CHAPTER 17
Crisis Management

By Matt Linton
with Nick Soda and Gary O'Connor

Once your systems are running, you'll want to keep those systems up, even when malicious actors are attacking your organization. The reliability of your systems is a measure of how well your organization can withstand a security crisis, and reliability directly impacts the happiness of your users.

This chapter starts by clarifying how to recognize a crisis, followed by a detailed plan of how to take command and maintain control of an incident—a topic that includes deep-dives into operational security and forensics. Communications are a critical but often overlooked part of crisis management. We guide you through some communication-related pitfalls to avoid and provide examples and templates. Finally, we walk through a sample crisis scenario to demonstrate how the pieces of incident response fit together.

Incident response is critical for both reliability and security incidents. Chapter 14 of the SRE book and Chapter 9 of the SRE workbook explore incident response as it relates to reliability outages. We use the same methodology—the Incident Management at Google (IMAG) framework—to respond to security incidents.

Security incidents are inevitable. A common maxim in the industry states that "There are only two types of companies: those that know they've been compromised, and those that don't know." The outcome of a security incident depends upon how well your organization prepares, and also how well you respond. To achieve a mature security posture, your organization needs to institute and practice an incident response (IR) capability, as discussed in the previous chapter.

In addition to the familiar pressures of ensuring that unauthorized parties can't access your systems and that your data stays where it should, IR teams today face new and difficult challenges. As the security industry trends toward greater transparency[1] and a need for increased openness with users, this expectation poses a unique challenge for any IR team accustomed to operating away from the spotlight of public attention. Additionally, regulations like the EU's General Data Protection Regulation (GDPR) and service contracts with security-conscious customers continually push the boundaries of how quickly investigations must begin, progress, and complete. Today, it's not unusual for a customer to ask for a notification of a potential security problem within 24 hours (or less) of initial detection.

Incident notification has become a core feature of the security domain, alongside technological advances such as easy and ubiquitous use of cloud computing, widespread adoption of "bring your own device" (BYOD) policies in the workplace, and the Internet of Things (IoT). Such advances have created new challenges for IT and security staff—for example, limited control over and visibility into all of an organization's assets.

Is It a Crisis or Not?

Not every incident is a crisis. In fact, if your organization is in good shape, relatively few incidents should turn into crises. Once an escalation occurs, a responder's first step in assessing the escalation is *triage*—using the knowledge and information available to them to make educated and informed assumptions about the severity and potential consequences of the incident.

Triage is a well-established skill in the emergency medical community. An emergency medical technician (EMT) arriving on the scene of a vehicle accident will first make sure there are no immediate risks of further injury to anyone at the scene, and then perform triage. For example, if a bus has collided with a car, a few pieces of information are already logically available. The people in the car may have sustained serious injuries, because a collision with a heavy bus can inflict a great deal of damage. A bus can hold many passengers, so there may be multiple injuries to passengers. It's unlikely that any dangerous chemicals are present, because neither vehicle would typically carry them. Within the first minute of arriving, the EMT knows that they'll need to call for more ambulances, possibly alert a critical care unit, and call the fire department to free any trapped occupants from the smaller vehicle. They probably don't need a hazardous materials cleanup crew.

1 See, e.g., Google's Transparency Report.

Your security response team should use these same assessment methods to triage incidents as they come in. As a first step, they must estimate the potential severity of the attack.

Triaging the Incident

When triaging, the engineer assigned to investigate must gather basic facts to help decide whether the escalation is one of the following:

- An error (i.e., a false positive)
- An easily correctable problem (an opportunistic compromise, perhaps)
- A complex and potentially damaging problem (such as a targeted compromise)

They should be able to triage predictable problems, bugs, and other straightforward issues using predetermined processes. Larger and more complex issues, such as targeted attacks, will likely require an organized and managed response.

Every team should have preplanned criteria to help determine what constitutes an incident. Ideally, they should identify what kinds of risks are severe versus acceptable in their environment before an incident happens. The response to an incident will depend on the type of environment where the incident happened, the state of the organization's preventative controls, and the sophistication of its response program. Consider how three organizations might respond to the same threat—a ransomware attack:

- *Organization 1* has a mature security process and layered defenses, including a restriction that permits only cryptographically signed and approved software to execute. In this environment, it's highly unlikely that well-known ransomware can infect a machine or spread throughout the network. If it does, the detection system raises an alert, and someone investigates. Because of the mature processes and layered defenses, a single engineer can handle the issue: they can check to make sure no suspicious activity has occurred beyond the attempted malware execution, and resolve the issue using a standard process. This scenario doesn't require a crisis-style incident response effort.
- *Organization 2* has a sales department that hosts customer demos in a cloud environment, where people who want to learn about the organization's software install and manage their test instances. The security team notices that these users tend to make security configuration mistakes that result in system compromises. So that these compromises don't require manual intervention from a human responder, the security team establishes a mechanism to automatically wipe and replace compromised cloud test instances. In this case, a ransomware worm would also not require much forensics or incident response attention. Although

Organization 2 doesn't prevent the ransomware from executing (as in Organization 1's case), Organization 2's automated mitigation tools can contain the risk.

- *Organization 3* has fewer layered defenses and limited visibility into whether its systems are compromised. The organization is at much greater risk of the ransomware spreading across its network and may not be able to respond quickly. In this case, a large number of business-critical systems may be affected if the worm spreads, and the organization will be severely impacted, requiring significant technical resources to rebuild the compromised networks and systems. This worm presents a serious risk for Organization 3.

While all three organizations are responding to the same source of risk (a ransomware attack), the differences in their layered defenses and level of process maturity affect the potential severity and impact of the attack. While Organization 1 may need to simply initiate a playbook-driven response, Organization 3 may face a crisis that requires coordinated incident management. As the likelihood that an incident will pose a serious risk to the organization increases, so does the likelihood that it will require an organized response by many participants.

Your team can perform some basic assessments to determine whether an escalation requires a standard playbook-driven approach or a crisis management approach. Ask yourself the following questions:

- What data do you store that might be accessible to someone on that system? What is the value or criticality of that data?
- What trust relationships does the potentially compromised system have with other systems?
- Are there compensating controls that an attacker would also have to penetrate (and that seem intact) in order to take advantage of their foothold?
- Does the attack seem to be commodity opportunistic malware (e.g., Adware), or does it appear more advanced or targeted (e.g., a phishing campaign seemingly crafted with your organization in mind)?

Think through all the relevant factors for your organization, and determine the highest likely level of organizational risk given those facts.

Compromises Versus Bugs

IR teams have long been tasked with responding to suspected intrusions and compromises. But what about software and hardware bugs, a.k.a. security vulnerabilities? Do you treat a newly discovered security vulnerability in your systems as a compromise that has yet to be discovered?

Software bugs are inevitable, and you can plan for them (as explained in Chapter 8). Good defensive practices remove or limit the potential negative consequences of vulnerabilities before they begin.[2] If you plan well and implement in-depth defenses with additional layers of security, you shouldn't need to handle vulnerability remediation the same way you handle incidents. That said, it may be appropriate to manage complicated or large-impact vulnerabilities with incident response processes, which can help you organize and respond quickly.

At Google, we typically treat vulnerabilities that carry extreme risk as incidents. Even if a bug isn't actively being exploited, a particularly severe one can still introduce extreme risk. If you're involved in fixing the vulnerability before it's publicly disclosed (these efforts are often called *coordinated vulnerability disclosures*, or CVDs), operational security and confidentiality concerns may warrant a heightened response . Alternatively, if you're hurrying to patch systems after a public disclosure, securing systems that have complex interdependencies may require urgent effort, and it may be difficult and time-consuming to deploy fixes.

Some examples of particularly risky vulnerabilities include Spectre and Meltdown (CVE-2017-5715 and 5753), glibc (CVE-2015-0235), Stagefright (CVE-2015-1538), Shellshock (CVE-2014-6271), and Heartbleed (CVE-2014-0160).

Coordinated Vulnerability Disclosure

There are many interpretations of what *CVD* means. The emergent ISO standard 29147:2018 provides some guidance. At Google, we generally define CVD as a process in which a team must maintain a careful balance between the amount of time it may take the vendor to issue security patches, the needs and wishes of the person who finds or reports the bug, and the needs of the user base and customers.

Taking Command of Your Incident

Now that we've discussed the process of triage and risk assessment, the next three sections assume a "big one" has happened: you've identified or suspect a targeted compromise, and you need to perform full-fledged incident response.

2 In one of our design reviews at Google, an engineer suggested that "There are two kinds of software developers: those who sandboxed Ghostscript, and those who should have sandboxed Ghostscript."

The First Step: Don't Panic!

Many responders associate a serious incident escalation with a rising sense of panic and an adrenaline rush. Emergency responders in the fire, rescue, and medical fields are warned in basic training not to run at the scene of an emergency. Not only does running risk making the problem worse by increasing the likelihood of an accident on the scene, it also instills a sense of panic in the responder and the public. In a similar fashion, during a security incident, the extra few seconds you gain by rushing are quickly eclipsed by the consequences of a failure to plan.

Although the SRE and security teams at Google perform incident management similarly, there is a difference between beginning a crisis management response for a security incident and for a reliability incident such as an outage. When an outage occurs, the on-call SRE prepares to step into action. Their goal is to quickly find the errors and fix them to restore the system to a good state. Most importantly, the system is not conscious of its actions and won't resist being fixed.

In a potential compromise, the attacker may be paying close attention to actions taken by the target organization and work against the responders as they try to fix things. It can be catastrophic to attempt to fix the system without first completing a full investigation. Because typical SRE work doesn't carry that risk, an SRE's usual response is to fix the system first and then document what they learned from the failure. For example, if an engineer submits a change list (CL) that breaks production, an SRE might take immediate action to revert the CL, making the problem go away. Once the problem is fixed, the SRE will begin investigating what happened. A security response instead requires the team to complete a full investigation into what happened prior to attempting to correct things.

As a security incident responder, your first task is to take control of your emotions. Take the first five minutes of an escalation to breathe deeply, allow any feelings of panic to pass, remind yourself that you need a plan, and begin thinking through the next steps. While the desire to react immediately is strong, in practice, postmortems rarely report that a security response would have been more effective if staff had responded five minutes sooner. It's more likely that some additional planning up front will add greater value.

Beginning Your Response

At Google, once an engineer decides that the issue they're facing is an Incident, we follow a standard process. That process, called Incident Management at Google, is fully described in Chapter 9 of the SRE workbook, along with case studies of outages and events where we've applied this protocol. This section describes how you can use IMAG as a standard framework to manage a security compromise.

First, a quick refresher: as mentioned in the previous chapter, IMAG is based on a formal process called the Incident Command System (ICS), which is used by fire, rescue, and police agencies around the world. Like ICS, IMAG is a flexible response framework that is lightweight enough to manage small incidents, but capable of expanding to encompass large and wide-ranging issues. Our IMAG program is tasked with formalizing processes to ensure maximum success in three key areas of incident handling: command, control, and communications.

The first step in managing an incident is to take *command*. In IMAG, we do this by issuing a declaration: "Our team is declaring an incident involving *X*, and I am the incident commander (IC)." Explicitly saying "this is an incident" may seem simple and perhaps unnecessary, but being explicit in order to avoid misunderstandings is a core principle of both command and communications. Beginning your incident with a declaration is the first step in aligning everyone's expectations. Incidents are complex and unusual, involve high amounts of tension, and happen at high speed. Those involved need to focus. Executives should be notified that teams might ignore or bypass normal processes until the incident is contained.

After a responder takes command and becomes the IC, their job is to keep *control* of the incident. The IC directs the response and ensures people are moving forward toward specific goals at all times, so that the chaos and uncertainty around a crisis doesn't throw teams off track. In order to maintain control, the IC and their leads must constantly maintain excellent *communications* with everyone involved.

Google uses IMAG as a general-purpose response framework for all sorts of incidents. All on-call engineers (ideally) are trained in the same set of fundamentals and taught how to use them to scale and professionally manage a response. While the focus of SRE and security teams may differ, ultimately, having the same framework for response enables both groups to seamlessly interoperate under stress, when working with unfamiliar teams may be at its most difficult.

Establishing Your Incident Team

Under the IMAG model, once an Incident is declared, the person declaring the incident either becomes the incident commander or selects an IC from among the other available staff. Regardless of which route you take, make this assignment explicit in order to avoid misunderstandings among responders. The person designated the IC must also explicitly acknowledge that they accept the assignment.

Next, the IC will assess what actions need to be taken immediately, and who can fill those roles. You'll likely need some skilled engineers to initiate the investigation. Large organizations may have a dedicated security team; very large organizations might even have a dedicated incident response team. A small organization may have one dedicated security person, or someone who handles security part time alongside other operational responsibilities.

Regardless of the size or makeup of the organization, the IC should locate staff who know the potentially affected systems well and deputize these individuals into an incident response team. If the incident grows larger and requires more staff, it's helpful to assign a few leaders to head up certain aspects of the investigation. Nearly every incident needs an *operations lead* (OL): the tactical counterpart and partner to the IC. While the IC focuses on setting the strategic goals needed to make progress on the incident response, the OL focuses on meeting those goals and determining how to do so. Most of the technical staff performing investigations, fixing and patching systems, and so on should report to the OL.

Some other lead roles you may need to fill include the following:

Management liaison
: You may need someone to make tough calls on the spot. Who from your organization can decide to shut down a revenue-generating service if necessary? Who can decide to send staff home or revoke the credentials of other engineers?

Legal lead
: Your incident may raise legal questions that you'll need help answering. Do your employees have an enhanced expectation of privacy? For example, if you think someone downloaded malware via their web browser, do you need extra permissions to examine their browser history? What if you think they downloaded malware via their personal browser profile?

Communications lead
: Depending on the nature of the incident, you may need to communicate with your customers, regulators, etc. A professional who is skilled at communicating could be a vital addition to your response team.

Operational Security

In the context of crisis management, *operational security (OpSec)* refers to the practice of keeping your response activity secret. Whether you are working on a suspected compromise, an insider abuse investigation, or a dangerous vulnerability whose existence could lead to widespread exploitation if publicized, you'll likely have information that you need to keep secret, at least for a limited amount of time. We strongly recommend that you establish an OpSec plan today—before you ever have an incident—so that you don't have to come up with this plan in a rush at the last minute. Once a secret is lost, it's hard to regain.

The IC is ultimately responsible for ensuring that rules around confidentiality are set, communicated, and followed. Every team member asked to work on an investigation should be briefed on what to expect. You may have specific rules for how to handle data, or expectations around which communication channels to use. For example, if you suspect your email server is within scope of a breach, you might prohibit

employees from emailing one another about the breach in case an attacker can see those conversations.

As a best practice, we recommend that your IC document specific guidance for each incident response team member. Each team member should review and acknowledge these guidelines before starting work on the incident. If you don't clearly communicate confidentiality rules to all relevant parties, you risk an information leak or premature disclosure.

In addition to protecting the response activity from your attacker, a good OpSec plan addresses how the response can proceed without further exposing the organization. Consider an attacker who compromises one of your employee's accounts and is trying to steal other passwords from memory on a server. They'd be delighted if a system administrator logged into the affected machine with administrative credentials during their investigation. Plan ahead for how you'll access data and machines without providing your attacker additional leverage. One way to accomplish this is to deploy remote forensic agents to all your systems in advance. These software packages maintain an access path for authorized responders from your company to obtain forensic artifacts without risking their own accounts by logging into the system.

The consequences of cluing in an attacker that you've discovered their attack can be high. A determined attacker who wants to persist beyond your investigation may go quiet. This deprives you of valuable insight into the extent of their compromise and can cause you to miss one (or more) of their footholds. And an attacker who has accomplished their objective and does not want to stay quiet may respond to your discovery by destroying as much of your organization as they can on their way out the door!

The following are some common OpSec mistakes:

- Communicating about or documenting the incident in a medium (such as email) that enables the attacker to monitor response activities.
- Logging into compromised servers. This exposes potentially useful authentication credentials to the attacker.
- Connecting to and interacting with the attacker's "command and control" servers. For example, don't try to access an attacker's malware by downloading it from a machine you're using to perform your investigation. Your actions will stand out in the attacker's logs as unusual and could warn them of your investigation. Also, don't perform port scanning or domain lookups for the attacker's machines (a common mistake made by novice responders).
- Locking accounts or changing passwords of affected users before your investigation is complete.
- Taking systems offline before you understand the full scope of the attack.

- Allowing your analysis workstations to be accessed with the same credentials an attacker may have stolen.

Consider the following good practices in your OpSec response:

- Conduct meetings and discussions in person where possible. If you need to use chat or email, use new machines and infrastructure. For example, an organization facing a compromise of unknown extent might build a new temporary cloud-based environment and deploy machines that differ from its regular fleet (e.g., Chromebooks) for responders to communicate. Ideally, this tactic provides a clean environment to chat, document, and communicate outside of the attacker's view.
- Wherever possible, ensure that your machines have remote agents or key-based access methods configured. This allows you to collect evidence without revealing login secrets.
- Be specific and explicit about confidentiality when asking people to help—they may not know particular information is supposed to be kept confidential unless you tell them.
- For each step of your investigation, consider the conclusions a shrewd attacker may draw from your actions. For example, an attacker who compromises a Windows server may notice a sudden rush of group policy tightening and conclude that they've been discovered.

When Your Tools Try to Be Helpful

Many modern communication and collaboration tools (e.g., email and chat clients, or collaborative document editors) try to be helpful by automatically detecting patterns that seem like internet content and creating links. Some tools even connect to those remote links and cache the content they find so they can display that content faster if it's requested. For example, writing "example.com" in a spreadsheet or email or chat window may result in content being downloaded from that site without any further interaction on your part.

Ordinarily this behavior is helpful, but if you're trying to practice good operational security while collecting data about your attacker, these tools can betray you by automatically talking to the attacker's infrastructure in easily observable ways.

If several analysts are sharing information in a private chat forum that automatically fetches content, something like the following may appear in an attacker's logs:

```
10.20.38.156 - - [03/Aug/2019:11:30:40 -0700] "GET /malware_c2.gif HTTP/
1.1" 206 36277 "-" "Chatbot-LinkExpanding 1.0"
```

If your attacker is paying attention, this is a pretty obvious signal that you've discovered their *malware_c2.gif* file.

Get in the habit of always writing out domains and other network-based indicators in ways that your tools will not match and autocomplete, even when you're not investigating an incident. Writing example.com as example[dot]com as a matter of routine will make a slipup during an investigation much less likely.

Trading Good OpSec for the Greater Good

There's one glaring exception to the general advice of keeping your incident response a secret: if you're faced with an imminent and clearly identifiable risk. If you suspect a compromise to a system so critical that vital data, systems, or even lives may be at risk, extreme measures may be justified. In the case of a vulnerability or bug that's being managed as an incident, sometimes that bug may be so easily exploited and so widely known (e.g., Shellshock[3]) that turning off or entirely disabling the system might be the best way to protect it. Doing so will, of course, make it obvious to your attacker and others that something is amiss.

Security and Reliability Tradeoff: Imminent Risk

You might have to endure product downtime and user anger in order to accomplish longer-term security and reliability objectives. In early 2019, Apple responded to a publicly disclosed, easily exploitable privacy bug in Facetime by turning off all access to the Facetime servers until it had deployed fixes. This day-long outage for a popular service was widely recognized in the security industry as the right course of action. In this case, Apple chose protecting its users from an easily exploitable issue over guaranteeing service availability.

These sorts of big decisions and organizational tradeoffs aren't likely to be solely made by the incident commander, except in extremely risky situations (e.g., shutting off a control system for a power grid to prevent a catastrophe). Typically, executives within the organization make such calls. However, the IC is the resident expert in the room when those decisions are debated, and their advice and security expertise are critical. At the end of the day, much of an organization's decision making in a crisis isn't about making the *right* call; it's about making the *best possible* call from among a range of suboptimal choices.

3 Shellshock was a remote exploit so simple to deploy that within a few days of its publication, millions of servers were already being actively attacked.

The Investigative Process

Investigating a security compromise involves attempting to backtrack through each phase of the attack to reconstruct the attacker's steps. Ideally, your IR team (composed of any engineers assigned to the job) will attempt to maintain a tight loop of effort among multiple tasks.[4] This effort focuses on identifying all the affected parts of the organization and learning as much as possible about what happened.

Digital forensics refers to the process of figuring out all of the actions an attacker may have taken on a device. A forensic analyst (the engineer performing forensics—ideally, someone with special training and experience) analyzes all the parts of the system, including available logs, to try to determine what happened. The analyst may perform some or all of the following investigative steps:

Forensic imaging
: Making a secure read-only copy (and checksum) of any data storage devices attached to a compromised system. This preserves the data in a known state at the exact time the copy is taken, without accidentally overwriting or damaging the original disk. Court proceedings often need forensic images of original disks as evidence.

Memory imaging
: Making a copy of a system's memory (or in some cases, the memory of a running binary). Memory can contain many pieces of digital evidence that may be useful during an investigation—for example, process trees, executables that were running, and even passwords to files that attackers may have encrypted.

File carving
: Extracting a disk's contents to see if you can recover certain file types, especially those that may have been deleted—for example, logs the attacker tried to delete. Some operating systems don't zero the contents of files when they're deleted. Instead, they only unlink the filename and mark the disk area as free for later reuse. As a result, you may be able to recover data an attacker attempted to remove.

Log analysis
: Investigating events relating to the system that appear in logs, whether on the system itself or from other sources. Network logs may show who talked to the system and when; logs from other servers and desktops may show other activities.

4 This tight loop of effort has a minimal delay between steps, such that no one is unable to do their part because they're waiting on someone else to complete their part.

Malware analysis
: Performing analysis on tools used by attackers to determine what those tools do, how they work, and what systems the tools might communicate with. The data from this analysis is usually fed back to teams doing forensic and detection work to provide better insight about potential indications that a system has been compromised.

In digital forensics, the relationships between events are as important as the events themselves.

Much of the work a forensic analyst does to obtain artifacts contributes to the goal of building a *forensic timeline*.[5] By collecting a chronologically ordered list of events, an analyst can determine correlation and causation of attacker activity, proving *why* these events happened.

Example: An Email Attack

Let's consider a fictional scenario: an unknown attacker has successfully compromised an engineer's workstation by sending a malicious attachment via email to a developer, who unwittingly opened it. This attachment installed a malicious browser extension onto the developer's workstation. The attacker, leveraging the malicious extension, then stole the developer's credentials and logged in to a file server. Once inside the server, the attacker proceeded to collect confidential files and copy them to their own remote server. Eventually, the developer detected the compromise when reviewing their installed browser extensions and reported the breach to security.

As an incident responder, your first instinct may be to lock the developer's account right away—but remember the operational security concerns mentioned earlier. You should always start your investigation with a hands-off approach, until you know enough about the attack to make informed decisions about how to react. At the beginning of the investigation, you have very little information. You know only that a malicious browser extension exists on the developer's machine.

Your first step is always to remain calm and not to panic. Next, declare that an incident is in progress and engage an operations lead to conduct a deeper investigation. From here, the OL should build a team to answer the following questions:

- How did the backdoor get installed?
- What are the capabilities of the backdoor?

5 A forensic timeline is a list of all the events that happened on a system, ideally centered on the events related to an investigation, ordered by the time at which those events occurred.

- On what other browsers in the organization does this backdoor exist?
- What did the attacker do on the developer's system?

We refer to these initial questions as *pivot points*: as you answer each question, new questions arise. For example, once the response team discovers that the attack progressed to a file share, that file share becomes the subject of a new forensic investigation. The investigation team must subsequently answer the same set of questions for the file share and any new leads that follow. As the team identifies each of the attacker's tools and techniques, they process the information to determine where to target any additional investigations.

Sharding the investigation

If you have enough personnel available to staff multiple efforts simultaneously (see "Parallelizing the Incident" on page 401), consider splitting your effort into three tracks, each tasked with a major piece of the investigation. For example, your OL might separate their makeshift team into three groups:

- A *Forensics* group to investigate systems and identify which ones the attacker has touched.
- A *Reversing* group to study suspicious binaries, determining the unique fingerprints that serve as indicators of compromise (IOCs).[6]
- A *Hunting* group to search all systems for those fingerprints. This group notifies the Forensics group any time they identify a suspect system.

Figure 17-1 shows the relationships between the groups. The OL is responsible for keeping a tight feedback loop between these teams.

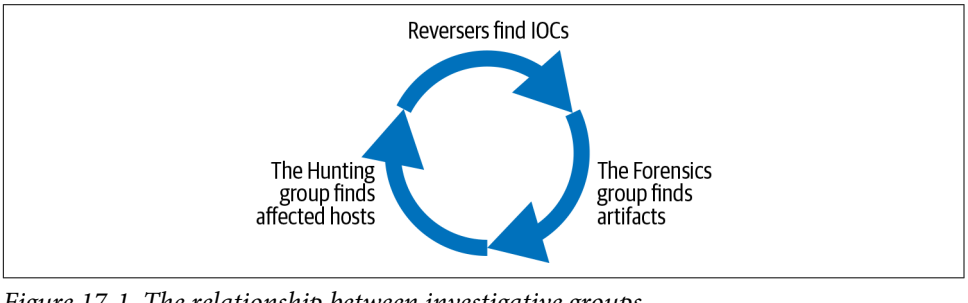

Figure 17-1. The relationship between investigative groups

6 Malware reversing is rather specialized work, and not all organizations have someone skilled in this practice.

Eventually, the rate at which you discover new leads will slow down. At this point, the IC decides it's time to move on to remediation. Have you learned all there is to learn? Probably not. But you may have learned all you need to know to successfully remove the attacker and protect the data they're after. It can be tough to determine where to draw this line because of all the unknowns involved. Making this decision is a lot like knowing when to stop microwaving a bag of popcorn: when the interval between pops noticeably increases, you should move along before the entire bag burns. Prior to remediating, a lot of planning and coordination needs to happen. Chapter 18 covers this in much more detail.

Digital Forensics at Scale

While this chapter's description of forensic analysis explains the basics, the topic of performing forensics at scale or in difficult environments like cloud deployments (where you may not have the tools you're used to, or your tools may work differently than normal) is a broad one. A series of forensic blog posts from Google covers these topics in more depth.

Keeping Control of the Incident

Once an incident is declared and your team members are assigned responsibilities, the IC's job is to keep the effort running smoothly. This task involves predicting the needs of the response team, and addressing those needs before they become problems. To be effective, the IC should devote all of their time to controlling and managing the incident. If, as IC, you find yourself jumping in to check logs, performing quick forensics tasks, or otherwise involving yourself with operations, it's time to step back and reassess your priorities. If no one is at the helm of the ship, it's guaranteed to veer off course or even crash.

Parallelizing the Incident

Ideally, a practiced IR team can *parallelize* the incident by breaking down all the pieces of the incident response process and running them simultaneously, to the greatest extent possible. If you anticipate an unassigned task or a piece of information you'll need during the incident lifecycle, assign someone to complete the task or prepare for the work. For example, you might not yet be ready to share your forensic findings with law enforcement or a third-party firm that will aid in your investigation, but if you plan to share your findings in the future, your raw investigation notes won't be helpful. Assign someone to prepare a redacted and shareable list of indicators as your investigation proceeds.

It may seem counterintuitive to begin preparations to clean up your environment during the early stages of your forensic investigation, but if you have available staff, it's a great time to assign this task. IMAG allows you to create custom roles at any time, so you assign someone the *remediation lead* (RL) role at any point during the incident. The RL can begin studying the areas of confirmed compromise as the ops team discovers them. Armed with this information, the RL can create a plan to clean up and fix these compromised areas later. When the ops team completes their investigation, the IC will already have the cleanup plan in place. At this point, rather than deciding next steps on the spot, they can initiate the next phase of work. Similarly, if you have the staff available, it's never too early to assign someone to begin the postmortem.

To borrow a programming construct, an IC's role in a large incident looks like a set of steps through a `while` loop:

```
(While the incident is still ongoing):
  1. Check with each of your leads:
    a. What's their status?
    b. Have they found new information that others may need to act upon?
    c. Have they hit any roadblocks?
    d. Do they need more personnel or resources?
    e. Are they overwhelmed?
    f. Do you spot any issues that may pop up later or cause problems?
    g. How's the fatigue level of each of your leads? Of each team?
  2. Update status documents, dashboards, or other information sources.
  3. Are relevant stakeholders (executives, legal, PR, etc.) informed?
  4. Do you need any help or additional resources?
  5. Are any tasks that could be done in parallel waiting?
  6. Do you have enough information to make remediation and cleanup
     decisions?
  7. How much time does the team need before they can give the next
     set of updates?
  8. Is the incident over?
Loop to beginning.
```

The *OODA loop* is another relevant framework for incident-related decision making: a pattern where the responder should *observe*, *orient*, *decide*, and *act*. This is a helpful mnemonic for reminding yourself to consider new information carefully, think about how it applies to the big picture for your incident, decide on a course of action with intent, and only then take action.

Handovers

No human can work nonstop on a single problem for long periods of time without experiencing problems. Most of the time, the crises we encounter take time to resolve. Smoothly passing work back and forth between responders is a must and helps to build a culture of security and reliability (see Chapter 21).

Fighting a large forest fire is physically and emotionally demanding and can take days, weeks, or even months. To fight such a fire, the California Department of Forestry and Fire Protection breaks it into "areas" and assigns an incident commander to each one. The IC for a given area establishes a long-term objective for their area and short-term goals to progress toward the objective. They split their available resources into shifts so that while one team works, another team rests and is ready to relieve the first team when they're exhausted.

When the crews working on the fire are nearing their limits, the IC collects status updates from all the team leads and assigns new team leads to replace them. The IC then briefs the new team leads on the overall objective, their assignment, specific goals they are expected to accomplish, resources available, safety hazards, and any other relevant information. Then the new team leads and their staff take over from the previous shift's staff. Each incoming lead communicates quickly with the outgoing lead to ensure they didn't miss any relevant facts. The fatigued crew can now rest while the firefighting continues.

Security incident response is not as physically demanding as fighting fires, but your team will experience similar emotional and stress-related exhaustion. It's crucial to have a plan to hand their work off to others when necessary. Eventually, a tired responder will start to make mistakes. If worked hard enough for long enough, your team will make more mistakes than they are correcting—a phenomenon often referred to as the *law of diminishing returns*. Staff fatigue isn't just a matter of treating your team kindly; fatigue can cripple your response through errors and low morale. To avoid overwork, we recommend limiting shifts (including IC shifts) to no more than 12 continuous hours.

If you have a large staff and want to accelerate your response, consider splitting your response team into two smaller groups. These teams can staff your response 24 hours a day until the incident is resolved. If you don't have a large staff, you might need to accept the risk of a slower response so your staff can go home and rest. While small teams can work extra-long hours in "hero mode," this mode is unsustainable and yields lower-quality results. We recommend that you use hero mode very sparingly.

Consider an organization with teams in the Americas, Asia-Pacific, and European regions. This kind of organization can staff a "follow-the-sun" rotation so that fresh responders continually cycle on the incident according to a schedule, as depicted in Figure 17-2. A smaller organization might have a similar schedule, but with fewer locations and more time between rotations, or perhaps a single location with half the operations staff working night shifts to keep the response going while the others rest.

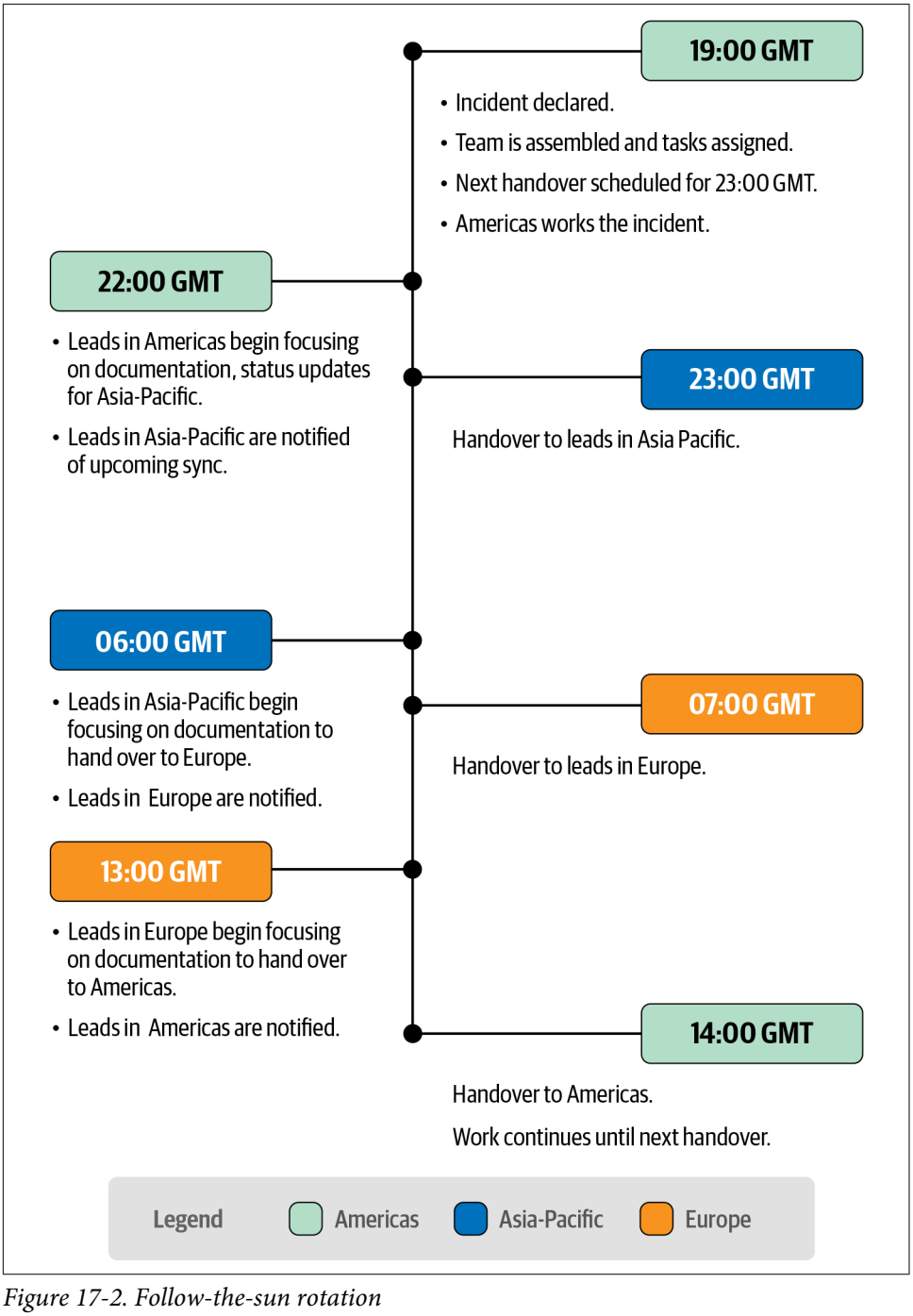

Figure 17-2. Follow-the-sun rotation

The IC should prepare for an incident handover ahead of time. Handovers include updating tracking documentation, evidence notes and files, and any other written records kept during the shift. Arrange handover logistics, time, and method of communication in advance. The meeting should begin with a summary of the current incident state and direction of the investigation. You should also include a formal handover from each lead (ops, comms, etc.) to their corresponding replacement.

The information you should communicate to the incoming team depends on the incident. At Google, we've found it consistently helpful for the IC of the outgoing team to ask themselves, "If I weren't handing this investigation over to you, what would I spend the next 12 hours working on?" The IC of the incoming relief team should have an answer to this question before the handover meeting ends.

For example, a handover meeting agenda might look something like this:

1. [Outgoing IC] Delegate one person to take notes, preferably from the incoming team.
2. [Outgoing IC] Summarize current status.
3. [Outgoing IC] Outline the tasks the IC would do over the next 12 hours if they weren't handing off the incident.
4. [All attendees] Discuss the issue.
5. [Incoming IC] Outline the tasks you expect to handle during the next 12 hours.
6. [Incoming IC] Establish time(s) of next meeting(s).

Morale

During any major incident, the IC needs to maintain team morale. This responsibility is often overlooked, but is critical. Incidents can be stressful, and each engineer will respond differently to these high-pressure situations. Some rise to the challenge and participate enthusiastically, while others find the combination of intense effort and ambiguity deeply frustrating, and would like nothing more than to leave the scene of the incident and go home.

As an IC, don't forget that motivating, encouraging, and keeping track of the general emotional state of your team are key factors in achieving a positive incident outcome. Here are some tips for maintaining morale in the midst of a crisis:

Eat

As your response team pushes itself to make progress, hunger will invariably strike. This decreases the team's effectiveness and can put people on edge. Plan in advance to take breaks and bring in food whenever possible. This helps keep the team happy and focused when you need them to be.

Sleep
: The law of diminishing returns applies to humans, too. Your staff will become less effective over time as exhaustion sets in and each member passes their own peak fatigue point. After this point, it's possible that continued work will result in more mistakes than progress, setting the incident response back. Keep watch for fatigue in your responders, and be sure that they are allowed rest periods as needed. Leaders may even need to intervene to ensure that people who don't recognize their own need for rest still take breaks.

Destress
: When the opportunity presents itself—for example, if the team is waiting for a file array to finish rebuilding and can't make parallel progress—gather everyone together for a destressing activity. Several years ago, during a particularly large and lengthy incident at Google, the response team took a one-hour break to smash a failed hard drive using a hammer and some liquid nitrogen. Years later, the team involved recalls that activity as a highlight of the response.

Watch for burnout
: As an IC, you should actively watch for signs of burnout in your team (and yourself). Is a critical engineer starting to become more cynical and defeatist? Are staff expressing fears that the challenge is unwinnable? This may be the first time they've had to handle a major incident, and they may have a lot of fears. Talk with them frankly and make sure they understand your expectations and how they can meet those expectations. If a team member needs a break and you have someone to replace them, offer a relief period.

Lead by example
: A realistic but positive outlook openly expressed by a leader goes a long way toward setting your team's expectations around their success. This kind of outlook encourages the team and makes them feel like they can achieve their goal. Additionally, members of a response team may be skeptical that taking time for self-care (e.g., eating and sleeping adequately) is truly encouraged until they see the IC or OL openly doing so as a best practice.

Communications

Of all the technical issues involved in incident response, communication remains the most challenging. Even in the best of circumstances, effective communication with coworkers and others outside your immediate group can be difficult. When subjected to stress, tight deadlines, and the high stakes of a security incident, these difficulties can intensify and quickly lead to a delayed or missed response. Here are a few major communications challenges you might encounter, and tips for managing them.

Many excellent books cover the topic of communications thoroughly. For a deeper understanding of communications, we recommend Nick Morgan's *Can You Hear Me?* (Harvard Business Review Press, 2018) and Alan Alda's *If I Understood You, Would I Have This Look on My Face?* (Random House, 2017).

Misunderstandings

Misunderstandings between the responders will likely make up the bulk of your communication problems. People who are used to working together may make assumptions about things that haven't been said. People who are not used to working together may use unfamiliar jargon, acronyms that mean different things to different teams, or assume a common frame of reference that isn't actually common.

For example, take the phrase "We can turn the service back on when the attack is mitigated." To the product team, this may mean that as soon as the attacker's tools are deleted, the system is safe to use. To the security team, this may mean that the system isn't safe to use until a complete investigation has concluded that the attacker is no longer able to exist in, or return to, the environment.

When you find yourself handling an incident, as a rule of thumb, it's helpful to always *be explicit and overcommunicate*. Explain what you mean when you ask for something or set an expectation, even if you think the other person ought to know what you mean. In the example from the previous paragraph, it couldn't hurt to say, "We can turn the service back on when we are confident that all paths back in have been fixed, and that the attacker is no longer able to access our systems." Keep in mind that the responsibility of communicating usually belongs to the communicator—only they can make sure the people they're communicating with receive the intended message.

Hedging

Hedging is another common communication mistake. When people are expected to give advice in a stressful situation but aren't feeling confident, they often tend to add qualifiers to their statements. Avoiding expressing certainty about an uncertain situation by saying something like "We're pretty sure we found all the malware" might feel safer, but hedging often muddies the situation and leads to uncertainty among decision makers. Again, being explicit and overcommunicating is the best remedy here. If your IC asks if you've identified all the attacker's tools, "We're pretty sure" is a weak answer. It would be better to answer, "We are sure about our servers, NAS, email, and file shares, but aren't feeling confident about our hosted systems because we have less log visibility there."

Meetings

To make progress on an incident, you'll need to get people together to coordinate their efforts. Regular, rapid syncs with key players in the incident response are a particularly effective way to maintain control and visibility of everything that's happening. The IC, the OL, a company attorney, and a few key executives might meet every two or four hours to make sure the team can quickly adapt to any new developments.

We recommend aggressively limiting attendees of these meetings. If you have a room or videoconference full of people, and only a handful are speaking while the rest listen or check email, your invite list is too large. Ideally, the incident leads should attend these meetings while everyone else works on tasks that need to be completed. Once the meeting ends, the leads can update their respective teams.

While meetings are often a necessary part of working together, improperly managed meetings risk derailing your progress. We strongly recommend that an IC open every meeting with a set agenda they've planned in advance. This may seem like an obvious step, but it can be easy to forget during the adrenaline rush and stress of an ongoing incident. Here's an example agenda used at Google for a security incident kickoff meeting:

1. [IC] Delegate one person to take notes.
2. [IC] All attendees introduce themselves, starting with the IC:
   a. Name
   b. Team
   c. Role
3. [IC] Rules of engagement:
   a. Do you need to take confidentiality into account?
   b. Are there specific operational security concerns to consider?
   c. Who owns making decisions? Who should be included in decision-making processes? Who should *not* be included?
   d. Let your lead know if you're switching tasks/finished doing something.
4. [IC] Describe current status: what problem are we solving?
5. [All] Discuss the issue.
6. [IC] Summarize actions and owners.
7. [IC] Ask the group:
   a. Are there any more resources we need?
   b. Are there any other groups we need to involve?
   c. What roadblocks are there?

8. [IC] Establish time of next sync meeting and expected attendance:
    a. Who is required?
    b. Who is optional?

And here's one for an in-progress sync meeting:

1. [IC] Delegate one person to take notes.
2. [IC or ops lead] Summarize current status.
3. [IC] Receive updates from each attendee/thread of activity.
4. [IC] Discuss next steps.
5. [Ops lead] Assign tasks, and get each person to repeat what they think they're going to do.
6. [IC] Establish time(s) of next meeting(s).
7. [Ops lead] Update action tracker.

In both example agendas, the IC assigns a note taker. Through experience, we've learned that a note taker is essential for keeping your investigation on track. Leads are busy dealing with all the issues that arise in the meeting and won't have the bandwidth to take good notes. During an incident, you often forget items you *think* you'll remember. These notes also become invaluable when writing your postmortem. Keep in mind that if the incident has any legal ramifications, you'll want to consult with your legal team on the best way to manage this information.

Keeping the Right People Informed with the Right Levels of Detail

Figuring out the right level of detail to communicate is a major challenge for incident communications. Multiple individuals at multiple levels will need to know *something* related to the incident, but for OpSec reasons, they likely cannot or should not know *all* the details. Be aware that employees who are only vaguely aware of the incident are likely to engage in gap-filling.[7] Rumors among employees can quickly become erroneous and damaging when relayed outside the organization.

A long-running investigation may involve frequent updates with different levels of detail to the following people:

Executives and senior leadership
: They should receive brief, succinct updates on progress, roadblocks, and potential consequences.

7 If an individual doesn't have access to the actual information, they're inclined to make something up.

The IR team
: This team requires up-to-date information about the investigation.

Organizational personnel not involved in the incident
: Your organization may need to decide what to tell personnel. If you tell all staff about the incident, you may be able to solicit their assistance. On the other hand, these people may spread information further than you intended. If you don't tell organizational personnel about an incident but they find out anyway, you may need to deal with the rumor mill.

Customers
: Customers may be legally entitled to be informed of the incident within a set period of time, or your organization may choose to voluntarily notify them. If a response involves shutting off services visible to customers, they may have questions and demand answers.

Legal/justice system participants
: If you escalated the incident to law enforcement, they may have questions and may begin requesting information you didn't originally intend to share.

To help manage all these demands without constantly distracting the IC, we recommend appointing a *communications lead* (CL). The core job of the CL is to stay informed about the incident as it unfolds and to prepare comms to the relevant stakeholders. The key responsibilities of the CL include the following:

- Working with sales, support, and other internal partner teams to answer any questions they may have.
- Preparing briefs for executives, legal, regulators, and others with oversight roles.
- Working with press and PR to ensure that people have the right information to make accurate and timely statements about the incident when necessary. Ensure that people outside of the response team don't make conflicting statements.
- Keeping constant and careful watch on the spread of information about the incident, so the incident staff respect any "need to know" guidelines.

The CL will want to consider reaching out to domain experts, external crisis communication consultants, or anyone else they need to help manage information related to the incident with minimal delay.

Putting It All Together

This section ties together the contents of this chapter by walking through a hypothetical response to a compromise that an organization of any size might encounter. Consider a scenario where an engineer discovers that a service account they don't recognize has been added to a cloud project they haven't seen before. At noon, they

escalate their concerns to the security team. After a preliminary investigation, the security team determines that an engineer's account has likely been compromised. Using the advice and best practices presented earlier, let's walk through how you might respond to such a compromise from beginning to end.

Triage

The first step in your response is to triage. Start with a worst-case assumption: the security team's suspicions are correct that the engineer's account is compromised. An attacker using privileged access to view sensitive internal information and/or user data would be a serious security breach, so you declare an incident.

Declaring an Incident

As the incident commander, you notify the rest of the security team of the following:

- An incident has occurred.
- You will be assuming the role of IC.
- You'll need additional support from the team to investigate.

Communications and Operational Security

Now that you've declared the incident, other people in the organization—executives, legal, etc.—need to know an incident is in progress. If the attacker compromised your organization's infrastructure, emailing or chatting with these people may be risky. Follow operational security best practices. Suppose your contingency plan calls for using an organization credit card to register a business account in a cloud-based environment not associated with your organization, and to create accounts for each person involved in the incident. To create and connect to this environment, you use freshly rebuilt laptops not connected to your organization's management infrastructure.

Using this new environment, you call your executives and key legal staff to advise them on how to obtain a secure laptop and a cloud account so they can participate via email and chats. Since all the current responders are local to the office, you use a nearby meeting room to discuss the particulars of the incident.

Beginning the Incident

As the IC, you need to assign engineers to investigate, so you ask an engineer on the security team with forensics experience to be the operations lead. Your new OL starts their forensic investigation immediately and recruits other engineers as needed. They begin by collecting logs from the cloud environment, focusing on the time period

when the service account credentials in question were added. After confirming that the credentials were added by the engineer's account during a time period when the engineer was definitely out of the office, the forensics team concludes that the account has been compromised.

The forensics team now pivots from investigating only the suspect account to investigating all other activities around the time the account was added. The team decides to collect all the system logs relating to the compromised account, as well as that engineer's laptop and workstation. The team determines that this investigation could take a single analyst quite some time, so they decide to add more staff and distribute the effort.

Your organization doesn't have a large security team, so you don't have enough skilled forensic analysts to adequately distribute the forensic task. However, you do have system administrators who understand their systems well and who can help analyze logs. You decide to assign these sysadmins to the forensics team. Your OL contacts them via email with a request to "discuss a few things" over a phone call, and briefs them fully during that call. The OL asks the sysadmins to collect all logs related to the compromised account from any system in the organization, while the forensics team analyzes the laptop and desktop.

By 5 p.m. it's clear that the investigation is going to last much longer than your team can continue working. As the IC, you correctly anticipate that your team will become fatigued and start to make mistakes before they can resolve the incident, so you need to come up with a handover or continuity plan. You notify your team that they have four hours to complete as much analysis of the logs and hosts as possible. During this time, you keep leadership and legal up to date and check in with the OL to see if their team needs additional help.

At the 9 p.m. team sync, the OL reveals that their team has found the attacker's initial entry point: a very well-crafted phishing email to the engineer, who was tricked into running a command that downloads the attacker's backdoor and establishes a persistent remote connection.

Handover

By the 9 p.m. team sync, many of the engineers working on the problem have been at work for 12 hours or longer. As a diligent IC, you know that continuing to work at this pace is risky, and that the incident will require a lot more effort. You decide to hand off some of the work. While your organization doesn't have a full security team outside of the main San Francisco office, you have an engineering office in London with some senior staff.

You tell your team to take the next hour to finish documenting their findings while you contact the London team. A senior engineer in the London office is appointed as

the next IC. As the outgoing IC, you brief the replacement IC on everything you've learned so far. After receiving ownership rights on all the incident-related documentation and making sure that the London team understands the next step, the London IC acknowledges that they are in charge until 9:00 PST the next morning. The San Francisco team is relieved and sent home to rest. Overnight, the London team continues the investigation, focusing on analyzing the backdoor scripts and actions performed by the attacker.

Handing Back the Incident

At 9:00 the next morning, the San Francisco and London teams hold a handover sync. Overnight, the London team made lots of progress. They determined that the script run on the compromised workstation installed a simple backdoor, enabling the attacker to log in from a remote machine and start looking around the system. Noticing that the engineer's shell history included logins to cloud service accounts, the adversary took advantage of the saved credentials and added their own service account key to the list of administrative tokens.

After doing so, they took no further action on the workstation. Instead, cloud service logs show that the attacker interacted directly with the service APIs. They uploaded a new machine image and launched dozens of copies of that virtual machine in a new cloud project. The London team hasn't yet analyzed any of the running images, but they audited all credentials in all existing projects and confirmed that the malicious service account and API tokens they know about are the only credentials that can't be verified as legitimate.

With this update from the London team, you acknowledge the new information and confirm that you are taking over as the IC. Next, you distill the new information and provide a concise update to executive leaders and legal. You also brief your team on the new findings.

Although you know that the attacker had administrative access to production services, you don't yet know whether user data was at risk or affected. You give your forensics team a new high-priority task: look at all other actions the attacker may have taken against existing production machines.

Preparing Communications and Remediation

As your investigation proceeds, you decide that it's time to parallelize a few more components of your incident response. If the attacker potentially accessed your organization's user data, you may need to inform users. You also need to mitigate the attack. You choose a colleague who is a strong technical writer to be the communications lead (CL). You ask one of the system admins who isn't working on forensics to become the remediation lead (RL).

In collaboration with the organization's attorney, the CL drafts a blog post that explains what happened and the potential customer impact. Although there are many blanks (such as "<*fill this in*> data was <*fill this in*>"), having the structure ready and approved ahead of time helps you to communicate your message much faster when you know the full details.

Meanwhile, the RL makes a list of every resource the organization knows to have been affected by the attacker, along with proposals for how to clean up each resource. Even if you know that the engineer's password wasn't exposed in the initial phishing email, you'll need to change their account credentials. To be on the safe side, you decide to guard against backdoors that you haven't yet found but that may appear later. You make a copy of the engineer's important data and then erase their home directory, creating a new one on a freshly installed workstation.

As your response progresses, your team learns that the attacker didn't access any production data—to everyone's great relief! The additional VMs the attacker launched appear to be a swarm of coin mining servers that mine digital currency and direct the funds to the attacker's virtual wallet. Your RL notes that you can delete these machines, or you can snapshot and archive the machines if you wish to report the incident to law enforcement later on. You can also delete the project the machines were created in.

Closure

Around mid-afternoon, your team has run out of leads. They've searched every resource the malicious API keys may have touched, built their mitigation document, and confirmed that the attacker didn't touch any sensitive data. Fortunately, this was an opportunistic coin mining exercise and the attacker wasn't interested in any of your data—just a lot of compute capability on someone else's bill. Your team decides that it's time to execute your remediation plan. After checking in with legal and leadership to ensure that the decision to close out the incident has their approval, you signal the team that it's time to act.

Freed from their forensic tasks, the operations team now divides up the tasks from the remediation plan and completes them as quickly as possible, ensuring that the attacker is shut off quickly and completely. They then spend the rest of the afternoon writing down their observations for a postmortem. Finally, you (the IC) hold an outbrief, where everyone on the team has the opportunity to discuss how the incident went. You clearly communicate that the incident is closed and no longer requires an emergency response. Your last task before everyone goes home to rest is to brief the London team so they also know that the incident is completed.

Conclusion

Incident management, when scaled, becomes its own art that is distinct and separate from general project management and smaller incident response work. By focusing on processes, tools, and proper organizational structures, it's possible to staff a team that can effectively respond to any crisis at the speed today's market requires. Whether you're a small organization whose engineers and core team become a temporary response team as needed, a giant-scale organization with response teams across the globe, or anything in between, you can apply the best practices described in this chapter to effectively and efficiently respond to a security compromise. Parallelizing your incident response and forensic work and professionally managing the team using ICS/IMAG will help you respond scalably and reliably to any incidents that arise.