STRIKE
By SecurityScorecard

REPORT

Catch Me If You Can

New Research Reveals CanOworms, a Proxy Network for Hire

Inside the Anonymization-for-Hire Network Serving Commodity Crime and Suspected Nation-State Hackers Alike

By Gilad Friedenreich Maizles

SecurityScorecard

Table of Contents

Executive Summary

SecurityScorecard's STRIKE Threat Intelligence Team identified a malicious 633-server "anonymity for rent" service, called CanOworms, spread across at least a dozen countries and rented to cybercriminals and suspected nation-state actors alike.

We are publishing these findings as black market services have decentralized, with different hacker groups specializing in a certain stage of the attack lifecycle, rather than the entire operation. Criminals lease infrastructure like this network the same way a business rents a commercial VPN. That lets them hide the true origin of malware, fraud, and intrusion attempts behind disposable IP addresses that get replaced faster than defenders can place them on a blocklist.

This infrastructure defeats the controls most organizations rely on by design. IP blocklists, geography, and hosting-provider reputation all fail against infrastructure built specifically to churn through and absorb takedowns. We traced previous iterations of this infrastructure, under a series of shells and fronts, back to at least 2017. It's an established, prolonged business venture, not a one-off campaign, and it's still expanding.

We developed and are publishing a fingerprinting method that identifies the network's servers regardless of which IP address they're currently using, giving defenders a detection signal that survives the churn. We could not identify who runs it: the operators built the network so that any attempt to trace the controller lands on an innocent third party. We partnered with FalconFeeds and Malanta to develop this research, and they have tied a suspected China-nexus actor, as well as two suspected North Korean (DPRK) actors, to this infrastructure, but only at low confidence and as one possible customer among many. The network itself is not any single group's infrastructure.

KEY FINDINGS

What it is:

A commercial anonymization network of approximately 633 rented VPN and proxy servers operating as a commodity service for cybercriminals, not one hacking group's private infrastructure.

Who's rending it:

Mostly operators of off-the-shelf malware, including Remcos, Quasar, NanoCore, NetWire, AsyncRAT, and Loki. Partner reporting also places a suspected China-nexus actor and North Korea-nexus actors on the same infrastructure, each at low confidence, and each as one tenant among many, not the network's owner.

Why IP blocking doesn't work:

The network is engineered to absorb takedowns. The threat actors burn and replace their leased address space continuously, with servers and providers spread thin across many countries to blunt reputation-based blocking.

How we can still detect it:

SecurityScorecard identified a technical fingerprint (a shared TLS certificate and two corroborating JARM and JA4X signatures) that identifies member servers regardless of IP address or location. A second, fleet-specific certificate (the nVpn CA family) independently marks membership as well. Full indicators are in Appendix A for direct use in detection tooling.

It carries live attack traffic:

Attack traffic transits the network. We observe netflow from many nodes that resembles SSH credential-spray against routers and cameras in commodity-crime geography. Because our netflow is pivoted on the mesh, we cannot confirm whether that activity originates with the operator or, more likely, with tenants routing their own tools through the relays. We describe it as suspected attack traffic exiting the fleet, not as the network itself conducting attacks.

The operator is unattributed by design:

We traced the business back to 2017 under an earlier series of shells, but the human controller is hidden behind innocent-looking rented servers. We investigated and ruled out one false lead.

Introduction

CanOworms is a certificate-defined network of roughly 633 proxy/VPN relays that operates as a commercial anonymization service for hire, not a single actor's command-and-control infrastructure. We identified it from one shared, self-signed TLS certificate (`O=kickass`) plus two corroborating JARM and JA4X fingerprints that together bind the fleet across roughly a dozen densely-populated /24s. It is spread deliberately thin over more than six unrelated hosting providers in as many countries.

The relays run a uniform Squid/SOCKS/OpenVPN/IPsec service stack and forward traffic outward to diverse backends. We assess with moderate confidence that the malware command-and-control indicators that other trackers publish against individual mesh IPs predominantly refer to this disposable front, not the true origin of the traffic. The IP range is leased and branded by a reseller identity, "PrivacyFirst" / MAXKO (Autonomous System (AS) Number `AS214366`). Independent prior reporting (abuse.ch, 2017; JPCERT/CC, 2021) traces this bulletproof-hosting lineage back nearly a decade under other cover identities.

Operators remotely manage the fleet from a small Czech control plane hidden on borrowed, benign-looking third-party boxes. The actual human operator remains hidden behind other procurement identities.

The clientele is a broad roster of commodity Remote Access Trojans (RATs), including Remcos, Quasar, NanoCore, NetWire, AsyncRAT, and Loki. Partner vendors Malanta and FalconFeeds, independently attributed several advanced actors to this activity, but only at low confidence, and no evidence points to potential ownership of one over the other, thus, in this report we will only refer to them as possible tenants among many, not as owners.

For defenders, the lesson is that IP-reputation, geolocation, and ASN controls are precisely what this network is built to defeat. The durable detection signal is the fleet's build fingerprints (the per-/24 certificate thumbprints, the JARM, and the JA4X), together with a possible chokepoint: a set of upstream “control plane” IPs that seem to be managing the mesh network but are not part of it.

1.

One certificate, six countries

Most investigations don't begin with a network. They begin with one detail that doesn't add up.

Ours began with a post. In early 2026, the threat-intelligence account FalconFeeds flagged a server it associated with an APT41-linked intrusion: a Cobalt Strike team server, reached by an XWorm sample, of the kind that surfaces and disappears in this space every week. On its own, it was unremarkable. We pulled on it anyway, because the host had one property that did not fit the pattern of a throwaway staging box: it was presenting a TLS certificate that was self-signed and deliberately malformed, and (this is the part that mattered) not unique to it.

The certificate's issuer and subject read `CN=*, O=kickass, L=Bar, ST=Foo, C=WS`. The `Foo/Bar` placeholders and the `kickass` organization name are the kind of values an operator types once into a certificate-generation script and forgets.

That is exactly why they are useful to a defender. A certificate meant to secure a real service carries a real name. A certificate meant only to wrap a socket in TLS so that a scanner sees “encrypted” carries whatever the operator feels like typing. And if the operator generates it once and deploys it across an entire fleet of their internet-facing assets, that string becomes a fingerprint that lets us tie the fleet together.

So we asked the obvious question: how many other hosts on the internet were presenting that same certificate? The answer was not “a few.” It was hundreds, arranged in tight blocks, across networks in half a dozen countries. (Although SecurityScorecard observed more than a dozen countries in the network, a handful of main hubs anchor the activity.) The single suspicious server was not a server. It was a visible edge of a much larger operation.

We call this set “CanOworms,” because SecurityScorecard’s STRIKE Threat Intelligence team recognizes it as persistent infrastructure rather than a one-off campaign. It represents sustained infrastructure across many months at a scale and running cost (leased address space, VPN/Tor storefronts, a live control plane) that imply substantial, ongoing investment.

The name is our label for the cluster of infrastructure we observed sharing this fingerprint and behavior. It is a convenience for discussion, not a claim that every host in it serves a single owner or purpose. Where the evidence supports unity, we identify it and share why. Where it does not, the name is just a name.

One certificate, six countries
(continued)

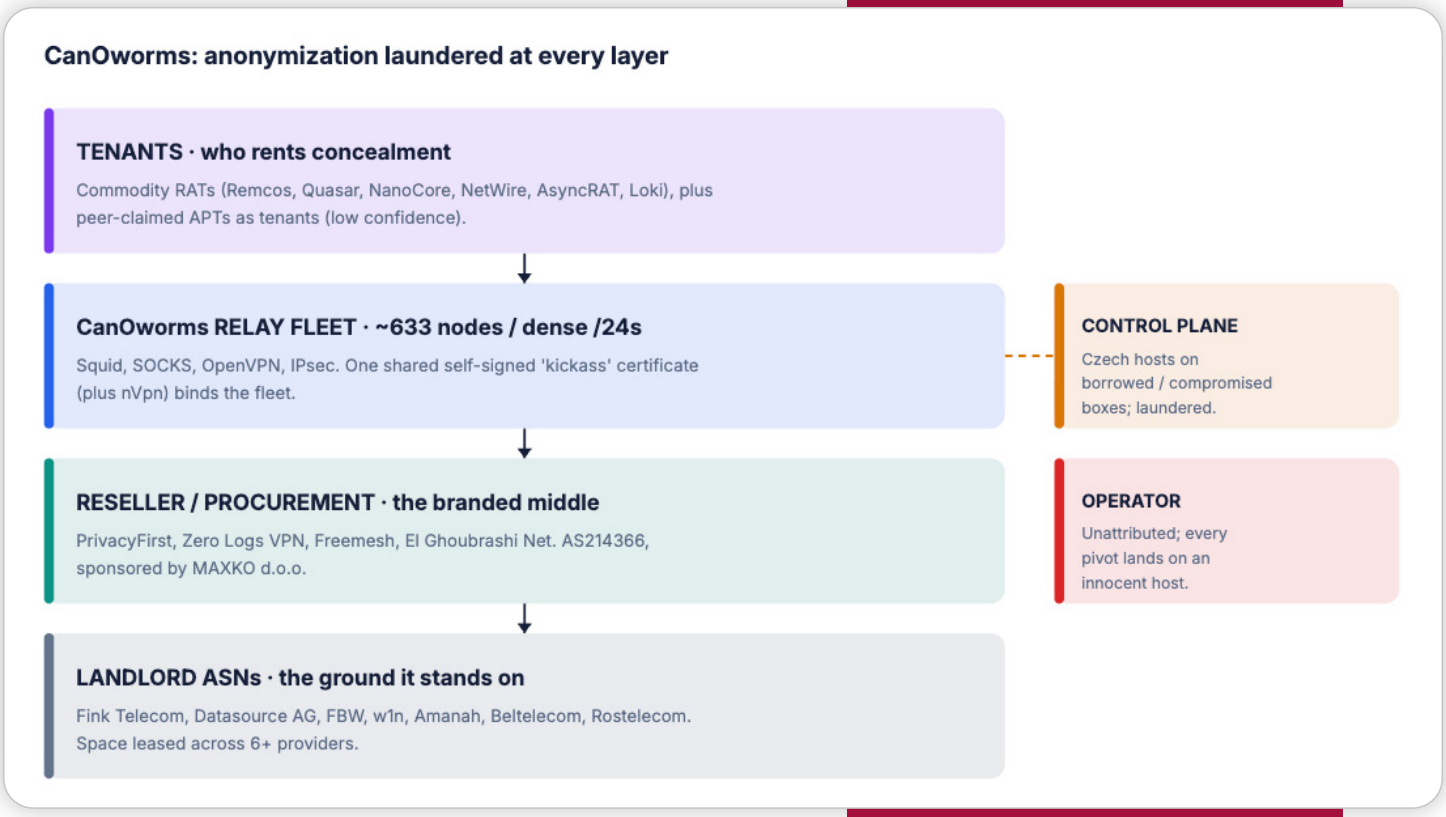


FIGURE 1
Architecture / four-layer model

2.

From one certificate to 633 hosts

Plenty of benign hosts share certificates, such as default appliance certs, load-balancer fleets, or hosting-provider boilerplates. What made this one worth trusting as a network fingerprint was that it was none of those things, and that it held up under investigation and corroboration.

First, the certificate was not a known default. It did not match the shipped certificate of any appliance, CDN, or common software package we could identify; the `kickass/Foo/Bar` issuer is not something a legitimate vendor ships.

Second, it is clustered in a way defaults do not. Rather than being sprinkled randomly across the internet, the hosts presenting it fell into densely populated address blocks, many contiguous /24s where a large fraction of the live hosts carried the same certificate. This is a trend we observe when one operator provisions ranges of machines from a template, not when unrelated parties happen to share a default.

We did not rely on the certificate alone. Two independent fingerprints corroborated the cluster. The hosts shared a JARM TLS-stack fingerprint of `21d10d00021d21d21c21d10d21d21de61583be67a70a87e745a4964791e063`, and a JA4X certificate fingerprint of `2bab15409345_2bab15409345_795797892f9c`.

JARM and JA4X describe how a server negotiates TLS and how its certificate is structured, respectively, and they are derived differently from the certificate string itself. When the certificate, the JARM, and the JA4X all agree across a set of hosts, you are no longer looking at a coincidence of one reused field. You are looking at a fleet built from one build process.

That agreement across three independently derived signals is what moved our investigation from "interesting" to high confidence that these hosts share a common provisioning origin.

How big is the fleet? SecurityScorecard's STRIKE team confirmed 633 CanOworms member servers concentrated in roughly a dozen densely-populated /24s using a three-signal membership test: a shared TLS certificate plus matching JARM and JA4X fingerprints.

Our scan data recorded 748 distinct IPs presenting the `kickass` issuer string, but 115 of those carried the string with mismatched fingerprints. We exclude these look-alikes because they fail the combination test, which is our threshold for inclusion in the tracked set.

Tellingly, whenever an IP address had only one single observation that correlated with the certificate metadata (as opposed to a continuous presence) it fell into that excluded group. No host carrying the full three-signal combination appeared only once, so we are not discarding real members that merely rotated quickly (a host that shows the full combination even once, we accept into our dataset of suspicious mesh nodes).

From one certificate to 633 hosts
(continued)

The densest blocks included `194.147.140[.]0/24`, `185.244.30[.]0/24`, `194.180.225[.]0/24`, `87.237.165[.]0/24`, `185.217.1[.]0/24`, `91.192.100[.]0/24`, and `79.134.225[.]0/24`.

We also observe a co-resident nVpn-certificate cluster, which we detail below. The co-resident nVpn-certificate cluster concerns largely the same machines rather than a separate fleet: most nVpn-certificate hosts also carry the kickass combination and are already inside the 633 member hosts, while a smaller set of certificate-only backbone nodes sits just outside it. The nVpn family widens the count modestly rather than doubling it.

These are a snapshot of what carried the fingerprint during our observation window of 180 days, not an all-time census; membership churns as blocks are added and retired, as well as occasional inter-block rotation to other IPs.

In other words, it is best to consider the fleet as a living membrane of proxy/SOCKS servers, rather than a static stack of IPs, with individual nodes rotating in and out of service.

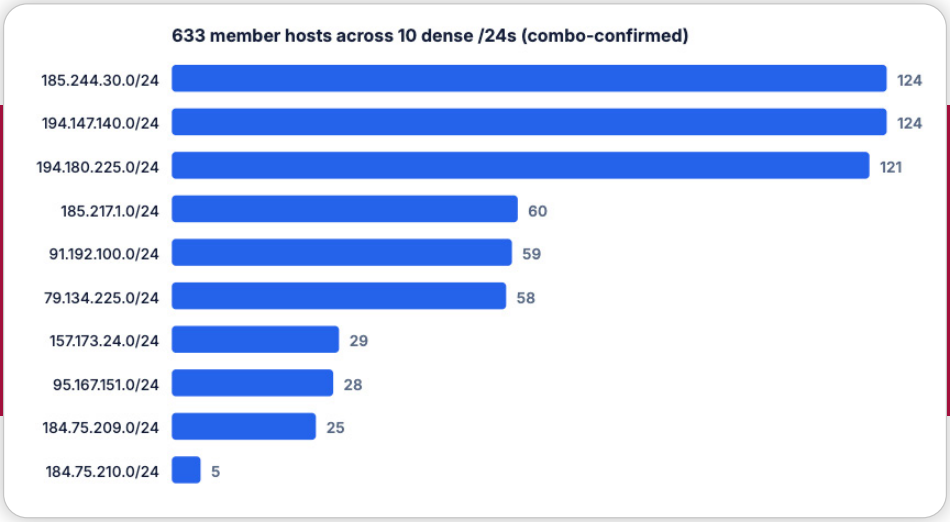


FIGURE 2
Fleet size by /24

Member-grade hosts per dense block (633 combo-confirmed total); issuer-string-only look-alikes are excluded, not counted as members.

As we expanded the cluster we found hosts that carried the kickass issuer string but whose fingerprints did not match. They instead carried a different certificate thumbprint or a different JARM or JA4X. Reusing a lazy issuer string is trivial, so a string match on its own is not membership, and we excluded those hosts.

The rule therefore is a combination of the following signals: shared certificate thumbprint, matching JARM and JA4X (on both squidCache and SOCKS services), and, as an extra layer of confirmation, the node-initiated heartbeat to the control plane (used as an added indication, not a threshold).

3.

What the machines actually are

With a defensible membership list, the next question was what was the intended purpose of the 633 machines.

They were not command-and-control servers in the ordinary sense. When we examined the services they exposed, we did not find malware panels or beacon listeners as the defining feature across the fleet; instead, we found a proxy and VPN service stack, repeated with striking consistency across the fleet: Squid HTTP proxy (commonly on 3127/3128), a SOCKS proxy (frequently 9998), OpenVPN (1194), IPsec/IKE (500/4500), and in places NFS (2049/20048).

This is the toolkit of an anonymization relay, an infrastructure whose product is passing other people's traffic, not the toolkit of a bespoke malware controller.

One detail in how that stack is served deserves close attention, because it can separate a hand-assembled box from a manufactured product. We found that the same TLS-negotiation fingerprint, the fleet JARM from the previous section, answers on both the Squid HTTP-proxy port (3127) and the SOCKS port (9998) on all but one of the 633 members. Neither proxy speaks TLS in its stock form, so for both to present the byte-identical fingerprint a single TLS-termination layer has to front both services. Each node is therefore a uniform, purpose-built relay image, more than a box with a few proxy packages installed side by side. That shared front is what wraps the services in the `kickass` certificate a scanner sees.

This co-residence with nVpn (a known provider adhering to “no logs” policy, that ensures no activity is recorded by the provider, irrespective of its legality or intent) raised a competing hypothesis: Are we simply looking at nVpn's ordinary commercial node fleet? The honest answer from the data is that it would appear not to be so. The nVpn certificate we track is not the commercial service's ordinary credential but a fleet-specific one, exactly as the kickass certificate is.

We have validated this claim using Driftnet’s 180 days entire data window. We found that this behavior is unique to our set and is not indicative of other nVpn nodes we detected, as a small set of nVpn nodes that returned in our search for a similarly structured certificate showed that separate nodes use individual certificates with a certain variation among them, and do not share between them. CanOworms nodes are the only ones reusing the same self-signed key — (sha1 - db6a990981b1a4020a7994a8258efa57bced2fb1).

We then profiled open TCP services, honeypots excluded, across three groups. Other genuine nVpn CA hosts were sparse: Squid on one of three, no SOCKS on 9998, no NFS. Our-certificate hosts that do not carry the kickass combo shared a private NFS storage backbone on 2049 (twelve of twelve) and ran Squid on about 58%, but ran no SOCKS. Our-certificate hosts that do carry the kickass combo ran the full front: Squid near 100%, SOCKS on 9998 about 93%, NFS about 97%.

What the machines actually are
(continued)

Through this cross service comparison we arrived at two conclusions:

1. **The NFS backbone that binds essentially every nVpn-certificate node to the rest of the fleet is absent from the other-nVpn hosts, so the nVpn certificate marks our set, not nVpn's general population**
2. **SOCKS on 9998 is the signature of the active relay fronts within that set, which is why it separates the kickass nodes from the certificate-only backbone nodes.**

The reading, at moderate confidence, is the reverse of "this is just nVpn": the nVpn CA certificate is a second membership fingerprint of CanOworms, like kickass, and nVpn-branding is one costume the operator wears rather than a commercial substrate the network merely rents. Only three other nVpn CA hosts existed in our window, a small comparison set, though that scarcity itself measures how completely this certificate space is the operator's.

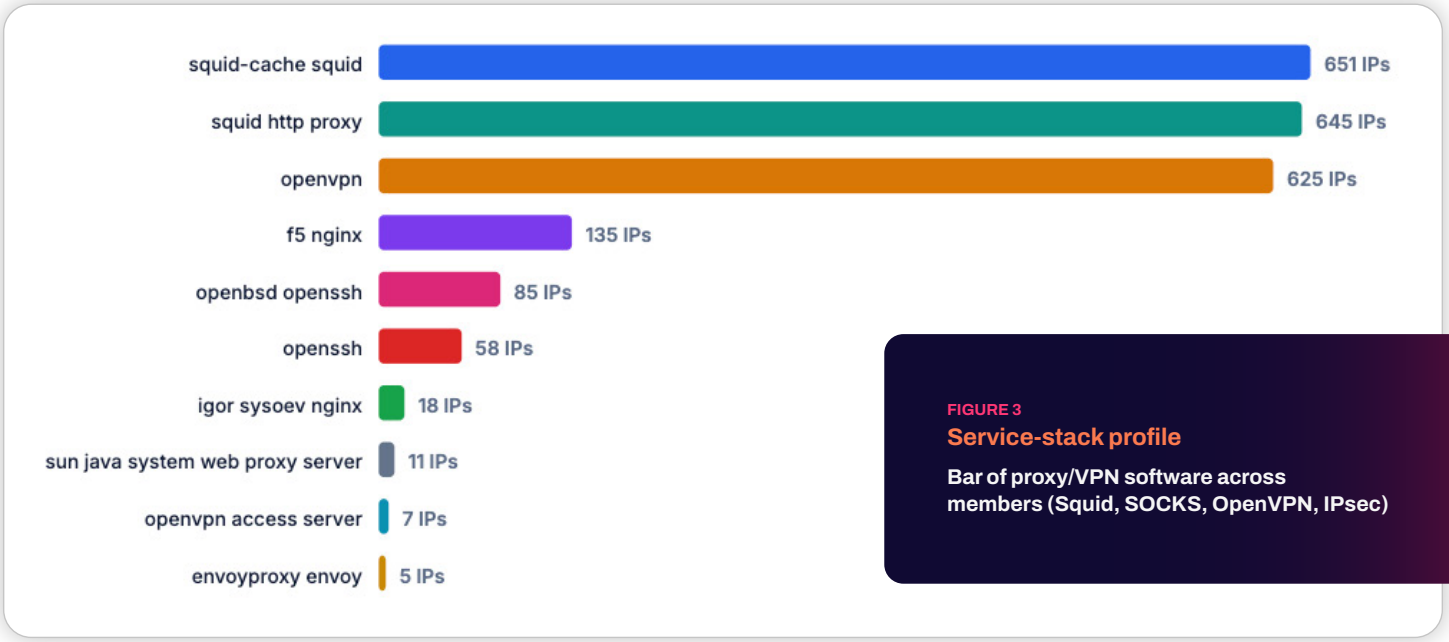


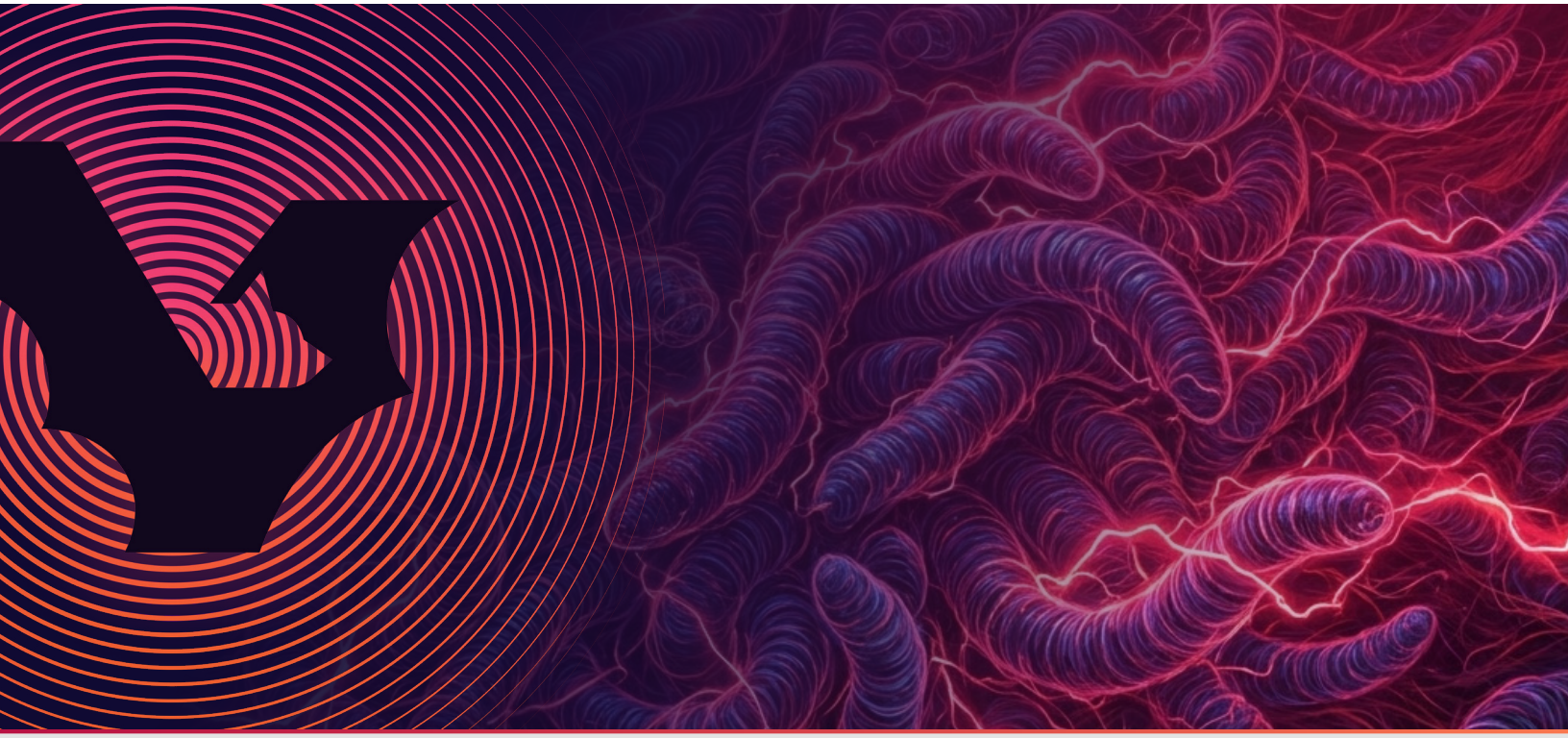
FIGURE 3
Service-stack profile
Bar of proxy/VPN software across members (Squid, SOCKS, OpenVPN, IPsec)

What the machines actually are
(continued)

That observation is the foundation for our detection and tracking method for CanOworms, so we will examine each aspect of it in turn. At the base is what we can simply see: across the fleet these hosts run a proxy/VPN service stack. On that foundation sits a first, near-certain reading: machines built as Squid/SOCKS/OpenVPN relays are built to relay traffic on behalf of others.

Resting on top is the working hypothesis that we tested next: that the fleet functions as a shared, for-hire anonymization layer. SecurityScorecard's Driftnet engine gave us the port-scanning visibility to test this directly: what was initially suspected as separate C2 machines, hosted on individual nodes did not return results that correlate with known C2 frontend infrastructure, but rather behave more as a relay standing in front of a hidden backend like a "bouncer" IP, not unlike a CDN service for malware.

Over the course of our investigation, the fleet's scale grew, rather than holding steady, and it was spread deliberately thin across many hosting providers and countries rather than concentrated with one, a distribution we return to when we look at who owns the address space. New blocks appeared during the months we watched, including the aforementioned co-resident certificate family associated with the "nVpn" branding (CN=nVpn CA, OU=nVpn, O=nVpn) deployed on most of the same machines.



4.

Front, not C2

As our investigation progressed, we assessed with moderate confidence that CanOworms is a front-facing relay layer. Throughout, we identify the evidence that would raise or lower our confidence in our assessment.

The first support is the service stack in the previous section: a machine built as a Squid/SOCKS/OpenVPN relay is built to forward traffic on behalf of others. The second is behavioral. Across six months of netflow in which one endpoint of every recorded flow is a CanOworms host, the hosts that other vendors may have labeled as a C2 are not traffic sinks. They forward outbound heavily and to diverse destinations. A host tagged as a Quasar controller, for example, showed hundreds to thousands of outbound flows fanning out to many external addresses on a mix of ordinary and ephemeral ports. That is behavior consistent with a proxy passing sessions through, and inconsistent with a terminal controller receiving calls home from malware.

The limitation is that our netflow was collected by pivoting on the mesh hosts, so it shows their flows but cannot, by construction, prove that any given outbound flow is the forwarded second leg of a specific inbound malicious session. We can say the hosts forward traffic at scale in a proxy-shaped pattern, but we cannot stitch individual sessions end to end. We hold the assessment at moderate confidence as a result.

While the absence of evidence is not evidence of absence, the hypothesis also sits well with the C2 shaped hole in the network configuration of all CanOworms IPs — as none of the IPs from the network that were labeled as a C2, returned any scan data that matches known configurations for the commodity malware that was used, or any deviation from the mold configuration to a similar extent, for that matter. It also explains why the malware families seen “on” the network are diverse and unrelated to each other, if indeed they belong to different end-users renting the same relays, not to one operator. It also explains the pattern, discussed in the tenants section, in which a host is reputationally clean by antivirus consensus yet relays many distinct malware samples. All converging into our leading hypothesis: traffic goes through, not to, nodes.

The competing hypothesis merits consideration: it remains possible that a subset of these hosts genuinely terminates C2 for specific families, with the proxy stack serving as a front. We do not claim the evidence forces a single answer for every host. But the weight of the evidence, in our opinion, seems to point in that way, and indicator lists that treat each mesh IP as a malware origin are, in most cases, cataloguing the front rather than the activity behind it.

For defenders, that distinction is the difference between blocking a symptom and finding the disease, which is why the rest of our research report keeps the front and the origin in separate boxes.

5.

The same story, told by the clock

Netflow cannot see payloads, but it can see who talks to whom, in what direction, and on what rhythm. Over six months the rhythm of CanOworms corroborates the relay reading and reveals how the fleet is run. The standing caveat from the last section applies throughout: this netflow is pivoted on mesh hosts, so it describes their conversations, not an unbiased view of every party's traffic.

The first surprise was what we did not see. Despite the "mesh"/"Freemesh" branding, in six months of flows from these nodes we never observed one mesh node talking to another. There was no peer relaying, no proxy chaining between the nodes themselves (an absence of evidence from our mesh-anchored vantage, not proof that none exists).

What the data shows is consistent with a centralized managed network: independent exit nodes that each phone home to a small central control plane and serve traffic outward. That distinction matters operationally because a centralized network has a bottleneck you can press on: a commodity VPN node that finds a local peer to be unreachable would (under normal conditions) reroute its traffic to another node. With CanOworms, not talking to one another means that verification of ownership, authentication of what gets through and what remains blocked would hypothetically demand direct instruction from the control plane, and might hamper the network's very functionality if the connection is severed.

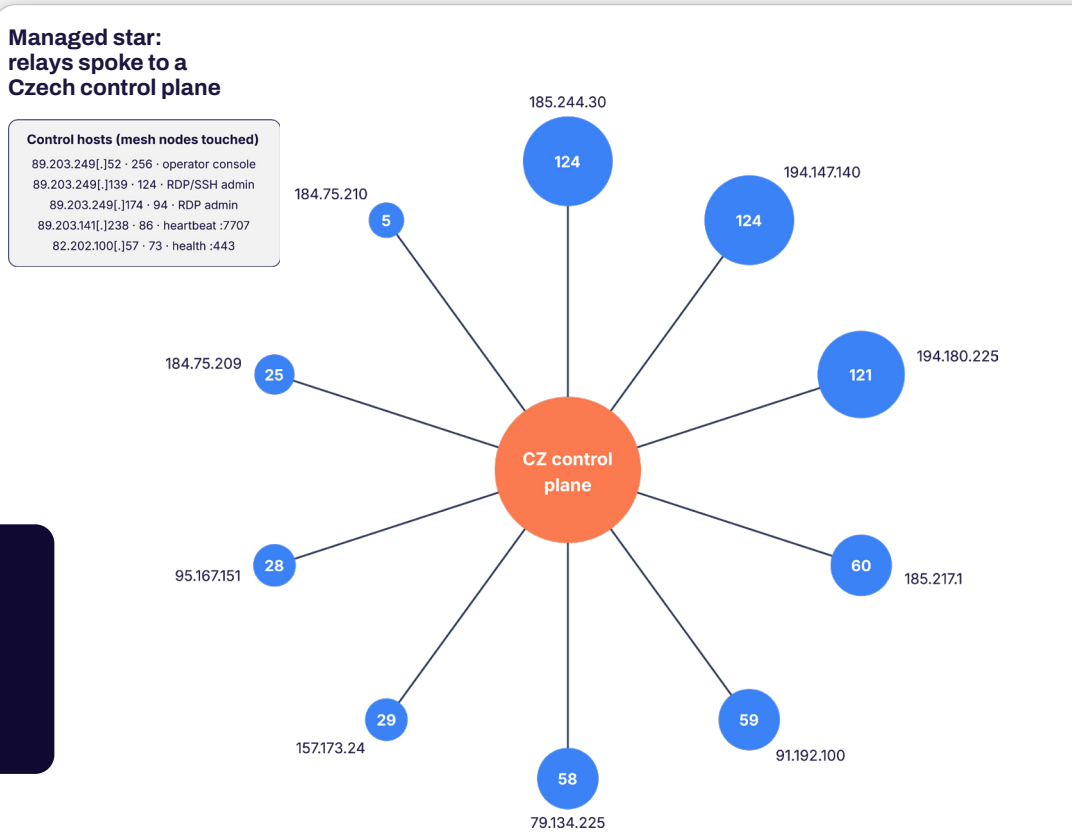


FIGURE 4

Topology: managed centralized mesh

Relay /24s spoking into the Czech control cluster, with the "zero node-to-node" callout

The same story, told by the clock
(continued)

The control plane resolves into a compact, consistent set of hosts, all in the Czech Republic, each converging on a large fraction of the fleet on stable private ports:

Control host	Mesh nodes touched	Role	Key port(s)
89.203.249[.]52	~256	operator console: phone-home sink plus interactive RDP/tasking	52335 in; 3389/mgmt out
89.203.249[.]139	~124	node administration	3389 (RDP), 22 (SSH)
89.203.249[.]174	~94	node administration	3389 (RDP)
89.203.141[.]238	~86	fleet heartbeat collector	7707 (~35 s beat)
82.202.100[.]57	~73	secondary health check	443 (~2 min beat)

The most telling single feature is that two clocks run at once in the same data. One is infrastructure telemetry: approximately 65 to 86 nodes send a regular approximately 35-second heartbeat to 89.203.141[.]238:7707, a signal indicating a node is online, exactly what you would build to monitor a fleet of relays. The other is malware behavior: scattered external hosts send low-jitter 60-to-125-second callbacks to RAT-specific ports on individual mesh nodes (for example a NetWire beacon from 45.187.92[.]54 to 79.134.225[.]25:2001 at a 58-second interval).

A fleet-wide 35-second beat pointed at one box is management; a low-jitter beat from scattered hosts pointed at RAT ports is a malware implant calling home. Both are present simultaneously. We catalogued 358 such external-to-node beacon pairs.

The nodes never chain to each other, but the clients move between them, and that is its own tell. Over the window, 301 external clients held sustained sessions with two to six different mesh nodes and rotated across them within seconds. One client handed off among three nodes with an average gap of 2 to 4 seconds while keeping its activity alive, and 1,922 clients spread their contact across nodes in two or more different /24s.

Servers that never talk to each other, yet clients steer across them like interchangeable exits behind a load balancer. This behavior is more consistent with a coordinated, CDN-like fleet (round-robin or failover across equivalent exits) than with a bag of wholly independent boxes. This is cadence-and-direction inference at moderate confidence: we can see the rotation, not the mechanism behind it.

When we restrict to the malware-labelled C2 ports specifically, we do not see clients rotating across nodes. That pattern lives in the fleet's general and BitTorrent-like traffic, not in the tenant C2. We looked precisely because a malware client hopping across fronts would be strong evidence that the nodes proxy to a hidden backend rather than hosting the C2 themselves. We did not find it, so we read the rotation as a property of the relay service as a whole, not as proof about any one tenant's C2.

Because our netflow is anchored on the mesh hosts, we only see the control hosts' conversations with the fleet, not what they do across the rest of the internet. We therefore make no claim that they are internet-wide port scanners, even where some of their behavior toward the nodes may resemble sweeping. That is a question we cannot answer in the scope of this report. What we can say is that the "C2" nodes' outbound traffic, where visible, fans out to diverse external destinations on ordinary and ephemeral ports and that it exhibits the forwarding profile of a proxy carrying many customers, not the tight callback of a single implant family.

6.

Who's renting?

We investigated the network's tenant base as a distinct research question: who rents this relay layer, and for what purpose. Answering it required cross-referencing multiple independent evidence types, since no single source carries sufficient weight on its own.

Public malware-tracking feeds place a recognizable roster of commodity RATs and stealers against mesh addresses, including Remcos, Quasar, NanoCore, NetWire, AsyncRAT, and Loki. abuse.ch's ThreatFox, for instance, recorded specific host:port C2 entries on mesh addresses: a Quasar entry on `91.192.100[.]36:8084`, a NanoCore entry on `79.134.225[.]18:3371`, and Remcos entries on hosts in `194.147.140[.]0/24`, some carrying near-maximal confidence and, in a couple of cases, an associated MalwareBazaar sample. This roster defines the network's clientele: commodity families sold and used broadly, not necessarily any one advanced actor.

The evidence carries one methodological limit in this section. We saw malware samples in contact with these IP-port pairs, often on ports that are themselves signatures of the named family, but we were never able to reach or authenticate to a control panel behind them. We therefore describe these as vendor-tagged C2 endpoints the malware used, which is consistent with, but not proof of, a live malware controller running on the host.

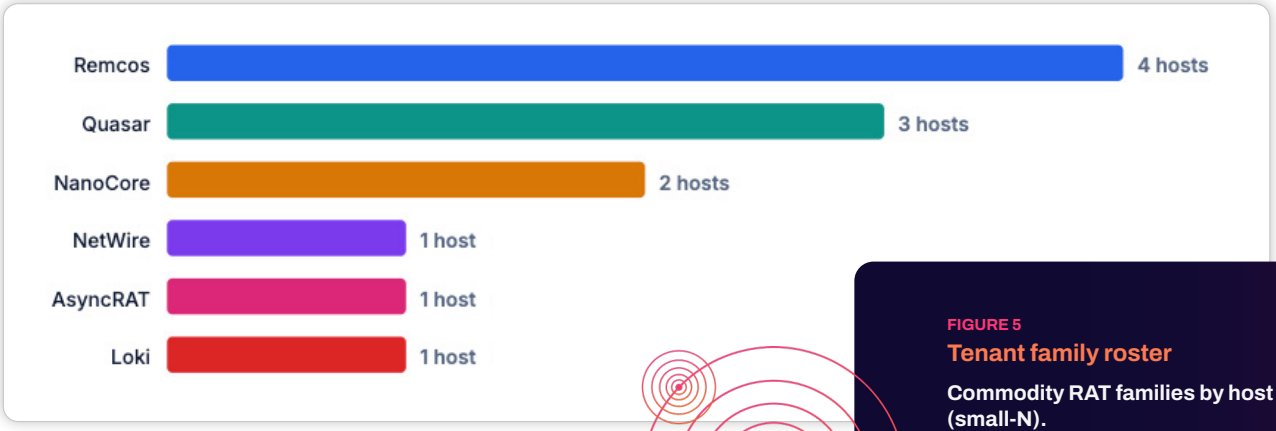


FIGURE 5

Tenant family roster

Commodity RAT families by host count (small-N).

Who's renting?
(continued)

We treated feed labels as leads, not verdicts, and graded each host by how many independent kinds of evidence agreed: a feed label, our own scan/netflow observation, a VirusTotal file relationship. Two feeds echoing each other is weaker than two different classes of evidence agreeing.

We also filtered for timing: a malware sample only supports "active on this host during its mesh membership" if the sample's activity overlaps the window in which we actually observed the certificate on that host. Applying that discipline, a handful of hosts reach high confidence as commodity-RAT C2 endpoints because a feed label, a live netflow observation of traffic on the named port, and VirusTotal file contact all line up. Examples are `91.192.100[.]36:8084` (Quasar; feed-named, 14 flows observed on `:8084`, in-window malware samples) and `194.147.140[.]242:1998` (Remcos; feed-named, flow on `:1998`). Others sit at moderate (feed plus one corroborating type) or low (single feed) confidence. The full graded table is in Appendix C with a confidence rubric.

One finding warrants particular emphasis, as it is where the tenant analysis and the "front, not C2" thesis converge on the same data. A number of hosts were reputationally clean, according to an antivirus-engine consensus, some even with slightly negative community scores, while simultaneously in contact with numerous distinct malware samples (frequently the maximum our query returned).

Read as "each host is a malware origin," that is a contradiction. Read as "these are relays carrying many customers' traffic," it is exactly what you would expect: the malware transits the proxy, the proxy itself is not the malware, and reputation engines (which score the endpoint) see nothing to flag. We present this as corroboration of the relay thesis, at moderate confidence, and note the caveat again: communicating-file relationships establish contact, not the direction or purpose of a specific session.

We also investigated a hypothesis that a second tier of the network is built on compromised edge devices, but we are not asserting this hypothesis. As the next section shows, the hosts the mesh reaches out to are strikingly edge-device-heavy, which is consistent with edge-device recruitment. But we could not separate genuine compromised-relay members from victims, honeypots, and scanner noise to a standard we would publish. We flag it as an open and unconfirmed question. Resolving it would require consistent membership signals (shared services suite, combined unique fingerprints, possibly with some form of management heartbeat) on device-class hosts, which we do not currently have.

7.

Who's on the other end

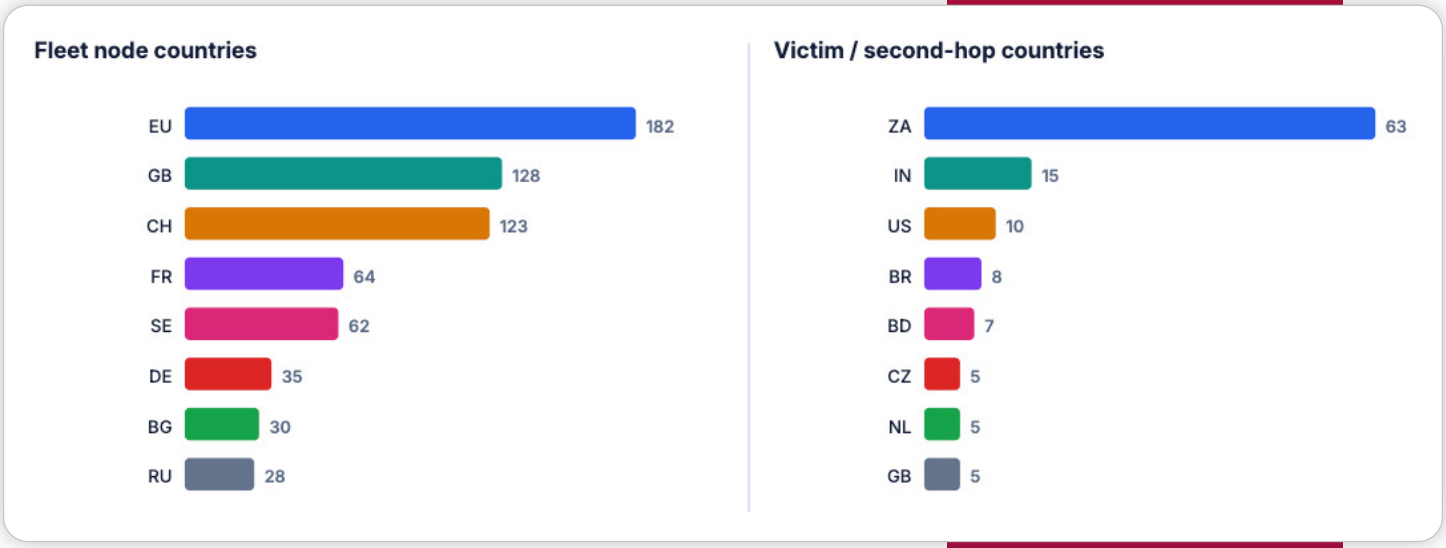
A relay network has two kinds of counterparties: the hosts it reaches out to and the hosts that connect into it. Both are visible in our data, and both point away from the "elite espionage platform" framing and toward commodity crime.

The hosts connecting in, the population beaconing to the RAT-specific ports on mesh nodes, sit overwhelmingly in South Africa, Bangladesh, India, Pakistan, and Brazil. That geography suggests commodity botnet victims and proxy customers, not European or East-Asian government espionage targets. Among the 358 beacon pairs we catalogued, the tightest-cadence, lowest-jitter beacons are the highest-confidence individual compromised hosts (for instance a 125-second, near-zero-jitter beacon to `194.180.225[.]240:18644`).

FIGURE 6

Geography: fleet vs. victims

Paired bars of node countries vs. counterparty countries: the commodity-crime geography



Who's on the other end
(continued)

The hosts the mesh reaches out to are more interesting, because they are how the network appears to grow and act. We observed two outbound patterns. The first is distributed SSH access activity, with many different mesh nodes each SSHing into roughly one host per target /24 inside a tight window and netflow behavior that is consistent with a coordinated credential-spray run from the fleet rather than from a single box.

This pattern is visible in the flow records. We did not capture the sessions, so we only describe the shape of the activity, not a confirmed brute-force payload. Concrete examples from June 2026 include 48 mesh sources hitting 50 hosts in `194.36.173[.]0/24` (between June 14 and 18) and 43 sources hitting 45 hosts in `194.36.172[.]0/24` (between June 16 and 18), with a similar shape against `91.243.119[.]0/24`, `87.229.95[.]0/24`,
and others. The pattern is that of sustained footholds: a small number of long-lived SSH sessions that look like a persistent access rather than spraying, for example `157.173.24[.]247` to `207.154.230[.]221:22`, with 236 flows over eight days.

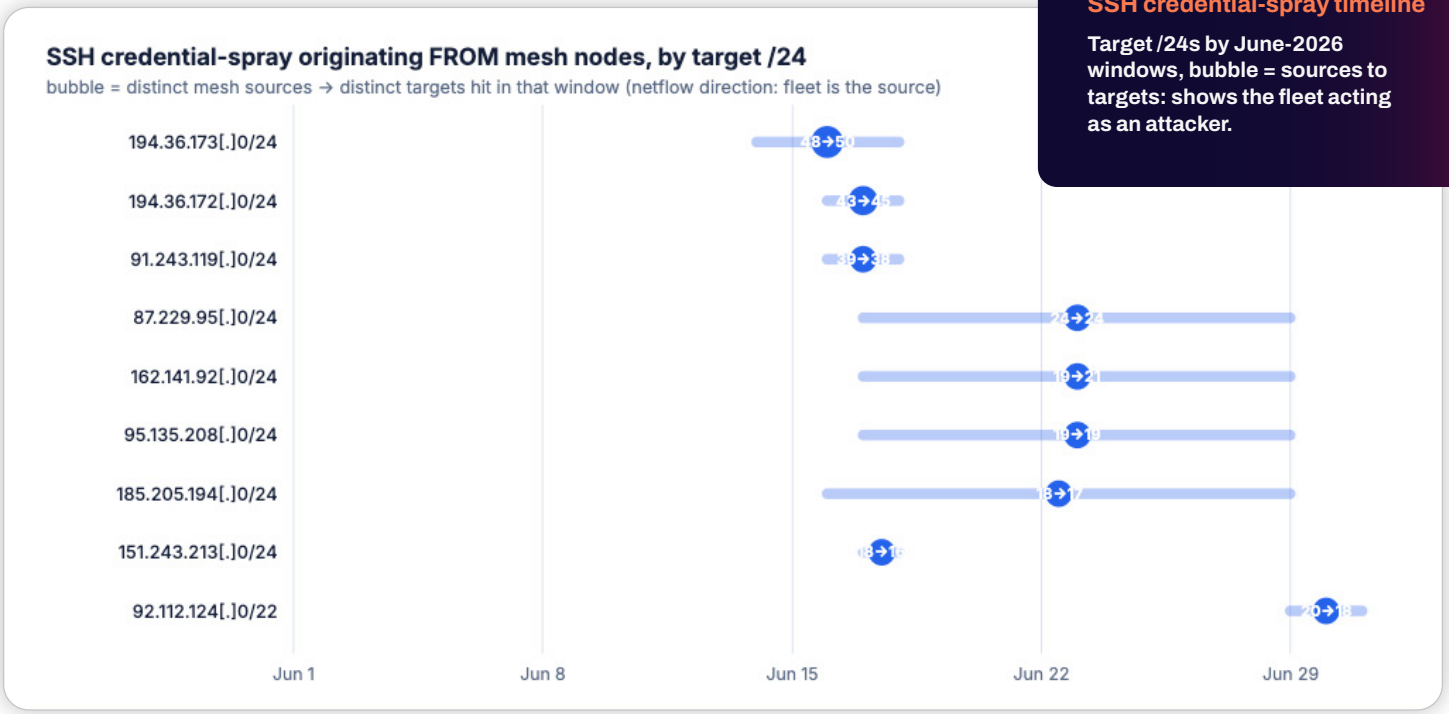


FIGURE 7
SSH credential-spray timeline

Target /24s by June-2026 windows, bubble = sources to targets: shows the fleet acting as an attacker.

Who's on the other end
(continued)

What are these outbound targets? We pulled a scan profile of approximately 154 second-hop hosts the fleet contacted, and the composition is instructive. About 41% sit in South Africa, with India, the United States, Brazil, and Bangladesh following. The device mix is dominated by home/SMB edge equipment, with MikroTik routers, TR-069 remote-management CPE, and Hikvision cameras making up the plurality, with Linux VPSs the other large share.

In sum, the fleet's outbound activity is heavily aimed at consumer and small-business edge devices in commodity-crime geography.

Attribution of this activity, however, remains an open question. The data supports three distinct readings, and we treat them as separate hypotheses rather than resolving them into one. It is consistent with the network recruiting additional relay capacity by compromising edge devices. It is also consistent with ordinary botnet victimization of those devices. Most economically of all — and directly downstream of the front-not-C2 thesis — it is consistent with a tenant's scanner running out through the proxy, which is exactly what a relay-for-hire is built to carry.

We have no evidence tying these SSH sessions to the control plane or to the operator, so we do not attribute them to the operator. The narrow pattern the data does establish, at moderate confidence, is that attack-shaped traffic exits through the fleet.

That is consistent with the relay model rather than proof that the operator is the attacker. We have the shape of the activity in flow records, not a packet capture of the sessions, so we can describe what leaves the nodes without confirming who initiated it.

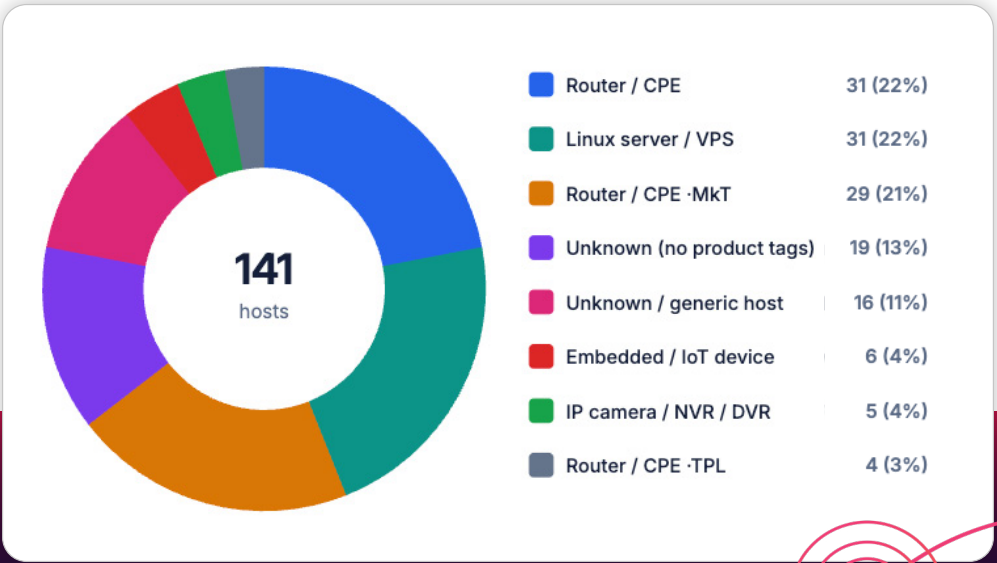


FIGURE 8
Second-hop target device mix

Donut of what the fleet reaches out to: the edge-device plurality (MikroTik, TR-069 CPE, Hikvision)

8.

Landlords, resellers, and the mesh

To understand a relay network you have to understand where it lives, and that means reading the registration records for every dense block in the fleet. The result is a layered picture, and we treat each one in turn.

The address space the fleet sits on belongs to a spread of unrelated hosting networks, among them Fink Telecom (79.134.225[.]0/24), Datasource AG (91.192.100[.]0/24), FBW Networks (194.180.225[.]0/24), a w1n ltd range (185.217.1[.]0/24), Amanah Tech (184.75.209[.]0/24), a Beltelecom range (178.124.140[.]0/24), and a Rostelecom range (95.167.151[.]0/24), alongside an autonomous system of the network's own. We did not find evidence to tie any specific hosting provider to the active operation of CanOworms, but rather only as the owners of the ranges from which it operates. These are like landlords, providers who lease the ground the relays use. Several have long-standing reputations in the abuse-tracking community, independent of our research.

Sitting between the landlords and the relays is a reseller identity that recurs across the fleet with a consistency that is itself a finding. Reading the customer-assignment records on the mesh blocks, the same brand family appears as the registered customer across six different, unrelated hosting networks: variants of "PrivacyFirst," "Zero Logs VPN," "El Ghoubrashi Net," "Freemesh Networks," and related labels, tied to the autonomous system AS214366 and sponsored by the Croatian registry member MAXKO d.o.o.

Yet again, registration data of this kind indicates the procurement identity and who leased and branded the space, not who operates the machines on it. For a bulletproof-style anonymization reseller, putting its own brand on rented space is the product: it absorbs the abuse complaints and shields whichever downstream customer actually runs the boxes. We therefore assess, at high confidence, that this PrivacyFirst/MAXKO identity is the reseller that procured much of the relay space. Still not precisely the operator.

Landlords, resellers, and the mesh
(continued)

Two measurements support that conclusion. First, SecurityScorecard found the CanOworms certificate is present on only a small carve-out of the reseller's footprint: of roughly 120 address blocks we could associate with the PrivacyFirst/MAXKO identity (on the order of 40,000 addresses), the mesh certificate appeared in only about sixteen, saturating a handful of small sub-allocations while the large majority of the reseller's space showed no mesh certificate at all.

Second, a majority of the certificate-bearing hosts sit outside that reseller's registered footprint, and instead sit in the space of the other landlords named above. Together, these measurements are inconsistent with the reseller and the mesh being coextensive. They are consistent, instead, with the mesh representing a single customer's deployment, drawn from this reseller alongside several other suppliers. We therefore describe PrivacyFirst/MAXKO as the most prominent and most closely tied supplier of relay space to CanOworms, not as CanOworms itself. (These proportions come from reverse-registration enumeration.)

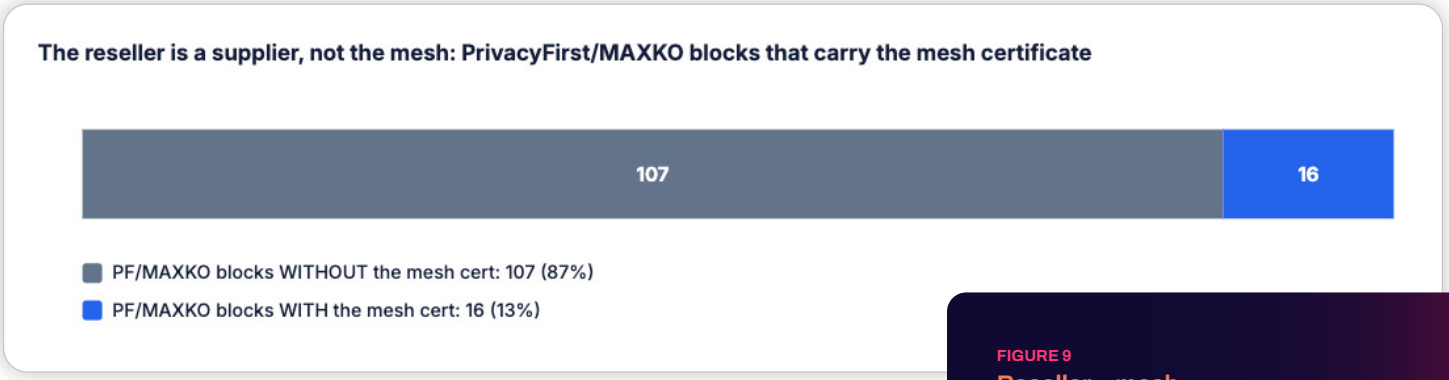


FIGURE 9

Reseller ≠ mesh

Proportion bar: of ~123 PrivacyFirst/MAXKO blocks only ~16 carry the cert, and ~60% of cert hosts sit outside PF space

9.

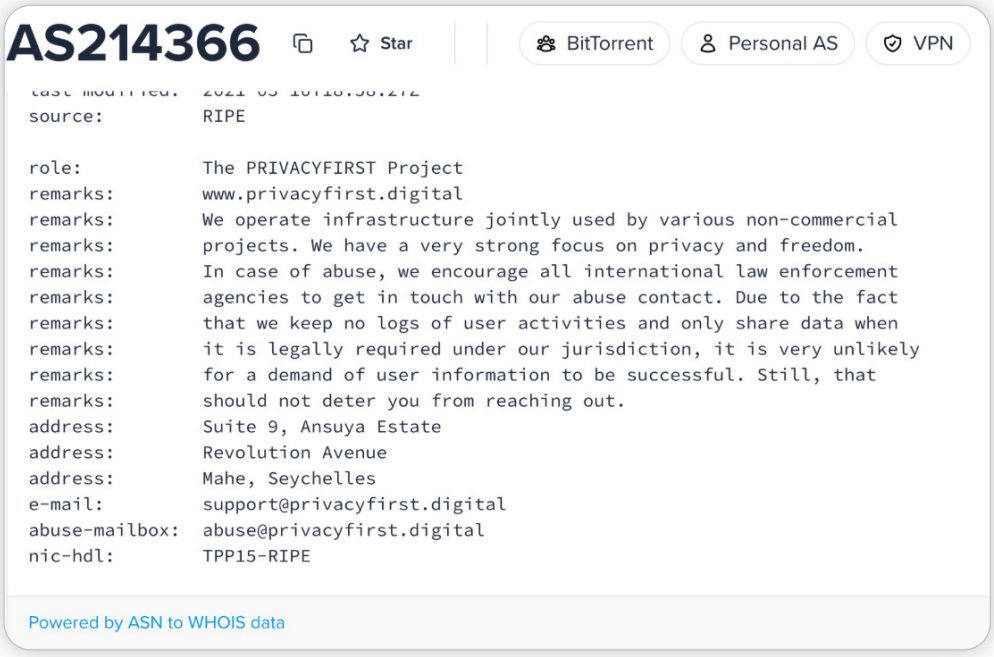
The commercial architecture

As part of this investigation, we examined whether the network's design reflects an improvised campaign or a sustained commercial operation. The evidence, addressed in turn below, indicates the latter.

Consider how CanOworms sources address space. The blocks are leased from the “gray” IPv4 market, a relatively fast-paced, changing-hands range that can be temporarily leased in and out of another entity's service. One of the core ranges, `194.147.140[.]0/24`, carries the registration lineage of a dormant Iranian allocation (originally Patron Technology Persia), repurposed gray-market space moved through IP-broker maintainer objects (`DeDServer`, `MARSHALL-MNT`, associated with IP Volume Inc.) and re-announced under the Emirati-fronted `AS214366` in 2024.

We assess that this is Iranian-registered space leased and operated by the reseller, not evidence of Iranian operators. Leasing repurposed, historically-isolated space is a deliberate economic hedge. When a leased block inevitably gets blocklisted by the abuse community, the operator drops the lease and rents another, leaving its core untouched.

Consider the storefronts. The same identity fronts a portfolio of consumer-facing anonymity brands ("Zero Logs VPN," "Freemesh Networks," "El Ghoubrashi Net," "The PRIVACYFIRST Project"), each marketed on absolute-privacy, no-logs rhetoric, with a Seychelles-registered abuse contact whose published notice essentially pre-declares that abuse requests will probably fail, though users are encouraged to try anyway.



The commercial architecture
(continued)

Consider as well that the parent, MAXKO, openly markets Tor-as-a-Service. It advertises privacy hosting for Tor nodes across its European and African autonomous systems and even offers a discount code for VPS bought to run Tor relays. On the same leased Iranian block (`194.147.140[.]0/24`), the reseller layer itself runs Tor relays whose contact information (`torNNN@ip-connect[.]info`) ties them to the Ukrainian upstream IP-Connect. Those relays are co-located with CanOworms proxy nodes on that operator-controlled /24 rather than being certificate-defined mesh members.

We note that the combination of Tor hosting, "no-logs" VPN branding, and proxy-for-hire in the same operator's space is consistent with a business model that affords its users plausible deniability.

The upstream fabric is likely chosen for resilience in the same way, with multi-homed transit through an Austrian provider (Evolus IT), the Ukrainian IP-Connect, and a German DDoS-mitigation network (combahton), so that the anonymization service stays reachable under pressure.

Considered in concert, the picture is coherent and, we assess with high confidence, deliberate. The core of CanOworms is not improvised infrastructure spun up for one campaign; it is a commercial anonymization service, and its design choices are the choices of a business optimizing for abuse-resistance and deniability.

CanOworms is the infrastructure expression of an anonymization-for-hire business whose every layer — from leased disposable IPs to VPN/Tor branding to the offshore abuse shell to resilient transit — is engineered to rent concealment to its clientele and to survive the takedown attempts that covert operations invite.

10.

The paper trail: 2017 to 2026

The reseller identity is worth dwelling on, because it does not begin in 2026. It has a paper trail, and following it changed how we look at CanOworms as a network.

The registration and contact records tie AS214366 to an individual named in the RIPE data, sponsored through the Croatian LIR MAXKO d.o.o., which itself maintains a dual corporate presence in Croatia and Egypt. The IPv6 assignments carry the label "Elghoubashi-Net," a label which reflects registry-label data only; we do not claim any individual bearing this name is knowingly involved. The assignments route their abuse contact through the Seychelles shell "The PRIVACYFIRST Project" (privacyfirst[.]digital).

Here we are careful to separate tiers. Observed (from RIPE and public corporate records): the ASN, the MAXKO sponsorship and shared maintainer objects, the brand family, the Seychelles abuse role. We infer with moderate confidence that the outward-facing registrant identity is a flag of convenience and the operation's centre of gravity is the MAXKO principals. This assessment relies on the sponsorship, the shared maintainer objects, and a surname match to the Egyptian corporate address, which is suggestive but not proof.

What makes the lineage compelling is that independent, prior reporting from other researchers lands on the same identities under earlier names, and our live data echoes it:

2017 2018 2019 2020 2021 2022 2023 2024 2025 2026

In 2017, abuse.ch documented a bulletproof-hosting operation ("AnMaXX," tied to a "no-logs" VPN front) reselling space through the same upstream ecosystem and hosting Adwind, NanoCore, and Remcos, the same family of commodity RATs we see today. Our own dataset contains a present-day "Anmax"-labelled assignment on 95.167.151[.]0/24.

In 2021, JPCERT/CC's JSAC presentation identified "The PRIVACYFIRST Project" as hosting a substantial share of tracked NanoCore command-and-control at the time. We independently observe NanoCore associated with mesh hosts in 79.134.225[.]0/24 today.

In 2021, Proofpoint attributed the "TA2722 / Balikbayan Foxes" campaigns (Remcos and NanoCore behind Philippine-government lures) to C2 sitting in this very space, `79.134.225[.]9, .92, .107` on Fink Telecom and `185.140.53[.]0/24` / `185.244.30[.]70` on the sibling AS208476 (netname PRIVACYFIRST). Notably, Proofpoint anchored that cluster to a registrant email, anthony.marshall.1986@gmail[.]com, that it tied back to the 2017 Adwind activity, a single procurement thread stretching from 2017 to 2021.

In 2024, Resecurity reported the JSOutProx financial-fraud framework striking banks across APAC and MENA with C2 including `185.244.30[.]218`, an address independently associated with "The Privacy First Project" / Freemesh, no-logs space.

In 2026, a CTX Threat News writeup recorded Remcos samples beaconing to mesh node `79.134.225[.]21`; we treat its technical data as usable and its "Gorgon Group" attribution as low-confidence and uncorroborated (see the certificate note below).

The paper trail: 2017 to 2026
(continued)

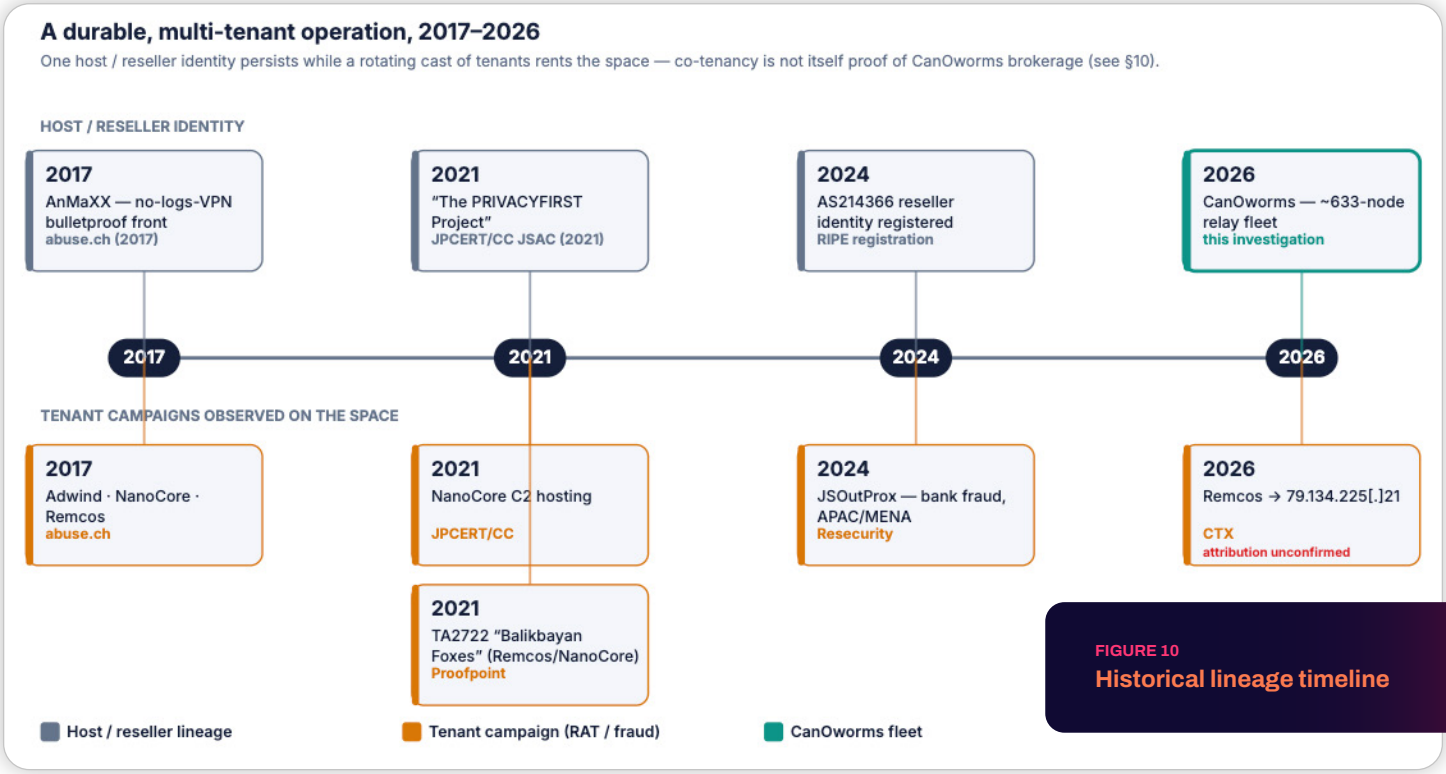


FIGURE 10
Historical lineage timeline

We attribute those prior findings to their originators (abuse.ch, JPCERT/CC, Proofpoint, and Resecurity respectively) and do not absorb their scope. We cite them because they corroborate, across nearly a decade and from different vantage points, that this is a durable, rebranding, bulletproof/anonymization business rather than a fresh campaign. That continuity — the same brand family, the same upstreams, and the same commodity-RAT clientele across 2017, 2021, 2024, and 2026 — is the strongest part of the operator picture, and we hold it at high confidence.

Who exactly sits at the keyboard operating the CanOworms relays today is a separate and unresolved question, as the next section shows.

One boundary has to be drawn around those tenant sightings, and it is a boundary of time, not of doubt about the hosting. Each of TA2722, JSOutProx, and the 2026 Remcos cluster is a confirmed tenant of the PrivacyFirst pool. What we cannot yet determine is whether they rented that pool directly or "farm to table," like when a threat actor rents the spaces directly from the source or one of its temporary resellers, without the "SaaS" layer that CanOworms provides. We also cannot determine whether CanOworms already existed as another middleman layer and brokered their access.

Our instruments have a horizon: the netflow is anchored to a 2026 window, the scan and certificate telemetry is a short rolling window, and the mesh's own `kickass` marker is a self-signed certificate that cannot be validated for its actual creation time. The same TA2722 and JSOutProx IPs are CanOworms `kickass`-fleet members today, which shows the pool's continuity and reuse, but because of a lack of data retention spanning multiple years, we could not firmly tie CanOworms' signature to those addresses at the time of the historical campaigns. We therefore record them as pool tenants with CanOworms brokerage undated (possible clients, or accidental neighbors drawn by the same lenient policy).

An open anomaly: the borrowed mail certificate

A single artifact sits oddly against this otherwise coherent picture and was flagged by our colleagues at Malanta: Mesh node `79.134.225[.]21` presents a seemingly legitimate German mail certificate (`CN=ciesnik[.]de`, a real Let's Encrypt wildcard cert) that it re-deploys on each monthly renewal, mirroring the owner's reissuances within about a day. We analyze it in three parts.

This is also not a one-off certificate loan. Just within the last six months of Driftnet data, we observed seven new Let's Encrypt `CN=ciesnik[.]de` certificates generated, replacing older ones among the SMTP services running them, and within a day, our mesh node (`79.134.225[.]21`) presented the new certificate as well on port 443.

First, this is a genuine anomaly across two axes: within CanOworms it is the only case we saw of a real third-party certificate on a node (every other member wears the operator's own `kickass` or `nVpn` cert), and of the approximately 58 internet hosts presenting this certificate, 57 are the owner's own Deutsche Telekom mail servers on SMTP/IMAPS (465/587/993) while `79.134.225[.]21` is the lone host serving it on an HTTPS/DTLS service (`:443`).

Second, we do not adopt the CTX writeup's conclusion that this certificate is the operative C2 component. The Remcos samples beacon to an entirely different port (`:60512`) while the certificate lives on `:443`, and the samples are years older than the observed certificate (2020-vintage payloads against a 2026 certificate generation). The parsimonious reading, at moderate confidence, is that the sample is an old relic resurfaced from an approximately 6-year-old campaign, its hard-coded C2 pointing at an address that has since changed hands, and perhaps more than once. This is precisely the churn a long-lived bulletproof pool produces.

That speaks to PrivacyFirst's history, not to an immediate indictment of `79.134.225[.]21` as a live Remcos controller. Making that indictment would require fresh samples with definitive sandbox detections resolving to this node while the certificate is present.

Third, none of this explains the behavior away. A relay node holding and renewing a stranger's mail certificate on a proxy service has no benign reading we can offer and is consistent with TLS impersonation and reputation-laundering. We therefore carry it as an open, actively-tracked anomaly at moderate confidence of malicious intent.

We emphasize that there is no evidence suggesting the legitimate third-party infrastructure owner, Deutsche Telekom, has any knowledge of or involvement in this activity; their infrastructure is more likely abused by the CanOworms relay node.

11.

Hunting the operator

If registration only reveals resellers, the one layer that cannot be resold anonymously is the control plane, because something has to manage the fleet. We therefore pursued this lead directly, even though our working hypothesis proved incorrect, because the manner of that failure constitutes a finding in its own right.

The control cluster in the Czech Republic (Section 5) was the obvious lead, and one host in it, the operator console `89.203.249[.]52`, also carried the web services of a small self-hosted project. A chain of shared-hosting and certificate-reuse signals initially looked like it might unmask the operator behind that project's identity. Pulled carefully, using only already-collected scan data and open-source search and without touching the live systems, the lead dissolved.

The project resolved to a small, early-stage community run by a named private individual, with a public presence entirely inconsistent with running a criminal relay network; and the control host carried the hosting provider's own infrastructure naming, indicating a shared or rented address rather than a single tenant's dedicated box. The "operator identity" we nearly grasped was almost certainly an innocent co-tenant or compromised third party sharing an IP with the actual controller.

We are reporting this dead end on purpose. It is methodological honesty, since we will not name the individual and the lead is excluded, but it is also the clearest illustration of the network's defining property. The controllers sit on the same kind of borrowed, shared, benign-looking infrastructure as the relays: small budget-hosting boxes belonging on paper to unrelated hobbyists and small sites, some plausibly compromised. Anyone who pivots on a control-plane IP lands on an innocent bystander, exactly as we briefly did.

That is not an accident of our investigation, it is the design of the infrastructure. We therefore report the control plane as real but laundered through third-party infrastructure, and the operator's real-world identity as unattributed, not for want of a signal, but because the point is to convert that signal into a false lead. On-demand collection against the control hosts was outstanding at the time of writing. If it returns a management panel or a certificate identity of their own, we will update.

12.

Attribution: Many flags on one pole

CanOworms entered our view through an APT41-linked report, and several vendors have associated parts of this space with advanced-actor activity. This question warrants a direct response, which we address at two levels of analysis.

At the level of infrastructure characterization, we hold high confidence that this is a shared, commercial, multi-tenant anonymization network with a bulletproof-hosting lineage, serving a broad commodity-malware clientele.

At the level of actor attribution, the most striking fact is not that one advanced group is present but that several unrelated ones are flagged at once. Our research partners at Malanta attribute different hosts to different, mutually-unrelated actors: APT43 on one Datasource host, APT37 and APT43 on a Fink host (both certificate-defined mesh members). On a separate case, our other research partners at FalconFeeds attributed APT41 subgroup (also known as Silver Dragon) on an adjacent cluster, alongside the commodity RAT families (Remcos, Quasar, NanoCore, NetWire, AsyncRAT, and Loki) that have been observed throughout a few open-source community malware sharing platforms.

Multiple unrelated nation-state and criminal actors converging on a single certificate-defined relay set does not require conclusion in favor of any one attribution. This convergence is precisely what a shared relay-for-hire network should look like. Many tenants, one set of relays. We represent each of those attributions at the confidence its originator assigned and neither adopt nor merge them.

The specific Silver Dragon APT link deserves its own scrutiny, because it is where we started. As the original lead, the FalconFeeds X post, suggested, the APT41-subgroup-attributed campaign leveraged more than a single tool against a single entity, yet, only the initial Xworm RAT tool that was used early in the campaign, linked the activity to a certificate-defined member of CanOworms, while others did not. It may suggest that different stages rely on different obfuscation methods, or indicate a handoff between smaller, internal teams — but we recognize that this weakens the attribution link overall.

In addition to that, part of the original bridge was a shared Cobalt Strike watermark, and cracked Cobalt Strike builds circulate so widely that a shared watermark is a shared-tooling artifact, not a fingerprint of one actor. This further weakens that initial attribution which in turn, lowers our confidence in the ties to APT41.

A narrower analysis is more defensible: an APT41 operator, like other actors, may rent these relays as one tenant among many (a low-to-moderate-confidence claim about a client). This is distinct from, and should not be conflated with, a claim that the network itself constitutes APT41 infrastructure. We therefore maintain intellectual caution, and will rely on our partners' assessment for further investigation, while lacking first party evidence to support the attribution to these threat actors.

13.

What this means for defenders

The uncomfortable lesson of CanOworms is that the controls most organizations lean on to make trust decisions about remote infrastructure, namely IP reputation, geolocation, and ASN or hosting-provider reputation, are precisely the controls CanOworms is built to defeat.

Reputation fails because the relays carry other people's malware without being flagged themselves — we saw hosts rated clean while relaying numerous malicious samples. Geolocation and ASN reputation fail because the fleet is spread thinly across many ordinary providers in many countries, and the control plane hides inside small, benign-looking third-party boxes. Blocking what initially seems to be a C2 address from a feed fails because that address is frequently the disposable front, not the origin. Take it down and the backend simply fronts through another relay.

What continues to hold true with this network is the observations and evidence we started from: its build artifacts. The most durable detection and hunting signals are the fleet's own fingerprints (the shared per-/24 certificate SHA-1 thumbprints, the JARM `21d10d00021d21d21c21d10d21d21de61583be67a70a87e745a4964791e063`, and the JA4X `2bab15409345_2bab15409345_795797892f9c`, all listed in full in Appendix A), which identify a host as a member regardless of its IP, country, or reputation.

What this means for defenders
(continued)

Those fingerprints, combined with the proxy/VPN service-stack profile and, where visible, the management-plane phone-home pattern (a fleet-wide approximately 35-second heartbeat to a single collector), may enable a defender to recognize CanOworms' infrastructure in their own telemetry even as individual IPs churn. We qualify this guidance with two operational findings:

1. The control plane is a pressure point, not a permanent chokepoint: Blocking or monitoring the current collectors raises the operator's cost and can disrupt tasking, but, as Section 11 shows, those hosts are cheap, rented, and movable in minutes, so a defender should expect the heartbeat to re-home and should keep tracking the fingerprints rather than treat any single control IP as structural. And because the set visibly adds blocks over time, fingerprint-based tracking is a useful leading indicator.
2. One caveat on the word "durable": the fingerprints are durable relative to IP churn, not permanent. JARM and JA4X follow the software build, so a re-image or upgrade will shift them and a tracker has to be re-derived periodically. The shared per-/24 certificate thumbprint, a reused private key, is the sturdier anchor while it lasts. The consolidated fingerprints and member indicators are in Appendix A. Treat the IP list as a snapshot that will age.

FIGURE 11

Certificate build-out curve

Cumulative member hosts by cert first-seen: the build-out-as-leading-indicator argument (with the observation-floor caveat printed on the chart).

None of this requires exploiting the network, and we have deliberately included nothing that would help reproduce or operate it. The goal is recognition and denial of trust.

14.

Methodology and evolving red flags

Because the conclusions above are only as good as the method behind them, and because the method is reusable against the next network of this kind, we document that method explicitly below, including two instances in which further investigation sharpened our conclusions.

Clustering from a fingerprint, not an IP list

The investigation is built on the principle that infrastructure is defined by how it was built, not by where it currently sits. We started from one shared self-signed certificate and expanded outward, but only under a strict membership test derived from an early false positive: shared certificate thumbprint, plus consistent JARM/JA4X, plus sustained presence and, when visible, a node-initiated heartbeat. The issuer string alone was insufficient for inclusion. The issuer-string-only look-alikes, hosts carrying the kickass string but not the matching thumbprint, JARM, and JA4X are why the string alone is disqualifying. Had we counted them, we would have inflated the fleet. It should be said that node-initiated heartbeat is a strong indication of membership, but not a prerequisite for it.

Triangulating three independent signals

Certificate thumbprint, JARM (TLS-negotiation stack), and JA4X (certificate structure) are derived differently, so their agreement across the 633 members is three independent measurements converging, the basis for our "common provisioning origin" call.

A single service's JARM, taken alone, is reproducible: deploy the same Squid or OpenVPN container behind the same TLS settings and the fingerprint marks the similarity, because JARM reflects on the software build. What is not reproducible by accident is what we observe here. The identical fleet JARM answers on two different services on the same hosts, the Squid HTTP-proxy port (3127) and the SOCKS port (9998), on 632 of the 633 members, and neither service speaks TLS in their default form.

For both to present TLS, and to present the byte-identical negotiation fingerprint, one TLS-termination layer has to front both services. That is a deliberate, purpose-built relay image, not a coincidence of two packages sharing a base container, and it makes the cross-service JARM corroborating evidence of a unified build. The single hardest-to-forge element is still the shared per-/24 certificate thumbprint, a reused private key a generic template would never mint identically across a block. Thumbprint plus the cross-service stack fingerprints are the membership test.

Bounding malware activity by membership window

Feed labels and malware samples are only meaningful if they were active while a host was a member. We recorded each host's certificate first- and last-seen window and kept only malware whose activity overlapped it. This is how we separated "malware on this relay during its mesh tenure" from "prior owner's baggage."

Methodology and evolving red flags
(continued)

The confidence rubric

Our confidence rubric is deliberately about diversity of evidence: a single feed is low; two sources of the same type (two feeds echoing) is moderate; two or more different classes of evidence agreeing (a feed label and our own netflow observation, or a feed and a VirusTotal file relationship) is high. That rubric is applied in Appendix C and throughout.

FIGURE 12

Evidence-corroboration heatmap

Hosts by evidence types, cells shaded where each source fired: the confidence rubric made visible; pairs with the Appendix C table.

Reading cadence, not payload

Netflow gave us direction and timing but never content, so every "beacon," "heartbeat," and "C2" in Section 5 is an inference from rhythm and port. The signal we relied on, a fleet-wide fixed heartbeat to one collector (infrastructure) versus scattered low-jitter callbacks to RAT ports (implants), is behavioral, not malware.

Registration as procurement, not operation

When we read WHOIS and reverse-registration data, we treated the customer-of-record as the leasing identity, never assumed it was the operating one, and used cert-coverage-versus-footprint measurements to keep the reseller and the operator distinct.

15.

What CanOworms tells us

As state-sponsored and financially motivated threat actors increasingly converge on shared, commercially operated infrastructure to obscure their origins, CanOworms is best understood not as a botnet or a command-and-control network but as an anonymization service for hire, laundered through other people's infrastructure at every layer.

The network sits alongside a broader pattern SecurityScorecard research has previously documented, including the LapDogs ORB and Operation WrtHug, in which China-nexus actors increasingly rent rather than build the infrastructure that hides them. This convergence of state and criminal tenants on identical relay space is precisely what makes the infrastructure durable: it is priced, marketed, and re-let, not staged for a single campaign, and it will outlast any one operator's use of it.

The relays run on address space leased across a spread of hosting providers under a reseller brand with a bulletproof-hosting lineage traceable through independent prior reporting to at least 2017. The suspected IPs tagged as C2 servers by other vendors are, in our assessment, predominantly the disposable front of that relay layer rather than the origin of the traffic. The clientele is a broad roster of commodity RAT operators and, at low confidence and as tenants among many rather than as the network's owner, plausibly several advanced actors at once, which is exactly why no single-actor attribution fits.

What CanOworms tells us
(continued)

The control plane, hosted separately from the network and carrying none of its fingerprints, remains hidden inside small, benign-looking third-party boxes. This, in our assessment, is an indication the space resellers (most prominently PrivacyFirst) and the CanOworms operators are not one and the same, as the owner of the space will most probably prefer to manage it from inside their purview, and not from the back yard of a separate entity provider. And it is not inert plumbing: attack-shaped traffic, including netflow that resembles SSH credential-spray against edge-device-heavy ranges, exits through its nodes, although whether that is the operator or, more likely, its tenant, remains to be determined.

For defenders, the takeaway is to stop trusting the signals this network is designed to launder, namely IP, geography, and provider reputation, and to hunt instead on what it cannot easily change, including the fingerprints of how it was built, and the pressure point of its managed control plane.

For the research community, CanOworms is a reminder that in a market where anonymization is a product, the hardest and most valuable analytic move is also the most disciplined one: keeping the operator, the reseller, and the client in three separate boxes, and refusing to collapse them.

This research establishes, at the confidence levels stated throughout, that a single relay network can simultaneously serve commodity criminals and advanced state-nexus actors without devolving into either category alone, and that operator, reseller, and client are distinct, verifiable layers of evidence rather than an arbitrary analytic convenience. That separation is the finding: attribution frameworks built around a single owner per network will misread infrastructure of this kind, and defenses built around IP and provider reputation will continue to fail against it until detection shifts to the fingerprints of how such infrastructure is built and operated.

APPENDIX A

Indicators

View the live fleet (Driftnet). Note this is a time-bound snapshot, not a static blocklist: members query, the kickass issuer plus JA4X combo.

Network fingerprints (durable):

- Certificate issuer/subject: `CN=*, O=kickass, L=Bar, ST=Foo, C=WS`
- JARM (fleet-dominant, full): `21d10d00021d21d21c21d10d21d21de61583be67a70a87e745a4964791e063`
- JA4X (fleet-dominant, full): `2bab15409345_2bab15409345_795797892f9c`
- Per-/24 certificate SHA-1 thumbprints (one shared key per block, the true membership signal):
  - `194.147.140[.]0/24: 40a77b63591410de545098eb1eda140338ee794e`
  - `185.244.30[.]0/24: 7a5b57d7fc3889064b6ac82682dcb6ed1dce5149`
  - `194.180.225[.]0/24: 273dd82e723ccd80581cc44c6f55e3aca6b73991`
  - `185.217.1[.]0/24: 2f586ebd3213dd08d4dae140ab7875f4d6030ba1`
  - `91.192.100[.]0/24: ccd8f4e4140e5e6d7eda11bfb945ccd13eb94b0a`
  - `79.134.225[.]0/24: e377bfee3c8f330e9641db3e20027083e1ce365e`
  - `157.173.24[.]0/24: d974ede731e1242aee8f97587a1abfdf4810ce5a`
  - `95.167.151[.]0/24: 5567395956179314e021e46f67fbc56c1d2c567e`
  - `184.75.209[.]0/24: 4e344edbbfb9f86d3ec097698b240fa5e98f0faf`
- Second, fleet-specific membership certificate (the "nVpn" family, not the commercial nVPN service's certificate): CN=nVpn CA, OU=nVpn, O=nVpn. SHA-1 db6a990981b1a4020a7994a8258efa57bced2fb1 accounts for about 96% (413 of 428) of all public-IPv4 hosts presenting any nVpn CA certificate in a recent window — one reused key deployed fleet-wide, a second membership signal alongside kickass. JA4X c9d784bbb12e_c9d784bbb12e_* (shared first two segments; observed third segments include 4f47b0ae8acc, bd03cf12d713, e0b3803da5e6).
- Excluded look-alike (honeypot/decoy) JA4X, for hunting hygiene: `d5d9105797f5_d5d9105797f5_000000000000`
- Membership rule: shared cert thumbprint plus consistent JARM/JA4X plus sustained presence. Not the issuer string, not a one-off sighting.

Densest relay blocks observed (snapshot): `194.147.140[.]0/24, 185.244.30[.]0/24, 194.180.225[.]0/24, 87.237.165[.]0/24, 185.217.1[.]0/24, 91.192.100[.]0/24, 79.134.225[.]0/24, 157.173.24[.]0/24, 95.167.151[.]0/24, 184.75.209[.]0/24, 178.124.140[.]0/24.`

Control-plane hosts (moderate confidence, role-based; all CZ): `89.203.249[.]52` (52335 phone-home sink plus RDP/tasking console), `89.203.249[.]139` (RDP/SSH admin), `89.203.249[.]174` (RDP admin), `89.203.141[.]238` (7707, ~35 s heartbeat), `82.202.100[.]57` (443 health check).

APPENDIX B

MITRE ATT&CK mapping

ATT&CK is victim-centric and CanOworms is infrastructure, so this maps from two vantage points: the “business process” itself (building and running the service) and a tenant (using it in an intrusion). Each entry ties to findings in the research.

Resource Development (TA0042)

Technique	How it appears in CanOworms
T1583.003 · Acquire Infrastructure: Virtual Private Server	~633 leased proxy/VPN VPS across six-plus hosting providers under a reseller identity; gray-market IPv4 (the `IR-`… block moved via IP brokers) leased as a disposable, blocklist-resistant hedge
T1587.003 · Develop Capabilities: Digital Certificates	A self-signed `kickass` certificate (and co-resident `nVpn CA` family) minted once and deployed fleet-wide to TLS-wrap the relay services
T1584.008 · Compromise Infrastructure: Network Devices	Suspected second tier of relays/redirectors on edge devices (MikroTik/CPE/Hikvision) and control-plane services on third-party budget-host boxes
T1588.002 · Obtain Capabilities: Tool	Tenant commodity RATs (Remcos, Quasar, NanoCore, NetWire, AsyncRAT, Loki); a shared cracked Cobalt Strike watermark in adjacent infrastructure
T1583.001 · Acquire Infrastructure: Domains	Dynamic-DNS (DuckDNS-style) C2 domains fronting some tenant activity

Command and Control (TA0011)

Technique	How it appears in CanOworms
T1090.002 · Proxy: External Proxy	The defining function: Squid/SOCKS/OpenVPN relays forwarding tenant traffic to hidden backends; feed-labelled "C2" hosts are the external proxy in front of the origin
T1571 · Non-Standard Port	RAT C2 reached on high/odd ports on mesh nodes (e.g. `:8084`, `:56098`, `:1998`, `:34574`)
T1573.002 · Encrypted Channel: Asymmetric Cryptography	Services wrapped in self-signed TLS (the membership certificate) to present as "encrypted" to scanners

Credential Access / Lateral Movement / Persistence of Access

Technique	How it appears in CanOworms
T1110.003 · Brute Force: Password Spraying	Netflow behavior resembling distributed SSH credential-spray from many nodes, ~one host per target /24, in tight windows (e.g. `194.36.173[.]0/24`, June 2026).
T1021.004 · Remote Services: SSH	Held SSH footholds (`157.173.24[.]247` to `207.154.230[.]221:22`, 8 days) and fleet SSH administration from the control plane
T1021.001 · Remote Services: RDP	Operator RDP into nodes from the Czech control cluster (`89.203.249[.]52/.139`)
T1133 · External Remote Services	Exposed remote-access services used to reach and manage relay nodes

APPENDIX C

Indicators of Compromise

Confidence reflects how many independent evidence types agree: High = two-plus different types (e.g. a feed label plus our own netflow/scan plus a VirusTotal file relationship); Moderate = a feed plus one corroborating type; Low (external) = reported by a collaborator (Malanta / FalconFeeds) and not independently observed by SecurityScorecard. A "C2" type names the relay endpoint the malware used, not necessarily the traffic's origin.

Indicator (IP : port)	Type	Confidence
91.192.100[.]36:8084	Quasar RAT C2 Front	High
194.147.140[.]242:1998	Remcos C2 Front	High
185.217.1[.]170:56098	Quasar RAT C2 Front	High
79.134.225[.]18:3371	NanoCore C2 Front	Moderate
194.147.140[.]197:34574	Remcos C2 Front	Moderate
79.134.225[.]25:2001	High-fan-in relay port (Consistent with NetWire port use)	High
91.192.100[.]62	Suspected APT43 infrastructure (per Malanta)	Assessed by other vendor
79.134.225[.]22	Suspected APT37 plus APT43 infrastructure (per Malanta)	Assessed by other vendor
194.180.225[.]137:10002	High-fan-in port, nature unconfirmed	Low

APPENDIX D

Sources

Open-source, published references:

- **Check Point Research,** "Silver Dragon targets organizations in Southeast Asia and Europe" (2026): https://research.checkpoint.com/2026/silver-dragon-targets-organizations-in-southeast-asia-and-europe/
- **JPCERT/CC,** JSAC 2021, "The PRIVACYFIRST Project" / NanoCore C2 hosting: https://jsac.jpcert.or.jp/archive/2021/pdf/JSAC2021_107_matsumoto_en.pdf
- **abuse.ch** (2017), "AnMaXX, Gerber EDV and the Qrypter Connection" (the bulletproof / "no-logs VPN" operation): https://abuse.ch/blog/anmaxx-gerber-edv-and-the-qrypter-connection/
- **FalconFeeds,** the public post that opened this investigation: https://x.com/FalconFeedsio/status/1955166371979071677
- **Proofpoint (2021),** "New Threat Actor Spoofs Philippine Government, COVID-19 Health Data in Widespread RAT Campaigns" — TA2722 / "Balikbayan Foxes"; Remcos & NanoCore C2 on the Fink Telecom (AS6775) and PRIVACYFIRST (AS208476) space, tied by the anthony.marshall.1986@gmail[.]com registrant back to the 2017 Adwind activity: https://www.proofpoint.com/us/blog/threat-insight/new-threat-actor-spoofs-philippine-government-covid-19-health-data-widespread
- **Resecurity** (2024), "The New Version of JsOutProx is Attacking Financial Institutions in APAC and MENA via GitLab Abuse" — JSOutProx C2 including 185.244.30[.]218, associated with the "Privacy First Project" / Freemesh no-logs space: https://www.resecurity.com/blog/article/the-new-version-of-jsoutprox-is-attacking-financial-institutions-in-apac-and-mena-via-gitlab-abuse
- **CTX Threat News** (SANDS Lab, 2026), "Gorgon Group's Remcos Campaign Hides Behind Three Evasion Layers and a Swiss No-Log VPN" — Remcos beaconing to 79.134.225[.]21. https://news.ctx.io/en/article/CTX6i3wxxjbid

Acknowledgements

We built this research in collaboration with FalconFeeds and Malanta, combining their intelligence with our own analysis. Some findings reflect their direct observations, reported to us and included as working hypotheses. We hold ourselves to a clear standard: we only present data as confirmed when we've verified it independently. That distinction protects the integrity of this research and respects the work our partners contributed.

We drew internet-scan, threat-feed, and registration data from SecurityScorecard's own Driftnet, as well as from VirusTotal, abuse.ch, and RIPE.

STRIKE
By SecurityScorecard

Discover how SecurityScorecard and STRIKE can identify and reduce exposure across your ecosystem.

Explore Internet Intelligence at SecurityScorecard.

For STRIKE media inquiries, contact us here.

SecurityScorecard

SecurityScorecard.com
info@securityscorecard.com