White Paper

How Corelight offers encrypted insights without decryption

Treat encrypted traffic like a gift you can't open: use ingenuity to find out what you need to know without breaking and inspecting the contents. Shake the box!

Introduction

Advanced security teams rely on network traffic as a fundamental data source, but encryption has made it much harder to obtain. Industry reports reveal that more than 72% of Internet-bound network traffic is now encrypted[1], and intruders continue to conduct their attacks cloaked by encryption.

Since encryption can foil traditional security technologies, organizations must adapt. Decrypting traffic seems like the obvious solution, but it can be cost prohibitive, violate privacy laws, raise overhead costs and/or degrade performance. Should security teams just admit defeat? Hardly. There's lots of useful information in the observable elements of encrypted traffic.

Remember when you were a kid, and you found a wrapped gift with your name on it? You developed a sixth sense for its contents by asking the right questions: How heavy is it? What shape is it? Who's it from? Does it make noise? You could learn a lot without getting in trouble.

Corelight's Encrypted Traffic Analysis Capabilities

Corelight draws its encrypted insights from Zeek, an open source technology that has proven to be an extremely adept tool for parsing encrypted traffic. Zeek operates out-of-band analyzing traffic captured by a packet broker, span port, or optical TAP–transforming raw traffic into comprehensive logs, extracted files, and automated insights.

Zeek parses more than 35 protocols, including encrypted ones like Kerberos, SSL, and SSH. As a connection-oriented logging tool, each Zeek log includes a unique TCP connection id number (UID) that allows security analysts to easily pivot and see the full protocol activity of a given connection.

Corelight's encrypted traffic analysis capabilities not only support Zeek's native capabilities, but also extend these with proprietary insights. Corelight does all this without decryption, leveraging the observable elements of encrypted connections to deliver four fundamental security capabilities:

1. Identification - Detect commonly-used encryption protocols wherever they occur, even if attackers attempt to obfuscate their activity
2. Parsing - Accurately parse the traffic's cryptographic characteristics, summarizing them in rich, yet compact logs that analysts can use for investigations
3. Context - As encrypted attacks often involve unencrypted activity, Corelight gives analysts the means to see and connect those dots
4. Analysis & Detection - Corelight's Encrypted Traffic Collection contains dozens of proprietary encrypted insights that extend Zeek's native capabilities with inferences and detections built around certificates as well as SSL, SSH, and RDP traffic.

Use Cases

1. Identification: spotting a wolf in SSL clothing

Attack scenario
An attacker uses port 443 to give their traffic the appearance of SSL since they know that many organizations and security tools will ignore it.

Capability
This evasion technique can't trick Corelight since we verify the traffic's identity via a dynamic protocol detection process, looking past the port and header information and using every protocol parser in our arsenal until all but one fails (revealing the true protocol). In addition to generating a protocol specific log, when Corelight identifies protocols on non-standard ports it generates a separate dpd.log that documents the traffic anomaly. In the previous port 443 attack scenario, Corelight would generate a dpd.log like the one below:

{"_path":"dpd","_write_ts":"2018-01- 15T17:11:57.552839Z","ts":"2018-01-15T17:11:57.552839Z","uid":"CpPNAD 4SAqvPZf0h5b","id.orig_h":"1.2.3.4","id.orig_p":3908,"id.resp_h":"5.6.7.8"," id.res_p":443,"proto":"tcp","analyzer":"SSL", "failure_reason":Invalid version late in TLS connection. Packet reporter version: 4753"}

The dpd.log captures the host (1.2.3.4) and server (5.6.7.8) SSL handshake and shows that each hello parses correctly as SSL, yet the subsequent packets fail to parse as SSL and reveal the traffic is not in fact SSL and thus warrants further investigation.

Outcome
Many security tools rely on well-known ports or simple signatures to identify traffic, or simply ignore traffic identified as encrypted since they cannot inspect its contents. Where other security tools would incorrectly identify this traffic as SSL and ignore it, Corelight's dynamic protocol detection caught the ruse, giving analysts the means to spot attackers disguising their communications as SSL on port 443. In reality, the traffic represents two attacker-controlled machines talking to one another using their own (non-SSL) protocol.

2. Parsing: finding malicious encrypted traffic with the ssl.log & x509.log

Attack scenario
An attacker uses a self-signed certificate to communicate over HTTPS with the intent of hiding their c2 activity amongst legitimate outbound web traffic.

Capability
Corelight performs full certificate chain validation for SSL and can automatically determine if a certificate is self signed. A threat hunter, recognizing that attackers commonly use self-signed certificates, can query their Corelight SSL logs to identify the use of self-signed certs ("validation_status":"self signed certificate") as the starting point for an investigation.

With this narrowed list of SSL connections, the hunter can then pivot into the corresponding x509.logs to examine the certificate details. Let's imagine they come across an x509.log where the certificate subject and issuer both share the same attributes, including the suspicious-sounding Common Name "CN=kingskillzzzz.ru". With a quick Google search of the domain they could pinpoint its role in a recent cyberattack, going from a hypothesis to threat discovery in a matter of minutes.

Outcome
Other security tools would likely have ignored the encrypted connection. Instead, Zeek has given threat hunters a shortcut to quickly filter out the noise and zero in on a malicious encrypted connection by automatically identifying self-signed certificates and parsing their details. For additional examples see "Examining aspects of encrypted traffic through Zeek logs".[2]

3. Context: investigating and resolving an incident

Attack scenario
An attacker injects malvertising into an unencrypted webpage that downloads an unencrypted payload on the devices of unsuspecting visitors. Once installed, the malicious payload establishes an SSL-based connection with its c2 server to receive encrypted instructions.

Capability
In this scenario Corelight would generate the following logs related to the attack:

• An http.log from the victim's web session
• A files.log from the malicious payload
• An ssl.log from the encrypted c2 connection

If an analyst backed into this investigation by filtering on SSL connections using self- signed certificates as described previously, they would next want to establish the attack chain that led to the encrypted c2 connection and to determine if other devices were compromised.

Since these logs all share the same UID, analysts could easily pivot backwards from the ssl.log UID into the associated http. log and files.log to see that the attack originated from a malware-laden website and take appropriate countermeasures.

Outcome
Instead of being stumped as to how attackers established an encrypted c2 communication within their environment, the analyst can use Corelight logs to identify the source of the breach and then quickly ascertain if any other IP addresses visited the malvertising domain or downloaded the malicious payload by querying Corelight's http.log and files.log

4. Analysis & Detection

Attack scenario
An attacker in a foreign country compromises a valid account to log into remote machines via SSH.

Capability
Corelight's Encrypted Traffic Collection contains more than 50 proprietary insights built around certificates and SSL, RDP and SSH traffic. With respect to SSH, these insights include:

- SSH client brute force success
- SSH client keystroke detection
- SSH large client file download
- And more..

Corelight appends these inferences as new fields to the ssh.log so an analyst investigating an alert or suspicious activity has direct access to these contextual clues about the connection.

Outcome
Where a compromised SSH connection might otherwise stand out as unremarkable, with Corelight's added context here an analyst would know that the login occurred via brute force, that a human was typing over the connection, and they requested and downloaded a large file from the remote machine,

raising suspicion to warrant a deeper investigation into this connection that might ultimately lead to an incident discovery.

The Corelight Labs team, led by company co-founder and Zeek's inventor, Dr. Vern Paxson, works closely with a select group of Corelight customers to analyze their live production traffic and develop novel network security insights. Given the prevalence of encryption in enterprise networks, developing cutting-edge insights around encrypted traffic has remained a core focus of research and development.

Consider the investigative use cases unlocked by the examples below of insights contained in the Encrypted Traffic Collection:

Custom Encryption Detection	Detects connections that are already encrypted without an observed handshake, which can indicate custom or pre-negotiated encryption.
DNS over HTTPs (DoH) Detection	Reveals when DNS queries are made to known DNS over HTTPS (DoH) providers to identify DNS traffic that would otherwise be obscured.
RDP Client Bruteforce Guessing	Reveals when an RDP client is guessing passwords.
RDP Password Authentication	Indicates that the client authenticated using an NTLM password that was provided before the connection was initiated.
SSH Agent Forwarding Detection	Reveals when SSH agent forwarding occurs between clients and servers, which may indicate lateral movement where adversaries have compromised SSH credential
SSH Scan Detection	Infer scanning activity based on how often a single service is scanned.
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 (e.g. TLS 1.0).
SSL Fingerprinting (JA3)	Create a hash of every SSL/TLS client and server negotiation for use in threat hunting or intel feed matching.

And dozens of additional encrypted insights…

For more detailed information about the Corelight Encrypted Traffic Collection, please contact us.

Conclusion

When organizations can't decrypt traffic due to cost, performance, privacy regulations or technical limitations, Corelight is the best way to derive insight. While encryption obscures payloads, it doesn't obscure the endpoints or timing of a communication, or the fact that a conversation took place, or didn't take place.

With Corelight, you can reliably detect commonly-used encryption protocols wherever they occur, comprehensively parse their cryptographic characteristics, and illuminate unencrypted traffic related to encrypted connections. Incident responders and threat hunters can use these insights to identify anomalies, detect suspicious activity, and fingerprint encrypted connections. You don't need to break and inspect - just shake the box.

[1] Maddison, John. Network Computing. Encrypted Traffic Reaches A New Threshold. Nov. 2018. https://www.networkcomputing.com/network-security/encrypted-traffic-reaches-new-threshold
[2] Bejtlich, Richard. Examining aspects of encrypted traffic through Zeek logs https://corelight.blog/2019/02/19/examining-aspects-of-encrypted-traffic-through- zeek-logs/

Defenders have always sought the high ground in order to see farther and turn back attacks. Corelight delivers a commanding view of your network so you can outsmart and outlast adversaries. We capture, interpret, and connect the data that means everything to defenders.

info@corelight.com | 888-547-9497