DETECTION ENGINEERING RESEARCH · NQ-DER-2026-001 · August 2026

Beacon Detection Without a SIEM: Multi-Signal Correlation as a Basis for Low-False-Positive C2 Identification

A methodology for distinguishing command-and-control beacons from benign periodic traffic using correlated, independently-scored network signals.

0xPersist · NorthQuinn Inc.

ABSTRACT

Beacon detection keyed on interval regularity alone produces an unmanageable false-positive rate, because benign periodic traffic—update polling, telemetry, time synchronization, keep-alives—is indistinguishable from command-and-control activity under a single timing signal. This paper presents a multi-signal correlation method that scores **ten independent signals** across Zeek connection, DNS, and TLS logs, spanning timing, volume, connection shape, certificate posture, and DNS behavior. Because a genuine C2 channel exhibits several of these characteristics simultaneously while benign services rarely do, requiring cross-signal agreement suppresses false positives without lowering sensitivity to the threat. Each signal is scored independently, weighted, ATT&CK-mapped, and reported with the evidence that produced it, yielding findings an analyst can audit rather than opaque scores. The method requires no SIEM, database, or agent and operates on a packet capture or Zeek log directory. It is implemented as the open-source tool **beacon-score**.

1. The False Positive Problem in Beacon Detection

Command and control beacons are among the most reliable footholds a defender can hunt. A compromised host that communicates with its operator must do so on some cadence, and that cadence leaves a recoverable pattern in network telemetry. The difficulty is not detecting periodicity. The difficulty is that the overwhelming majority of periodic patterns on a real network are benign.

The intuitive approach measures the interval between successive connections to a destination and flags destinations whose intervals are highly regular. Applied to production traffic, this approach produces an unmanageable false positive rate. Software update checks poll on a schedule. Telemetry and crash reporting fire on timers. Network time synchronization is periodic by definition. Application keep-alives, certificate revocation checks, mail polling, monitoring agents, and cloud SDK heartbeats are all regular. The modern endpoint generates a continuous stream of well-behaved periodic traffic to legitimate infrastructure, and a detector keyed on interval regularity alone cannot distinguish that traffic from a beacon.

This is the central tension of the discipline. A signal sensitive enough to catch the adversary is generally sensitive enough to catch a large volume of benign activity that resembles the adversary. Reducing sensitivity to suppress the noise reduces it for the threat as well. The problem cannot be solved by tuning a single signal, because the threat and the noise occupy the same region of that signal's output.

2. Multi-Signal Correlation as the Method

The resolution is not a superior single signal but a refusal to rely on any single signal. A genuine C2 channel typically exhibits several independent characteristics simultaneously. Its timing is regular. The volume of data exchanged per check-in is uniform, reflecting an encoded status poll rather than human-driven activity. The connection may be persistent if the channel is tunneled. The TLS certificate may be self-signed or short-lived because the operator provisioned it. The destination may be one the network has never previously contacted. The DNS may present high-entropy subdomains where a domain generation algorithm is in use.

Considered individually, each characteristic is noisy. Many legitimate services exhibit uniform byte ratios, self-signed certificates, or novel destinations. Legitimate services rarely exhibit all of these properties at once; a C2 channel frequently does. The detection question therefore shifts from “is the timing regular” to “how many independent beacon characteristics does this destination exhibit, and how strongly.” A destination that trips regular intervals, uniform byte ratios, and a self-signed certificate is a categorically different finding from one that trips regular intervals alone. The former warrants analyst time; the latter is a telemetry agent.

This is the operative thesis: single-signal detection produces noise, and multi-signal correlation produces findings. The engineering effort lies in selecting signals that are genuinely independent, scoring each honestly, and combining them so that agreement across independent signals is what elevates the score.

3. Signal Independence

The value of correlation depends entirely on the independence of the constituent signals. Five signals that all measure timing in slightly different ways do not constitute a multi-signal detector; they constitute one signal expressed five ways, and they produce correlated false positives. Effective signals are drawn from different layers of the traffic and different stages of the adversary's infrastructure. The method described here draws ten signals from three Zeek log sources, spanning timing, volume, connection shape, TLS posture, and DNS behavior.

Signal	ATT&CK	Basis
interval_regularity	T1071	Low coefficient of variation across connection intervals; automated origin.
jitter_low	T1571	Mean absolute deviation relative to interval mean; non-human timing.
byte_ratio_uniform	T1095	Uniform originator/responder byte ratios; encoded keep-alive.
session_frequency	T1071.001	High session rate relative to window; automated poll loop.
long_connection	T1071.004	Persistent long-lived connections; tunneled channel.
sni_cert_mismatch	T1573.002	TLS SNI does not match certificate CN/SAN; fronting or evasion.
dns_entropy_high	T1568.002	High Shannon entropy on subdomain labels; DGA or DNS tunneling.
short_cert_lifetime	T1587.003	Short certificate validity window; ephemeral infrastructure.
self_signed_cert	T1587.003	Issuer matches subject; attacker-controlled TLS endpoint.
low_ttl_variance	T1568	Low DNS TTL variance; fast-flux adjacent behavior.

Table 1. The ten correlated signals, their source layer, and ATT&CK technique mapping.

Each of these measures a property the adversary controls independently. Timing is a function of the malware's configuration; the certificate is a function of how the operator provisioned infrastructure; DNS entropy is a function of whether a generation algorithm was used. An adversary can defeat any one signal in isolation. Defeating all of them simultaneously, without introducing some compensating anomaly, is substantially harder, and that difficulty is precisely what the correlation exploits.

4. Defensible Scoring

A score is useful only if an analyst can determine why it holds the value it does. A model that emits a bare probability provides no basis for an investigator to distinguish a score grounded in strong evidence from an artifact of the scoring arithmetic, and the analyst will consequently either over-trust or ignore it. In the method described here each signal is scored independently on a zero-to-one scale, weighted, and summed into a final probability. Weights are configurable and a weight of zero disables a signal entirely, accommodating the differing noise profiles of different environments.

Critically, every signal that fired is reported alongside the evidence that caused it to fire. The output does not state merely that interval regularity scored 0.98; it states that interval regularity scored 0.98 on a coefficient of variation of 0.019 and a mean interval of 60.1 seconds across 179 sessions. An analyst reading that line knows without further inquiry that the destination was contacted almost exactly every sixty seconds, one hundred and eighty times, with negligible deviation. The per-signal breakdown converts the score from a verdict into an argument, and an argument is a thing a human can evaluate, override, or carry into an incident review.

5. Representative Output

```
$ beacon-score --conn conn.log --dns dns.log --ssl ssl.log --detail

#  Destination        Score   Confidence  Sessions  ATT&CK
-------------------------------------------------------------------
1  185.220.101.45     0.7830  HIGH        180       T1071 T1587.003 T1071.001
2  194.165.16.11      0.5446  MEDIUM      72        T1095 T1071 T1573.002
3  45.142.212.100     0.2930  LOW         60        T1071.001 T1568.002
```

Figure 1. Ranked beacon candidates. Scores derive from correlated per-signal contributions.

The leading candidate scored HIGH because multiple independent signals aligned: regular intervals, uniform byte ratios, and a self-signed certificate, each contributing to the total and each mapped to its ATT&CK technique. The third candidate scored LOW despite a comparable session count, having tripped only one or two weak signals. That separation, HIGH against LOW on superficially similar traffic, is the objective of the method. The score does not assert that all listed destinations are beacons; it directs analyst attention to the first.

Confidence bands (CRITICAL at 0.80 and above, HIGH at 0.60, MEDIUM at 0.40, LOW at 0.20, INFORMATIONAL below) translate the continuous score into triage priority. The ATT&CK mapping ensures the finding enters the vocabulary the surrounding security program already uses, serving analyst, detection engineer, and threat intelligence functions from a single output.

6. Operating Without Standing Infrastructure

Most beacon analysis tooling presupposes a mature platform: a database, ingestion pipelines, and endpoint agents. That assumption is reasonable for an established security operations center and unreasonable during incident response on an unfamiliar network, in a laboratory validating a detection hypothesis, or against a sensor capture from an unmonitored segment. The method described here operates on available artifacts, a packet capture or a directory of Zeek logs, with no database, agent, or standing infrastructure. When a log source is absent, the signals dependent on it are skipped and the remainder still score, so a connection log alone yields timing, volume, and connection-shape signals in the absence of TLS or DNS data. This portability is a deliberate design decision: the most useful triage is the one that can run the moment the data is in hand.

7. Conclusion

Beacon detection is an instructive case of a general principle in detection engineering: the difficult task is rarely detecting the pattern but separating the pattern of interest from identical-looking patterns of no interest. Sensitivity alone cannot achieve that separation, because the threat and the noise occupy the same sensitivity. Correlation can, because the threat and the noise differ in how many independent characteristics they exhibit concurrently. The practitioner should select signals drawn from genuinely distinct portions of the traffic and the adversary's infrastructure, score each honestly and retain its evidence, combine them so that cross-signal agreement drives the result, and surface the reasoning so that the human at the end of the pipeline receives an argument to act upon rather than a verdict to accept on faith.

Availability. The method is implemented as **beacon-score**, an open-source tool correlating the ten signals above across Zeek conn.log, dns.log, and ssl.log, producing ranked candidates with per-signal evidence and ATT&CK mapping, operating on a PCAP or log directory with no database or agent. MIT licensed. Source: github.com/0xPersist/beacon-score