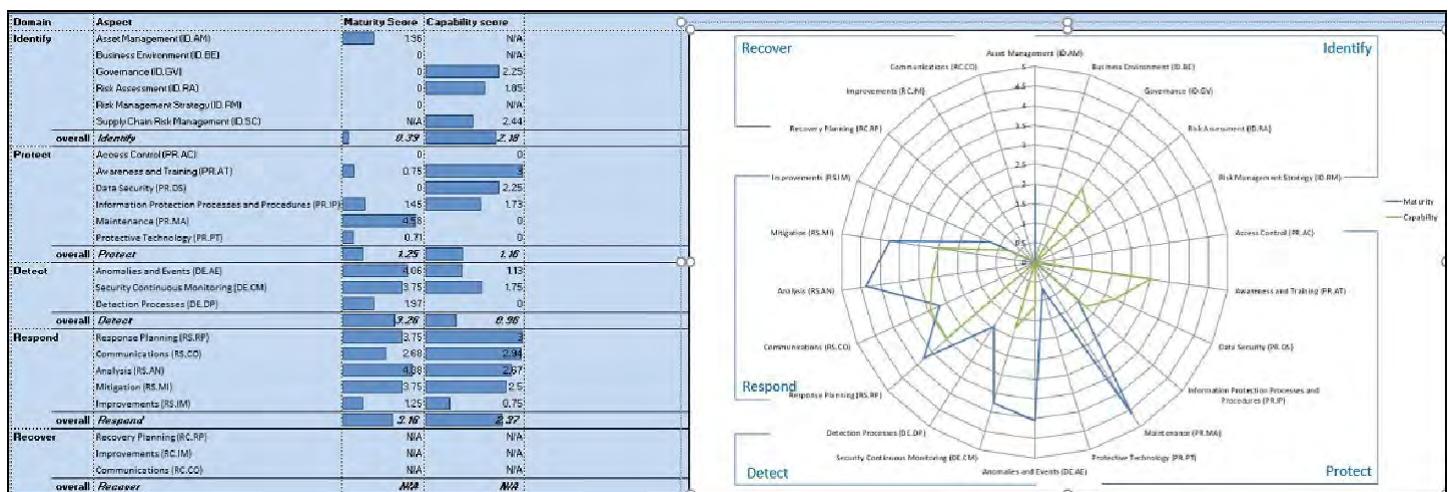
Review CSF Scoring



Click the  icon at the top of the spreadsheet to skip to the results. Under Results, you will see maturity and capability scoring for each domain. You will also see a comparison of your current score to the Maturity Target scores we set in the last step.



The SOC-CMM is aligned to the NIST Cyber Security Framework (CSF) and includes reporting aligned to that framework. Click on NIST CSF 1.1 Scoring to view it. Since we have only populated the People domain, most of the CSF domains will not be scored. However, you should already see some maturity scores filled in based on your inputs.



We can use these maturity scores to help us baseline our staffing and identify organizational needs to be filled by training and career development - in this case, primarily around training and security awareness. Completing the other domains and aspects in the SOC-CMM will give you even more insights into capability and maturity improvements that might be made via additional training.

Note

As you can see in the scoring results, SOC-CMM scores along two scales: maturity and capability. While things like documented processes and structure can increase your maturity score, they don't necessarily mean more **capability** - performance measurement and continuous improvement demonstrate capability and will improve scoring on that scale.

Conduct a Needs Assessment

We now have at least two sources of information that should help us highlight skill, knowledge, or process gaps in the SOC:

1. Our SOC-CMM assessment
2. The output of our case quality review from Exercise 4.3.

But in order to develop detailed training plans specific to each role and each team member, we need to dig a bit deeper. Performing a needs assessment will help us gather additional inputs to our training plan.

Organizational assessment

Compare the gaps you perceive with the SOC's strategic goals and target metrics. Capture the resulting items - tasks, processes, etc. - in discrete, singular terms. For example, let's say static analysis of malware falls within the SOC's

purview. You have noticed that despite formal work descriptions, SOPs, new hire training, and the availability of tools and a testing environment, the team's static analysis work tends to be incomplete or incorrect. Express this training need as static analysis or a similarly specific requirement.

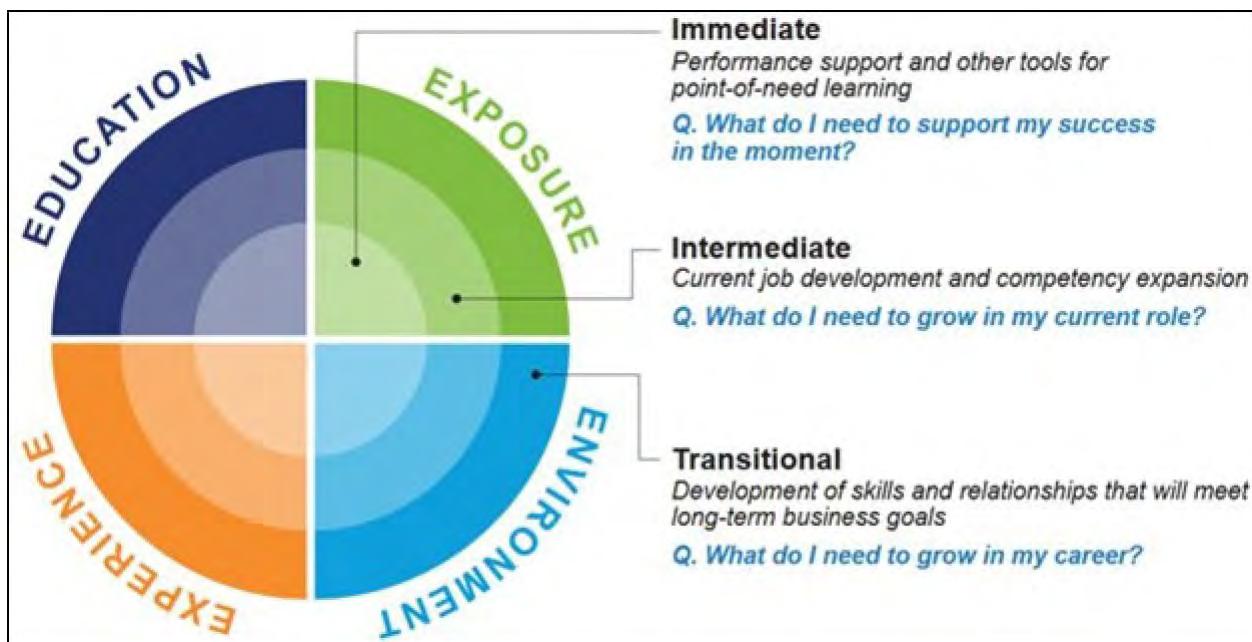
Keep in mind that this is a *team* assessment. There may be individuals in the SOC who have the skills and abilities you need, but if that knowledge is not effectively shared and/or does not come through consistently in the team's work products, the gap remains. We'll come back to team training needs in a moment.

Occupational assessment

This assessment covers the specific tasks, skills knowledge, and abilities required to do jobs within the organization. If you scored highly in Roles and Hierarchy maturity in the SOC-CMM assessment, you likely have formal job descriptions and a well-defined team structure - these will be the primary sources of information for this portion of your assessment. The third assessment we will do is at the individual level, after which we will be ready to complete this framework.

Individual assessment

Now it is time to assess training needs within your team on an individual level. Recall the Continuous Learning Model we discussed in the lecture on day 1:



Training requirements for each team member can be organized into three categories:

1. Things they need to be successful in their jobs right now
2. Things they need to grow in their current role
3. Things they need to advance their career beyond their current role

This structure should lend itself to defining clear requirements for individual, intermediate, and transitional learning for each individual on your team.

Info

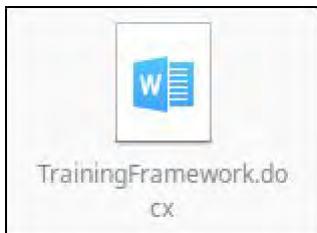
Educator and researcher David Orr and others developed a model for transformative learning referred to as *Head, Heart, and Hands*. These terms refer to different levels of learner engagement which can be addressed by various kinds of training. The "head" refers to a lack of knowledge, the "heart" refers to lack of belief or motivation, and "hands" is the lack of practical skills or experience. Technical skills can be addressed through mentorship and on-the-job training, foundational knowledge can be built through formal instruction and memorization, and motivation can be improved by empowerment, seeing teammates succeed, and positive messaging from leadership. Keep this model in mind as you develop training plans for your SOC team - you will definitely encounter all three of these scenarios!

Analyst self-assessment

Judging someone's skills, knowledge, and abilities can be difficult to do objectively. Just as bias can creep into our analysis without our awareness, bias can influence our perception of our own team. Having your analysts complete a self-assessment can give you insights into how they view their own skills and abilities, areas in which they wish to improve, and their own goals and objectives. This input will be a key part of your individual planning efforts as a manager.

Build a Training Framework

Click on the `TrainingFramework.docx` file in the Lab 5.1 Files folder:



The Training Framework document is a Word version of this table:

Training Framework	Plan
Needs Assessment	<i>The skill or knowledge needed</i>
Delivery Mode	<i>Format of the training</i>
Budget (per person)	<i>Cost of the training</i>
Delivery Style	<i>The way the knowledge is transferred</i>
Audience	<i>Training recipient - usually the role requiring the training</i>
Goals and Learning Objectives	<i>What we want out of the training</i>
Timeline	<i>Training schedule</i>
Communication	<i>How the training will be announced/coordinated</i>
Measurement Method	<i>How will we know the training has been effective</i>

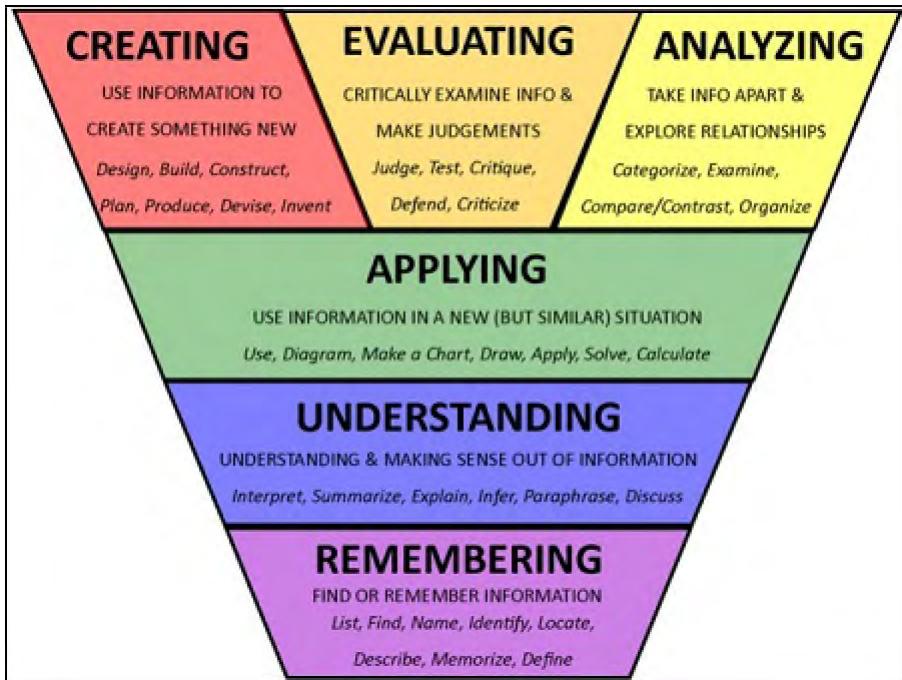
This is a simple framework developed by the [University of Minnesota](#) that outlines the training need, delivery mode and constraints, learning objectives, timeline, and other details. Let's use an example of the Tier 1 Security Analyst role, with Alert Investigation and Case Management as two items we identified based on our Needs Assessment. Enter these items in right-hand cell of the first row in the first and second tables next to "Needs Assessment":

Training Framework	Plan
Needs Assessment	
Delivery Mode	Choose an item.
Budget (per person)	
Delivery Style	Choose an item.
Audience	

Fill in Delivery Mode, Budget, and Delivery Style with what you think is most appropriate for each of these skills. Audience will be "All Tier 1 Security Analysts".

Define Learning Objectives

In the 1950s, psychologist Benjamin Bloom and a group of colleagues created what is known as Bloom's Taxonomy, which is a framework for levels of understanding. It's a great reference model that can help you organize your thinking about the knowledge your team needs. The model contains six levels of learning, or "cognitive gain," that start at basic understanding and get progressively more in-depth as they move upward towards creating and evaluating.



We obviously want our Tier One Security Analysts to be at the "Applying" level of Bloom's taxonomy, so we can use this phrasing in our learning objective:

Goals and Learning Objectives	Apply alert triage techniques to consistently and accurately prioritize critical and high-impact alerts for investigation.
--------------------------------------	----------------------------------------------------------------------------------------------------------------------------

For this exercise, we can leave the timeline and communication items blank or just put arbitrary text in those boxes. For "measurement method," select the most appropriate choice from the drop down menu embedded in the cell.

We now have simple, tailored training frameworks defined for two key skills for our Tier 1 SOC Analyst!

Develop a Career Development Plan

Now that you have identified training needs and created a framework to address those needs, it is time to focus on strategic planning to continue developing our team. Many organizations use third party software for their review process and career planning, so we want to use something that is simple, straightforward, and can be ported into any form or tool relatively easily. Here is an example career development plan for our Tier 1 Security Analyst based on our assessment and training framework:

Today's Date	March 12, 2021
Employee	Samus Aran
Current Job Title	Tier 1 Security Analyst
Goals	<ul style="list-style-type: none"> - Apply alert triage and investigation skills - Ensure investigations are well-documented and resolved in a timely manner according to SOC SOP
Training Needed	Tier 1 shadowing, offline SOP study
Next Milestone	100% investigations/escalations completed within SLA with minimal defects
Estimated Costs	\$0
Completion Date	December 2021
Manager Notes:	Following completion of in-house desk side training, Samus is eligible for the SANS SEC 450 course. Tier 2 Analyst Kung Lao will supervise, SOC Manager Cassie Cage will sign off on SLA compliance and quality checks. Next steps: assume monitoring and triage duty.

Both our Tier 1 Analyst and his manager would sign off on this plan and have something to revisit at their next progress check.

Collect Training Metrics

Of course, we won't get those Capability scores up unless we are measuring and improving! Once the training and development plans have been finalized and put into practice, ask the following questions as you monitor the process:

1. **Reaction:** How did the participants react to the training program?
2. **Learning:** To what extent did participants improve knowledge and skills?
3. **Behavior:** Did behavior change as a result of the training?
4. **Results:** What benefits to the organization resulted from the training?

Answers to these questions should feed into your next organizational and individual assessments, and give you the opportunity to adjust training content, delivery style, or format.

Exercise - Key Takeaways

In this exercise, we have:

- Used the SOC-CMM to baseline staffing, knowledge, and skills within our team
- Conducted a needs assessment on an organizational/team level and an individual level
- Used Bloom's Taxonomy to define learning objectives for SOC training needs
- Built custom training and career development plans based on defined learning objectives

Exercise 5.1 is now complete!

Exercise 5.2: Creating, Classifying, and Communicating Your Metrics

Background

Finding the right metrics for your SOC can be an incredibly difficult task. First you must identify metrics that are useful, and even within those metrics, you must find ones that are practical and easy to collect. In this exercise we will tackle this problem. In the first step, we will take a top-down approach to develop a list of metrics tied directly to goals the SOC has set. In the second portion, we will take these metrics and further refine and classify them so that you will have a deeper insight into which options are the best choices, and why. The goal is to emerge from this exercise with a good sense of what makes metrics meaningful, and which options will be *best* for your unique team.

Objectives

- Learn how to derive which metrics you should (and should not) be collecting
- Use the GQM system to map goals to collected metrics
- Justify the business value and justify the effort required to collect your metrics
- Explain the factors that make a chosen metric useful to measure
- Classify and gain insight into the nature of your metrics
- Suggestions for effective presentation and communication of metrics

Exercise Preparation

This exercise is completed in your MGT551 Linux VM

Launch the MGT551 Linux VM and log in.

```
- LOGIN = `student`  
- PASSWORD = `mgt551`
```

Exercise Steps

Which Metrics Should You Collect?

One of the biggest errors made in many SOC (and other) teams is collecting measurements simply because they are convenient to collect or are automatically produced and reported by our tools. Are these numbers truly helpful though? How do we define helpful in this sense to even answer this question? If we back up and think about the goal of metrics - feedback for continuous improvement and operations, it stands to reason that, if a measurement is meaningfully

contributing to feedback, then it is a useful and good thing to collect. The overarching goal of any metrics collection effort should be to aid in decision making. Therefore, if a metric isn't contributing to any meaningful decision-making process, it may not be worth collecting.

This information brings us to a core principle of metrics selection, and that is that the **measurements you collect must be chosen in a "top-down" fashion**. Metrics collected should flow directly from a specified goal you want to hit, or operational objective you want to maintain, and assist directly in ensuring measuring progress or sustainment of that objective. If you look at a metric and wonder "What am I supposed to do with this?" it's likely that metric isn't contributing to decision making in any meaningful way. In other words, you should not simply look at the numbers you are given and only answer questions based on the data you have available. You should first define the goals you need to hit, then seek to produce and collect the metrics that tell you if you are on track.

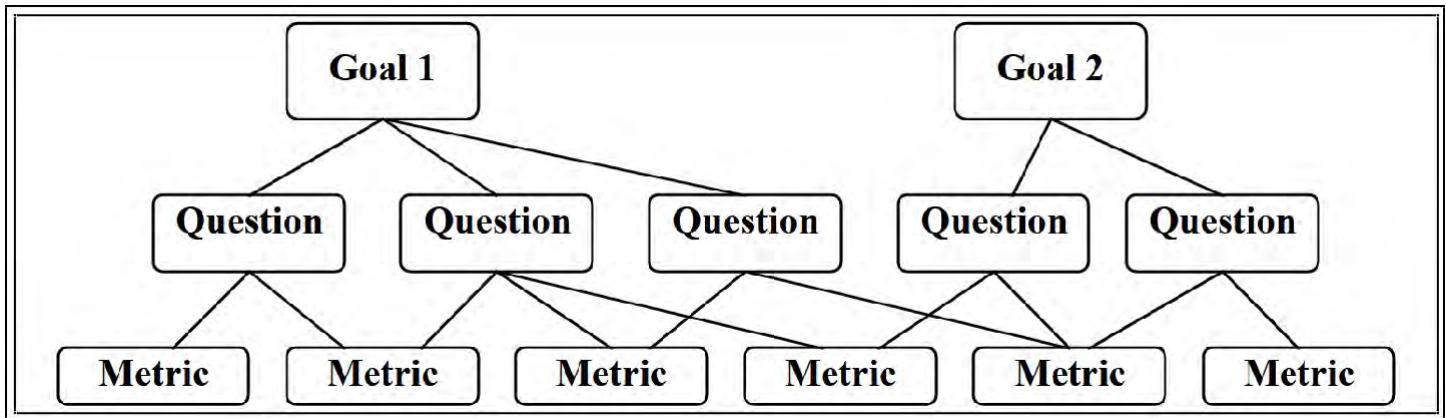
This top-down approach is the process suggested by the well-known "[Goal, Question, Metric](#)" (GQM) system, an approach first developed to derive important metrics for software development by Victor Basili of the University of Maryland, College Park and the Software Engineering Laboratory at the NASA Goddard Space Flight Center. As you will see in this step of the exercise, although GQM was originally designed for software, its usefulness has been successfully adapted to use far beyond the software engineering world. The GQM framework and its later derivatives form a highly useful defined and repeatable process for organizations to derive metrics that directly support the goals they wish to achieve.

The GQM Process

GQM helps us answer the question of which metrics to collect by directly tying those metrics to a purpose they serve. It provides an important mental framework to understand *why* you are collecting a given metric and weed out those that are not contributing in a meaningful way. To do so, GQM has you start not with the metrics themselves, but first with the GOALS or OBJECTIVES you have for your team - meaningful, important actions that deliver clear value to the business. Next, QUESTIONS that support answering whether or not the goal is being achieved are listed. Finally, the source data (METRICS) needed to answer these questions is listed. This final step is where your list of important metrics to collect is created.

Since all METRICS assist in answering a chosen QUESTION, which in turn helps the team know if they are hitting their defined GOAL, by necessity, each metric will be an important measurement to collect! Additionally, this system helps us map out what metrics feed which questions and goals, which in turn identifies if a single measurement may help us answer multiple questions, meaning it is useful in making decisions in more than one area. Using the GQM system for deriving metrics complements the KPI/OKR paradigm because as you should recall from the slides, while both KPIs and OKRs are used differently, both require a well-chosen metric. GQM helps us explicitly define the use of each metric and aids traceability from the measure, and supporting KPI or Key Result, to the overall goal it is to support achieving.

To make this clear, here is how the original GQM paper linked above represents the Goal-Question-Metric hierarchy of items.



As you can see, a goal is linked to related questions. Each question, in turn, is linked to metrics required that answer that question. In some cases, metrics may contribute to answering more than one question, a condition the GQM system can help you identify.

Note

After GQM was initially defined, the system was modified by others to use it in a more general sense beyond software engineering. One of those derivative goal-based measurement systems is [GQIM \(Goal, Question, Indicator, Metric\) from the Software Engineering Institute at Carnegie Mellon University](#). We suggest as follow-on work to this exercise that you read this linked document on what SEI calls the "GQIM" method and the notes on a workshop that was led based on it. It provides outstanding additional info on this system and examples of how it can help define cybersecurity-related goals. GQIM takes GQM a step further by specifically aligning goals to business objectives and inserting a layer of "indicators" between questions and metrics. While GQIM is a useful addition to the GQM system, in the interest of simplicity we will use the GQM model for this exercise as it supports the key desired takeaways for this lesson without introducing more detail than necessary to the exercise.*

Defining Goals, Questions and Metrics - GQM Example

How does GQM work in practice? Let's walk through a simple example.

Perhaps you have an objective to keep your car in good operating condition, what goals can be derived from this?

Objective	Goals
O1. Keep car in a reliable operating condition	G1. Ensure the car has enough oil at all times
	G2. Ensure the oil in the car is always within specific operating parameters for mileage and age

From these goals, we must now come up with some key questions that relate to it. The idea of the questions layer is to derive a set of questions that, if answered, lead you to know if you met that goal. The literature on the GQM method suggests you consider questions like "What is the process for..." for the creation of implementation metrics, and "How effective is..." for effectiveness metrics. Questions like this will help identify the data required to answer the question in the next step.

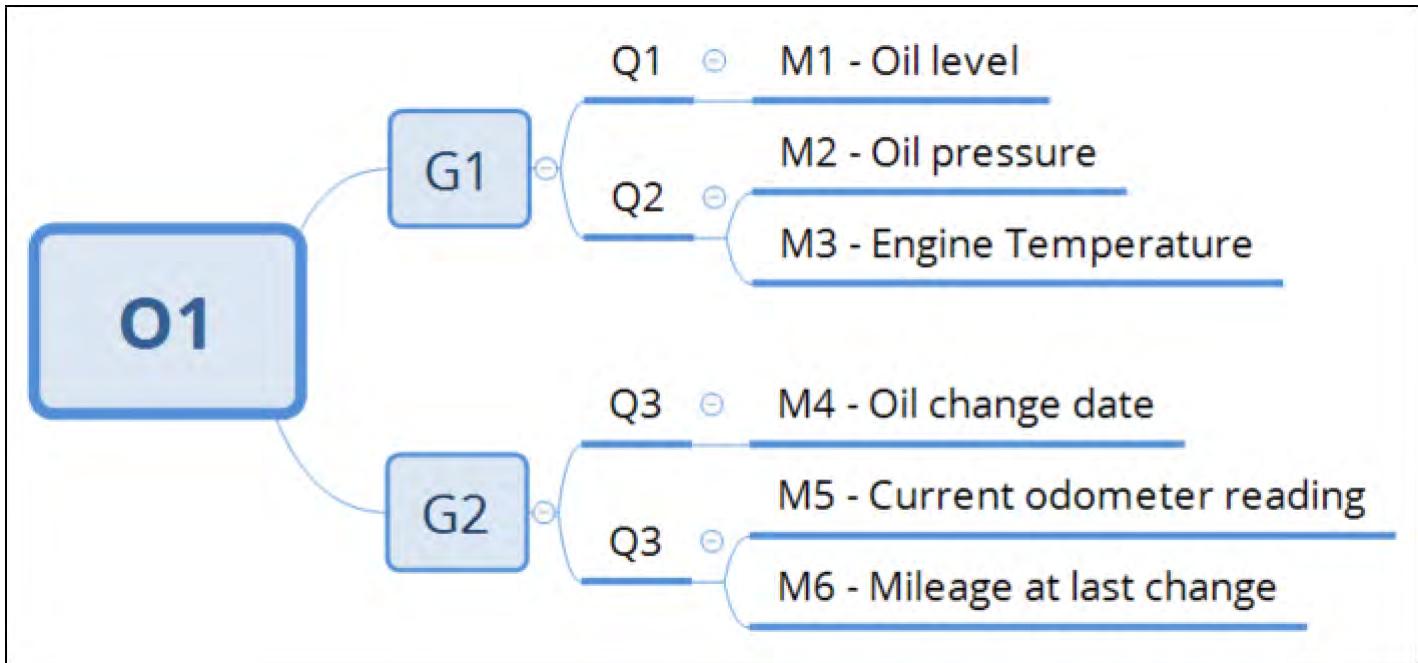
Goals	Questions
G1. Ensure the car has enough oil at all times	Q1. What is the process for checking the oil level in the car?
	Q2. How will we know if the car becomes unexpectedly low on oil?
G2. Ensure the oil in the car is always within specific operating parameters for mileage and age	Q3. What is the process for knowing when the last oil change occurred?
	Q4. How will we know when we reach the mileage limit of the oil?

Now that we have our goal and questions defined, we can consider the operational processes and systems behind these questions and start to derive the specific data (metrics) we must seek to answer the questions. The [SEI GQIM literature suggests](#) you qualify measures by asking "if you have this data, will you be able to answer some aspect of the question?"

For the oil example, this may be:

Questions	Metrics
Q1. What is the process for checking the oil level in the car?	M1. Dipstick oil level
Q2. How will we know if the car becomes unexpectedly low on oil?	M2. Oil pressure
	M3. Engine Temperature
Q3. What is the process for knowing when the last oil change occurred?	M4. Oil change date on window sticker
Q4. How will we know when we reach the mileage limit of the oil?	M5. Current odometer reading
	M6. Mileage on window sticker

At this point, we have now taken a high-level objective and goal, and from the top-down developed a list of key metrics to track (M1 - M6). Here is how the GQM hierarchy looks for this example:



Since we have started at the left from our objectives and goals and moved towards the right, all metrics should directly support key initiatives and operational goals we have set for ourselves, and this is the point of the GQM system. While all metrics will indeed support our goals, that doesn't mean every metric is equally important, a key second step to the process is reviewing metrics from this step against several considerations, but before we do that, let's use this process to develop some key metrics for our own SOCs from the top down.

GQM for theSOC

In this step you will walk through the same process we just demonstrated but use a SOC specific objective and goal to start from and develop the metrics required to support it using the GQM paradigm. For those that do not have a SOC yet, we'll provide a cyber security-specific example to read through simultaneously.

For your work in this step, a goal-driven measurement worksheet has been created in your ~/labs/5.2 folder. Enter the following command on a terminal to open the goal-driven measurement worksheet:

```
libreoffice /home/student/labs/5.2/goal-driven_measurement_worksheet.odt &
```

Note

You will receive an error that says "Could not find the Java Runtime Environment ..." You can ignore this error.

You should now have the worksheet open in LibreOffice Writer:

MGT551 Lab 2.2 – Goal-driven Measurement Worksheet

1 Objective to Goals:

SOC Objective: "Minimize impact of cyber attacks on the organization"

Goals related to your objective:

- G1:
- G2:
- ...

Use this sheet to record your work through the following steps.

OBJECTIVES AND GOALS

To start, first, consider a current OKR objective (or Wildly Important Goal) for your SOC that you would like to develop or check your metrics for. This could be anything from a current initiative to a longer-term definition of success for your SOC. After selecting an objective, use it to think of several clear goals associated with the achievement of that objective.

Example: Since nearly every SOC has the high-level objective to "Minimize the impact of cyberattacks on the organization" we'll use this as a generic example. From this objective, we could pick several specific goals such as G1 and G2 below.

- O1: Minimize the impact of cyber attacks on the organization
- G1: Reduce attacker capability to perform lateral movement
- G2: Reduce the impact of phishing through detection and blocking of malicious email

Action: Note your chosen Objective and related goals in the designated area in section 1 of your worksheet. Once you have completed listing your goals, move on to the next step.

QUESTIONS

From the goals you have noted down, the next step is to come up with some key questions that relate to them. The idea of the questions layer is to derive a set of questions that, if answered, lead you to know if you met that goal. The literature on this method suggests you ask questions in the form of "What is the process for..." for the creation of implementation metrics, and "How effective is..." for effectiveness metrics.

Example: For our example goals, here is what an organization may come up with:

- G1: Reduce attacker capability to perform lateral movement

- Q1: How do we monitor the network for signs of lateral movement?
- Q2: How do we restrict lateral movement between hosts and network segments?
- Q3: What process do we use to identify suspicious connections within the internal network?
- Q4: Which protocols are allowed/blocked from use on the network?
- G2: Reduce the impact and frequency of phishing incidents
- Q5: How many malicious emails are delivered?
- Q6: How are phishing attempts reported to the security team?
- Q7: What is the makeup of phishing messages that get past filters?
- Q8: What is the process to record the impact of phishing?

Of course, you could take each of these further, but for the sake of simplicity for the exercise, we'll stop at this list.

Action: In section 2 of your worksheet, copy and paste the goals from section 1 into a bulleted list, and nest under each one the relevant questions that relate to it. Be sure to give every question a unique numeric label. Keep in mind that these questions should probe the processes and systems that relate to your stated goals and inspire the development of good data measures in the next step. Once you have completed brainstorming feel free to go back through and edit questions out that might not work well, the idea is to inspire ideas that will lead to good measurement methods. Once you are happy with the set of questions that relate to your goals, move on to the next step.

METRICS

Next, we must think about what measures can be generated that will answer these questions. The [SEI literature suggests](#) when coming up with this list you ask, "if you have this data, will you be able to answer some aspect of the question?"

Example: For the example goals and questions, lets now think of some of the measurements we can take to either answer the questions directly, or address the processes or systems behind the question. Here are some metrics you might come up with for each of these questions.

- G1: Reduce attacker capability to perform lateral movement
- Q1: How do we monitor the network for signs of lateral movement?
 - M1: Number of deployed network metadata sensors and IDS appliances deployed
 - M2: Percentage of subnets with traffic visible to sensors
- Q2: How do we restrict lateral movement between hosts and network segments?
 - M3: Percentage of hosts with host firewall enabled blocking ports used for lateral movement
 - M4: Percentage of firewalls with lateral movement prevention/detection rules enabled
- Q3: What process do we use to identify suspicious connections within the internal network?
 - M5: Number of suspicious traffic alerts per month

- M6: Number of false-positive suspicious traffic alerts per month
- Q4: Which protocols are allowed/blocked from use on the network?
- M7: Percent of explicitly disallowed protocols with detection analytic coverage
- M8: Verified test cases for suspicious traffic detection
- G2: Reduce the impact and frequency of phishing incidents
- Q5: How many malicious emails are delivered?
 - M9: Total number of incoming messages per month
 - M10: Total number of incoming messages marked as malicious upon delivery
 - M11: Count of file extensions blocked by email filter
- Q6: How are phishing attempts reported to the security team?
 - M12: Phishing emails reported to the security team per month
 - M13: User-reported, verified phishing attempts per month (spam filtering misses at minimum)
- Q7: What is the makeup of phishing messages that get past filters?
 - M14: Verified phishing attempts with malicious links
 - M15: Verified phishing attempts with malicious attachments
- Q8: What is the process to record the impact of phishing?
 - M14: Number of incidents with phishing as delivery phase
 - M15: Impact of phishing related incidents (machines compromised, dollars, or other)

This shortlist of questions easily produced 15 metrics and likely could have easily gone further. If we step back and ask ourselves "does this make sense?", I believe the answer is a clear yes. Measurements such as the visibility of sensors and sensor count would clearly have a direct impact on the ability to spot lateral movement, as would hosts with firewalls enabled and network firewalls blocking commonly abused protocols. To minimize phishing, the numbers chosen would help scope the problem and classify the types of failures that are occurring so that they can be focused on in the future, they also give us a sense of the size of the problem, so we know how to prioritize it against other defense initiatives.

Hopefully reading through this list gives you an idea for the relationship between the metrics you pick and the goals and questions you have defined so that you can create a similar list for yourself.

Action: In section 3 of your worksheet, copy and paste the questions from section 2 into a bulleted list, and nest under each one metrics that relate to, or help you answer each question. Ask yourself if knowing that measurement will have a direct bearing on answering your question, and therefore help in achieving the goal and stated objective it connects to. Be sure to use a unique numeric name for each metric for tracking purposes. Once you've completed listing your metrics, continue to the next step in the exercise - reviewing your choices.

Reviewing Your Chosen Metrics

You've completed the first step! You've now used the GQM framework as a way to brainstorm meaningful metrics for your SOC that are tied to your high-level goals, objectives, OKRs, or WIGs! Take a look at the list you've created. Is it similar to the metrics you *are- collecting right now? If not, hopefully, this activity has given you some ideas of where you can improve your metrics collection, as well as how to evaluate what you are collecting and connect it upstream to a defined purpose.

We aren't done just yet though; we can take the list of metrics you've developed and look at it from some additional angles to help highlight strengths and weaknesses. While your list is a great start, that doesn't mean everything on it is equally useful or dependable. Other factors must be considered when deciding on a data collection strategy.

High Level Review

To start the review step, the GQIM literature suggests you look back at the metrics you've now decided upon and double check them by considering the following:

- **What decision will the metric inform?**- This should easily be determined by reviewing your GQM hierarchy
- **What actions would I take based on this metric?**
- Note that if you do not have a direct answer to this question, the metric likely has no target/range and may be part of a larger equation. Likewise, not all metrics are KPI material since they don't have a target or range, but this does not mean it isn't useful in supporting the goal. An example of this is the odometer reading in the oil example, in of itself it does not tell you what to do, only when used in the context of the mileage at which the next oil change is required does it become a number you can act on. The current odometer reading is still useful though as part of the greater KPI "Miles until oil change required" in which it plays a part in calculating the target. In that case, the equation for the target is `Mileage at next oil change - Current odometer reading >= 0`. If you didn't know the metric, you wouldn't be able to calculate the KPI, so it is still useful.
- **What behaviors would the metric affect?**
- Can you map this metric to a specific action you would expect your team or the organization to take as a result?
Which "lever" could be pulled to influence this measurement?
- **What would improvements look like?**
- Can you define "good" and "bad" areas or directions for the measurement? Again, answering this question may give you a clue whether data is purely a measurement, or if it could be a KPI or part of an OKR.

Action: Go to section 4.1 in your worksheet and ask yourself the above questions. You don't need to write down a response for every metric, but for each metric in your list, consider the commentary above on each and whether that measure will be helpful to you as a metric or KPI. Note any comments or thoughts you have regarding these metrics in your worksheet. If you find that any metrics no longer seem to make sense in light of this review, or could be replaced with something better, modify your list. Once this is complete, move on to the next step in the exercise.

Metric Classification

Now that you have a sanity-checked list of metrics that seem to make sense for your organization, consider taking it a step further. Section 4.2 in your worksheet has additional methods for classifying your metrics. Consider this step optional for this exercise since walking through each in this much detail may take an exceedingly long time. Do, however, at least read through the below descriptions of additional dimensions to classify your metrics on to understand them more deeply and align your expectations to what each metric may be able to deliver. These are also good questions to consider for any metrics your team already reports, and evaluation of your current metrics against this list is another suggested activity for when you return to the office.

For your consideration, here are some additional factors to help you sort and prioritize your metrics:

- **Lead or Lag?**
 - Consider whether each of the metrics defined is a lead or lag indicator. In the oil change example, the oil level, and pressure is a **lead** indicator, while the engine temperature is a **lag** indicator (since the engine will overheat after the oil level or pressure becomes out of spec). Classifications such as the lead and lag status of each metric may help prioritize which is the most important to focus on within your list of options.
- **Performance or Effectiveness Measure**
 - Does this metric measure whether you are "Doing things right" or in other words, performing within specification on the tasks you have chosen to do? (i.e. changing your oil at the determined interval) Alternatively, does it measure whether you are "doing the right things", or in other words, does the metric measure that the activity you are engaging in is achieving the intended outcome? Using GQM, you are more likely to produce metrics of the former variety, but they should align with actions that affect the latter as well. The whole point of GQM is to ensure that since you have aligned metrics with your goals, what you measure should follow with movement towards the desired outcome. Note we did not list an effectiveness metric in the previous car oil example metrics. An example of one might be "instances of engine overheating in the last year", if this measure is 0, we are seemingly "doing the right things". A **complete metrics program requires the inclusion of both types of measures - measuring that what you're doing is *actually* working is a key part of success.**
- **Quantitative or Qualitative**
 - Does this metric address a counting, grouping, or statistic of some event, or is it measuring accuracy, completeness or other qualitative data?
- **Subjectivity**
 - How consistent will the data be. If two different people gather it, will they produce the same results? Qualitative measures tend to be more subjective than quantitative measures.
- **Frequency of Collection**
 - How quickly must you sample this measurement to get meaningful data?
- **Difficulty to Produce**

- How complex or time-consuming is it to produce this data? Is it worth it for the value it delivers, or could you get the same information more easily?
- Priority / Usefulness
- Is this an important and unique measure or just one of the multiple ways of getting to the same answer?

Once you have considered your metrics list in light of these additional factors, move on to the next step.

Conclusion

Congratulations, you have now finished the main portion of the metrics exercise, but there is one step we haven't yet touched on - Communication of metrics. The final portion is marked as an appendix as it is a purely informational step, but we *highly* recommend you read through it, if not now, later on when you have time. Step 3 covers important, in-depth information on how to collect and express the information that you have now determined you need to achieve your goals. If you finish this exercise in class with enough time left over, continue to the next section where we will discuss methods for collecting metrics as well as tools for storing and displaying data in the most effective way possible.

Exercise Conclusion – Key Takeaways

In this exercise, you have:

- Been introduced to the GQM(I)M system for goal-driven measurement
- Developed a set of goals, questions, and associated metrics for your own team
- Reviewed and classified your metrics on a set of additional useful dimensions
- Learned best practice for effective presentation and communication of metrics

Exercise 5.2 is now complete!

Appendix: Tools and Tips for Effective Metrics Presentation

While creating and tracking important measurements is a requirement for running an effective SOC, do not forget the second half of the equation - dissemination of those metrics to stakeholders. Metrics, KPIs, and OKRs and the improvements they drive are effectively invisible to those outside the group if you are not communicating your goals and efforts towards achieving them. Therefore, it is important to pay attention not just to which metrics to collect and how to collect them, but also the most effective way to communicate and display those metrics to others.

The two main concerns for effective communication of your SOC metrics are the actual visual presentation and accessibility of those metrics. In this section, we'll do a quick rundown of some of the methods and tools for metrics collection and display that can help in your efforts, as well as principles to keep in mind when rendering the visuals that will help you get the intended message across in the best way possible.

Tools for Metrics Collection and Presentation

To run an effective metrics collection process, you must be aware of the options for the collection of the data. Often the metrics we decide may be important are things that the team has never pursued collecting before, and a new method for interfacing with the application that generates the data needs to be found that can retrieve that data in an automated way. That collected data must then be sent to a system that can store and render it. Storage may be simply writing information to a file, or as complex as sending it to a database. The stored data must then be accessible and used with some sort of presentation layer that ideally supports taking the data and cutting and grouping it in various flexible ways into visualizations and dashboards. In this section, we'll cover some of the most popular options for each of these phases.

Generation and Collection of Metrics

APPLICATION PROGRAMMING INTERFACES (APIS)

APIs are the standard way to programmatically interface with an application from the outside, and one method of pulling metrics info from any application that offers an API. A ticketing system such as TheHive gives API access, for example, so that you can write a script or use a SOAR platform to automatically create a new alert, pull metrics, poll statistics, and more. If you need metrics that you suspect are stored by an application, look for API access to that information in the developer documentation, there is likely a way you can request it and pull it into one of the tools we will soon discuss below.

WEBHOOKS

Webhooks are another method of generating metrics best explained as "APIs in reverse". Tools such as TheHive and many other programs support [Webhooks](#) as well, which is effectively a way for the program that generates the events you want to track to *push* information to another system via standard format HTTP requests (instead of the "pulling" of information done via API). They are like APIs in reverse in that the designer of an application that supports webhooks is essentially saying "I'm willing to send this information out to you when something happens, you have to write a tool or API for your application to receive it."

Where supported, Webhooks provide a preferable and highly efficient way of monitoring everything happening within an application without the hassle of polling an API. If you'd like to track new case creation in TheHive for example, you could use the API to periodically ask for case creation information, and update in your tracking tool whenever it changes. The problem is most of the time it is polled there may be no difference to record. Using Webhooks, you can have TheHive outwardly send an HTTP request to you in real-time every time something of interest happens, a much more efficient and real-time method of tracking events.

AGENTS AND MORE

Where there is no built-in API for a system or application, many times there are SIEM vendor created, or other agents that can be installed on a system to monitor for events of interest, similar to a log Agent. In the Elastic ecosystem, for example, there is Metricbeat, Heartbeat.

There are also other tools like Fluentd, Graylog, Sysdig, Prometheus, and more that can all play a part in watching a system for events of interest.

Tools for Collection and Display of Collected Metrics

Here is a list of some tools for you to explore for data storage and visualization. Each of them has its own specialty and strengths and should be evaluated against the criteria and data sources you have for the best fit.

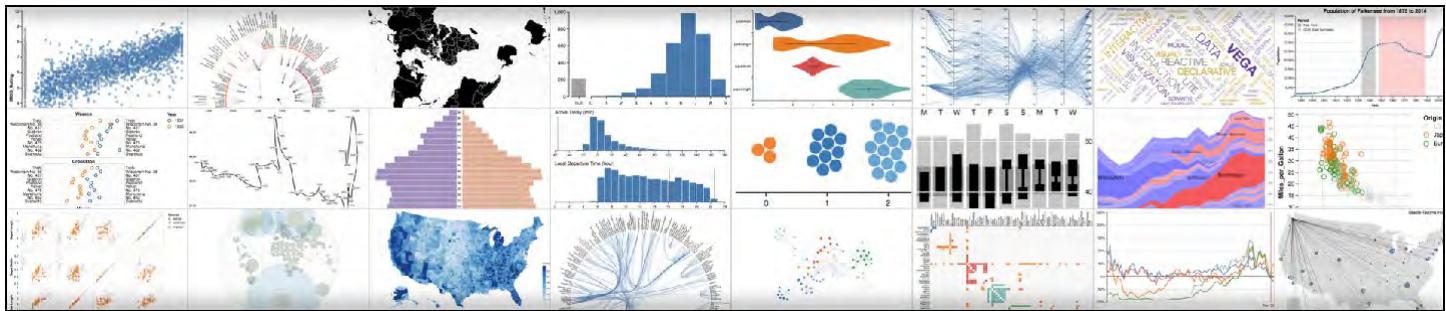
- Microsoft PowerBI
- Tableau
- Grafana
- Prometheus
- Graphite
- Kibana / Elasticsearch
- Vega

What You're Looking for In a Data Visualization Program

When choosing which program to best present your data with, here are three main considerations to keep in mind:

Data Source Support - Does the program support pulling in information from CSV, JSON, XML, Excel, Databases, SharePoint, SQL Servers, Cloud Storage, and other applications? Having a program that can directly poll the application you're trying to make metrics for can greatly ease the difficulty in metric generation. Programs such as [Tableau](#) and [Microsoft PowerBI](#) excel in this area, making it easy to directly source data from a variety of formats, locations, and applications.

Visualization Options - While your SIEM may support the collection of many of the data types you're interested in, it might not be the best application for rendering that data into charts. Many SIEMs have extremely limited chart and graph capabilities that barely extend beyond tables, pie, bar, and line graphs. For a team looking to do advanced metrics, this can be extremely limiting. If your SIEM is not delivering in this category, look to business intelligence tools or other supplementary options for graphics rendering. Some of the free options here include Kibana, Grafana, and Prometheus, all of which support a wealth of outstanding and visually appealing graph types. There are also supplementary data visualization libraries such as Vega (pictured below) that can be used, with some learning curve, to make incredible visuals. [Click here for an example of what Vega can do](#). Compare this with the list of graphics your SIEM can produce, do you think you could present a better story with these extra visuals? If so, spend some time figuring out how to make it happen! The takeaway here is that you should not limit yourself to only what your SIEM or Excel can provide for you, numerous free tools can help take your visualization game to the next level!



Accessibility - The final consideration for a data visualization program is the ability to have the program keep a live view of that data, accessible to all stakeholders using RBAC-style controls. Microsoft PowerBI, for example, can be used directly to make a dashboard accessible only to those with permission, which can give upper management view-only level access to SOC data on demand that is up to the minute accurate. Giving consumers and stakeholders of your data live, continuous access can go a long way in selling the value of the SOC and makes for great presentations and material for SOC tours as well!

References for Effective Communication of Visual Information

As a final consideration for metrics, understand that not all charts and graphs are created equal. For any metric you'd like to display there is likely a very objectively "right" and "wrong" way to do it. As security professionals, we are not often trained in the principles of effective display of visual information, but there's nothing worse than doing lots of amazing back end complicated work, only to sabotage it by using a chart type that undermines your message. Therefore, it is important as a manager to understand *how* to display any given metric or piece of information in the most effective way - the way that makes it clear to the reader what the message or state of that system is.

Principles for Effective Visuals and Dashboards

Data visualization is an art to master all on its own, but fortunately, we don't need to be full-on data viz. geeks to pick up enough of what we need to know to get our point across in a more effective way.

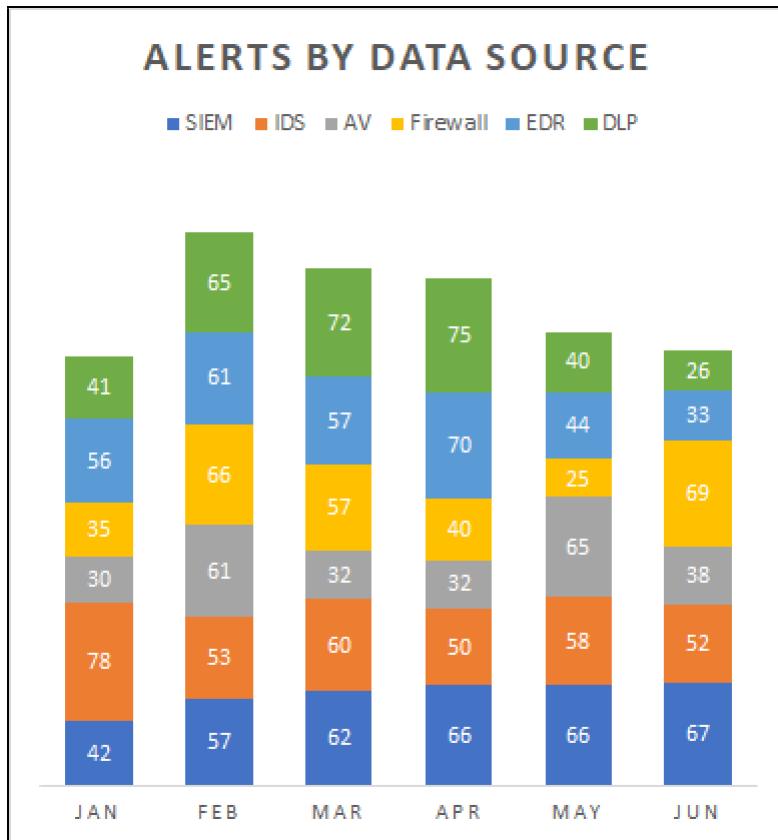
Note these principles are sourced from some of the best minds in visual communication design. [Edward Tufte](#) and his multitude of books, such as "[The Visual Display of Quantitative Information](#)" are considered to be the most influential and authoritative writings on the subject.

Some principles to remember for effective visualization from Tufte and others:

1. **Choosing the Right Graph Type** - Each type of chart has its strengths and weaknesses, and you should know, based on the message you are trying to convey, which is the best fit for your purpose. Before picking a graph type, consider what the reader is supposed to take from it. Are you comparing values? Are you explaining the composition of a whole? Are you trying to see a distribution of your data or find trends? Each type of graph has a specialty and picking the wrong one for the message you're trying to send can lead to sub-optimal results.

Here are some of the most common graph types and what they are best used for:

- **Vertical Bar** - One of the most common graphs, best used for comparing values of multiple items against each other or one variable over time.
- **Horizontal Bar** - Similar to vertical bar charts, but can be better if data labels are long, if there are many (10+) items to chart, or you need to display negative numbers, not a good choice for time-based variables.
- **Stacked Bar** - Shows the relationship between two different variables. Good for similarity and anomaly identification and seeing the distribution of data.

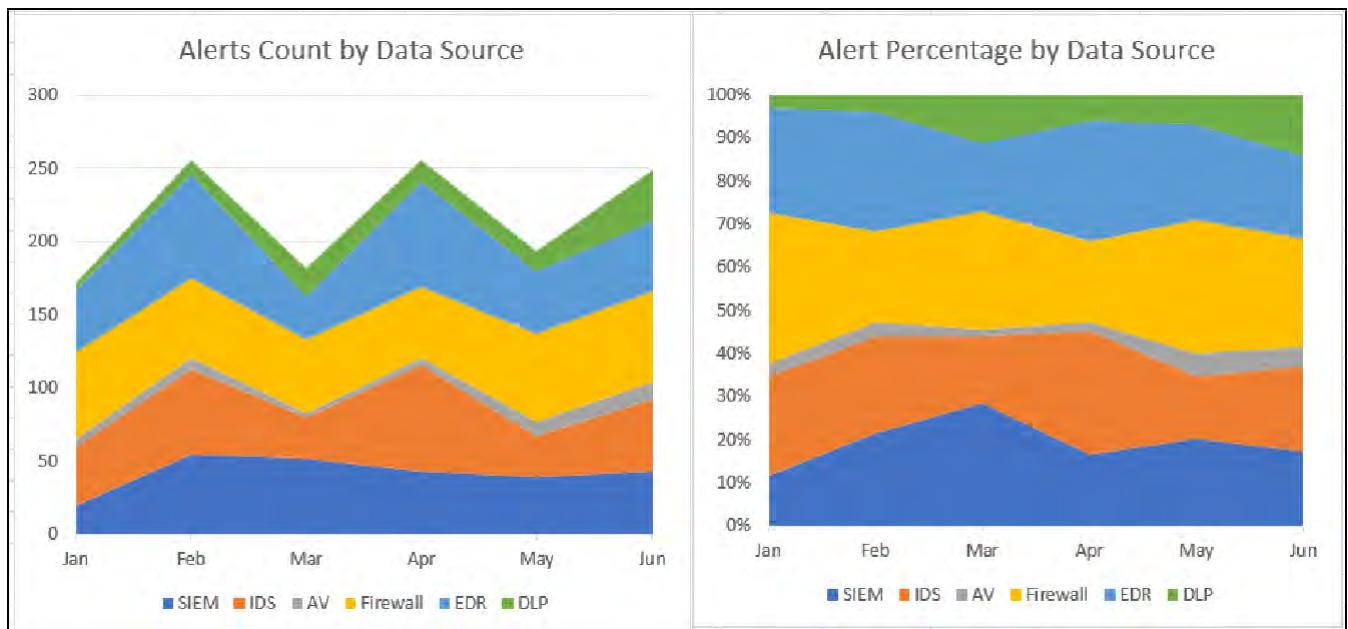


- **Line** - Good for showing progress over time, better than a bar chart for continuously sampled (instead of once per month, for example) datasets.
- **Pie** - Best for showing how the pieces of something make up the whole 100%. Be cautious with use - order slices according to size, do **not** use 3D pie charts, and make sure all pieces add to 100%.
- **Heat Map** - Best for showing the relationship between two variables and a 3rd axis of information about the specific combination, such as a rating or volume.

		Destination Subnet			
		Users	Int. Servers	DMZ	External
Source Subnet	Users	100	150	200	400
	Int. Servers	0	125	10	0
	DMZ	0	0	25	0
	External	0	0	300	0

Traffic in GB (July 2020)

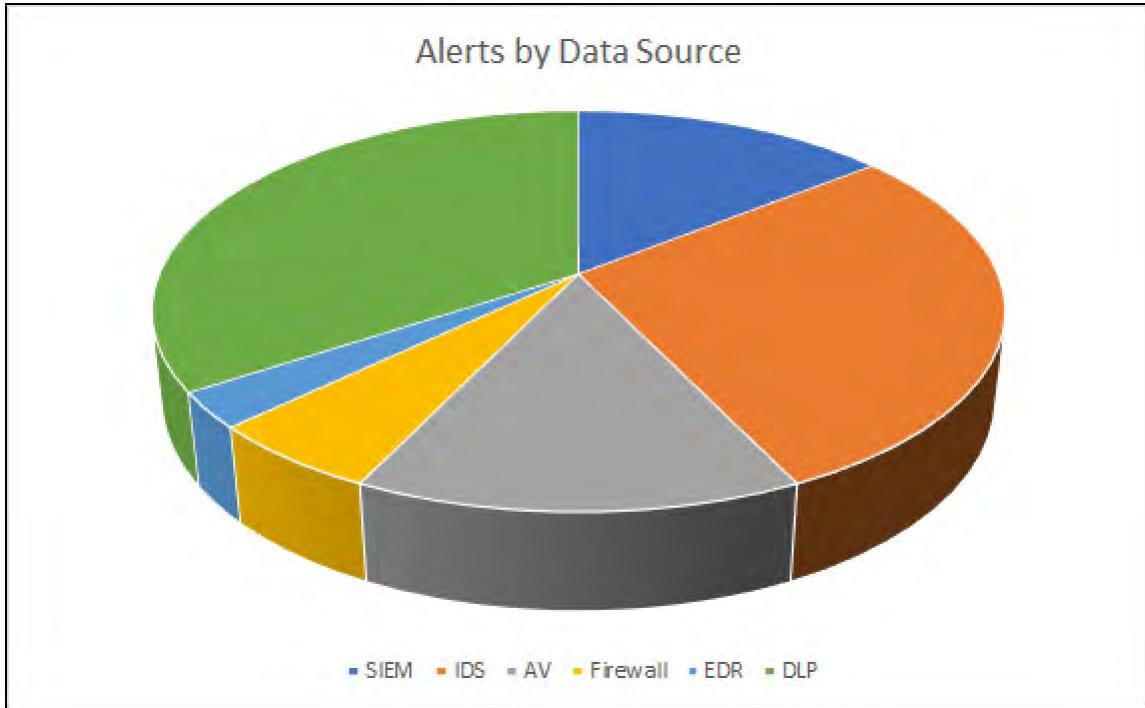
- **Scatter** - Best for plotting the relationship between two variables, highlights groups or outliers within the data and helps understand the distribution of the information.
- **Area** - Similar to the line chart, but better for emphasizing the part-to-whole relationship between a component of data and the total. These can either display the total count or the percentage each piece makes of the whole (the charts below depict the same source data).



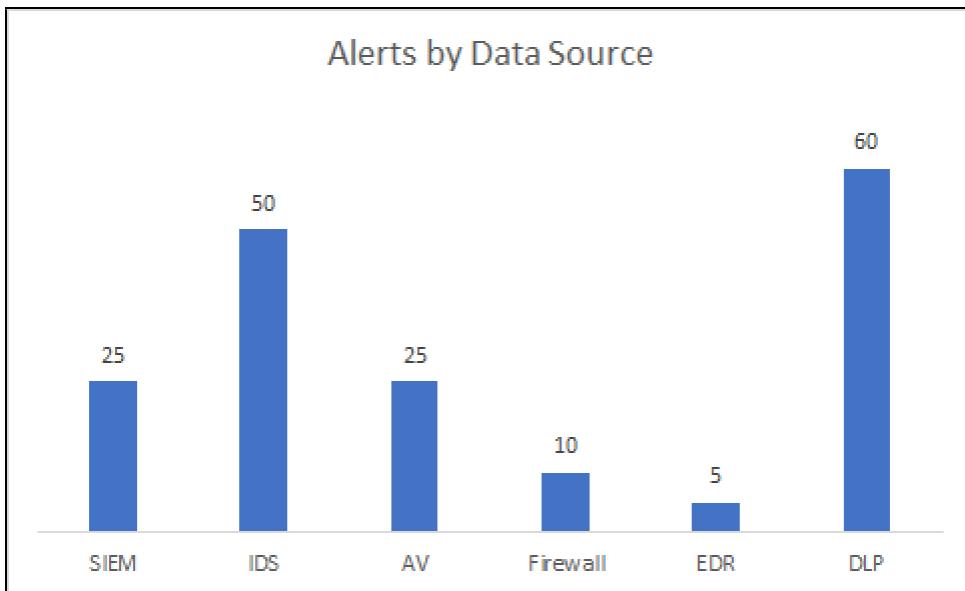
There is an amazing PDF chart guide [here](#) by Andrew Abela that summarizes this information as well as other chart types and what they can help you communicate.

2. **Labeling** - Use clear, detailed labels to point out explanations or important points in the data.
3. **Maximize data to ink ratio** - Charts should minimize any "ink" not directly depicting data - "chartjunk" as Tufte called it. This would be any unnecessary grid lines, 3D bars where they serve no function, etc.
4. **Data Density** - Maximize the real estate on the chart used to display data. If the chart can be smaller and still tell the same story, consider changing scales or shrinking the chart

5. **Visual Integrity** - The chosen display should not distort or misrepresent the data. [3D pie charts are guilty of this](#), for example, and as a result are often one of worst choices for effective visual communication. How do you feel about the pie chart below? Can you tell the number of alerts generated by the SIEM and AV is actually the same? It is, but the chart distorts it due to slice placement.



What if we represented the same data in a much different way?



Which chart lets you more easily compare the number of alerts generated per source? The answer is clear.

6. Spatial Arrangement - For dashboards, arrange charts such that the most important items are displayed in the order you normally read the text in. For English, the most important charts would be at the top left, moving right, then on to the next line.

As you can see metrics are not simply about collecting the data into your tool of choice. The design decisions you make can also have a big impact on your message and either help emphasize it while making your group look knowledgeable and professional, or undercut you and render your efforts useless.

If you'd like to dig deeper on principles of dashboarding and data visualization, here are some additional resources on the topic:

- "[Lessons from Edward Tufte](#)" - SlideShare Summary of these principles
- [Information Dashboard Design](#)
- [Storytelling with Data](#)
- [The Big Book of Dashboards: Visualizing Your Data Using Real-World Business Scenarios](#)
- [Encyclopedia of Slide Layouts: Inspiration for Visual Communication](#)

Exercise 5.3: Purple Team Assessment Planning, Execution, and Tracking

Background

In this exercise, you will be leveraging the data you've used throughout the rest of the course to plan a purple team assessment. It is crucial to not only test your signatures and analytics at the time of creation and with automated tools, but also against real (simulated) human attackers in order to verify detection and prevention mechanisms will function as expected under varying conditions. Planning and executing a purple team will go a long way towards these ends, and the software you will learn to use in this exercise - Vectr, will make planning, execution, and tracking of purple team tests fast and easy.

Objectives

- Learn how to plan a purple team assessment for your SOC
- Organize your purple team strategy into assessments and campaigns
- Use threat intelligence to guide testing and track results using Navigator
- Identify your most and least effective security tools
- Learn how to objectively demonstrate SOC improvement over time

Exercise Preparation

This exercise is completed in your MGT551 Linux VM, if you have trouble with the setup, see the troubleshooting information in the wiki, or reach out to an instructor or SANS support.

Launch the MGT551 Linux VM and log in.

```
- LOGIN = `student`  
- PASSWORD = `mgt551`
```

Before starting this exercise, you must start the required services. To do this, open a command terminal from the start bar.



Once the window is open, start the services by copying and pasting the following command at the command line:

```
cd /home/student/labs/5.3
docker-compose up -d
```

You should see output similar to the following, the list order of container startup may vary. If you receive an error message inform your instructor, or run the reset script from the "troubleshooting" page in the wiki and then run the previous command again.

```
Creating network "sandbox1_vectr_bridge" with the default driver
Creating sandbox1_redis_1 ... done
Creating sandbox1_mongo_1 ... done
Creating sandbox1_builder_1 ... done
Creating sandbox1_webserver_1 ... done
Creating sandbox1_tomcat_1 ... done
```

Keep the terminal open in the background, we will use it to shut these services down at the end of the lab.

Exercise Steps

Learn How to Use Vectr To Plan and Track Assessments

In this first step we'll introduce you to an outstanding piece of free, open-source software for planning and tracking purple team assessments named "[Vectr](#)" from [Security Risk Advisors](#) (SRA). Vectr comes preloaded with all of the configuration necessary and many of the tests most organizations would want to run for a wide variety of purple team testing scenarios and makes it easy to have both the red and blue team use it to perform the test and track results. Vectr is designed not only to facilitate a single purple team test but track the outcome of *all* purple team tests your organization does across all of time, showing performance metrics and (hopefully) improvement across each additional assessment. It's without a doubt one of the best, easiest, and most complete free solutions out there that we're aware of.

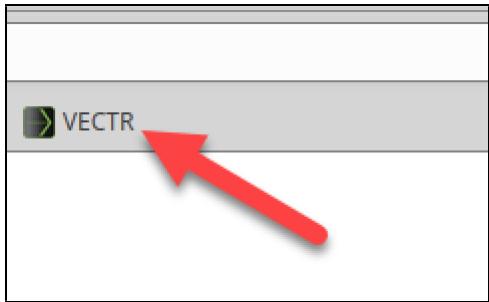
Vectr Introduction

To begin, we'll do a quick introduction to the Vectr software, which is loaded in a docker container in your virtual machine.

Note

Once you run the docker-compose command above it may take a minute or two for Vectr containers to be ready before the login page is functional. If you do not see the page in the following steps, wait a few moments before trying again. If the page never becomes available, run the troubleshooting script from the wiki and run the docker-compose command again, or inform your instructor.

To open Vectr, first, open a Firefox browser window by clicking on the icon in the top bar in your virtual machine, then click on the Vectr link in the bookmark bar, or just click [here](#) to open it in a new tab.



You will see a certificate warning page from Firefox. Click through it by clicking "Advanced" then "Accept the Risk and Continue":



Warning: Potential Security Risk Ahead

Firefox detected a potential security threat and did not continue to localhost. If you visit this site, attackers could try to steal information like your passwords, emails, or credit card details.

What can you do about it?

The issue is most likely with the website, and there is nothing you can do to resolve it.

If you are on a corporate network or using anti-virus software, you can read [1](#) to the support teams for assistance. You can also notify the website's administrator about the problem.

[Learn more...](#)

[Go Back \(Recommended\)](#)

[Advanced...](#)

[2](#)

[View Certificate](#)

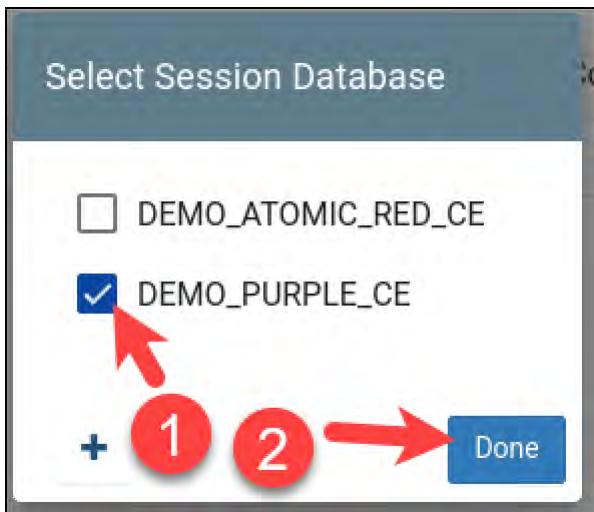
[Go Back \(Recommended\)](#)

[Accept the Risk and Continue](#)

You should now see the Vectr login screen. Use the credentials `student / mgt551` to login.



If you are asked to select a session database immediately upon login, select DEMO_PURPLE_CE and press done.



You should now see the logged-in Vectr application.

	Name	Create Date	Status
1	Enterprise Purple – 2017 Q1	01/02/2017	Completed
2	Enterprise Purple – 2017 Q3	08/02/2017	Completed
3	Enterprise Purple – 2018 Q1	01/02/2018	Completed
4	Enterprise Purple – 2018 Q3	08/02/2018	Completed
5	Enterprise Purple – 2019 Q1	01/02/2019	Completed

Session Databases

To begin, we must first learn some terminology in how Vectr is organized. Since Vectr is designed to segment and track purple team assessments for multiple groups over time, the first choice you must make when looking at Vectr is the SESSION DATABASE. There can be multiple session databases, and each one is meant to track all the purple team assessments for a single group/organization, or of a single type over time.

The currently selected session database is shown in the upper left of the top bar in the Vectr application as shown in the photo below (Vectr comes with staged demonstration data).

Name
Enterprise Purple – 2017 Q1

If needed, you select a new session database by clicking the database icon in the upper right of the application interface, then click "Select Session Database". For this exercise, the option we will be using is the example "DEMO_PURPLE_CE" session database, so there is no need to change it for this exercise. We will use the staged example data in this session database to further familiarize you with the organization of Vectr.

Assessments

You should now see the Vectr application with the "DEMO_PURPLE_CE" session database selected. Once a session database is selected Vectr will display all of the ASSESSMENTS that have been performed and are stored in that session database. In this case, the DEMO_PURPLE_CE database contains 5 simulated assessments that the example organization has performed between Q1 2017 and Q1 2019, each of which is listed with a creation date and status.

	Name	Create Date	Status
1	Enterprise Purple - 2017 Q1	01/02/2017	Completed
2	Enterprise Purple - 2017 Q3	08/02/2017	Completed
3	Enterprise Purple - 2018 Q1	01/02/2018	Completed
4	Enterprise Purple - 2018 Q3	08/02/2018	Completed
5	Enterprise Purple - 2019 Q1	01/02/2019	Completed

Each of these assessments is a fully contained set of items that were tested as part of that individual assessment. To see how assessments are organized, click on the "Enterprise Purple - 2019 Q1" assessment at the bottom of the screen.



Campaigns

On the following screen, you will be brought to a list of CAMPAIGNS that were run as part of that individual ASSESSMENT. Note that once you click, you can see your location within the session database and assessment on the top navigation bar as shown in the photo below. CAMPAIGNS are how ASSESSMENTS are organized. Each campaign is a customization set of TEST CASES that can be run all organized into a similar theme or objective. Before running any tests, all available unit tests are organized into campaigns as desired (there are a great set of defaults built into the application). Then, each new assessment is designed and run by selecting the campaigns you want to run as part of that assessment, depending on what you would like to include in the test. Each campaign has its own progress and outcome indicators as shown in the photo below.

Name	Progress	Outcome	Tags	Action
External Port Scans	100%	100%		⋮
External Web App Profiling	100%	67% 33%		⋮
External Password Attacks	100%	75% 25%		⋮
External Automated Scans	100%	67% 33%		⋮
Register Phishing Domains	100%	50% 50%		⋮
Email With Malicious Attachments	100%	73% 27%		⋮

As we can see, in the "Enterprise Purple - 2019 Q1" assessment, every campaign has been completed and the outcome of each TEST CASE that was part of that campaign is summarized in a colored bar next to it. Here is how to interpret them:

- Blue - Percent of unit tests that resulted in a **Blocked** result - the best result for unit tests since the attack failed.
- Green - Percent of unit tests that resulted in a **Detected** result - the 2nd best result, the attack wasn't stopped, but it was identified.
- Red - Percent of unit tests that resulted in a **Not Detected** result - the worst result, the attack was missed.

To see the individual makeup of a campaign, let's click one to enter it. Click on the "Email with Malicious Attachments" campaign to see the details.

Register Phishing Domains	100%	50% 50%	⋮
Email With Malicious Attachments	100%	73% 27%	⋮
Email with Malicious Links	100%	55% 45%	⋮
Malicious Document Execution	100%	42% 50% 8%	⋮

Test Cases

You should now see the campaign details page that shows all the TEST CASES in the lower half of the screen under the "Escalation Path" and "Timeline" windows. The "Escalation Path" window shows how test cases across multiple kill chain

stages can connect to form a full kill chain, in this case though, the campaign is purely focused on a single phase (Delivery), so this graph is not as useful for this type of campaign. The Timeline view shows all changes that were recorded as part of this campaign across time by both the red and blue team members(red and blue timeline dots).

Note that the navigation bar now shows the session database, assessment, and campaign you have drilled into:

Of most interest on this screen is the "Test Case" list on the lower half of the screen. These are the individual unit tests that were run in this assessment as part of this campaign. They are organized with a kill chain phase, technique, test case name, status, and outcome. The categorization items such as technique and phase are set ahead of time for each test case through the administrator interface, while the outcome and status are updated by team members as each test case is run in real-time. We are looking at a completed test and scrolling through the test cases will reveal which test cases had an outcome of "Blocked" vs. "Not Detected".

	Delivery	Phishing Payload	14 - Macro - Remote Template	Completed	Not Detected
	Delivery	Phishing Payload	15 - Archive - Macro	Completed	Blocked
	Delivery	Phishing Payload	16 - Archive - DDE	Completed	Blocked
	Delivery	Phishing Payload	17 - Macro - LuckyStrike PowerShell CellEmbed + Sandbox Evasion	Completed	Blocked
	Delivery	Phishing Payload	18 - Encrypted Archive - Macro	Completed	Not Detected
	Delivery	Phishing Payload	19 - Encrypted Archive - DDE	Completed	Not Detected
	Delivery	Phishing Payload	20 - Password-protected Office Doc - Standard CS payload	Completed	Not Detected

The idea of a single campaign like this is to take one assumed attacker goal or action - "Email with Malicious Attachments" in this case, and exhaustively test every possible option that the adversary might attempt while recording your ability to detect that specific attack implementation. While all of these test cases are done differently, each one is a unique implementation of what would fall under a single MITRE ATT&CK Technique, which is why testing and tracking detection at this granular a level is an *incredibly powerful and thorough* method of doing assessments and gives you much better confidence in the truth of your detection coverage map. (This is also one of the reasons MITRE is breaking techniques into individual sub-techniques). This campaign shows that during the 2019 Q1 assessment, some of these methods were blocked, some were Not Detected. This outcome set for this SOC should obviously result in a post-assessment effort to remedy all "Not Detected" technique implementations.

Test Case RecordEntry

There is one final level of depth to Vectr - the test case details editor. While performing the assessment, this is the screen both the red and blue team will be referencing to run and record the results of each test. To see it, click on an individual test case such as "1 - Macro - Cobalt Strike Standard".

Test Cases					
	Phase	Technique	Test Case	Status	Outcome
	All	search ...	search ...	All	All
☰	Delivery	Phishing Payload	1 - Macro - Cobalt Strike Standard	Completed	Blocked
☰	Delivery	Phishing Payload	2 - Macro - CobaltStrike Standard (MMG)	Completed	Blocked
☰	Delivery	Phishing Payload	3 - Macro - Cobalt Strike Standard as HREF (URL Rewrite Update)	Completed	Blocked

You should now see the Test Case editing screen, which displays the details of the test case split into a red team side (in red title bars) and a blue team side (in blue title bars). Let's explain each half.

The red team side, as shown below, has several important areas:

- A status bar that is meant to be clicked when the test starts and stopped when the attack test case has completed.
- Red Team Details - Which informs the red team what the test is about, the technique and phase settings, and operator guidance for how to perform the test.
- Source IPs, Attacker Tools, and Target Assets - If you'd like to track these individually for each test, details can be entered here.

Edit 1 - Macro - Cobalt Strike Standard Test Case

Status: Completed

Attack Start

01/04/2019
18:01:12
status changed to
InProgress

Attack Stop

01/04/2019
18:56:37
status changed to
Completed

Source IPs

Red Team Details

Name
1 - Macro - Cobalt Strike Standard

Description
Send phishing email to victim containing link with malicious document. This is intended to test the mail gateway (and sandbox analysis) in isolation.

Technique Phishing Payload Phase Delivery

Operator Guidance
Attacks → Packages → MS Office Macro
Copy code to office document

References

Attacker Tools
Cobalt Strike

Target Assets

On the right side of the screen, the blue team can enter details on numerous relevant items such as:

- Outcome
- Vendor/tools that detected the attack (which ultimately turns into metrics, as we will see later)
- Outcome notes

- Expected "defense layers" that should detect the attack (pre-set with a list created by Vectr app administrator)
- Detection and prevention methods
- Test case-related evidence files

Blue Team Details

Outcome
 TBD Blocked Detected NotDetected

Detecting Blue Tool(s):

FireEye Email Security

Was an alert triggered?
 Yes TBD No

Outcome Notes
outcomeNotes

Tags

Rules

Detection Time

01/04/2019 19:23:00
outcome changed to
Blocked

Expected Detection Layers

Email Gateway

Detection

- 1) Malicious email delivery alerted by email gateway (ideally blocked/quarantined too with attachment stripped and URLs rewritten)



Prevention

- 1) Malicious document blocked/quarantined by email gateway, or email delivered but filtered to junk folder. Malicious URLs are rewritten and behavior analyzed in sandbox consistent with attachments.

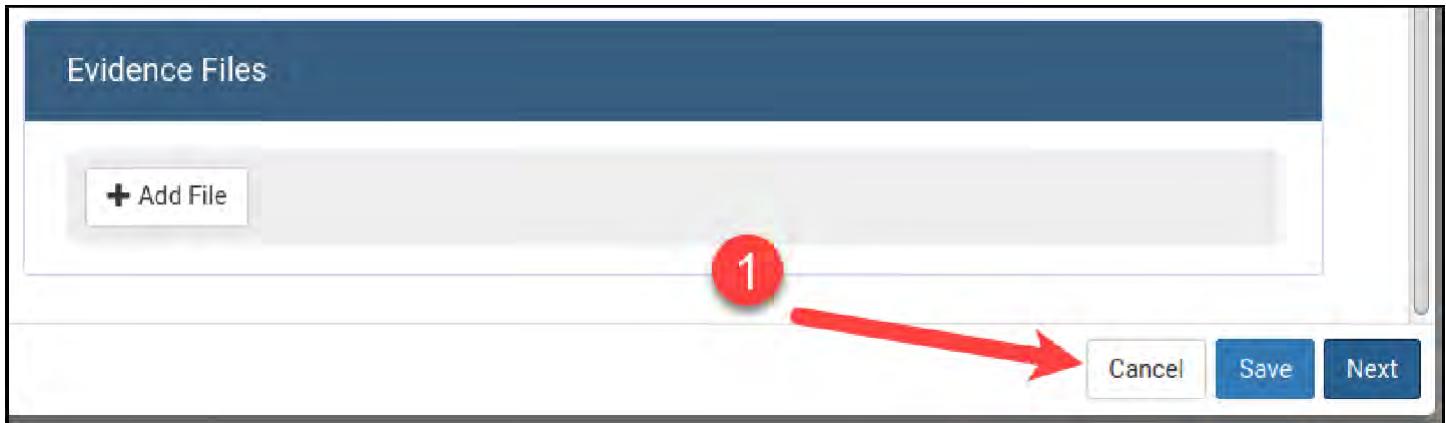


Evidence Files

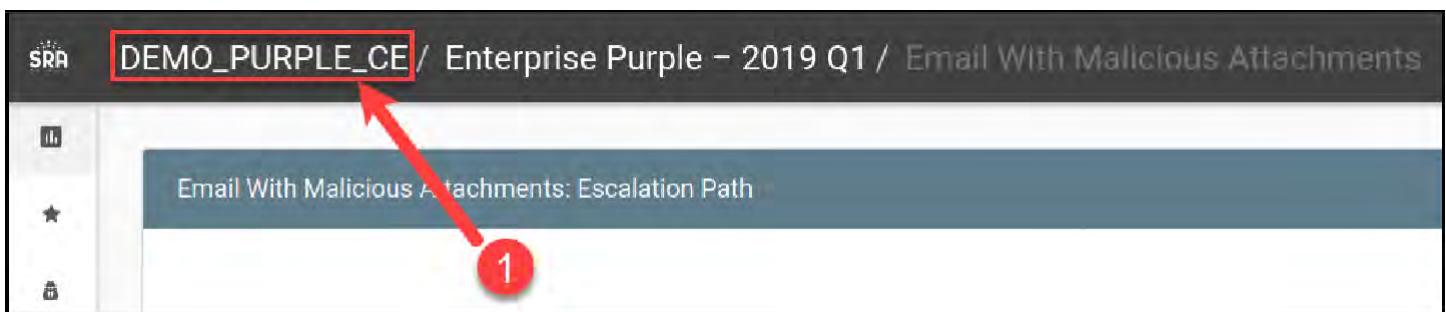
+ Add File

Any changes in either side of this screen will cause a record to be generated in the campaign timeline on the main campaign screen. As each unit test in the campaign is started, red team members should enter into the test case and press start. Once the test is run, they press stop, creating the timestamp that signifies when the event occurred. The blue team then has a chance to go look for any evidence generated as a result and notate the results as necessary. This is one of the fantastic things about a purple team assessment, not only does it test all of your analytics in a very thorough way, but blue team members will also be trained in how to spot "real" attacks on their real tools, inside their actual network. This double purpose means purple teams are one of the most valuable training exercises you can run! Hopefully, now the process of using Vectr is becoming clear and you are starting to see how it can unleash the potential in your SOC team!

Press cancel to exit out of the test case details screen.



This concludes the explanation of assessments, campaigns, test cases, and how to record the details of each. Click the name of the session database at the top of the screen to exit the campaign details screen and return to the assessment list for this session database.



Cross-Assessment Metrics

Now that you've seen the idea of how Vectr is laid out and the workflow for designing a single purple team assessment it's time to move in the other direction. Let's "zoom out" for a moment and show how Vectr also can summarize *all* assessments within a session database to show the results of multiple assessments across time. Assuming your SOC is following up on the results of each of your purple team assessments, the summary metrics capability is the function of Vectr that can help objectively show the improvement of your SOC over time - the most important part of running purple team assessments!

To see statistics for all assessments over time, from the Assessment list screen, click on the Reporting tab on the left side of the screen.

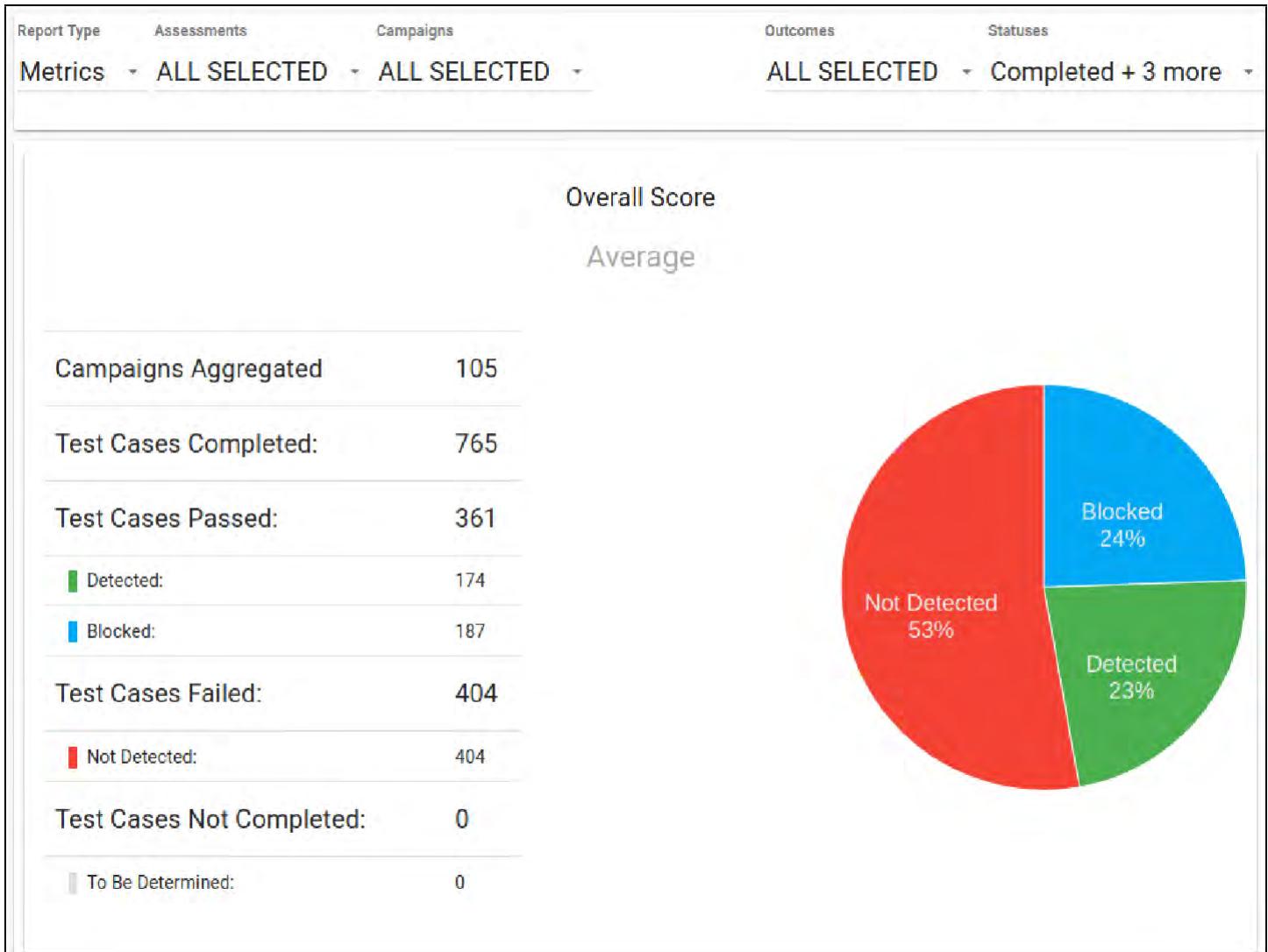
The screenshot shows the SRA Vector ECE application interface. At the top, there is a dark header bar with the SRA logo and the Vector ECE title. Below the header is a navigation sidebar with the following items:

- Assessments
- Reporting** (highlighted with a red arrow)
- Vendor & Tools
- Defensive Layers
- Target Assets
- Source IPs
- Administration

The main content area has a table with the following data:

Name
Enterprise Purple – 2017 Q1
Enterprise Purple – 2017 Q3
Enterprise Purple – 2018 Q1
Enterprise Purple – 2018 Q3
Enterprise Purple – 2019 Q1

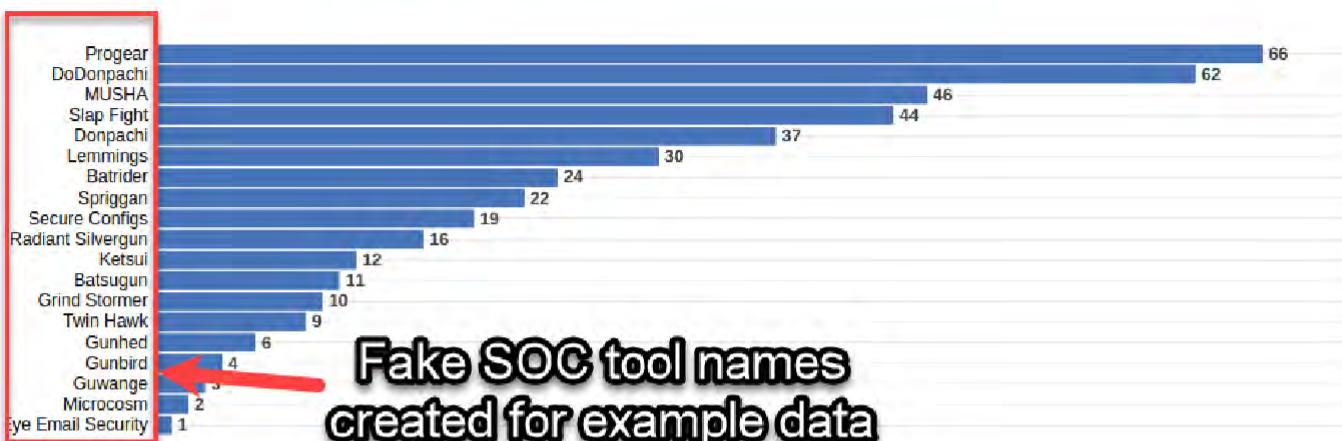
The reporting feature has multiple different pages that can show you the results of all your assessments across multiple dimensions. The default view you'll be brought to will show you the pie chart of not blocked, detected, and blocked test cases across all assessments and all campaigns.



Even more interesting on this page though, is the breakdown of detection and prevention by *tool* and the statistics by *kill chain* phase that is below. (Note that since this is example data, SRA made up pretend SOC tool names such as "Progear" and "DoDonpachi" that you'll see in these reports. In a normal situation, this is where your SIEM, IDS, host tools, and other items product names would go.)

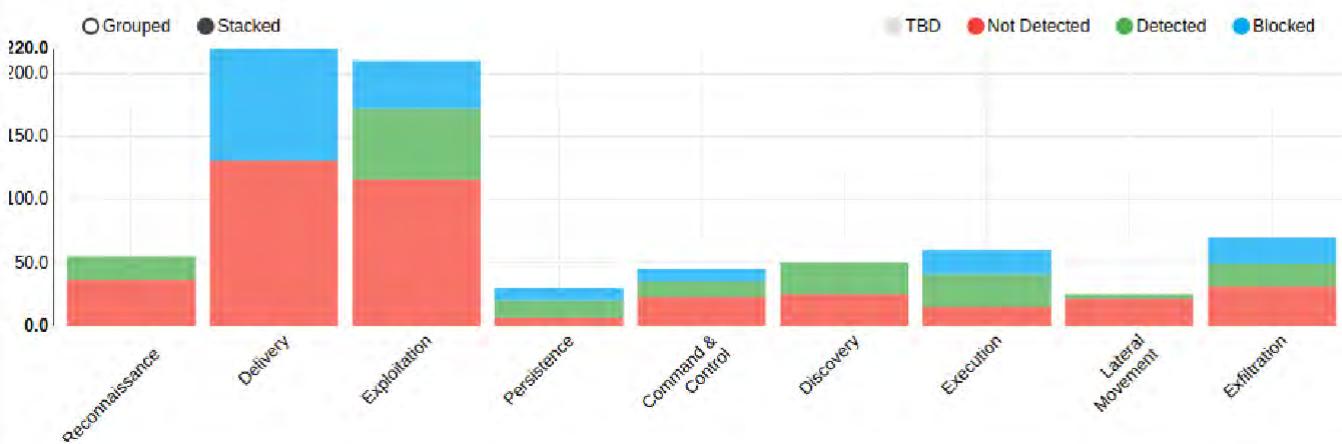
Statistics by Detection/Prevention Tool

Blocked and detected test cases for detection/prevention tools employed



Statistics by Kill Chain Phase

Test case detection status distribution with respect to attack lifecycle phases



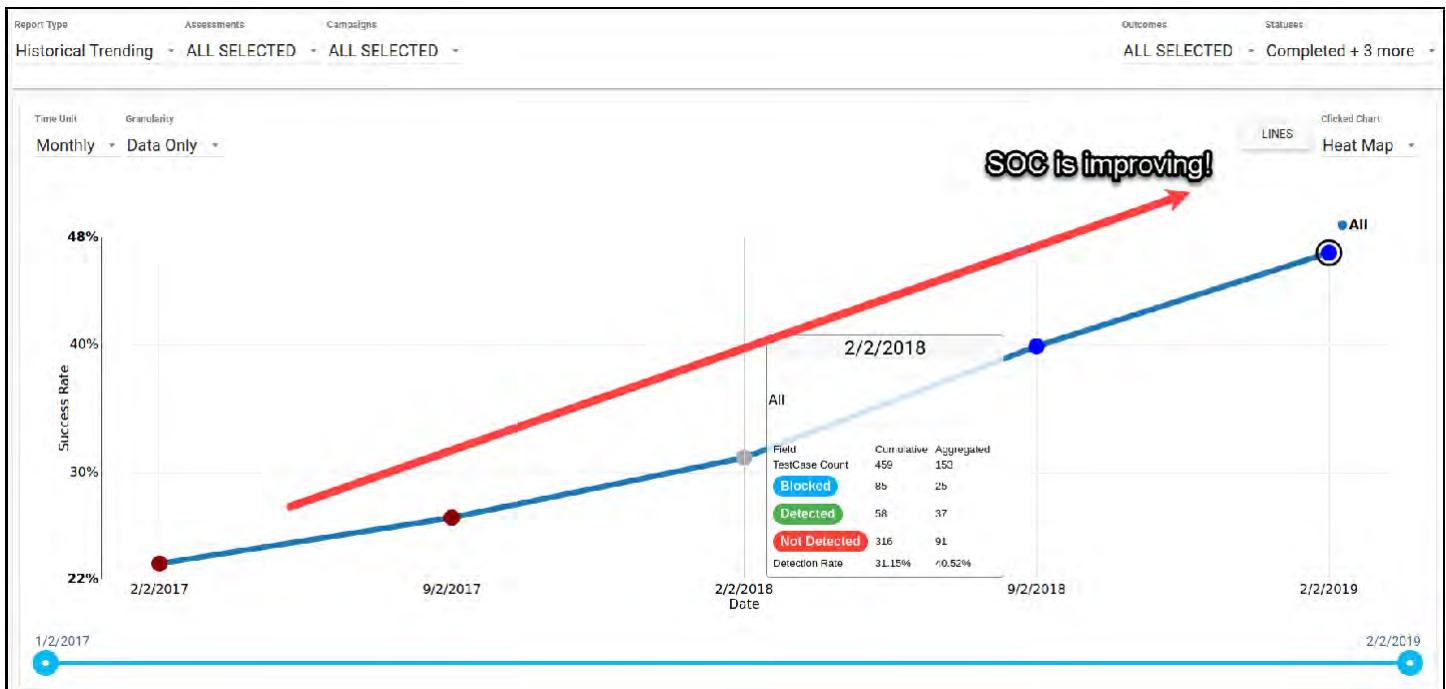
By analyzing these two charts, you should start to get an immediate sense of which tools are doing the most work detecting the important test cases, as well as how your defense stacks up across the kill chain. If the way you've done multiple assessments focusing on a single phase is warping the charts you can use the drop-downs at the top of the screen to only show reports for certain assessments or campaigns.

Report Type	Assessments	Campaigns	Outcomes	Statuses
Metrics	ALL SELECTED	ALL SELECTED	ALL SELECTED	Completed + 3 more

Let's check out some of the other report options. Click on the "Report Type" drop-down and select "Historical Trending".

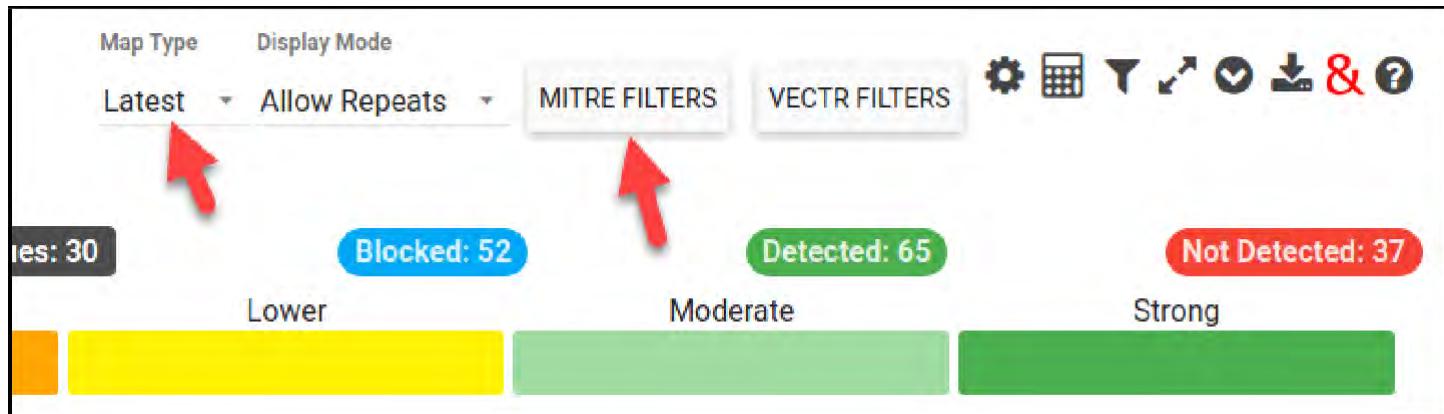
The screenshot shows the ATT&CK Navigator interface with the 'Metrics' report type selected. The top navigation bar includes 'Assessments' and 'Campaigns' tabs, and dropdown menus for 'CTED' and 'ALL SELECTED'. Below the navigation, there's a list of options: 'Test Case Drilldown', 'Historical Trending...', 'Heat Map', and 'Toolset Drilldown'. A red arrow points to the 'Historical Trending...' option. At the bottom of the list, it says 'Campaigns Aggregated'.

This here may be the most important chart of all. This graph shows, across all dated assessments, the total success rate for the blue team detecting and blocking test cases. An upward trajectory here is your evidence that the blue team is indeed becoming better at defending across time!

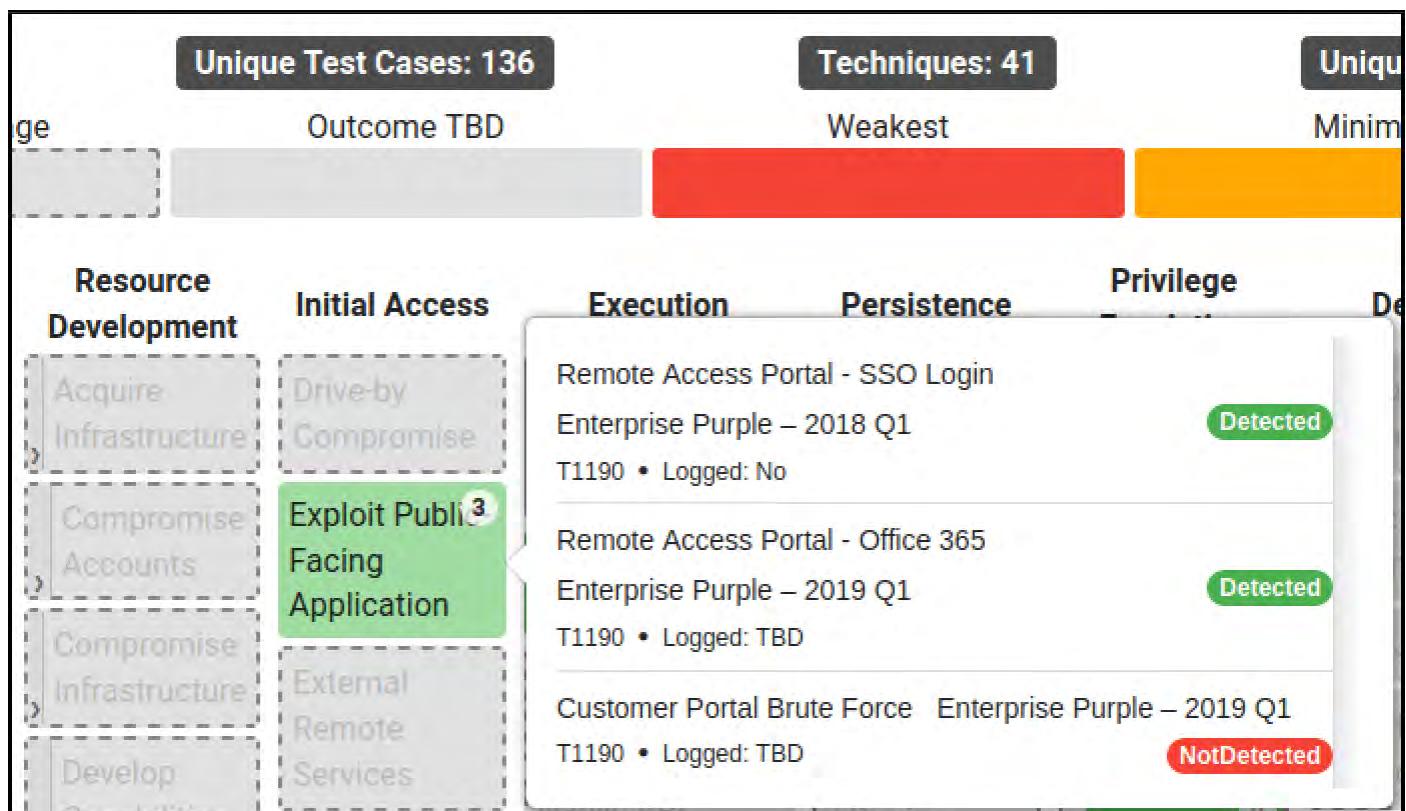


It gets even better - if you scroll beneath the historical trend line you'll be given an ATT&CK Navigator style printout of each ATT&CK Technique and the results of each test of that technique from either the most recent assessment, or from all

assessments in the session database! You can even apply the same filters we used in the Navigator exercise to show layers by Threat Groups, Software, Data Sources, Platforms, and more, as well as export what you see to a new ATT&CK Navigator layer! Feel free to click around on these buttons and experiment with the options.



Back to the heat map - here's an example of the data shown when mousing over the "Exploit Public Facing Application" technique. It shows the most recent time per test case that an evaluation was done that falls under this ATT&CK technique, and what the results were. In the photo, we see two items from 2019 Q1, one that was detected and one that was not, and also a 3rd test case (Remote Access Port - SSO Login) that was last tested in the 2018 Q1 assessment, and that it was detected.



Using this view combined with what we learned about Navigator in the previous exercise ,you can now match up threat groups with *actual* tests of multiple methods of attack that fall under each technique, a truly powerful way of aligning your threat intelligence with defense assessment data!

Before moving into the next step, where we'll learn to create our own campaigns, explore some of the other "Report Type" tabs and check out the wealth of data Vectr can produce on your purple team assessments. Other recommended reports you should investigate are **Toolset Summary**, **Scorecard**, and **Kill Chain Summary**.

Once you are done exploring report types, move on to the next step.

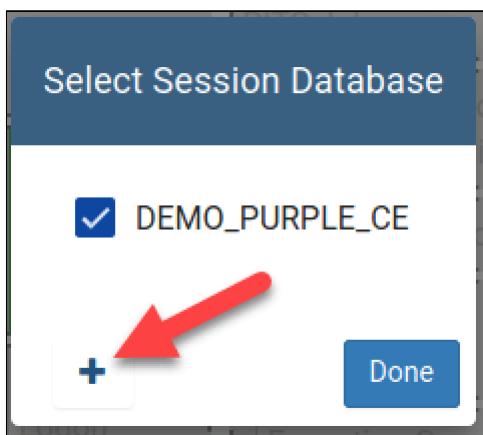
Design Your Own Purple Team Test

Now that you've had the overview of Vectr and how it works, it's time to learn how to set up our own purple team assessments that we can use for our SOC.

The first step in setting up a clean environment is creating a new session database. As a first step, click on the database icon in the upper right of the application and click "Select Session Database".



Click the plus the create a new session database



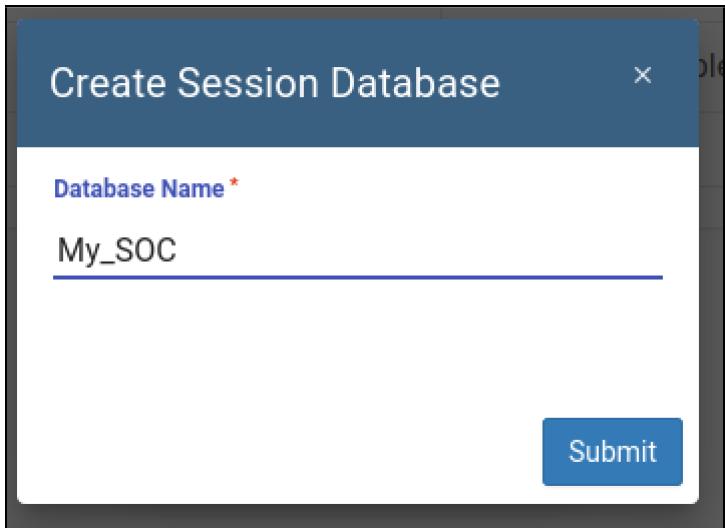
Enter a name for it, we can use "My_SOC" for now, click Submit to create the database.

Create Session Database

Database Name *

My_SOC

Submit



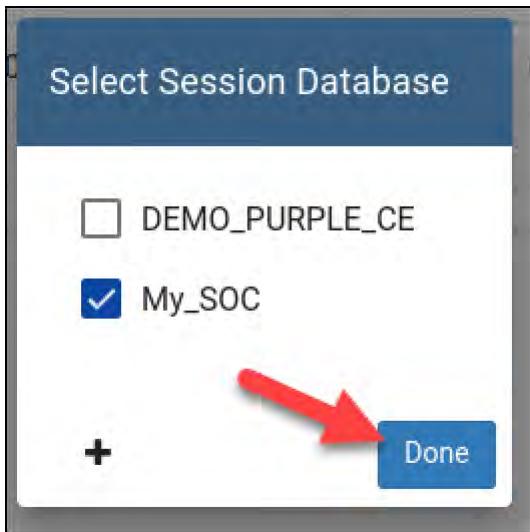
Then click Done to select it.

Select Session Database

DEMO_PURPLE_CE

My_SOC

+ Done

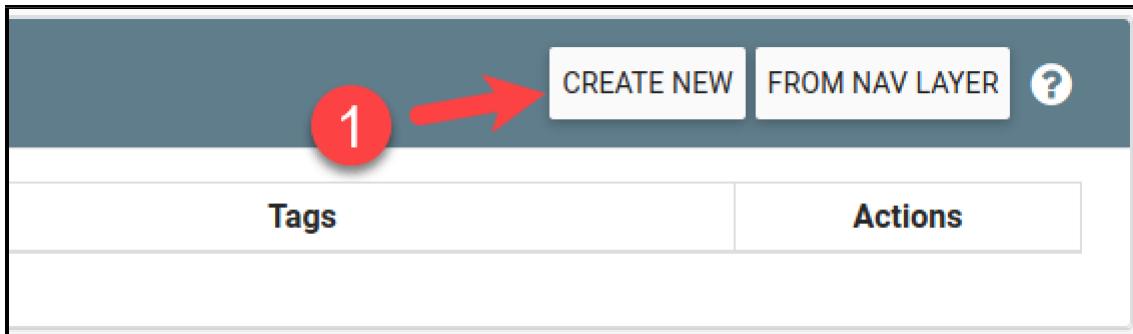


You should now be looking at a fresh session database with no assessments currently in the list.

Assessments			
	Name	Create Date	Status

Create a Custom Focused Test

We'll start by making a very simple single-purpose assessment. Let's say we'd like to run a purple team to test our phishing defenses for example. To do this, click the "Create New" button in the upper right corner of the screen.



On the following screen, fill out the name and description:

- **Name:** Phishing Assessment
- **Description:** A test of our phishing defenses
- **From template:** (Leave blank)
- **Kill Chain:** Default

Below the metadata, for the assessment, you will see a list of all pre-created campaigns that can be added to the assessment campaign list. Since this test is meant to test phishing attacks, we'll look for campaigns related to email attacks by searching the word "email" in the search box as shown below. Type email in the search box then select the three campaigns as shown in the photo below.

- Email Spoofing
- Email with Malicious Attachments
- Email with Malicious Links

New Assessment

Name:	Phishing Assessment 1			
Description:	A test of our phishing defenses 2			
From Template:				
Kill Chain:	Default			
Select	Organization	Campaign	Description	# TestCases
<input type="checkbox"/>	All	email 3		
<input type="checkbox"/>	SPA 4	Email Spoofing	The objective of this campaign is to test mail gateway controls regarding emails with spoofed characteristics	3
<input type="checkbox"/>	SRA	Data Exfil Methods - Email	Includes test cases for data exfiltration attempts using email and popular websites/cloud storage services	5
<input type="checkbox"/>	SRA 5	Email With Malicious Attachments	Includes a variety of malicious attachments containing different backdoor payloads. The objective of this campaign is to assess the mail gateway's success in blocking malicious attachments.	14
<input type="checkbox"/>	SRA	Register Phishing Domains	Activities include registration of phishing domains in preparation of email-based attacks and C2 testing, which tests possible brand monitoring/DNS reputation services and more importantly the ability to block domains both inbound and outbound based on reputation	2
<input type="checkbox"/>	SRA 6	Email with Malicious Links	This campaign is similar to the malicious attachment delivery, but delivers the same payloads as direct-download links rather than attachments. The purpose of this campaign is to assess URL rewriting and sandbox analysis.	17 7

Cancel Save

Click Save when done to save the campaigns to the new assessment.

Note

Instead of selecting individual tests on this page, you could use a pre-made Template. If we hadn't left the template box blank and instead selected one of the default Vectr templates a series of standardized campaigns would be added to the assessment for us. SRA includes several options that are great default campaign lists for full-scope purple team tests.

You should now see a session database screen with our single "Phishing Assessment" listed. Click on it to enter the assessment as if we were ready to start working on it.

Assessments			
	Name	Create Date	Status
	Phishing Assessment	03/31/2021	Not Performed

You should now see the three campaigns you selected in the Assessment creation screen, each with 0% progress, which makes sense since this is a brand-new assessment.

Campaign Dashboard						ASSESSMENT ACTIONS
	Name	Progress	Outcome	Tags	Action	
1	Email Spoofing	0%	0%			
2	Email With Malicious Attachments	0%	0%			
3	Email with Malicious Links	0%	0%			

Click on the "Email Spoofing" campaign to see the associated test cases.

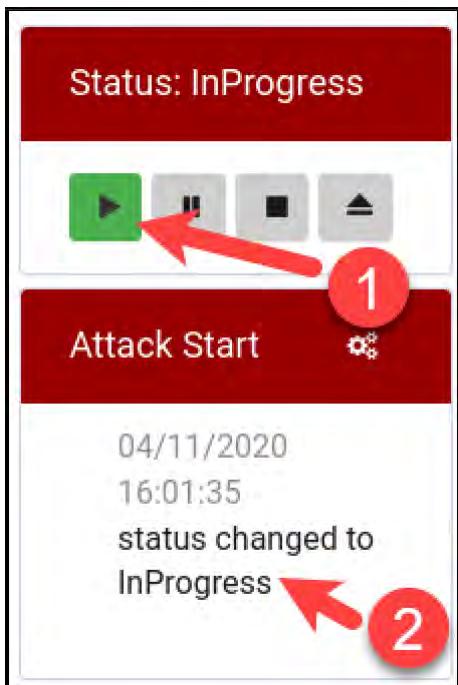
Campaign Dashboard	
	Name
1	Email Spoofing
2	Email With Malicious Attachments
3	Email with Malicious Links

Once inside the email spoofing campaign, you can see there are 3 test cases that make it up. Let's pretend we're ready to start this test and run the first use case - click the line of the test case for Spoof Sender Address.

Test Cases

Phase	Technique	Test Case	Status	Outcome	Tags
All	search ...	search ...	All	All	All
Initial Access	Spearphishing Link	Spoof Sender Address	NotPerformed	TBD	
Initial Access	Trusted Relationship	Open Relay	NotPerformed	TBD	
Initial Access	Trusted Relationship	Office 365 Tenant SMTP	NotPerformed	TBD	

In the Edit Test Case screen, designate the start of the test by pressing the Play button as shown below, after you click it, you should see the Attack Start box filled with the current time.



Let's switch perspectives to the blue team. Pretend the spoofed email was sent, and it was blocked and also silently dropped without an. We would mark this by placing a check in the "Detected box" on the right half of the screen, and a "No" in the alert triggered box since the alert wasn't triggered (since the email was silently dropped.)

Blue Team Details

Outcome

TBD Blocked Detected NotDetected

Detecting Blue Tool(s):

Was an alert triggered?

Yes TBD No

We can also select the gear icon to mark the tool that detected and blocked the attack for us. This is where the metrics for which tool is most useful will be sourced, so you should fill this in. Click the icon then type "exchange" in the box and check the box for "Exchange Online Protection (EOP)", we'll pretend this was the tool that detected and stopped the spoofed email. Press Close.

Select blue Tools

exchange

Include	Tool	Vendor
<input checked="" type="checkbox"/>	Exchange Online Protection (EOP)	Microsoft

You should now see a screen that looks like below. The blue team could also note additional info about the outcome or upload an evidence file if desired, we'll skip that for this demonstration.

Blue Team Details 

Outcome

TBD Blocked Detected NotDetected

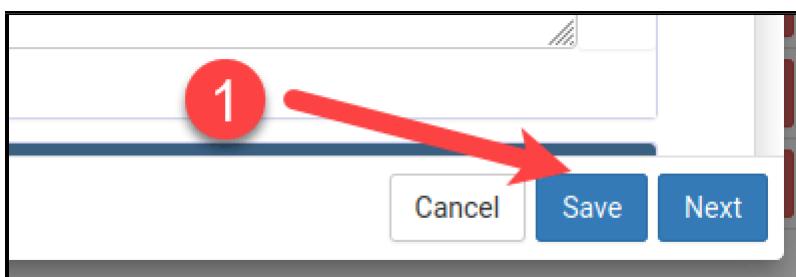
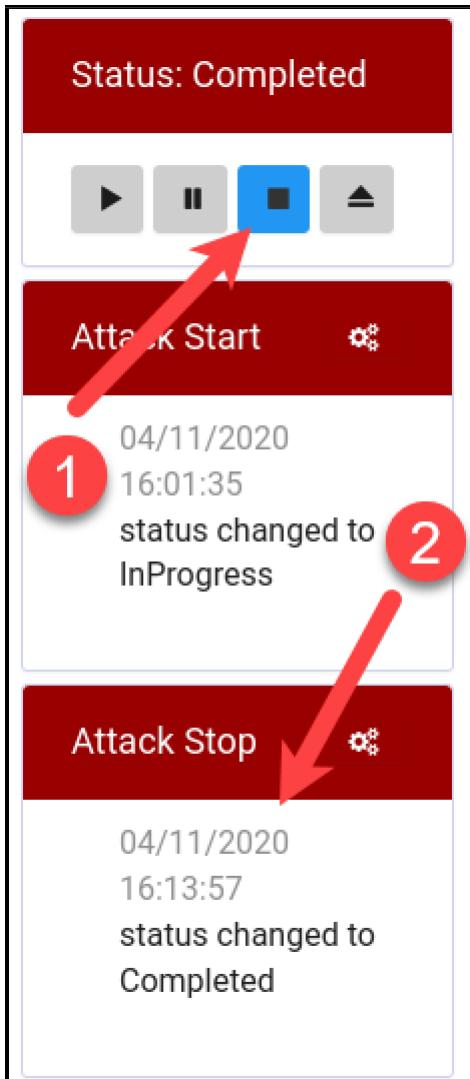
Detecting Blue Tool(s): 

Exchange Online Protection (EOP)

Was an alert triggered?

Yes TBD No

Click the Stop button on the left half the test case info to mark the test case as completed then press "Save" in the bottom right corner.

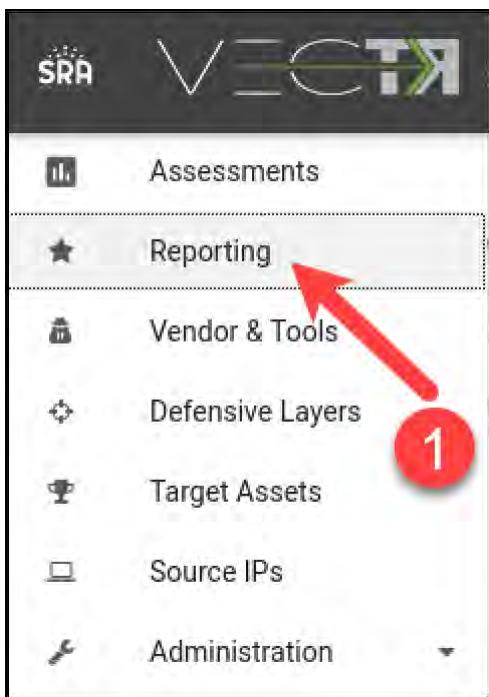


You'll notice the timeline in the upper right has all of the events we just performed recorded and the test case line will now switch to "Complete" with an outcome of "Blocked" as we had stated.

Phase	Technique	Test Case	Status	Outcome
All	search ...	search ...	All	All
Initial Access	Spearphishing Link	Spoof Sender Address	Completed	Blocked
Initial Access	Trusted Relationship	Open Relay	NotPerformed	TBD
Initial Access	Trusted Relationship	Office 365 Tenant SMTP	NotPerformed	TBD

To run this test, the red and blue team would keep working through tests in this and all other campaigns in the assessment until all were complete and the outcomes recorded. Once done, you can view all of the outcomes in the reports section.

Click on "Reports" on the left sidebar to navigate back to the screen that summarizes your assessment performance.



Since most of the test cases are not complete in our assessment and throughout our new session database there is minimal information on this screen so far, but you can see the impact of the single test case we just completed. Scroll down and you can see Vectr is already building the visualization showing that EOP has blocked one test case and that it was an "Initial Access" stage test case.

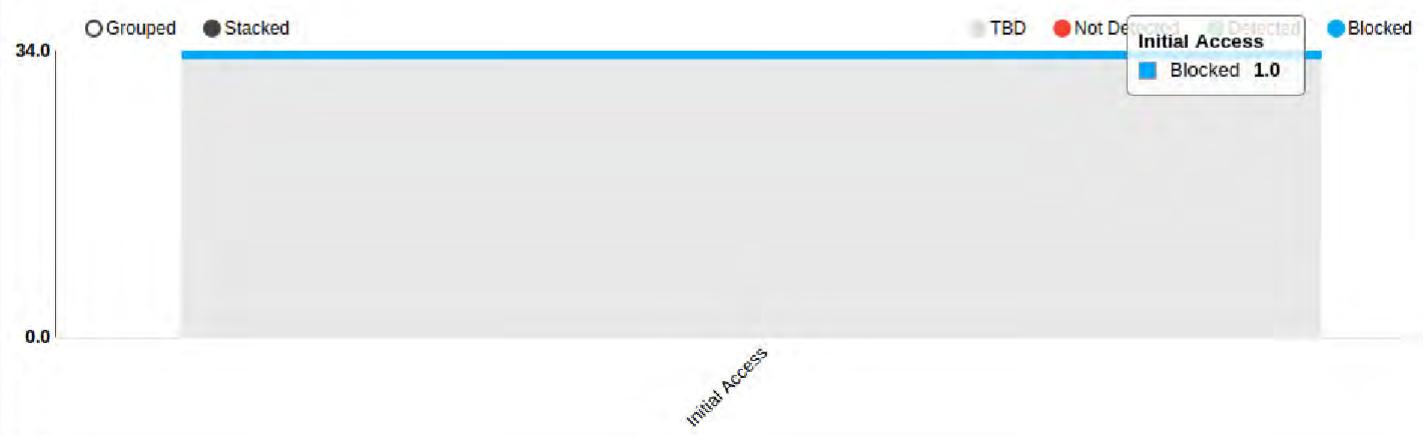
Statistics by Detection/Prevention Tool

Blocked and detected test cases for detection/prevention tools employed



Statistics by Kill Chain Phase

Test case detection status distribution with respect to attack lifecycle phases



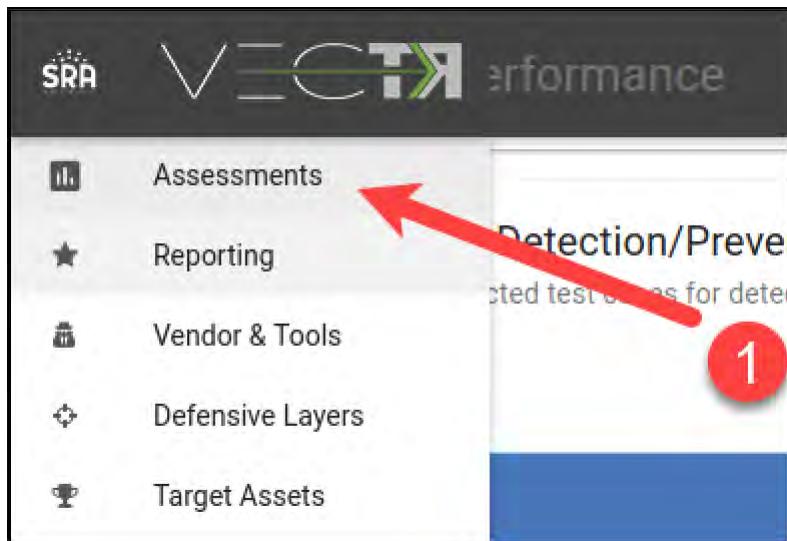
Whether this was a focused test, or one of the more full-scope assessments with multiple campaigns across all kill chain stages, by continuing through the rest of the test cases, your team will end with the assessment with some incredibly useful information:

- The different types of methods an attacker may attempt for each campaign type
- Whether each method will be missed, detected, or blocked
- How to spot those attacks on your actual tools
- Which tools were most effective in those detections
- A clear plan of action for where improvements must be made

This step showed you how to create either a fully custom test made of hand-selected campaigns or an assessment template. To wrap this exercise up, in the last step we'll step through the creation of one final crucial type of test configuration.

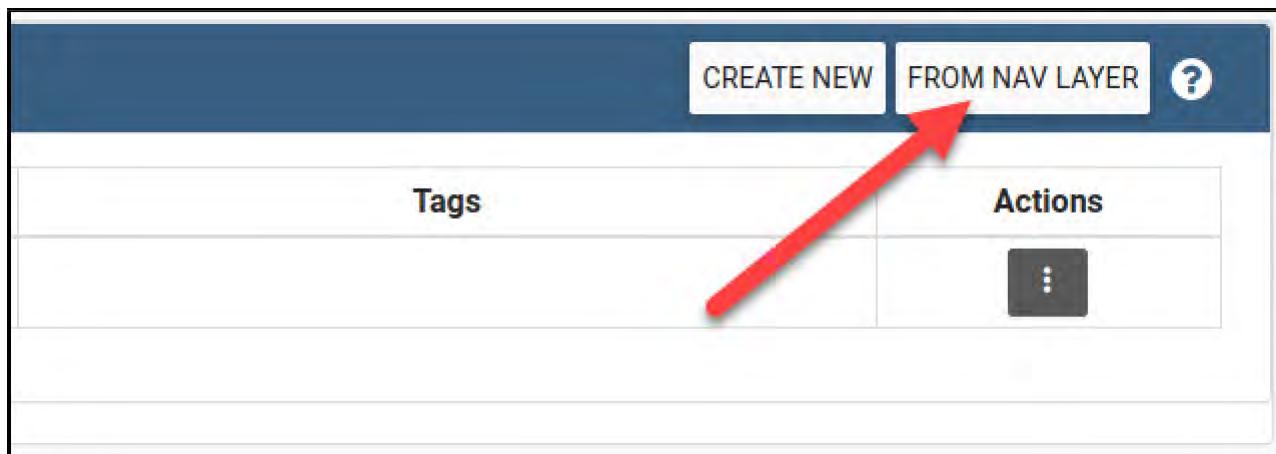
Create A Custom Full Assessment from a Template

Click back on the Assessments area on the left side of the web application to go back to your assessment list.



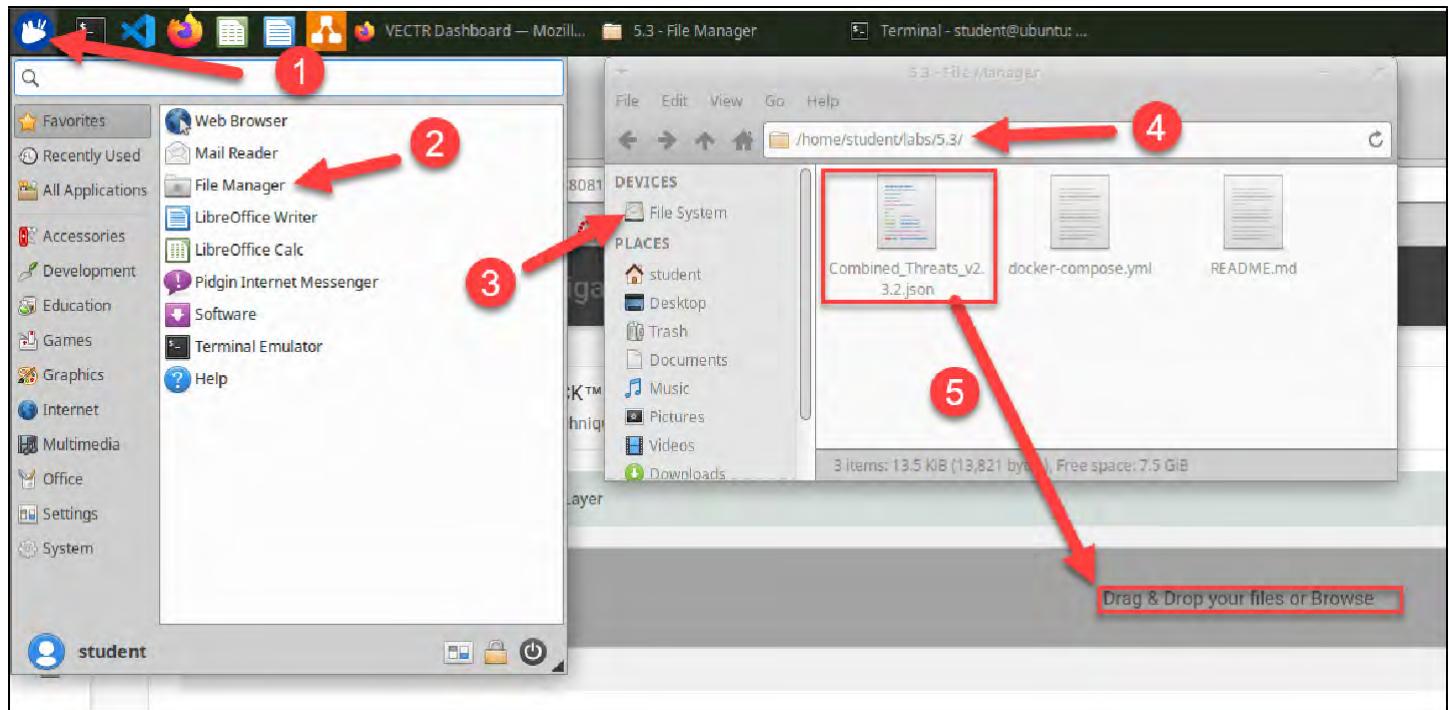
We're going to make one more type of new assessment, but instead of using the builder like we did last time, this time we're going to use a unique source. Remember back in the navigator exercise where we exported the "Combined_Threats.json" layer, built of all the known interesting attack groups techniques? Vectr can take an ATT&CK Navigator layer and use it as a way to build a list of tests as well!

To get this started, click "From Nav Layer" on the assessments page.



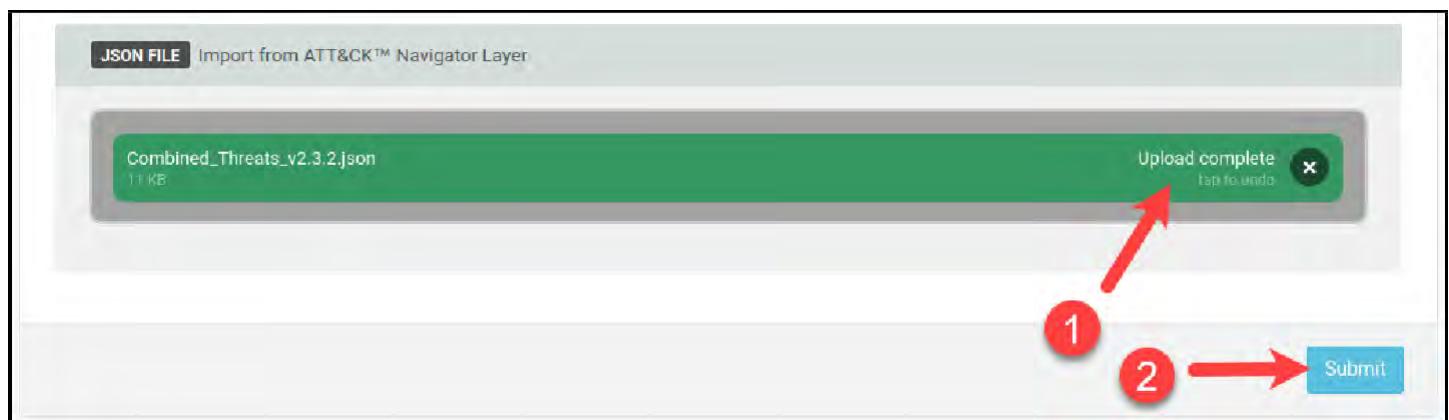
You should now see the "Create New Assessment" screen with a gray bar to select a Navigator Layer.

Open up the File Manager Application in the virtual machine and navigate to the /home/student/labs/5.3/ folder then select the file named `Combined_Threats_v2.3.2.json` and click and drag it over the gray bar (you may need to hover for a moment to make sure it recognizes it correctly).



Once the file is dragged over you will see a progress bar as it imports, and should be left with a green bar that says "Upload Complete" as shown below. If it did not work correctly, try dragging the file over the gray bar again (sometimes Firefox gets confused during the drag and drop).

Press the Submit button to continue.



You will then be asked a question about how you'd like to import the file. You can choose either option here but Vectr will structure the assessment in two different ways depending on which you choose.

Import Mode

How do you want the campaign(s) generated?

TACTICS AS CAMPAIGNS END TO END

- **Tactics as Campaigns** - In this mode, Vectr will look at all the highlighted items in each column (tactic) of the ATT&CK Matrix and create a campaign for each tactic, with test cases for each item that aligns with that tactic. In most cases, this will result in an assessment with multiple campaigns grouped by attack tactic. Each campaign will contain many test cases for all techniques that fall under that tactic that had a score.
- **End to End** - In this mode, a single campaign will be generated, but that campaign will have items across all MITRE ATT&CK tactics that were scored in the uploaded layer. In other words, this will make one large campaign that tests all tactics, whereas the "Tactics as Campaigns" import option will result in multiple campaigns with fewer test cases each.

Select "Tactics as Campaigns" for this example. You will then see a screen with additional options on exactly what you'd like to import as shown below. To see the details for what will be created click the unfold arrow on the right side of the screen. This shows that you will be creating 104 Test Cases across 8 different campaigns for this assessment. Place a check mark in the top checkbox to accept all of these test cases then press "Submit" to finalize the new assessment creation.

The screenshot shows a user interface for selecting campaigns. At the top, there are buttons for 'SELECT ALL', 'DESELECT ALL', and 'Submit'. Below these are status indicators: '0 Assessment Group Templates', '8 Campaigns', '104 Total Test Cases Selected', and a 'Submit' button. A red arrow labeled '1' points to the 'List of All Campaigns' header, which includes a checked checkbox and the text 'Total Campaigns: 8 Total Test Cases: 104'. Another red arrow labeled '2' points to the page navigation buttons 'First', 'Previous', '1', 'Next', and 'Last'. A third red arrow labeled '3' points to the 'Submit' button.

Campaign	Total Test Cases
Campaign: Collection	5 Total Test Cases
Campaign: Command & Control	7 Total Test Cases
Campaign: Credential Access	21 Total Test Cases
Campaign: Defense Evasion	23 Total Test Cases
Campaign: Discovery	21 Total Test Cases
Campaign: Execution	17 Total Test Cases
Campaign: Impact	4 Total Test Cases
Campaign: Persistence	6 Total Test Cases

You should now see your new assessment named "Combined Threats" in the assessment list, click on it to see the campaign list.

The screenshot shows a table of assessments. The columns are 'Name', 'Create Date', 'Status', 'Tags', and 'Actions'. There are two rows: 'Phishing Assessment' (Create Date 03/31/2021, Status In Progress) and 'Combined Threats' (Create Date 03/31/2021, Status Not Performed). A red box highlights the 'Combined Threats' row, and a red arrow points to the 'Name' column of that row.

Name	Create Date	Status	Tags	Actions
Phishing Assessment	03/31/2021	In Progress		⋮
Combined Threats	03/31/2021	Not Performed		⋮

On the next screen, you can now see that you have, as you had selected, a campaign list for each tactic.

Campaign Dashboard						NEW CAMPAIGN	EDIT
	Name	Progress	Outcome	Tags	Action		
≡	Execution	<div style="width: 0%; background-color: #ccc;"></div>	<div style="width: 0%; background-color: #ccc;"></div>	0%			
≡	Discovery	<div style="width: 0%; background-color: #ccc;"></div>	<div style="width: 0%; background-color: #ccc;"></div>	0%			
≡	Persistence	<div style="width: 0%; background-color: #ccc;"></div>	<div style="width: 0%; background-color: #ccc;"></div>	0%			
≡	Defense Evasion	<div style="width: 0%; background-color: #ccc;"></div>	<div style="width: 0%; background-color: #ccc;"></div>	0%			
≡	Credential Access	<div style="width: 0%; background-color: #ccc;"></div>	<div style="width: 0%; background-color: #ccc;"></div>	0%			
≡	Impact	<div style="width: 0%; background-color: #ccc;"></div>	<div style="width: 0%; background-color: #ccc;"></div>	0%			
≡	Collection	<div style="width: 0%; background-color: #ccc;"></div>	<div style="width: 0%; background-color: #ccc;"></div>	0%			
≡	Command & Control	<div style="width: 0%; background-color: #ccc;"></div>	<div style="width: 0%; background-color: #ccc;"></div>	0%			
≡	Lateral Movement	<div style="width: 0%; background-color: #ccc;"></div>	<div style="width: 0%; background-color: #ccc;"></div>	0%			
≡	Exfiltration	<div style="width: 0%; background-color: #ccc;"></div>	<div style="width: 0%; background-color: #ccc;"></div>	0%			
≡	Initial Access	<div style="width: 0%; background-color: #ccc;"></div>	<div style="width: 0%; background-color: #ccc;"></div>	0%			

Which test cases will be included in these tactics? Click on the Impact campaign for example to take a look.

Test Cases

<input type="checkbox"/>	Phase	Technique	Test Case
	All	search ...	search ...
◆ <input type="checkbox"/>	Impact	Data Encrypted for Impact	Files with Ransomware Extensions #1 (LOCKY)
◆ <input type="checkbox"/>	Impact	Data Encrypted for Impact	Files with Ransomware Extensions #2 (KRATOS)
◆ <input type="checkbox"/>	Impact	Data Encrypted for Impact	Encrypt a Large Amount of Files
◆ <input type="checkbox"/>	Impact	Resource Hijacking	Upload and Execute a Cryptominer

Here is a snippet of what the ATT&CK Navigator layer that we uploaded looks like, notice that Impact had only two items scored. "Data Encrypted for Impact" and "Resource Hijacking".

Evasion	Credential Access	Discovery	Lateral Movement	Collection	Command And Control	Exfiltration	Impact
	21 items	23 items	18 items	13 items	22 items	9 items	16 items
Sniffing	Brute Force	Network Service Scanning	Remote Desktop Protocol	Input Capture	Connection Proxy	Data Compressed	Data Encrypted for Impact
HTML File Injection	Credential Dumping	Network Share Discovery	Remote File Copy	Screen Capture	Domain Generation Algorithms	Automated Exfiltration	Resource Hijacking
Red Files or on	Credentials from Web Browsers	System Network Configuration Discovery	AppleScript	Video Capture	Fallback Channels	Data Encrypted	Account Access Removal
Command	Credentials in Files	System Network Connections Discovery	Application Deployment Software	Audio Capture	Remote File Copy	Data Transfer Size Limits	Data Destruction
on Proxy	Input Capture	System Owner/User Discovery	Component Object Model and Distributed COM	Automated Collection	Standard Application Layer Protocol	Exfiltration Over Alternative Protocol	Defacement
Loading	Account Manipulation	Account Discovery	Clipboard Data	Clipboard	Web Service	Disk Content Wipe	Disk Structure Wipe
	Bash History						Endpoint Denial of

On the list of test cases that was automatically generated by Vectr, notice the only test cases added were ones that directly tested one of the scored techniques.

Test Cases			
	Phase	Technique	Test Case
	All	search ...	search ...
◆ <input type="checkbox"/>	Impact	Data Encrypted for Impact	Files with Ransomware Extensions #1 (LOCKY)
◆ <input type="checkbox"/>	Impact	Data Encrypted for Impact	Files with Ransomware Extensions #2 (KRATOS)
◆ <input type="checkbox"/>	Impact	Data Encrypted for Impact	Encrypt a Large Amount of Files
◆ <input type="checkbox"/>	Impact	Resource Hijacking	Upload and Execute a Cryptominer

This brings us to the key point of this exercise and culmination of many concepts in this course: You can and should use this method to use pre-created ATT&CK Navigator layers, that were based on real threat intelligence for high-risk threat groups, to design a purple team test ideal for **your** organization! Don't stop at one assessment either! You can continue to run the same assessment template over and over again, quarter after quarter to check that:

1. All analytics continue to work as expected
2. Your team is always improving!

This is a true intelligence-driven threat defense backed up by continuous assessments and objective measurements and improvement. With checks like this done regularly, you can rest easy at night knowing your SOC delivers on what it promises and is using its resources in an ideal fashion to deal with the highest importance and most likely threat groups. How awesome is that?!!

This has been a quick whirlwind tour of the main features of Vectr, but it supports much more customization than we have time to touch on here. For additional information on Vectr administration, see the documentation at docs.vectr.io.

Exercise Conclusion – Key Takeaways

In this exercise, you have:

- Used free, open-source software to learn how to plan and execute a purple team test
- Seen how to run a custom and template-driven assessment

- Used an ATT&CK Navigator layer to facilitate testing of high important attack tactics.

!!! Note: If you'd like to redo this exercise, be aware that the newly created session database will not automatically delete itself. You can use the "Delete Databases" option in the session database menu to delete it, or just create a new one for that portion of the exercise.

To shut down the services used for this exercise go back to your terminal window (or open a new one) and enter the commands below:

```
cd /home/student/labs/5.3
docker-compose down
```

You should a response similar to the following, if you do not, please alert your instructor:

```
Stopping sandbox1_tomcat_1 ... done
Stopping sandbox1_builder_1 ... done
Stopping sandbox1_webserver_1 ... done
Stopping sandbox1_mongo_1 ... done
Stopping sandbox1_redis_1 ... done
Removing sandbox1_tomcat_1 ... done
Removing sandbox1_builder_1 ... done
Removing sandbox1_webserver_1 ... done
Removing sandbox1_mongo_1 ... done
Removing sandbox1_redis_1 ... done
Removing network sandbox1_vectr_bridge
```

Exercise 5.3 is now complete!

Appendix: Vectr Installation Instructions

Vectr is *incredibly* quick and easy to get running with docker-compose. Visit the docs at docs.vectr.io and follow the installation instructions. In short, they are:

1. Download the zip file of the [latest release on GitHub](#) to a folder on the machine you will host Vectr on, unzip the file, and change into the new vectr folder.
2. Edit the ".env" file in the vectr folder, setting the settings as referenced in the documentation. The key item to set for functionality is the hostname, which must match the DNS hostname you will use to access the application. For security purposes, you should also change the default passwords set in this file.
3. Type `docker-compose up -d` inside the vectr folder, and wait several minutes for the application to download, setup, and start running.
4. Access the application at [https://\[hostname\]:8081](https://[hostname]:8081)

Troubleshooting

If you're having any issues with internet connectivity, exercise container launching, or otherwise, run the following command in a terminal window:

```
/home/student/labs/reset.sh
```

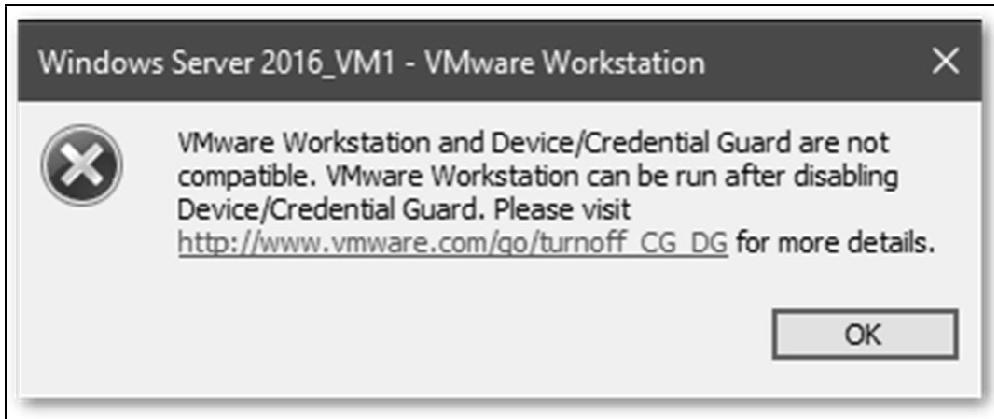
Note: When you run this script, you will see a password prompt that says `[sudo] password for student:`, at this prompt you have to type the password (`mgt551`) but it will NOT SHOW ON THE SCREEN, just type it and hit enter to proceed.

This will stop all running containers leaving the VM in a clean state, reset docker networking, and reestablish a connection to the internet by asking for a new DHCP lease.

VMware Workstation/Credential Guard Incompatibility

If your Windows host system has Credential Guard enabled and you attempt to run VMware Workstation, there is an issue that may prevent you from using your VMware in class..

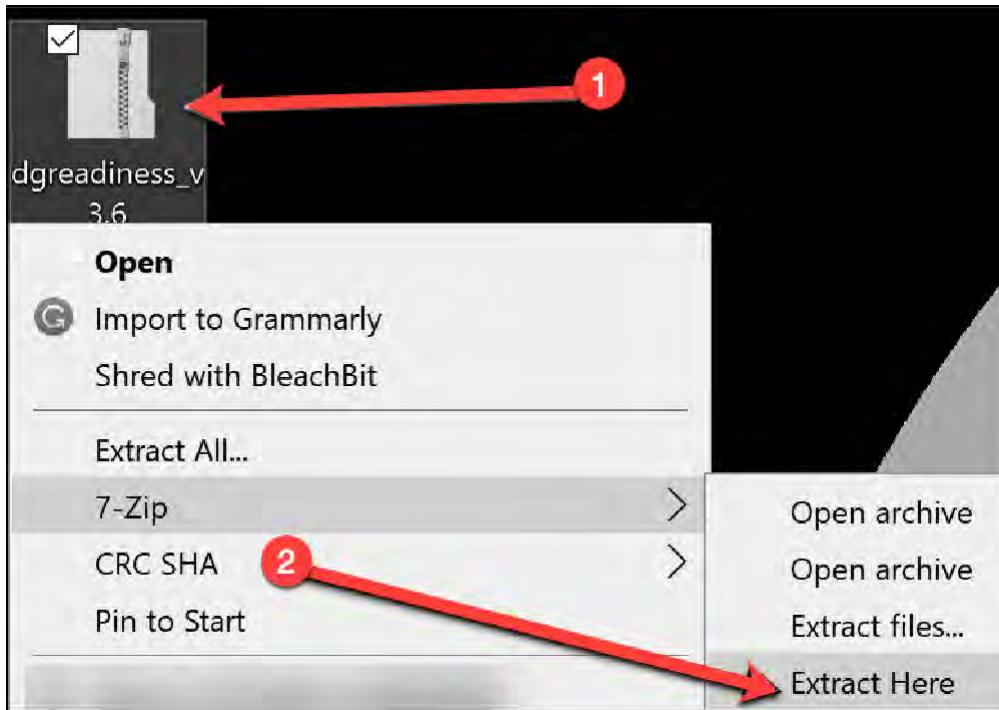
Upon running VMware Workstation, you may encounter a dialog such as below. You will not be able to start the application.



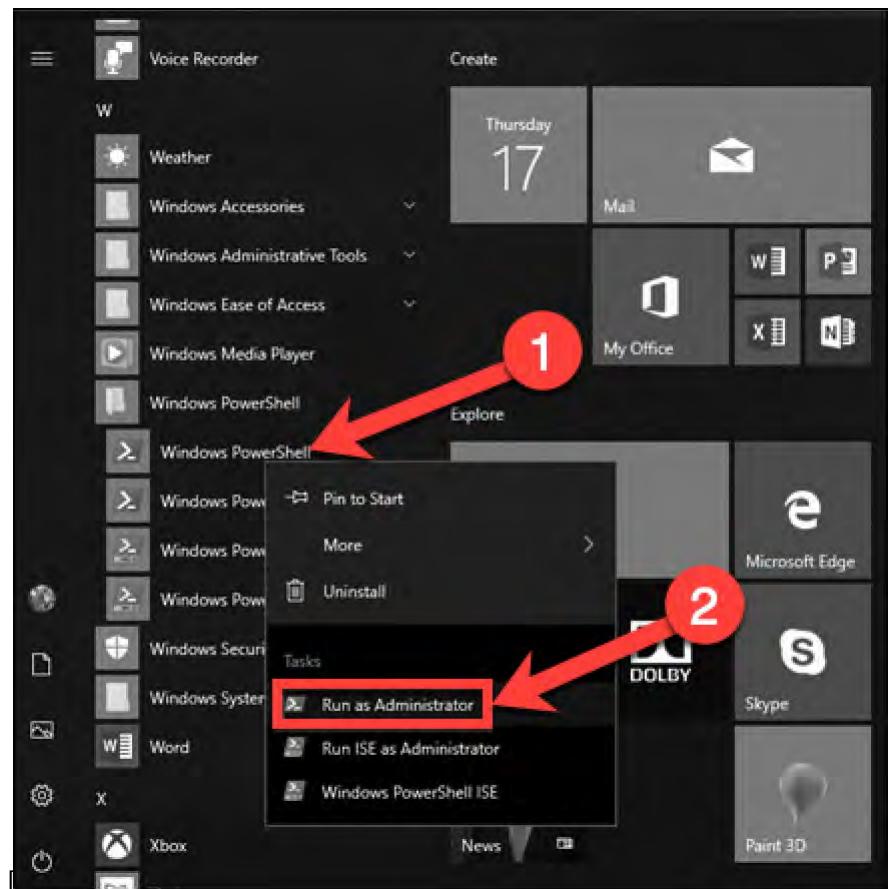
To correct this, take the following steps.

Disabling Credential Guard for Class

1. From your **host operating system**, [Download the "Device Guard and Credential Guard hardware readiness tool" from Microsoft](#).
2. Move the downloaded zip file to your desktop and extract the zip file to your Desktop.



3. Run PowerShell as Administrator.



4. In the PowerShell window, change the directory to the folder where the script is extracted and run the following PowerShell commands. For example, in the command below, the zip file was extracted to the Desktop folder. You may need to reboot your host system for the changes to take effect.

Note:

The exact version might change over time. In this example, the version is 3.6, but that might change if Microsoft updates the tool. If it does, in each command below, the folder path might change slightly based on the version number.

Command lines

```
cd ~\Desktop\dgreadiness_v3.6\  
Set-ExecutionPolicy Unrestricted
```

Expected Results

Execution Policy Change

```
Do you want to change the execution policy?  
[Y] Yes [A] Yes to All [N] No [L] No to All [S] Suspend [?] Help (default is "N"):
```

Type **A** and press the Enter/Return key.

Command lines

```
.\DG_Readiness_Tool_v3.6.ps1 -Disable
```

Expected Results

Security Warning

```
Do you want to run C:\Users\<%YOUR_USERNAME%>\Desktop\dgreadiness_v3.6\DG_Readiness_Tool_v3.6.ps1?  
[D] Do not run [R] Run once [S] Suspend [?] Help (Default is "D"):
```

Type **R** and press the Enter/Return key.

Expected Results

```
#####
Readiness Tool Version 3.4 Release
Tool to check if your device is capable to run Device Guard and Credential Guard
#####
Disabling Device Guard and Credential Guard
Deleting RegKeys to disable DG/CG

Disabling Hyper-V and IOMMU
Disabling Hyper-V and IOMMU successful

Please reboot the machine, for settings to be applied.
```

Reboot as directed and your system should be ready for use.

Re-enabling Credential Guard After Class

When class is over, if you no longer need to use VMware Workstation and/or require Credential Guard to be enabled, follow these steps.

1. Run PowerShell as Administrator as shown above.
2. Run the following commands. You may need to reboot your host system for the changes to take effect.

Note:

The exact version might change over time. In this example, the version is 3.6, but that might change if Microsoft updates the tool. If it does, in each command below, the folder path might change slightly based on the version number.

□ Command lines

```
cd ~\Desktop\dgreadiness_v3.6\  
.\\DG_Readiness_Tool_v3.6.ps1 -Enable -CG
```

A Expected Results

Security warning

```
Do you want to run C:\\Users\\<%YOUR_USERNAME%>\\Desktop\\dgreadiness_v3.6\\DG_Readiness_Tool_v3.6.ps1?  
[D] Do not run [R] Run once [S] Suspend [?] Help (Default is "D"):
```

Type R and press the Enter/Return key.

A Expected Results

```
#####
Readiness Tool Version 3.4 Release
Tool to check if your device is capable to run Device Guard and Credential Guard
#####
#####
OS and Hardware requirements for enabling Device Guard and Credential Guard
1.OS SKUs: Available only on these OS Skus - Enterprise, Server, Education, Enterprise IoT, Pro, and
Home
2.Hardware: Recent hardware that supports virtualization extension with SLAT
To learn more, please visit: https://aka.ms/dgwhcr
#####

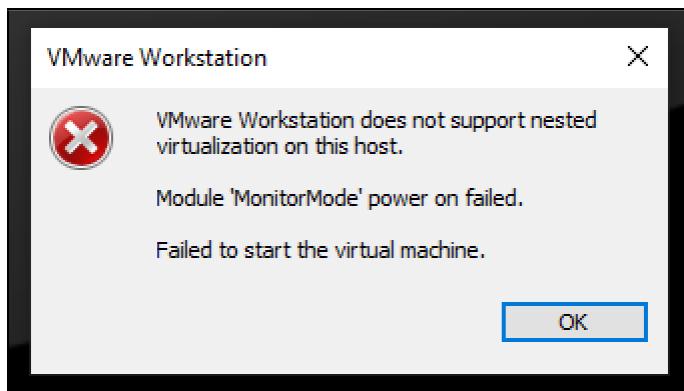
Enabling Device Guard and Credential Guard
Setting RegKeys to enable DG/CG
Enabling Hyper-V and IOMMU
Enabling Hyper-V and IOMMU successful
Please reboot the machine, for settings to be applied.
```

Reboot as directed and your system should be ready for use.

VMware Workstation/Hyper-V Incompatibility

If your Windows host system has Hyper-V enabled and you are running Windows 10 version 2004, there is an issue that may prevent you from using your class VM(s) in VMware Workstation 15.5.5.

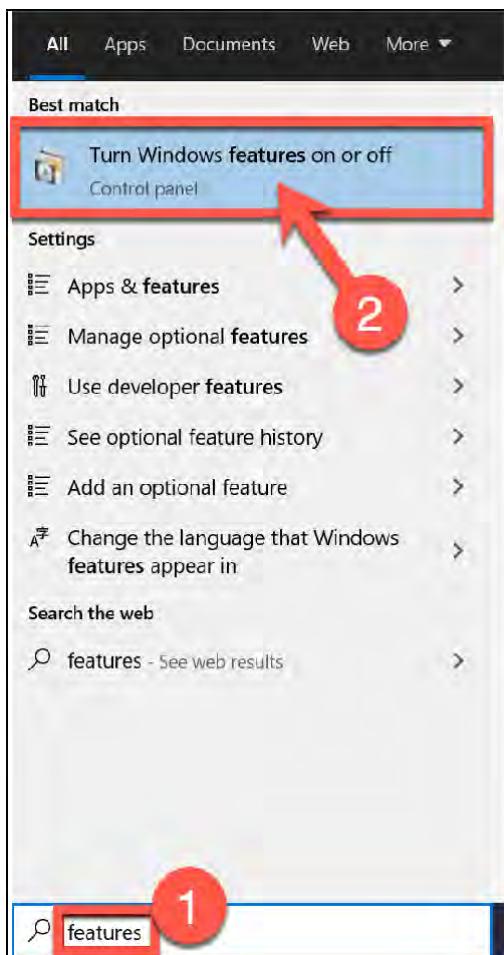
Upon starting your class virtual machine(s), you may encounter a dialog such as below. You will not be able to start the virtual machine.



To correct this, take the following steps.

Disabling Hyper-V Features for Class

1. If needed, disable Credential Guard using these instructions
2. Click the Windows button and type `features`. Then click on the result titled `Turn Windows Features on or off`.



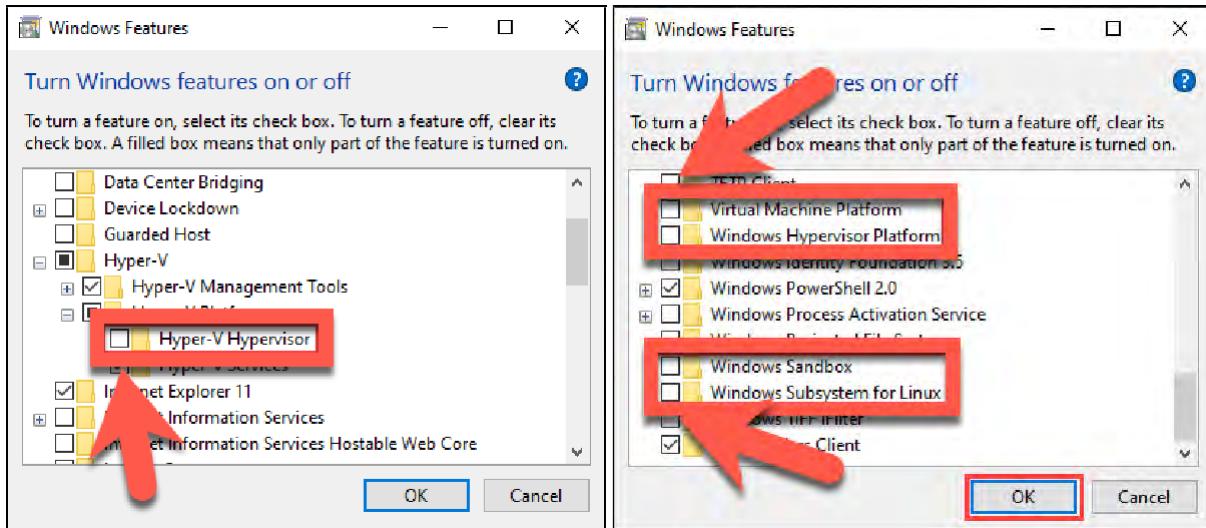
3. Ensure that the following options are unchecked:

WARNING!

Keep track of which of the following options you need to change for the class. When class is over, you'll need to re-enable any options you have disabled.

- Hyper-V Hypervisor
 - Windows Hypervisor Platform
 - Virtual Machine Platform
 - Windows Sandbox
- If you are using the Windows Subsystem for Linux 2 (WSL2), ensure that the Windows Subsystem for Linux option is also unchecked.

Click **OK**.



4. Your system will ask to reboot so the changes will take effect.

Re-enabling Hyper-V Features After Class

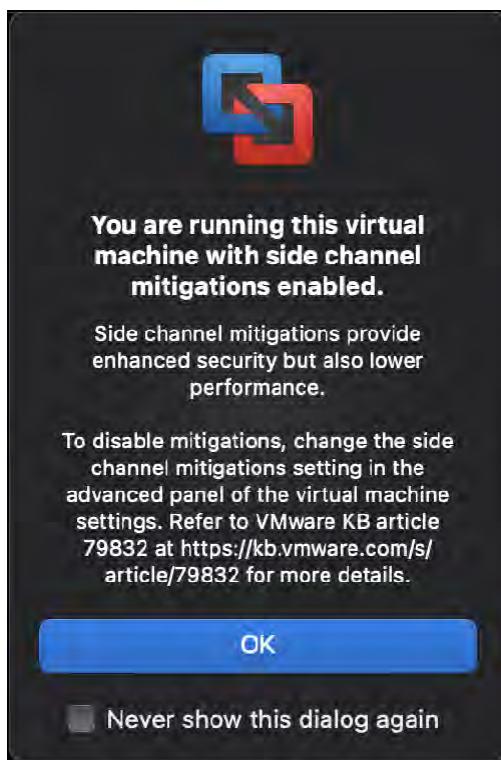
When class is over and you no longer need to use the class virtual machine, re-enable the options that you disabled above. If you disabled Credential Guard, [re-enable it with the instructions provided here](#).

VMware Fusion Issues with macOS 11 (Big Sur)

With the update to macOS 11 (Big Sur), there are a few issues that may prevent you from using your class VM(s) in VMware Fusion 12. This document addresses these issues.

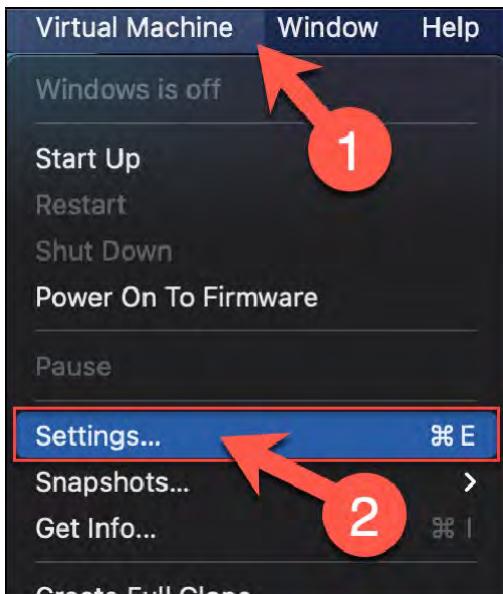
"Side Channel Mitigations" Error Message

Upon starting your class virtual machine(s), you may encounter a dialog such as below. You can safely click OK in order to continue running the affected virtual machine, however you may see degraded performance as a result.



To overcome any performance issues take the following steps.

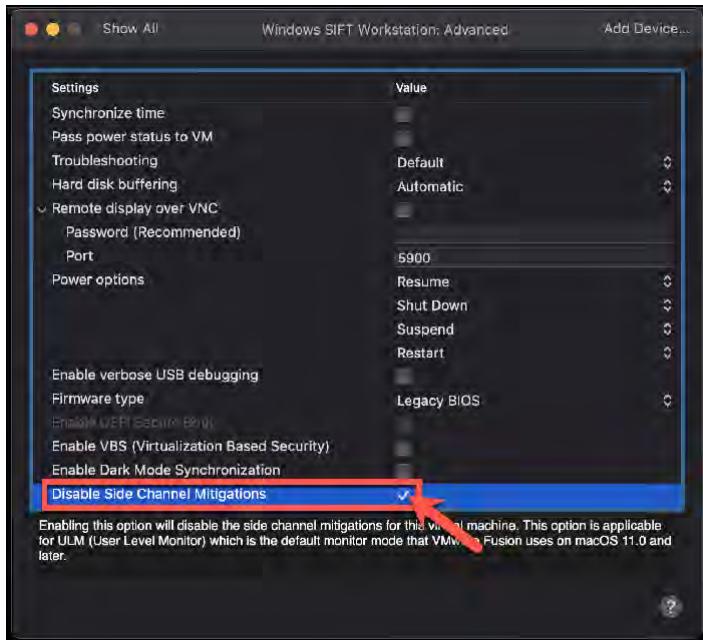
1. Shut down the virtual machine. (Not "Suspend".)
2. Click on the `Virtual Machine` menu item. Then click `Settings...`.



3. Click the `Advanced` icon.



4. Check the box next to `Disable Side Channel Mitigations`



5. Close the Settings dialog and start the virtual machine.

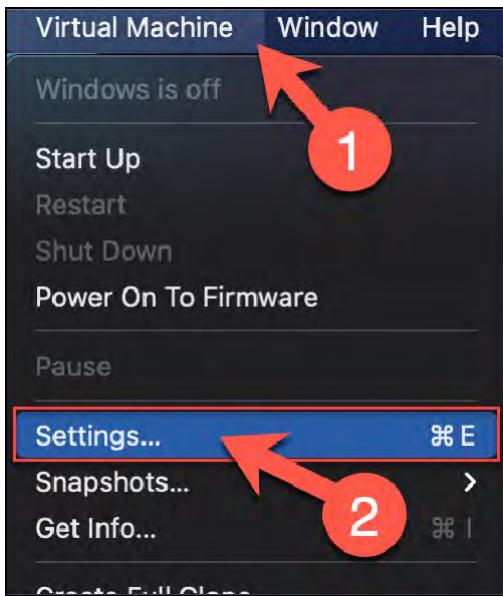
"Virtualized Performance Counters" Error Message

Upon starting your class virtual machine(s), you may encounter a dialog such as the one below. You will not be able to start the virtual machine.



To correct this, take the following steps.

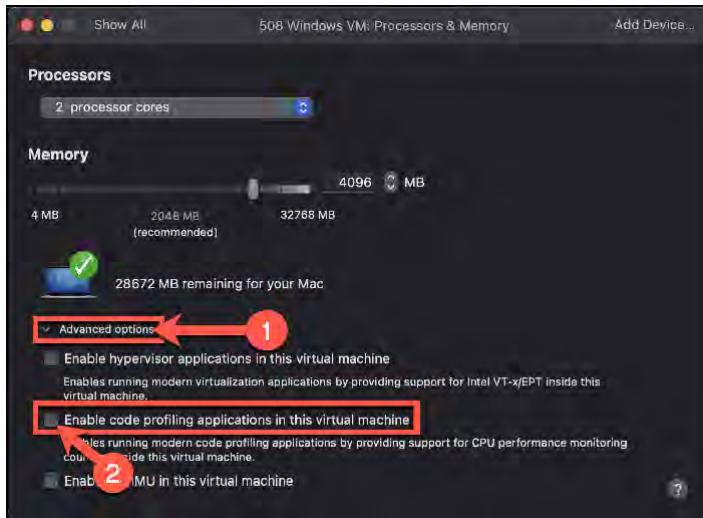
1. Click on the `Virtual Machine` menu item. Then click `Settings...`.



2. Click the `Processors & Memory` icon.



3. Click the arrow to expand the `Advanced options` section. Then un-check the box next to `Enable code profiling applications in this virtual machine`.



4. Close the Settings dialog and start the virtual machine.

"Nested Virtualization" Error Message

Upon starting your class virtual machine(s), you may encounter a dialog such as the one below. You will not be able to start the virtual machine.

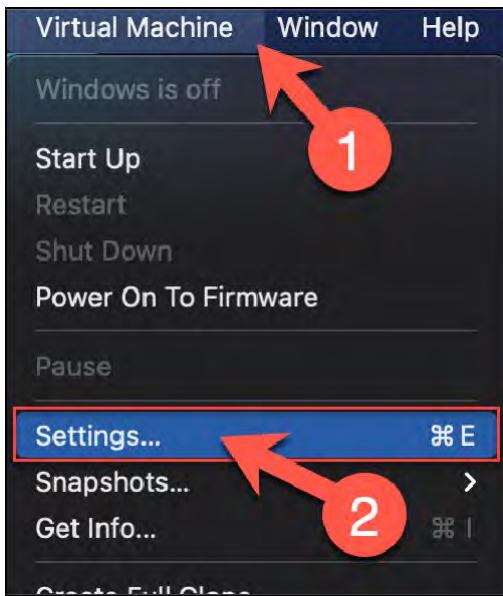


To correct this, take the following steps.

WARNING!

While taking these steps will allow you to boot the virtual machine, you may not be able to complete any labs that rely on nested virtualization features. Contact your instructor or OnDemand support to determine if this affects your class.

1. Click on the `Virtual Machine` menu item. Then click `Settings...`.



2. Click the `Processors & Memory` icon.



3. Click the arrow to expand the `Advanced options` section. Then un-check the box next to `Enable hypervisor applications in this virtual machine`.

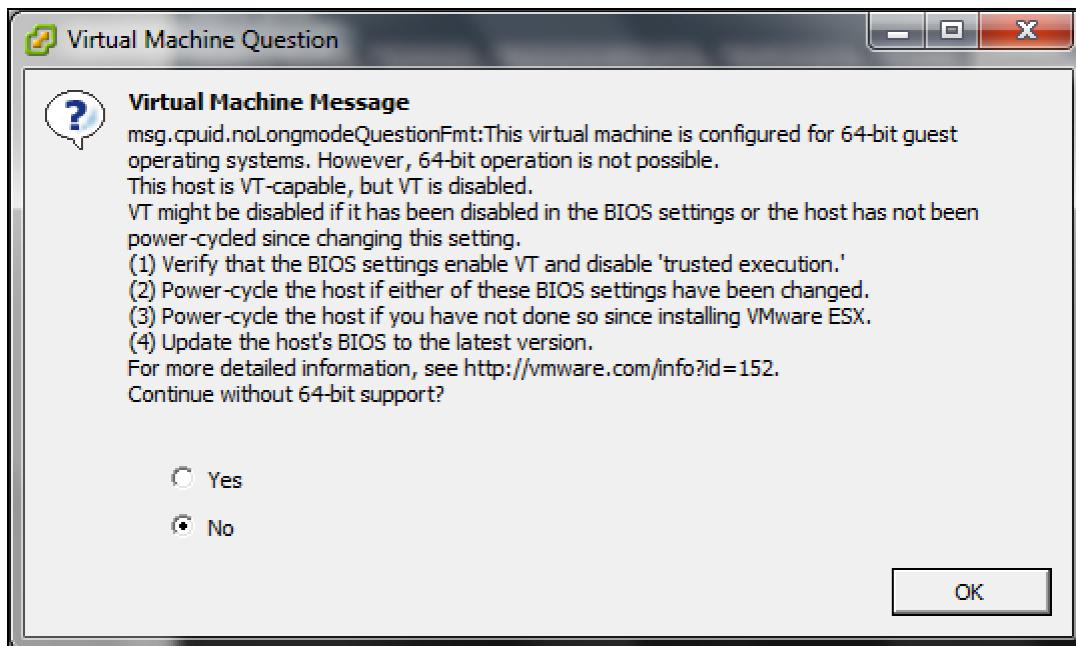


4. Close the Settings dialog and start the virtual machine.

Enabling Virtualization Technology Extensions (VTx) in Intel and AMDBIOS

On Intel and AMD systems, there is a BIOS extension that must be enabled or you will not be able to boot your class VM(s) in VMware.

Upon starting your class virtual machine(s), you may encounter a dialog similar to the one below. Starting the virtual machine without 64-bit support will result in a non-functional VM.



To correct this, take the following steps.

Enabling VTx for Class

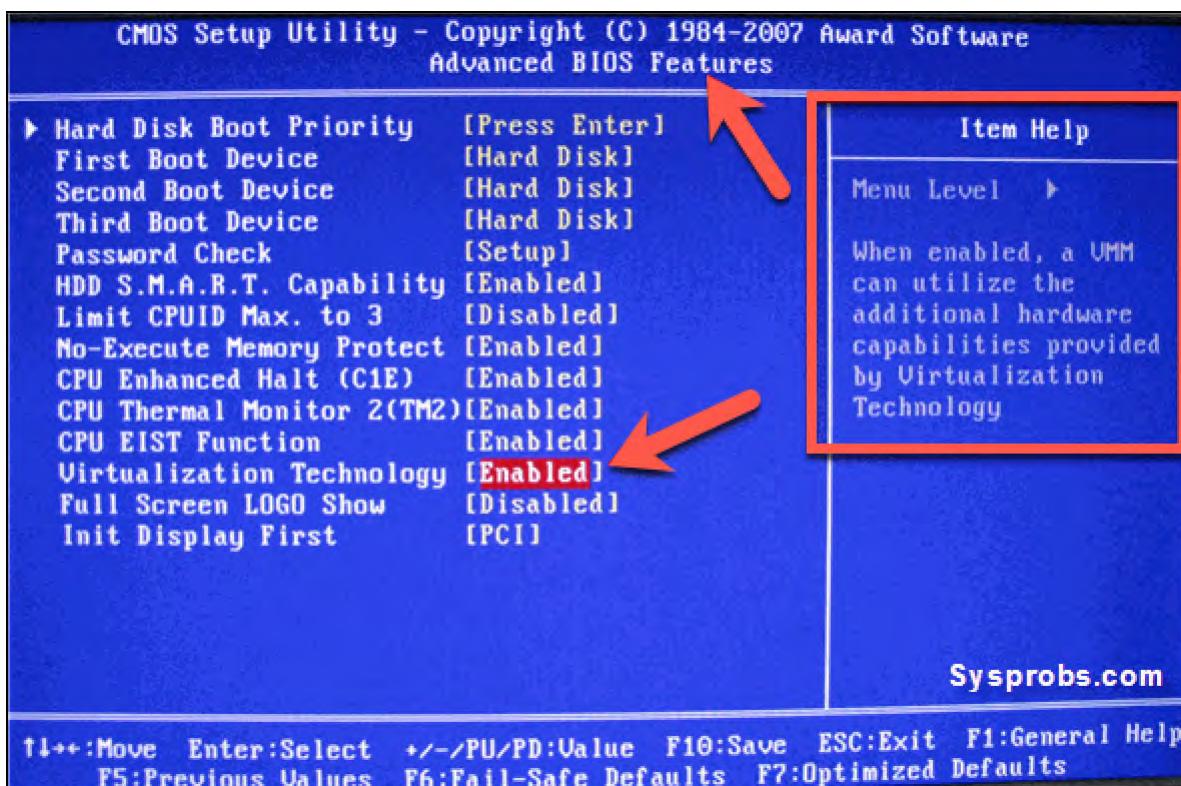
1. Enter your system's BIOS configuration menus. This requires pressing a designated key immediately upon booting/rebooting your system, but the exact key depends on the system and BIOS manufacturers. Most systems use one of the following five keys:

- F1
- F2
- DEL
- ESC
- F10

- Older computers may require multiple keys to be pressed simultaneously, or keys other than those listed above:

- CTRL+ALT+ESC
- CTRL+ALT+INS
- CTRL+ALT+ENTER
- CTRL+ALT+S
- PGUP
- PGDN

2. Identify the BIOS menu that controls the VTx settings. This is also dependent on the specific version of BIOS that your system uses. The screenshots below represent the Award BIOS, but you may need to explore the various BIOS menus on your system to find the proper menu and setting. Different BIOS versions also have varying keyboard controls - some use the space bar to change settings, others use the PGUP and PGDN keys, etc.



Saving the settings may require pressing F10 or other keys or menu sequences.

3. Exit the BIOS settings and reboot the system. Ideally, keep the power off for approximately one minute before powering it on to clear any residual configuration settings. The reboot is critical, as the BIOS settings are essentially a configuration file that is only read at boot time.

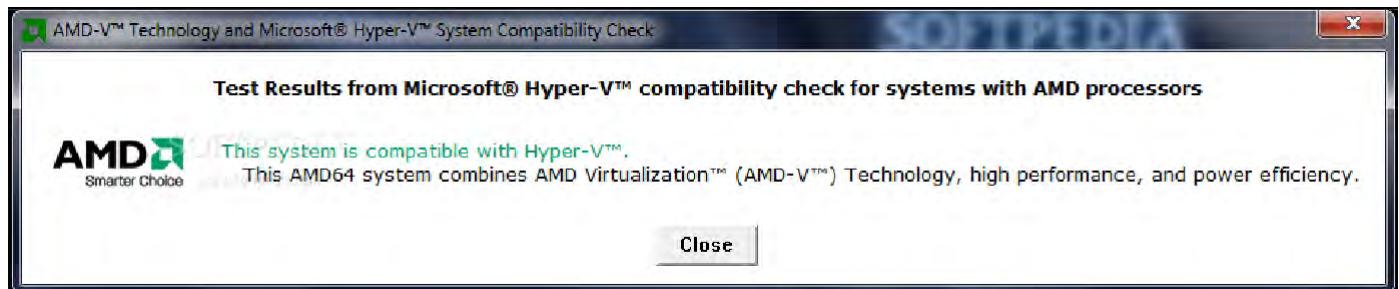
Verifying That VTx Settings are Correct

There are several ways to verify that the VTx settings above have been set correctly.

1. Boot your class VM(s) to ensure the VTx error at the beginning of this document is not displayed.
2. For Intel processors, you may [download the Intel Processor Identification Utility](#). Run the utility and click the **CPU Technologies** tab to confirm if VTx is enabled or not.



3. For AMD processors, you may [download the AMD Virtualization Technology and Microsoft Hyper-V System Compatibility Check Utility](#). Run the utility to confirm if VTx is enabled or not.



4. For both Intel and AMD processors, you may download Microsoft's Hardware-Assisted Virtualization Detection Tool. Run the utility to confirm if VTx is enabled or not.

