Brought to you by:

corelight

Open NDR

for dummies®
A Wiley Brand

Understand your network

Respond to attacks

Protect your organization

Corelight Special Edition

Alan Saldich

About Corelight

Corelight gives defenders unparalleled insight into networks to help them protect the world's most critical organizations and companies. Corelight's global customers include Fortune 500 companies, major government agencies, and large research universities. Based in San Francisco, Corelight is an open-core security company founded by the creators of Zeek, the widely used network security technology. For more information, visit www.corelight.com or follow @corelight_inc.

Open NDR

Open NDR

Corelight Special Edition

by Alan Saldich

for dummies®
A Wiley Brand

Open NDR For Dummies®, Corelight Special Edition

Published by
John Wiley & Sons, Inc.
111 River St.
Hoboken, NJ 07030-5774
www.wiley.com

ISBN 978-1-119-81796-3 (pbk); ISBN 978-1-119-81797-0 (ebk)

Manufactured in the United States of America

10 9 8 7 6 5 4 3 2 1

For general information on our other products and services, or how to create a custom *For Dummies* book for your business or organization, please contact our Business Development Department in the U.S. at 877-409-4177, contact info@dummies.biz, or visit www.wiley.com/go/custompub. For information about licensing the *For Dummies* brand for products or services, contact BrandedRights&Licenses@Wiley.com.

Publisher's Acknowledgments

Some of the people who helped bring this book to market include the following:

Project Editor: Elizabeth Kuball
Acquisitions Editor: Ashley Coffey
Editorial Manager: Rev Mengle
Business Development Representative: Cynthia Tweed
Production Editor: Mohammed Zafar Ali
Special Help: Nicole Sholly, John Gamble, Charles Strauss, Kylie Heintz

Table of Contents

Introduction

Networks are the veins of modern organizations, and data is the lifeblood that flows through them. Networks carry all the applications and data required to operate in today's digital world, but it wasn't long ago that interconnecting computers was a new concept (sneaker-net anyone?).

Networking has transformed enterprises, but it has also opened a pathway for criminal groups, corporate insiders, and nation-states to easily traverse an organization in search of critical data such as intellectual property, personal information, financial data, credit card numbers, health records, and more. Virtually all cyberattacks require getting from point A to point B, and these points are connected in almost all cases by a network.

Networks create an unavoidable vulnerability, but when properly monitored they also provide a great source of evidence for investigation when you're attacked. According to author and security expert Richard Bejtlich, "The defender only needs to detect one of the indicators of the intruder's presence in order to initiate incident response within the enterprise" (`https://taosecurity.blogspot.com/2009/05/defenders-dilemma-and-intruders-dilemma.html`).

Defending an organization *is* a tough job, but often all you need is *one clue* that something looks weird or has gone awry to lead you to the path to unravel an attack. Finding that one clue isn't easy, by any means, but attackers are human — they aren't perfect, and they make mistakes. Just like the burglar who leaves behind a single fingerprint, sometimes a single clue is all you need.

That's the fundamental idea of network detection and response (NDR): To find an intruder, you need to be collecting evidence all the time, and one of the best sources for that evidence is the traffic in your network. If you aren't monitoring your network 24/7 and keeping the relevant evidence on hand for months (or ideally, years), your *blue team* (network defenders, incident responders, and threat hunters) will be at a huge disadvantage when they're called upon to resolve a serious security incident or to embark on proactive threat-hunting missions.

Network data is ground truth. Unlike applications or servers whose data can be overwritten, network traffic contains innumerable clues about people, devices, applications, and data that are critical to successful incident response and threat hunting. Attackers use a wide variety of techniques to try to hide in the network, but ultimately they can't avoid pushing packets across the wire, leaving behind an immutable record of their activity. If you're there, ready to capture that record and turn it into evidence, you and your team will have the high ground.

About This Book

You'll find this book important if you want to

- Improve the speed and operational effectiveness of your security operations team with better data
- See how data-driven, data-first approaches to enterprise security can transform incident response and threat hunting
- Understand why metadata and other information extracted from your network is instrumental for incident response and threat hunting
- See why traditional network monitoring tools like NetFlow and packet capture (PCAP) don't give your security teams the information they need to do their jobs
- Enable your security teams to understand the context around incidents by providing them with the evidence they need to conclude investigations
- Give your blue team the data they need to enable effective threat hunting

Foolish Assumptions

I assume you're a security professional in business, government, or academia. This isn't an implementation guide — it's a book for security managers, architects, and others who may be deeply familiar with many aspects of modern enterprise security. At the same time, you may not be as well versed in network security

monitoring (also referred to as network traffic analysis, network detection and response, network analysis and visibility, and a host of other overlapping terms). You also may not be familiar with the open-source project called Zeek (which was known as Bro from 1995 to late 2018). If that's you, you're in the right place.

Icons Used in This Book

Throughout this book, I occasionally use icons to call out important information. Here's what to expect:


TIP

Tips are appreciated, never expected, and I sure hope you'll appreciate these useful nuggets of information.


REMEMBER

This icon points out important information you should commit to your nonvolatile memory, your gray matter, or your noggin — along with anniversaries and birthdays!


WARNING

These alerts point out the stuff your mother warned you about. Well, probably not, but they do offer practical advice to help you avoid potentially costly or frustrating mistakes.

Where to Go from Here

If you like what you read in this book, visit `www.corelight.com` for more information about its Open NDR solutions, arrange for a demo, or learn how Corelight can help organizations like yours improve their security operations.

IN THIS CHAPTER

- Learning the history of network monitoring and Zeek (formerly Bro)
- Understanding why perimeter-based security is no longer sufficient
- Exploring the power and importance of "ground truth"
- Digging into the "SOC Triad" and NDR

Chapter 1
Building Resilient Security with Open NDR

As soon as organizations started using networks for operational purposes and sharing data that had operational, technical, or personal value, bad actors started to try to exploit them. Very early on, monitoring networks to see what was going on seemed like a pretty good idea. What does "monitoring" mean, exactly?

Is it keeping a copy of all the network traffic? Would monitoring specific links be enough? Maybe just logging data from routers and switches? How about focusing on specific ports or protocols?

For a modern enterprise with thousands of employees, dozens or hundreds of sites, and thousands of servers, routers, switches, databases, and other infrastructure, monitoring networks for mischief gets complicated quickly. Add smartphones, cloud, bring your own device (BYOD), Internet of Things (IoT), working from home (WFH), software as a service (SaaS), and a host of other acronyms, and it seems impossible. But it doesn't have to be.

In this chapter, I explain the basics of a network monitoring–based approach as the foundation for enterprise security and discuss why the strategy of keeping attackers out is hopeless.

Why Can't We Just Keep the Bad Guys Out?

Enterprise security is a never-ending battle, one that gets more difficult every year as attackers get more sophisticated, invent new techniques, and build new tools and malware. Of course, the easiest approach is to just keep them out in the first place! After all, thousands of companies deliver firewalls (and advanced firewalls!), intrusion detection, intrusion prevention, application monitoring, end-point detection products, threat intelligence, end-user detection, and many more.

Yet despite decades of work and billions of dollars spent on security solutions every year by people like you, cyberattacks are constant, and many of them are successful.

Attackers are creative, smart, determined, often highly motivated, and often well funded. Whether it's industrial espionage, disgruntled employees, nation-states searching for strategic or military advantage, or just a hacker looking for a challenge, they all pose serious problems for blue teams who are charged with defending their organizations.

"Just keeping them out" isn't realistic for several reasons:

- Too many attacks generate too many alerts, swamping more security teams.
- Too many alerts mean lots of false alarms, incidents that aren't run to ground completely, or attacks that are missed.
- Blue teams (defenders) often don't have the data they need to understand security incidents quickly and accurately.
- Attackers are adept at hiding in plain sight, using "normal" applications, traffic, and tools to move around networks.
- After breaching an organization, attackers often lay low for weeks or months, before they act, making them challenging to detect.

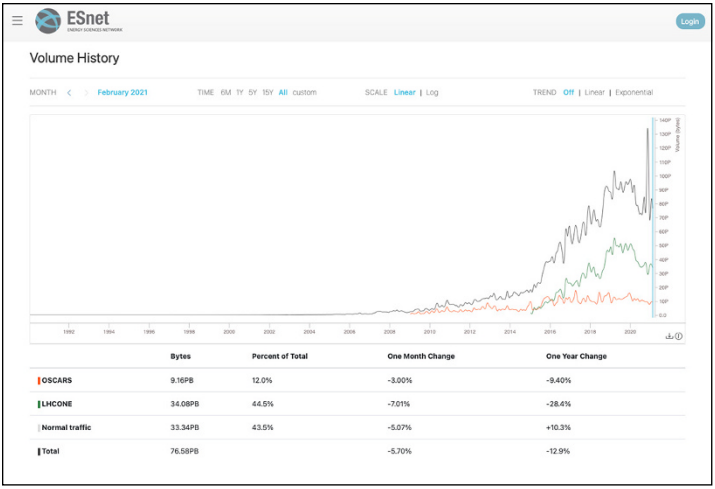
The Birth of Zeek (Formerly Bro)

Put yourself in 1995 (if you're old enough). In the technology world, that wasn't long after the introduction of NCSA Mosaic (the world's first web browser) in 1993. Believe it or not, many companies were just adopting email widely and were still trying to figure out what do with a "web server."

Remote offices were often barely connected to their headquarters by computer networks. If they were, those connections were usually low-bandwidth dedicated telephone lines, ISDN, fractional T1 lines, or just dial-up. Fax machines were still in wide use, and two-way paging was cutting-edge! The iPhone and cloud services like Amazon Web Services were still more than a decade away.

In that era, cyberattacks against businesses were still pretty rare. Many companies were just beginning their digital transformations, so it was harder to penetrate them and steal much of consequence over networks. And because there weren't many hosts online, monitoring them was easier in many ways, even though the tools were primitive.

The year 1995 looked very different at UC Berkeley. A grad student named Vern Paxson was working at Lawrence Berkeley Lab (LBL, now known simply as the Berkeley Lab), a Department of Energy (DoE) facility in the Berkeley hills overlooking San Francisco Bay. He was monitoring the massive networks that interconnect the various labs run by the DoE which include basic research labs like LBL, but also the nuclear weapons complex with famous facilities like Los Alamos, Sandia, and Oak Ridge. They were, and still are, interconnected by the Energy Sciences Network (ESnet), which is usually the largest network in the world at any given time. Figure 1-1 shows a chart of traffic growth on ESnet since the early 1990s. Back then, traffic was negligible compared to today, but it was growing exponentially!



Source: https://my.es.net/traffic-volume?scale=linear

FIGURE 1-1: 1990s traffic was low but growing fast.

Paxson developed a tool to monitor the traffic in support of experiments at LBL. He named the tool "Bro," which was an allusion to Big Brother in George Orwell's novel *1984*. Because Bro was so powerful and enabled Paxson to see the network traffic in so much detail, he immediately recognized the potential for the misuse or abuse of this powerful monitoring tool. (Bro was renamed Zeek in 2018. In this book, I use Zeek, even though it was known as Bro for more than 20 years. If you see Zeek in this book, know that it refers to the open-source project, which is distinct from products like Corelight's that incorporate Zeek into their solutions.)

Why Perimeter-Based Security Is No Longer Sufficient

LBL, like many labs and universities, was a vastly different environment compared to most businesses, even large enterprises, at the time.

For one thing, it was a research facility in a large public university that worked cooperatively with other scientific labs around the world. Roughly 25 percent of their students turned over every year, and the lab had scientists coming and going constantly.

They would, of course, be using their own computers and applications, so controlling those in the way a business might was inconceivable. The scientists and students at the lab would be doing all manner of experiments, from high-energy physics to energy conservation to biology, and much of that was unpredictable. The data sets could be massive, with things like the Advanced Light Source (a particle accelerator) generating terabytes or petabytes of data regularly. There was no way to enforce the use of specific devices, applications, or data sets.

This environment required something different from a perimeter-based security approach that was the default at the time.

Zeek evolved under the harsh conditions at LBL, as well as some powerful trends in computing that resulted in its unique characteristics:

- **Zeek is nonjudgmental.** Because LBL didn't control the applications and data that users implemented, a signature-based system was a nonstarter. If you don't know what to look for, then there's no known signature to trigger an alert. The underlying philosophy of Zeek is to neutrally observe and record network traffic, not to try to decide whether a particular packet, IP address, DNS query, or file is good or bad, malicious or benign.
- **Zeek is a real-time event processing engine.** Zeek isn't a static filter or list of things to watch out for. An *event,* however, can be defined in many ways, so over the last 25 years, Vern and the subsequent developers who built Zeek have created many different packages (sometimes called *scripts*) to extract different elements of data, metadata, or observed behavior from network traffic. Zeek continues to evolve today as new protocols, applications, and behaviors become common.
- **Zeek logs are the output of that engine.** There are dozens of "standard" Zeek logs, hundreds or possibly thousands if you count custom or community-contributed scripts. Each Zeek log is the product of a particular script — for example, scripts and logs for email traffic (Simple Mail Transfer Protocol [SMTP] log), Transmission Control Protocol (TCP) connections (connection log), web pages (Hypertext Transfer Protocol [HTTP] log), Microsoft Office traffic (Server Message Block [SMB] log), and dozens of others, including a "Weird log" to catch odd or unknown traffic.

- **Zeek logs are compact, curated, structured, and interconnected.** Those attributes relate to the data collected in Zeek logs:
  - **Compact:** Typically, in aggregate, all the data contained in Zeek logs is roughly 1 percent of the volume of the monitored traffic. If you were monitoring a 10 Gbps link, you would expect roughly 100 Mbps of Zeek logs. That's still a lot of data, but their relatively compact size means the logs can be kept for many months or even years.
  - **Curated:** Zeek extracts only relevant data that's necessary for security investigations. Unlike packet capture (PCAP) solutions, Zeek is producing a curated set of data, designed by and for incident responders and threat hunters.
  - **Structured:** Zeek logs can be imported into virtually any security analytics stack (typically, security and information event management [SIEM]) in a variety of formats.
  - **Interconnected:** Zeek logs contain key fields that are common across many logs (for example, they share a common time stamp [ts] and the unique ID [UID] that I discuss later, among others). That allows incident responders to pivot quickly from the conn log with data about a TCP connection, to the HTTP log to see relevant web traffic, to the SMTP log to investigate concurrent email exchanges, and so on.
- **Zeek is an open-source project.** It has been built and improved over many years by a team of people across various companies and academia. So, different organizations have developed their own packages to extract data or generate alerts. (See `www.zeek.org` for more information about it.)

With this high-level understanding of Zeek's origins, you can see how it's different from other security technologies designed to detect and stop intruders. Zeek users recognize that "signature-based" strategy isn't 100 percent reliable and attackers will be successful in penetrating an organization, and by default its network.

REMEMBER

The challenge for defenders is collecting the right evidence needed to investigate a suspected attack or breach, whether it happened just now, or a year ago. Regardless of the recency, incident responders need the right data to assess the damage, and to decide what to do about it. They need to know the "ground truth," and the best place to get that is from network data.

The Meaning, Value, and Enduring Value of "Ground Truth"

Attackers are crafty and motivated. They're adept not only at evading detections by hiding in normal traffic and using admin tools to move around, but also at obscuring their presence by overwriting device logs that recorded what they did.

A Zeek sensor is deployed *out-of-band,* which means that when you install one, you're connecting it to a tap in your network (usually via a packet broker or TAP/SPAN port) and then it's ingesting a copy of the network traffic. Why is that important? Because then attackers who are traversing your organization's network have no way of knowing Zeek is watching, and no way to evade it. Your Zeek sensor is always evaluating a live copy of the network traffic, not the live in-path traffic where the sensor could be detected and possibly avoided.

If an attacker accesses a file server by traversing a network that's being monitored by Zeek, you'll automatically have collected a record of their activity. An intruder has no way to go back later and overwrite a file or otherwise hide their actions. Zeek has collected the "ground truth" that can't be changed.

TIP

Often people describe what Zeek does as being a bit like a flight data recorder (FDR) on an airplane. When something goes awry, the first thing investigators do is recover that device. Why? Because during the flight, it was recording the relevant data that would be needed in an investigation — things like: Airspeed, throttle position, engine speed, aileron and flap angles, rudder angles, fuel level, altitude, attitude and other critical data.

The FDR isn't recording what everyone on the plane was wearing, what movies were shown, or what each passenger ate for the in-flight meal; it's recording only the relevant data required to

piece together what happened in the event of an accident or other mishap. Unfortunately, using that metaphor leads people to think about plane crashes, so I use it judiciously.

The SOC Visibility Triad

In 2019, Gartner published a paper entitled "Applying Network-Centric Approaches for Threat Detection and Response" (`www.gartner.com/en/documents/3904768/applying-network-centric-approaches-for-threat-detection`). In that paper, they outline the idea of the SOC Visibility Triad. The basic concept is that for an enterprise security operations center (SOC) to have a complete picture of the security environment, it needs three key elements:

- Endpoint detection and response (EDR)
- Network detection and response (NDR)
- Security incident and event management (SIEM)

REMEMBER

Network security monitoring has been around for decades. In the 2010s, security professionals rightly decided there was a tremendous amount of security information in endpoints (PCs, laptops, servers, and other hosts). So, the security pendulum swung hard, with several new EDR vendors emerging that still provide powerful tools for blue teams today.

But, like all pendulums, it eventually swung back. EDR data is valuable and essential, but it isn't nearly enough to paint the complete picture. One simple challenge is that instrumenting *every single endpoint* isn't possible. In a large enterprise or government agency with tens or even hundreds of thousands of employees, complex IT infrastructure, instrumenting every endpoint is challenging and often impossible.

To complete the picture, Gartner (and many other security analysts) recommend monitoring networks as well. The nice thing about monitoring networks is that if you pick your sensor locations strategically, you can get a complete picture of activity in each segment with one sensor (as opposed to trying to deploy many thousands of endpoint monitoring solutions).

TIP

Zeek sensors can be deployed to monitor *north–south traffic* (typically traffic entering or exiting your organization at an egress point or primary Internet connection) or *east–west traffic* (usually defined as internal traffic to/from or within data centers or other "on-premises" locations), or h*igh-value locations* (like specific research labs, high-performance computing, specific applications or databases, and so on).

The two systems combined — EDR and NDR — provide a more complete picture, and all that data is typically ingested into and stored in a SIEM platform (or data lake). Although you have many ways to monitor network traffic, Zeek has been the standard for security monitoring at the world's largest, most sensitive, and most-attacked networks for many years.

They're Already In: What Can You Do?

You have to assume your organization will be breached or that it already has been. In that case, you may be wondering how an NDR solution built on Zeek would help. NDR is a somewhat new category name, but it certainly has several predecessors, and they overlap. Network security monitoring (NSM), network traffic analysis (NTA), network analysis and visibility (NAV), and network intrusion detection systems (NIDS) have all been used over the years to describe the basic functionality of monitoring networks for security purposes.

REMEMBER

Zeek produces efficient compact logs that provide the evidence that a security professional needs to investigate an incident quickly. But that's only true if your team has deployed sensors and has been collecting and keeping logs from some time before the relevant breach started.

That's one of the most powerful capabilities of Zeek: It's a simple step any organization can take to start collecting that security evidence, in preparation for that day in the future when you'll be grateful you have that critical data.

If you're fortunate enough to have Zeek logs available going back months or years, then when you hear about a new indicator of compromise (IOC) that emerges from threat intelligence services or government agencies, the first step is a simple one:

Just search your SIEM platform or data lake for those IOCs in the Zeek logs. Whether you use Backstory, Databricks, Devo, Elastic, Hadoop, Splunk, Sumo, or something else, if you have the logs then searching for specific strings is quick.

And, more important, your team can start pulling on threads as they come across clues of malicious behavior. Maybe the IOC is the signature of a particular piece of malware. If a search shows that malware first showed up in your network seven months ago via a phishing attack, you may wonder what websites were visited just before that.

Then you may notice that in addition to the specific piece of malware you're searching for, other files that arrived at the same time are visible. What are those files? Who downloaded them? Where did they come from? Why doesn't the file extension match the file type? Why was that SMB traffic on an unexpected port?

An investigator can answer these questions quickly with Zeek logs. There's simply no better tool for security teams when it comes to understanding what happened on your network.

IN THIS CHAPTER

- Learning key capabilities of Open NDR built on Zeek and Suricata
- Comparing Open NDR to other network monitoring approaches
- Looking at the UID, the FUID, and other pivot points in Zeek logs
- Integrating Zeek logs with Suricata Alerts and PCAP

Chapter 2
Using Open NDR in Your Enterprise

This chapter covers the advantages of Open NDR — because of its inherent characteristics — versus other types of network monitoring approaches. It also dives into some of the finer points of Zeek logs, integrating data with Suricata Alerts and packet capture (PCAP), and the 2020 SolarWinds/SUNBURST breach.

THE 2020 SOLARWINDS/SUNBURST BREACH: A CASE STUDY FOR ZEEK

In mid-December 2020, it was announced that a major security breach had occurred. A year earlier, in the fall of 2019 (or earlier), attackers had infected the widely used Orion software of the infrastructure monitoring vendor SolarWinds. The attack was called SUNBURST. In the spring of 2020, the attackers used the SolarWinds normal software update mechanism to spread malware across the networks of thousands of companies and government agencies.

The attackers were sophisticated. Not only did they execute a successful supply-chain attack, but after their malware was deployed at the

(continued)

(continued)

targets, it lay dormant for a period of several weeks to ensure it hadn't been detected. This was a well-planned attack that caused extensive damage, the extent of which still wasn't completely understood at the time of publication of this book.

Stopping such a novel attack is exceedingly difficult, if not impossible. However, organizations that deployed an Open NDR solution based on Zeek had a huge advantage in understanding the scope and severity of the attack. In fact, one of the senior consultants at Mandiant who unraveled the attack mentioned on Twitter (`https://twitter.com/srunnels/status/1338329916304199680`) his use of Zeek during the investigation:

> We leveraged a *lot* of tech and this investigation only solidified my belief that a [network security monitoring] stack isn't complete without Zeek. Obfuscatory attacker actions had a hard time hiding from all the research done by the folks at @corelight_inc."

Organizations that had Zeek sensors deployed could execute a simple search when the SUNBURST indicators of compromise (IOCs) were published. Assuming their security teams had kept the logs in a security and information event management (SIEM) platform or a similar analytics platform, a quick text search would confirm whether the malware had been seen on the network.

This is hugely important because if an organization was a SolarWinds customer but didn't have access to security-specific logs going back a year, they would have no way to know if they had been affected, and would've had to assume the worst. Many government and enterprise shops talked of having to "burn down the network" to ensure they had eradicated the malware. According to a *Business Insider* article (`https://www.msn.com/en-us/news/politics/it-could-take-years-to-evict-russia-from-the-us-networks-it-hacked-leaving-it-free-to-destroy-or-tamper-with-data-ex-white-house-official-warns/ar-BB1c0ouw`):

> Tom Bossert, a former homeland security advisor to President Trump, in a *New York Times* op-ed, sounded the alarm about a recent Russian hack of U.S. systems. "The Russians have had access to a considerable number of important and sensitive networks for six to nine months," he wrote. He said that it could take years to remove the hackers, and Russia could use its access to monitor or alter government data, and spread chaos.

Exploring Zeek's Key Capabilities

Open NDR deployments built on open-source tools like Zeek and Suricata can be easily integrated into the security infrastructure of almost any modern enterprise. But, before you get started, consider these characteristics of organizations that have the most successful deployments:

- **A data-first approach to security:** You could think of this as a "big data" mentality: The first step is collecting the data, even if you're not exactly sure how it will be used or needed in the future. Not every organization is prepared for, or wants to keep, terabytes or even petabytes of security log data. If your organization thinks of security infrastructure as *alert-first* (meaning your team would be satisfied with an intrusion-detection system that alerts you to malware, but that doesn't provide access to the underlying evidence of intrusion), then perhaps Zeek isn't right for you.
- **A modern security operations center (SOC) and SIEM platform:** Zeek logs need to go somewhere, and most successful deployments involve integration of the Zeek sensors with a modern SIEM platform. Popular choices include Chronicle Backstory (Google), Elastic (ELK Stack, typically), Splunk, and sometimes newer entrants like Databricks, Devo, Securonix, Sumo, or even data lakes built on Hadoop and related technologies.

  
  REMEMBER

  Wherever you store the logs, they need to be accessible and searchable. Some organizations have older SIEM platforms and will have trouble ingesting and storing the volume and variety of Zeek log data.
- **Appropriate network tapping capability:** As I explain earlier, Zeek is typically connected via a packet broker to your network, so it can ingest a copy of your network traffic at that location. You'll need an available port on your packet broker, or if you don't have one, you'll still need a TAP/SPAN port available. If your security organization isn't able to tap the network, then Zeek is of no use to you.

Some of the key capabilities of Corelight Sensors include the following:

- **High-volume traffic monitoring:** A single Corelight Sensor can handle up to 100 Gbps of ingested traffic in a 1 rack unit (RU) physical appliance (open-source Zeek sensors typically start to struggle at 3 to 5 Gbps).
- **Flexible deployment options:** Corelight Sensors can be deployed as physical appliances in your data center, as virtual machines (VMs), or in the cloud (Amazon Web Services [AWS], Microsoft Azure, or Google Cloud Platform [GCP]). A version called the Software Sensor can be deployed as an application on any Linux platform.
- **Integration of Suricata:** This book focuses mostly on Zeek, but Corelight has also integrated Suricata, which is often used side-by-side with Zeek in open-source deployments. For smaller deployments, integrated solutions like Security Onion include both (plus other security solutions).
- **High-speed file extraction (also known as file carving):** Keeping files on hand for future forensic examination is often useful. Corelight Sensors can *extract* (carve) files from network traffic at extremely high speed — thousands of files per minute — and operators can control which types of files are extracted. The combination of Zeek logs, Suricata alerts, and extracted files can help security teams resolve and understand 80 percent to 90 percent of all security incidents without examining PCAP files.
- **Insight into encrypted traffic:** More and more network traffic is encrypted, even inside organizations. That obviously makes it harder to examine the traffic for malicious content or behavior. Corelight offers a group of Zeek packages called the *Encrypted Traffic Collection* (ETC) that offers insight into what's going on.

  Although any security investigator would rather have access to unencrypted traffic, and although that's technically possible with break-and-inspect solutions, those aren't always feasible for legal, policy, or operational reasons.
- **Traffic shunting:** Some traffic is just not worth inspecting because that portion of the network traffic is exceptionally large and potentially repetitive. *Shunting* allows the operator to divert the payload of *elephant flows* (very large data flows,

for example video traffic, DNA data sets, and so on) and to retain only some of the connection metadata, increasing the effective monitoring capacity of the sensor.

- **Community ID:** Corelight supports this feature that allows easy pivots on network connections between tools like Zeek and Suricata. The Community ID is a hash of the five-tuple (composed of five values: the source and destination IP addresses, source and destination ports, and the transport protocol). Because several other key security solutions also support Community ID, it's a powerful way to see correlations and behavioral or temporal relationships between the two systems for a given TCP session. For more information about community ID, Christian Kreibich of Corelight published a paper about it in 2018, which you can read here at `http://icir.org/christian/talks/2018-11-suricon-communityid.pdf`.
- **Data reduction:** Zeek is very efficient at extracting compact, structured logs from network flows, but it still pumps out a lot of data. Because some SIEM platforms are priced based on the amount of data ingested, adding Zeek log data can increase the bill for your SIEM platform, sometimes significantly. Corelight Sensors can reduce the volume of key logs compared to Zeek by 30 percent to 50 percent, which can make a material difference in your bill.
- **Fork-and-filter:** Sometimes a SOC team wants to send some Zeek logs into its SIEM platform for live analysis but send other logs to longer-term "colder" (and cheaper) storage. Corelight Sensors allow fine-grained control of log destinations to *fork* different logs to different destinations, and to *filter* out some logs altogether if they aren't needed.
- **Support for the Zeek Input Framework:** Often it's useful to automatically append or "decorate" Zeek logs with more human-readable information to speed up investigations. That can be done using the *Input Framework,* which allows the addition of third-party data to be inserted into Zeek logs. It might be useful, for example, to include department names, location information, machine names, and so on, so that incident responders do not have to constantly look up trivial (yet important) data to understand the context around an alert.

Note: Corelight is the company behind Zeek, and some of these features are available only with Corelight Sensors, not with DIY Zeek sensors. I address some of the challenges with building your own Zeek sensors in Chapter 5.

Other Elements of Zeek Logs

One of the most powerful capabilities of Zeek is the structured and linked nature of the data. Dozens of logs come "out of the box" with Zeek, plus other packages from the community are available from GitHub (`https://github.com/zeek`). And power users (Zeek admins) can write their own packages for custom applications or use cases.

Because all Zeek logs are created using the real-time event processing engine that we describe in Chapter 1, they share certain common elements. A trivial but important example is the timestamp. Any incident responder who doesn't have access to Zeek logs can tell you that before they can piece together the circumstances around an incident, they need to sync up whatever data they've been able to collect from sources that are usually not tightly synchronized like NetFlow collectors, PCAP files, and device health logs (for example, email servers, file servers, web servers, Domain Name Server [DNS] logs, and so on).

Imagine you're a detective investigating a crime, and you have a bunch of clues, but they're not on a common timeline. That makes the investigation harder because time is a critical dimension of any incident. This is just one example of the tedious and time-consuming part of incident response. Zeek logs, by contrast, all share a common timestamp that allows instant investigation across all Zeek logs with temporal data included.

Another critical field is the Unique ID (UID) field. Every TCP connection logged by Zeek is assigned a unique identifier known as the UID. This is a critical pivot point in Zeek logs that allows an investigator to "pull the threads" as they follow clues in the Zeek logs.

For example, the UID allows an analyst to start with a clue triggered by an email, with that evidence collected in the SMTP log. The analyst can copy the UID from that log and then pivot to the

Files log to see any file downloads that occurred during the same session and determine whether anyone else was affected. They may then examine any DNS queries and responses in the DNS log for more clues, again pivoting off the UID, and examine the certificates used during the session. Cross-log linking is one of the most powerful capabilities of Zeek.

Similarly, files are also assigned a unique ID when observed traversing a network, which is known as the File Unique ID (FUID). All files are assigned a FUID, which makes searching them a snap. For example, if an analyst determines that malware was downloaded after a user clicked on a malicious link, that FUID would be visible in the Files log. The analyst could simply copy the FUID and search for it across the network to see if other users had also downloaded that file or if it had made its way into other hosts in the network.

Integrating Zeek Data with Suricata Alerts and PCAP

Earlier I describe why relying on perimeter-based security solutions is insufficient, but that doesn't mean they aren't necessary. Any enterprise security team wants to detect known malicious software or other IOCs. The problem is that lots of attacks use novel techniques or methods like social engineering to evade IDS solutions. That said, many Open NDR deployments built with Zeek also rely on a complementary IDS called Suricata, deployed in parallel and used separately, providing powerful network monitoring.

Corelight has taken it one step further and integrated Suricata onto its sensors. It has done considerable integration at the hardware level to ensure that, as traffic load increases, the compute power required to execute both Suricata alerting and Zeek event processing can scale gracefully. Corelight also provides a modified Suricata alert that includes the UID I discuss in the preceding section so that investigators can pivot quickly from the Suricata alert right into the relevant Zeek log, which saves time with every investigation.

Two other tools frequently used alongside Zeek and Suricata are PCAP and NetFlow data. Normally, if you don't find what you're looking for in the thin data Netflow stream, you have to dive into a PCAP file where 99 percent of the data probably isn't relevant to a security investigation. Security teams that use Zeek and Suricata, combined with file extraction, typically find that 80 percent to 90 percent of security incidents can be resolved using those tools alone.


REMEMBER

PCAP files of large network flows are so large that they're typically only kept for a few days or maybe a week or two, continually overwritten by newer PCAP files. If you're investigating a six-month-old breach, PCAP can't help you!

Why Not Build DIY Sensors?

Hopefully, with this brief introduction you can appreciate the power of Zeek logs, especially when they're integrated with Suricata alerts and a modern SIEM platform. Perhaps you're also thinking to yourself, "This is open-source software — I should be able to build these things myself!" And you can. But this section is intended to help you understand what you may be getting yourself into, especially if you work at a large enterprise or government agency.

First of all, it's true: Like all open-source software, Zeek and Suricata are both free and can be downloaded from GitHub and installed on off-the-shelf hardware. Thousands of individuals and organizations around the world have done so.

In fact, Zeek is also incorporated into quite a few commercially available network security products, though many of those solutions don't allow operators to access the raw, underlying Zeek logs. Instead, they use the Zeek data as inputs into their own alerting logic.

TIP

Before you decide to build your own open-source/DIY Zeek sensors, think about these considerations:

- **It's free (like a puppy).** Any open-source software comes with costs even though you're not paying directly for the software. Mostly you pay with your time (and frustration) — time spent by your security experts specifying, buying, assembling, configuring, tuning, debugging, upgrading, and just maintaining the tools.
- **You have lots of choices.** Building an enterprise-grade Zeek sensor requires your team to select the version of Zeek, the hardware platform, an appropriate network interface card (NIC), and a version of Linux. Your team then has make it all work (not to mention ongoing upgrades, maintenance, debugging, and so on).
- **You may not be making the best use of your people.** In most organizations, the people building Zeek sensors are the very same folks whose day job is incident response or threat hunting. Serving as Zeek and Suricata system administrators is not a good use of their time.
- **You don't get support.** Like any open-source tool, Zeek doesn't come with technical support other than what you can get from message boards, community members, or your Zeek-expert friends. That may not matter when you're dealing with a single small-scale sensor in a lab, but it's a big issue when you're deploying dozens (or hundreds) of sensors around the world.
- **There's a risk of turnover.** Capable, Zeek-knowledgeable security people are hard to find, and even harder to replace on short notice. What if your resident Zeek expert leaves after doing a great job building and deploying your sensors? If Zeek has become mission-critical, you could be in a tough spot.
- **You need plenty of space, power, and cooling.** DIY Zeek Sensors typically start to struggle at 3 to 5 Gbps. That means if you're a big shop and you want to monitor a 100 Gbps link, you need a rack full of Zeek sensors. Corelight's AP 5000 appliance, by comparison, can handle up to 100 Gbps in a 1 RU form factor, with no packet loss.

- **You may face tuning issues and packet loss.** Zeek sensors built with off-the-shelf components are notorious for high and often devilishly invisible packet loss. That means your team may be seeing only 60 percent to 80 percent of the packets — you're essentially looking for security evidence with one eye closed. Corelight has optimized its NIC and software to deliver zero packet loss, which means less headaches and better performance.

REMEMBER

Open-source software is free, but it isn't cheap!

IN THIS CHAPTER

- Helping SOCs operate faster with Open NDR based on Zeek and Suricata
- Extending functionality with Zeek and Corelight packages
- Integrating with your SIEM platform
- Integrating with SOAR platforms

Chapter 3
Improving Operational Security with Open NDR

This chapter is about implementing Zeek in your enterprise from a high level. It's not the Zeek manual or installation guide (also known as *Book of Zeek*), which is available at `https://docs.zeek.org/en/current/install.html`. Rather, it's a big-picture explanation of how building an Open NDR capability based on Zeek and Suricata can have a big impact on your security operations center (SOC), how it operates, and how effective it can be.

The Power of Open NDR

You may wonder, what is *Open* NDR, and why do we need yet another category? Three things characterize Open NDR:

- **Open-source:** Enterprise solutions built on open-source software ensure that customers aren't dependent on a single vendor for technical advances, and they're also not beholden to the vendor financially. Customers have a lot more freedom to move off the platform if they're not happy, and they benefit

from innovation from the vendor, from other vendors and even individual contributors to the open-source project. It's a more powerful way to develop software.

- **Open data:** Incident response and threat hunting is detective work, and teams depend on access to the evidence. NDR solutions that just give you the "answers" in the form of "the right alerts" are probably not what a sophisticated blue team is looking for. If someone at a crime scene tells you, "The butler did it," but doesn't let you see any fingerprints, cell phone records, or other evidence, you'd be hard pressed to take their word for it.
- **Open architecture:** Your needs change over time, and perhaps your organization has specialized requirements. In that case, the flexibility of an Open NDR platform is critical: You can modify it with third-party packages or with packages your own security team develops to meet your needs.

Using Zeek Collections with Your Open NDR Platform

Some vendors like Corelight include curated collections of Zeek packages to extend the performance of the sensors. Here are just a few of the common open-source packages available:

- **Lateral movement detection (MITRE BZAR):** Detect lateral movement techniques in MITRE ATT&CK related to Server Message Block (SMB) and Distributed Computing Environment/Remote Procedure Call (DCE/RPC) traffic, such as indicators targeting Windows Admin Shares and Remote File Copy.
- **Cryptomining detection:** Generate a notice when Bitcoin or Litecoin mining traffic is detected over Transmission Control Protocol (TCP) or Hypertext Transfer Protocol (HTTP).
- **HTTP stalling detection:** Detect when a web client executes a resource exhaustion attack on a web server.

- **Long connections detection:** Generate a notice when long-running connections occur, providing early visibility into a possible attack in progress.
- **Port scanning detection:** Identify port scanning behavior involving hosts (horizontal) or ports (vertical) across a variety of protocols.

Analyzing encrypted traffic

As you get more familiar with your sensors, you may be curious about all that encrypted traffic on your network, and with good reason. Short of *break-and-inspect* (decrypting the traffic, inspecting it, and then re-encrypting it), it might appear you can't do much for visibility. Although many products enable break-and-inspect, implementing them isn't always possible. Legal restrictions or internal policies might prohibit their use. Or deploying such a solution everywhere you'd like may not be operationally feasible.

As encryption becomes more common, even for east–west (internal) enterprise traffic, it's nice to know that with some Open NDR solutions, you have options to gain insight into potentially malign behavior on your network without decryption.

Here are a few examples from Corelight's Encrypted Traffic Collection of Zeek packages:

- **Secure Sockets Layer (SSL) fingerprinting (JA3):** Create a hash of every SSL/Transport Layer Security (TLS) client and server negotiation for use in threat hunting or intel feed matching.
- **Secure Shell (SSH) fingerprinting (HASSH):** Create a hash of every SSH client and server negotiation for use in threat hunting or intel feed matching.
- **SSL certificate monitoring:** Track expired and soon-to-expire certs, newly issued certs, self-signed certs, invalid certs, change-validation errors, old versions, weak ciphers, weak key-lengths, and bad versions (for example, TLS 1.0).
- **SSH client brute force detection:** Reveal when a client makes excessive authentication attempts.

- **SSH authentication bypass detection:** Reveal when a client and server switch to a non-SSH protocol.
- **SSH client keystroke detection:** Reveal an interactive session when a client sends user-driven keystrokes to the server.
- **SSH client file activity detection:** Reveal a file transfer occurring during the session when the client sends a sequence of bytes to the server or vice versa.
- **SSH scan detection:** Infer scanning activity based on how often a single service is scanned.
- **Custom encryption detection:** Detect connections that are already encrypted without an observed handshake, which can indicate custom or prenegotiated encryption.
- **Expected encryption detection:** Identify unencrypted connections running on ports when encryption is expected.
- **SSH agent forwarding detection:** See when SSH agent forwarding occurs between clients and servers, which may indicate lateral movement when adversaries have compromised SSH credentials.
- **SSH multifactor authentication (MFA) detection:** See when SSH connections use MFA, which can help analysts rule out other explanations for observed timing discrepancies in SSH connections. This detection can also help teams monitor external SSH servers for MFA compliance.
- **Noninteractive SSH detection:** Reveal when SSH connections don't request an interactive terminal and instead use SSH as a port forwarding tunnel, which may indicate malicious SSH tunneling.
- **SSH reverse tunnel detection:** Reveal when a client connects to an SSH server and sends the server an interactive terminal, establishing a reverse SSH tunnel that may indicate malicious SSH tunnelling.
- **Domain Name System (DNS) over HTTP Secure (HTTPS) (DoH) detection:** Reveal when DNS queries are made to known DoH providers to provide insight into DNS traffic that would otherwise be hidden.

You can see the power in taking advantage of the extensibility of Open NDR solutions when it comes to encrypted traffic. Of course, any investigator would like to be examining unencrypted traffic, but that's not always possible.

Detecting command and control (C2) activity

After attackers are inside your organization, typically they need to communicate with the outside world. The attacker needs to maneuver across your infrastructure. To do that, they need to have communication with, and control over, assets inside your network via an external *command and control server* (C2).

Detecting C2 communications can be very difficult because attackers are adept at covering their tracks and hiding in normal enterprise traffic. It's akin to a terrorist dressing up like a sports fan and blending in with the crowd as thousands of people enter and exit the stadium.

Corelight has developed dozens of detections, insights, and inferences to discover C2 activity preloaded on Corelight Sensors. There are too many insights (about 50) to detail here. Think of them as clues that C2 activity may be going on that warrants further investigation. Here are a few examples:

- Detections of common families of malware over HTTP
- For Meterpreter, specific detections for Metasploit's CLI
- List-based detections for domain generation algorithms
- Specific and generic detections for DNS tunneling
- Specific and generic detections for ICMP tunneling

The focus for these C2 detections built on top of Zeek is to build durable detections with a high signal-to-noise ratio (not to generate more alerts and false positives) by finding known tools that attackers use that indicate probable malicious behavior. It's a bit like keeping an eye out for a guy walking around your neighborhood with bolt cutters.

Integrating Open NDR into your security architecture

With any NDR solution, open or not, you must understand the scope of the solution. There isn't strict agreement as to what capabilities are essential, required, or optional. But generally, the flow goes something like what's shown in Figure 3-1.

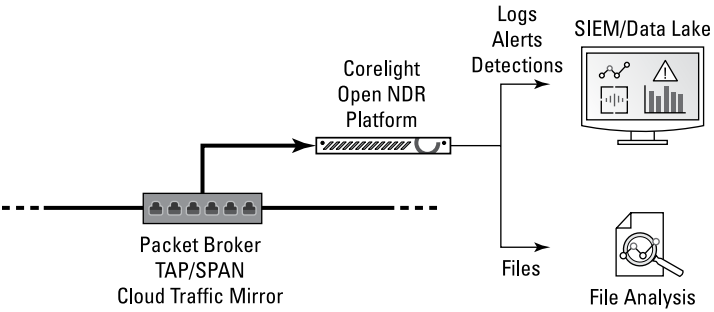


FIGURE 3-1: Integrating an Open NDR sensor into your network.

The core functionality of the system as a whole is to:

- **Extract** useful data from the network traffic, including logs, alerts, files, and other metadata.
- **Filter, search, sort, and prioritize** the alerts and log data to look for anomalies. This can be done by humans, or in combination with some sort of artificial intelligence (AI)–driven tool, or security orchestration, automation, and response (SOAR) platforms.
- **Export** the data, logs, and alerts in real time to the security information and event management (SIEM) platform, data lake, or other analytics platform for further analysis and long-term storage (in case a year or two from now a new attack is divulged and the data from today becomes useful).

REMEMBER

Some NDR solutions are all-in-one and require the customer to convert from their existing analytics platform or SIEM platform to the new NDR vendor's analytics platform. Others simply deliver the data to the customer's existing SIEM platform or data lake for analysis.

The choice about what approach you want to take depends on a couple factors:

- **Your existing SIEM platform:** Are you happy with it? Are you planning to upgrade now (or soon)? What about in a year or two? Can your SIEM handle the demands of a modern big data approach to security?
- **The sophistication of your team:** Does your team want the data? Will they analyze it? Do they understand how to take advantage of the power of the data? If not, then maybe you'd be better off with an all-in-one solution, more of an off-the-shelf NDR solution.

If you have a "data-first" philosophy, one that seeks answers in the data that may not be apparent, then you probably have a big-data-oriented approach to security, and you probably have a modern SIEM platform and don't want to change. In that case, you'll be better served with an Open NDR solution that puts the power of the data in your hands.

Using Zeek Data for Incident Response and Threat Hunting

You've heard of "garbage in, garbage out." That concept pertains to security data as well. Zeek is the gold standard for security-relevant network metadata. Integrating it into your existing security processes makes them more effective and efficient.

Zeek data can be used for incident response or for threat hunting, and ideally for both. One of the fundamental principles to keep in mind is that you only get one chance to capture the relevant data lurking in your network traffic. After the data passes over your network, if you didn't capture it, it's lost forever. The first step is to begin collecting it using an Open NDR Sensor.


TIP

As the saying goes, the best time to plant a tree was 20 years ago; the next best time is now. The same holds true for deploying network sensors!

After your sensors are deployed, you can start thinking about making use of the data, a topic too big for this book. I cover a few considerations in the following sections, and at www.corelight.com where you can request Corelight's Threat Hunting Guide.

Pivoting through the data

Anyone who has used Zeek logs for incident response or threat hunting can tell you that one of the most powerful features of Zeek logs are the key data fields that allow analysts to pivot quickly from one log to another.


REMEMBER

The simplest pivot point is time — all Zeek logs share a common timestamp that allows the investigation of web traffic, email, file downloads or shares, DNS queries and responses, and everything else across a common time boundary. That may sound trivial, but when you're gathering data from many unrelated sources like system health logs, network performance logs, NetFlow, and so on, and then trying to sync them up, that's just drudgery that Zeek avoids.

The most powerful Zeek pivot point though is the Unique ID (UID). Zeek assigns a unique number to every TCP session that it observes, and that UID field is present in most Zeek logs. If a suspected phishing attack occurs, and the analyst observes something in the Simple Mail Transfer Protocol (SMTP) log, they can simply copy the UID and search for it across the Zeek data — if something else happened during that session, it will show up. Then, in seconds, the analyst has a pretty complete picture of activities across a range of interactions, protocols, ports, applications, and other factors that otherwise might take hours to assemble.

Integrating Suricata alerts with Zeek logs

Zeek is often used in parallel with Suricata, the powerful signature-based IDS alerting framework. Although the two can exist side-by-side in different sensors, Corelight has uniquely integrated the two open-source tools into one sensor, which boosts performance and saves analysts time.

Notably, the Suricata alerts generated by Corelight include the UID I discuss in the preceding section. An analyst can copy the UID from the Suricata alert and pivot directly to the Zeek files log with that UID to find the hash of that file. The analyst can then

simply search Virus Total to see if that file hash is a malicious file that required further investigation.

Incorporating PCAP into Zeek investigations

The metadata that Zeek collects in real time from network traffic is so relevant, concise, and useful that most large-scale enterprise SOCs can tell you that 80 percent to 90 percent of security investigations can be resolved using that evidence alone. But that still leaves 10 percent to 20 percent that requires something extra, and usually it's resorting to examining packet capture (PCAP) files ("Let's go to the videotape!").

PCAP is really like a surveillance video — if your business is burgled, but you don't know exactly when or how, then sometimes you have no choice but to watch a lot of video. Hopefully the cameras were on, the recording was functioning properly, and the contents were stored long enough and not written over! But even though it's laborious and time-consuming, sometimes the answer is there.

In the world of network security, Zeek can help you pinpoint a lot of clues around a point in time and a specific TCP connection or file transfer, and usually that's enough to figure out what happened and what to do about it. ("Footprints show they jumped the back fence, guessed the security code on the loading dock, and stole our truck!")


TIP

Sometimes it's still necessary to replay an entire traffic session to really unravel a complex attack. Some Open NDR solutions incorporate selective, or *smart,* PCAP into their solutions to extend the time window over which PCAP is useful by only keeping some of the network traffic, usually controlled by policies that limit collection:

- From specific ports
- When defined Suricata alerts trigger
- Over specific protocols
- Of unencrypted traffic (no point in capturing and storing encrypted data that you can't make sense of later)
- From unknown protocols/ports/traffic types (to make sure you capture anything weird you'll want to have)

REMEMBER

By being selective (or "smart") about which packets you capture, you can make more efficient use of the given storage available, which means your team can go back farther in time using PCAP to resolve investigations. That increases the odds of success for those thorniest cases.

Using SOAR playbooks

Zeek data is great for SOAR. One of the challenges of implementing SOAR over the last few years has been the quality and variety of the inbound data: If the data your SOAR system is ingesting isn't clean, standardized, and structured, then no automatic system will be very effective.

With modern SOAR playbooks, vendors like Corelight have written the rules to automate mundane and repetitive investigative tasks to sort through the Zeek logs and help separate the wheat from the chaff. Using SOAR playbooks in conjunction with Zeek data can be a force multiplier for your SOC. Find Corelight's playbook for Splunk's Phantom SOAR offering at `https://github.com/corelight/phantom-playbooks`.

Data reduction

Zeek is a very compact data structure, which means you can keep Zeek logs around for years.

Remember: Typically, the output of a Zeek sensor is roughly 1 percent of the monitored traffic. So, if you're monitoring a 10 Gbps fully utilized link, you would expect roughly 100 Mbps of Zeek logs to be created. Although that's much more efficient than 100 percent PCAP, it's still a lot of data to ingest and store! Some vendors, like Corelight, have implemented techniques to reduce the volume of logs without giving up the relevant information.

REMEMBER

Zeek was created in 1995 in an academic environment. Commercial considerations like the cost of SIEM licenses weren't really on the radar — in fact, SIEM platforms didn't exist, so they *really* weren't on the radar!

With data reduction implemented, it's possible to reduce the volume of Zeek logs as they're exported to your SIEM platform by 30 percent to 50 percent compared to open-source implementations.

Fork and filter

Sometimes, possibly also for cost considerations (like per-gigabyte SIEM licensing models), you may not want to export all your Zeek data into your SIEM platform. Maybe you want the most commonly accessed logs to go there, but you want to send everything else to long term, low-cost cloud storage. Or perhaps your company is moving to a new SIEM platform, so migrating some logs to the new system while retaining your old system during a transition makes sense.

Forking and filtering simply means to designate different targets for different logs, and perhaps even filtering out specific logs you're not interested in.

Speeding up investigations

Using Zeek for the first time can be overwhelming even for experienced security analysts. With a ton of data that may be unfamiliar, analysts new to Zeek may not know where to start. It's not unusual for new users to only scratch the surface of what's possible because they're not sure where to go next:

- **Using pre-built dashboards:** Zeek produces a lot of data! Out of the box, it has approximately 50 log types and hundreds of data elements. For new users or teams, even knowing where to start can be a challenge.

  Luckily, some prebuilt SIEM dashboards for vendors like Splunk and Elastic (for Kibana, their visualization tool) make it easier to get started, and for your team to get more familiar with Zeek data. When they do, no doubt they'll want to explore the data directly by using their tool's data exploration and querying tools.

- **Appending third-party data:** One of the laborious, repetitive, and mundane tasks any security analyst has to do is continuously look up IP addresses and other data to see what they mean in human terms. Is that a server in Romania or something in our Dallas data center? It might matter!

  With Zeek, you can decorate the logs with human-readable data from third-party databases of locations, equipment, functions, or whatever else makes your analyst's job easier.

IN THIS CHAPTER

- Learning how the most sophisticated security teams think about monitoring
- Finding out why perimeter-based security is no longer sufficient
- Knowing what to consider as you look at NDR solutions

Chapter 4
Rethinking Your Security Posture

This chapter steps back and takes a look at the philosophy behind Zeek and discusses the power of Open NDR. These topics may be unfamiliar ground to you, depending on your past experience and the tools you're accustomed to using. Regardless, I hope it helps you consider thinking differently about your security architecture and approach.

Perimeter-Based Security: Necessary but Insufficient

Seatbelts. Water. Good barbecue. Many things in our lives are necessary but not sufficient (alone) to sustain life. Perimeter-based security is one of them. For decades, the need to keep attackers out of our facilities, systems, databases, networks, and other infrastructure has been obvious. It's also required. It's a no-brainer.

Just like locking your front door and setting the house alarm when you leave, ensuring that you have correctly configured firewalls, advanced firewalls, intrusion detection and prevention systems, multifactor authentication, physical security, and all the rest is fundamental. Unfortunately, in today's environment, it's not enough.

I cover the basic idea in Chapter 1. Revisiting why this is the case in enterprise security today is worthwhile. Recall that Zeek was developed and evolved in the late 1990s at Lawrence Berkeley Lab and then spread throughout the federal government before making its way into the world of enterprise security.

Today, in 2021, as we (hopefully) exit the COVID-19 pandemic, the world is a very different place. Enterprises of all sizes are dynamic, dispersed, decentralized, and dependent on their employees working wherever and whenever it makes sense. Many organizations have to adopt a zero-trust posture — no matter where someone is, even on the headquarters campus, you can't be sure users are who they say they are. Authentication is a must.

WARNING

Adversaries are determined, creative, smart, and often highly motivated to penetrate your organization. Whether they do that through brute-force password guessing, phishing, supply-chain attacks like the SolarWinds/SUNBURST incident (see Chapter 2), or social engineering, you have to assume attackers will sometimes be successful.

REMEMBER

Therefore, if those adversaries will get in, you need to rethink your approach to security. As shown in Chapter 1, a critical component of a multilayer defense is the Gartner SOC Visibility Triad that combines evidence collected from networks and endpoints and delivers it to a security information and event management (SIEM) platform or whatever your analytics stack happens to be.

WARNING

You need to start collecting the data yesterday (actually yesteryear). If you miss the chance to collect that evidence when the attack was perpetrated, you can't get it back!

The other thing to keep in mind is the point made by Richard Bejtlich: "You only need to trip up an attacker only once to discover them and take action." Attackers aren't perfect, but it may

seem overwhelming as a member of a blue team — because thousands of combinations of tactics, techniques, and procedures can be launched by an unknown number of adversaries over any time period. The reality is that anticipating and preventing all possible attacks at all times is not possible. Monitoring your key networks 24 hours a day, 365 days a year *is* possible. If your organization isn't doing that simple first step, you're missing critical pieces of information that your security teams will need to do their jobs at some point in the future!

Why Open NDR Is More Powerful Than Proprietary NDR

In recent analyst reports, the network detection and response category includes many vendors that make a range of capability and design choices when it comes to NDR. I discuss a few in this section.

Trust the vendor (versus trust but verify)

In a proprietary (or "closed") NDR solution, the vendor owns the data format. It may be "Zeek-like" or a totally different methodology for extracting meaning and insights from network flows. If it is, then you just have to take the vendor's word for it. In an Open NDR solution, the data format is wide open, documented, and available for inspection because open-source software underpins the data.

A widely used "design pattern"

When you go with Open NDR, you're following in the footsteps of the world's most advanced blue teams, the *Apex Defenders* (who defend their organizations against apex predators). You're implementing best-of-breed open-source solutions that generally fit the design pattern shown in Figure 4-1.

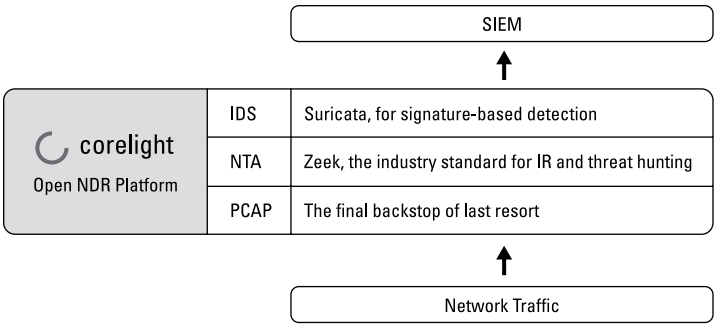


FIGURE 4-1: The Open NDR design pattern.

The flavor of the data

All network analytics solutions produce data, whether for security, network performance, or other operational reasons. But not all data is created equal. Data produced by Open NDR solutions typically scores high in these dimensions:

- **Exportability:** Can the data be exported to any analytics platform, or are you stuck using the built-in analytics capabilities of the NDR solution?
- **Extensibility:** Out-of-the-box data provided by an NDR solution often isn't exactly what you're looking for. Can you extend, modify, append, and integrate the data easily to meet your needs?
- **Filterability:** Sometimes NDR solutions produce data you're not interested in. Does the product allow you to control the volume and types of data produced? Can you easily send the data where you want?

Analytics included (or not)

You may be an employee of a large global corporation, or you may work at a small business and maybe even operate as a one-person shop where you also have to make sure routers, switches, and printers are working!

If you're in the latter category, you probably would be better served by an all-in-one NDR solution, where you're getting a pretty good first approximation of nefarious activity on your network, have a nice graphical user interface (GUI), tunable alerts

based in signatures, and more — solutions like Security Onion (an all-in-one tool including Zeek, Suricata, and other security tools combined with a nice GUI built on the ELK stack [Elastic, Logstash, and Kibana] from Elastic) would probably be worth investigating.

If you're in the former category, perhaps an incident responder or security architect at a large global company, then those types of solutions are probably not appropriate. For one thing, the networks you need to be monitoring may be measured in the tens of gigabits per second (Gbps); in aggregate, you may need to monitor hundreds of Gbps, which requires dedicated, high-performance sensors.

Plus, at a large company, you probably require access to the underlying data. It's simply not enough in a large organization for a buzzer to sound, a light to flash, or a map to blink. You want to investigate the *why* of an incident, especially if you think it's serious.

Access to underlying data

If you're at a large organization, in order to understand not only what happened, but also to whom, when, why, and how, you need access to the data from which the alert was generated.

Networks at large enterprises are full of a massive amount of highly complex data, unusual traffic patterns; many thousands of users; hundreds of applications; and lots of simply weird and unexpected behavior and activity that's difficult for even an experienced security analyst to make sense of. Getting an alert that says "something bad happened" or "this is an anomaly" just isn't enough. You must have access to relevant evidence to be sure.

Cloud only or hybrid

Cloud computing is an inexorable force, a massive phenomenon driving phenomenal change in enterprise computing. However, the fact is there's a ton of data, applications, and other infrastructure on-premises, and there will be for a long time to come. When you're looking at NDR solutions, cloud monitoring is critical and a fundamental requirement, but you need to think about the whole picture and not forget about the on-premises networks that must be monitored, too.

Onboard storage or flexible options

Even with systems like Zeek, which are super-efficient at pulling out the relevant metadata from network traffic, it's still a ton of data. Some NDR solutions store the extracted metadata onboard, whether that includes packet capture, Zeek data, Suricata alerts, or some combination. Even in a large heterogeneous environment there simply won't be enough onboard storage to take advantage of one of the most powerful capabilities of Zeek: to go back in time, by years if necessary. To do that, you need to have kept the log data, and if you're constrained to keeping the data local in the sensor, you'll run out of space very quickly!

Fancy maps and charts

Look, we all love those fancy graphs, charts, and maps. They're interesting to look at, can save time, help analysts focus on what's important, and tell a story efficiently. But don't get distracted by them — sometimes all you need is summary data.

When a patient talks to an ER doctor, it's not enough for the patient to say, "I'm sick, admit me!" The doctor has to run a bunch of tests to ascertain the patient's temperature, blood pressure, kidney function, heart function, brain function . . . you get the idea. They need the underlying data to decide whether to send the patient home with the proverbial two aspirin or admit them to the hospital.

Security investigations are similar. Analysts need to gather evidence from many sources to paint a complete picture of an alert and decide if it's malicious or benign.

Revisiting the Core Components of Open NDR

In Chapter 3, I outline the core components of Open NDR, but I cover them here, too, in the context of the power of Open NDR and rethinking your security approach.

Open source versus proprietary

Open-source software isn't the only way to develop technology and build products, but it's a pretty powerful approach. From the buyers' perspective, there are a couple of key advantages:

- **With open-source software and open-core products (solutions built on open source at the core with perhaps proprietary add-ons), you're benefitting from the ingenuity and know-how of project contributors, no matter where they work.** With proprietary solutions of any sort, you're dependent on the ability of a particular vendor to hire and retain the best people in a particular field.
- **With proprietary solutions, you become accustomed to that gnawing, sinking feeling when it comes time to renew the license or subscription.** If you've become dependent on the solution and it's deeply integrated into your IT infrastructure, it's difficult to switch. MBAs even have a name for it: *switching cost.*

REMEMBER

With proprietary software you're at a disadvantage because often another viable vendor with a comparable solution just doesn't exist. With open-core solutions, if push comes to shove you can always decide to implement the open-source software yourself — in this case, build your own Zeek and/or Suricata sensors. That's real leverage over vendors. (But remember the costs of "free" open-source software I cover in Chapter 2 — they're real.)

Open data versus alerts only

We cover this pretty extensively because it's so important. Any security analyst or investigator worth their salt wants the data. If you buy an NDR solution that doesn't allow that, eventually you'll be caught flat footed.

Open architecture versus closed

Life isn't static, and neither are security threats (or solutions). If the NDR solution you buy is closed, if its features can't be extended or added to, then you're stuck with what you buy.

Open NDR solutions' open architecture means you can extend its functionality with new packages. You can create new packages yourself to extract data or other evidence that's specific to your company or industry. And you benefit from the work of people in other organizations (see `https://github.com/zeek/packages`) who may benefit from yours.

REMEMBER

In the end, every company has to develop the security posture that makes sense for their situation. But no matter who you are, monitoring network traffic is fundamental. Counting on perimeter-based security alone is dangerous. You must assume that attackers have penetrated your network or will do so soon. If you're not gathering evidence before that happens, well, good luck to you.

IN THIS CHAPTER

- Considering questions security teams often have trouble answering
- Finding areas to investigate in your own organization

Chapter 5
Ten Questions to Ask Yourself about Your Network

This book explains how Open NDR can help your cybersecurity teams operate more efficiently and effectively, closing investigations using the evidence collected by Zeek and Suricata. With a grounding in Open NDR, you might wonder, "So what? What can this really do for me?"

This chapter focuses on a key question for any cybersecurity analyst, incident responder, or threat hunter: "How do you know?"

When faced with any incident, the problem for most members of a security operations center (SOC) is the seemingly infinite amount of data available to explore in an effort to understand what happened (or didn't happen). At the same time, the universe of available data is somewhat unknown. Analysts may wonder

- Do I have all the relevant information? What is all the relevant information, and where is it?
- When did this incident start? How will I know?
- Who else may have been affected? How will I know?

And many, many others. Here are ten things your security team should know — and *can* know — with an Open NDR platform.

Are You Collecting All the Relevant Evidence?

You can tap many sources of data to help security analysts to do their jobs. Unfortunately, it's not so easy to actually collect and keep it. Here are some things to consider:


REMEMBER

- **If you monitor endpoints (and you should), do you have every single endpoint instrumented across your company?** Every laptop and host, every smartphone in every office (or coffee shop) worldwide?

  If you're part of a large enterprise with hundreds of thousands of devices, not all of them will accept the deployment of endpoint detection and response (EDR) solutions. There's almost no way to instrument every endpoint.
- **Are all your relevant pieces of infrastructure being monitored using appropriate logs?** Is logging turned on everywhere? Are you sure? On Domain Name System (DNS) servers? File servers? Email servers? Databases? Data storage systems? Routers? Switches? Legacy infrastructure? In all sites and facilities? In every country where you have people or facilities? Really?
- **Are those logs being kept long enough to be useful?** How long is that? Hours? Days? Weeks? Months? Years?
- **Does your security team have instant access to that data in the event of a breach or other incident?** Do they have to ask for permission? How long does it take them to gather all the data they need?
- **Is access to that data permitted?** Is it constrained by local privacy laws? Does another group have oversight over the data that may delay access to it?
- **What about cloud-based systems, applications, and data?** Is that traffic being monitored? By what? Where is the data?
- **Is the data you're collecting continuous or sampled?** Many logging systems don't keep everything; they only sample

the traffic periodically, or capture only some portions of an interaction but not others. Or they may log once every minute. Or they may collect data on a query but not the associated response (for example, collecting DNS queries but not responses).

You may not have the visibility you think you have. In the middle of the next breach is not the right time to realize you have massive visibility holes!

Do You Understand Your Network and What's Connected to It?

Most IT organizations think they do, but some simple questions sometimes may poke holes in that view:

- **Can you create a complete picture of your global network?** How confident are you that the picture would be complete?
- **How many devices are connected to it right now?** Are they all legitimate and authorized? How do you know?
- **Is every device on your network running the minimum required version of its operating system?** If they aren't, is that a vulnerability?

Who Was Affected by This Attack and When?

Often attacks involve downloading malware, sometimes by email due to a successful phishing attack. If that happens, or you suspect it has happened, then you'll have questions:

- **What's the hash of that file?** Is it malicious?
- **Who else has downloaded it?** How can you figure that out? How sure are you of the results?
- **When did it first appear on our network?** Are you sure about that? Can you prove it's never been on your network?

How Far Back in Time Can You Go?

Hours? Days? Months? Years? Most enterprise security teams rely on a combination of data sources to investigate breaches. Common sources are device health logs, NetFlow logs, packet capture (PCAP), and endpoint detection and response (EDR). That means analysts spend a *lot* of time pulling together information from many sources and building a coherent picture of what happened.

The problem is often that the data they have to work with is spotty and incomplete (as highlighted by the previous questions in this chapter).

Even more problematic is that sophisticated attackers, especially advanced persistent threats (APTs), are patient and willing to wait. Sometimes they remain undetected for weeks (very common), sometimes months (for example, in the SolarWinds/SUNBURST breach), and sometimes attackers go undetected for years (for example, in the case of Marriott).

WARNING

If your SOC isn't collecting and storing security-relevant evidence for at least a couple of years (and ideally longer), when you're subject to one of these attacks, your analysts literally have nothing to work with. How will they know when an attack really started, and who was affected? The answer is, they won't.

What's in That Encrypted Traffic Anyway?

A fundamental challenge for any SOC is the growing proportion of internal traffic that's encrypted. Sure, lots of break-and-inspect solutions are out there, and any security analyst would much prefer to have access to unencrypted traffic during an investigation.

But the reality is that reality intrudes. Although deploying break-and-inspect solutions is technically feasible, it's usually not possible, at least not everywhere, for these reasons:

- **The security team doesn't have control over that portion of the network infrastructure.**
- **It could be illegal in some jurisdictions for privacy reasons.**
- **Your organization may have policies that don't allow traffic to be unencrypted for security purposes.** This may vary from country to country.
- **Networks contain a tremendous amount of very detailed, very specific, and often extremely sensitive information.** (Think about salaries, proprietary information, email traffic that isn't intended for public consumption, web page visits, and so on.) So, naturally some companies are extremely reluctant to make that available by decrypting it.

Some commercial Open NDR solutions include techniques for gaining insight or developing inferences about encrypted traffic. By analyzing network-level traffic patterns, it's possible to infer that malicious behavior may be happening, giving investigators leads to follow instead of being completely blinded by encryption. Can you do that?

How Would You Detect Off-Protocol Port Use?

Wouldn't it be great if all terrorists were known and used their real passports when they made travel arrangements? We all know they use fake information to travel on commercial airlines hidden among the general public. The same is true of many attackers: They've become very clever at using nonstandard protocols and ports to hide in "standard" or "normal" traffic. Can you detect that? How do you know?

Techniques like dynamic protocol detection can spot potentially malicious traffic even if it's using nonstandard ports.

Can You Find Malicious Files Hiding in Plain Sight?

A simple technique attackers use is simply to change the file extension to make an executable look like a GIF or some other innocuous file type. If that happens, would you have a way to look for it in the ocean of benign traffic? How would you go about it? What if you miss one?

Are You Sure You See Every Packet?

Lots of network monitoring sensors struggle with packet loss at higher bandwidth. Even Zeek sensors can suffer from 30 percent, or even 50 percent packet loss in a way that isn't detectable to system administrators.

That's like having half of your surveillance cameras tricked like they do in the movies where the bank robber puts up a precisely sized picture of the room in front of the surveillance camera. The poor security guard who's trying to stay awake all night just sees a static, empty room. Nothing going on here!

If your network sensors aren't inspecting every packet, your team is flying partially blind, and they may not even realize it.

Is Your Security Team as Efficient as Possible?

Ask any analyst, and they'll tell you they spend a lot of time searching, gathering, organizing, synching, correlating, cleaning, transforming, and doing all manner of things to try to do their jobs. A lot of it is, time-consuming, repetitive, drudgery.

And even after all that, they're probably not sure they have all the evidence they need. It would be like trying to solve a jigsaw puzzle when you don't know what the picture is, and you're not sure if

all the pieces are in the box, or even if the pieces in the box are for the right puzzle! Are there other puzzles? Other boxes? Other pieces? For most security teams, there's room for improvement and increased efficiency.

What Does It All Mean?

The modern world is driven by data, and today, so is security. For many SOCs at large enterprises, cybersecurity is a big data problem. With thousands, tens of thousands, or even hundreds of thousands of employees using all manner of devices and applications from locations around the world, trying to make sense of suspected attacks isn't easy. By implementing an Open NDR solution in your organization, you can transform your security posture into one that's faster and more effective. Here are a few advantages:

- **Taking a systematic, data-driven approach to security by deploying an Open NDR platform means your team has the best, most relevant data at their fingertips.** No more scrounging around to assemble data from many different sources.
- **Because Open NDR data (Zeek logs) is compact, it can be kept for years.** This means your team can go back in time as far as necessary to get to the bottom of things.
- **Zeek logs are concise, relevant, and interconnected.** That translates to your incident responders and threat hunters being able to find the edges of an attack and understanding the context around alerts more quickly.
- **Understanding the context means they can close out incidents more quickly, whether benign or malicious.** If it's the latter, they can decide what needs to be done.
- **Closing out tickets more quickly means lower costs per incident (or more capacity per team).** Users of Zeek frequently report that incidents can be closed out 10 to 20 times faster compared to incident response without Zeek data.

- **Closing more tickets per hour, per day, per week, and per month is better.** It means analysts are more likely to get to the serious one (faster) among the thousands of benign alerts, reducing the existential or strategic risk to your organization.
- **Every large organization lives with the fear of a major breach.** A major breach leads to loss of revenue, confidential information, market value, reputation, and brand equity, or worse, legal and regulatory jeopardy and penalties.

Data is evidence, and evidence is what security teams need to resolve investigations. Without it, they're not able to live up to their potential. Open NDR can be a fundamental piece of your security infrastructure to ensure your SOC is doing all it can to defend your organization against clever, determined, and sophisticated attackers.

Notes

Notes

Notes

Notes

Notes

Notes

Empower your security teams with next-level threat hunting

Network data is ground truth. Unlike endpoints or servers whose data can be overwritten, network traffic contains innumerable clues about people, devices, applications, and data that are critical to successful incident response and threat hunting. This book is designed to introduce the fundamental idea of Open NDR (network detection and response): To find an intruder, you need to be collecting evidence all the time, and one of the best sources for that evidence is the traffic in your network.

Inside...

- Improve the speed and operational effectiveness of your security operations team
- Understand why metadata is critical for incident response and threat hunting
- Learn why traditional perimeter defenses aren't enough
- Enable your security team to understand the context around incident
- Give your blue team the data they need to enable effective threat hunting

Alan Saldich is the chief marketing officer at Corelight. He has been a marketing leader and adviser at several enterprise technology companies, including Cloudera and Riverbed. He has a B.S. in mechanical engineering from UC Berkeley and an MBA from Harvard Business School.

Go to Dummies.com™
for videos, step-by-step photos, how-to articles, or to shop!

ISBN: 978-1-119-81796-3
Not For Resale

for dummies®
A Wiley Brand

WILEY END USER LICENSE AGREEMENT

Go to www.wiley.com/go/eula to access Wiley’s ebook EULA.