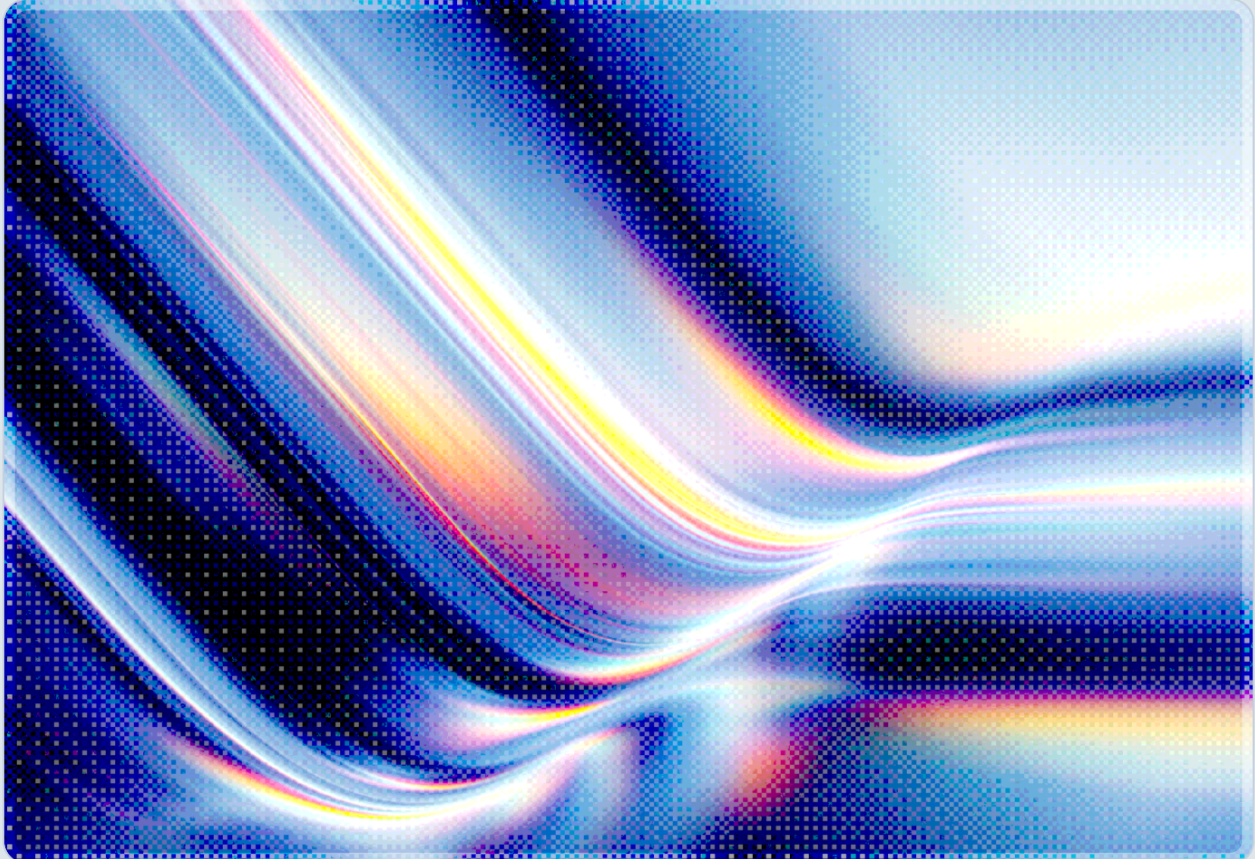


# KeyDrift: SD-WAN Controllers as Key-Harvesting Platforms

Security Implications and Guidance for Catalyst SD-WAN Operators

2026-10-30

 AI-assisted Rapid Research



CSAI

cloud  
security  
alliance®

**© 2026 Cloud Security Alliance. Some rights reserved.**

You may download, store, display, view, print, redistribute, and link to this document in its original, unmodified form, provided that attribution to the Cloud Security Alliance is maintained and all trademark and copyright notices remain intact.

This document may not be modified or altered. You may quote portions of the document as permitted by the Fair Use provisions of the United States Copyright Act, provided that attribution is given to the Cloud Security Alliance.

This document may be shared on professional and social media platforms in its original form with attribution.

*This document was generated with AI assistance and has not undergone official CSA review and approval processes.*

---

## Key Takeaways

On October 28, 2026, Cisco Talos published an analysis of KeyDrift, a custom malware framework that an advanced persistent threat actor it tracks as UAT-11473 deployed on Cisco Catalyst SD-WAN control components after exploiting CVE-2026-20127, a CVSS 10.0 authentication bypass [1][2]. KeyDrift runs inside the legitimate `vdaemon` process. It reads control-plane messages after they are decrypted and extracts a 32-byte field that Talos assesses to be AES-256 IPsec data-plane key material distributed through the Overlay Management Protocol (OMP) [1]. Talos describes this field as assessed rather than confirmed key material. The campaign shows that an SD-WAN controller is more than a management target: it is the point where an entire fabric's encryption keys pass through. Anyone controlling it can, in principle, collect the keying material needed to decrypt inter-site traffic. Combined with on-path capture of that traffic, this would allow decryption without touching a single branch router. Public reporting confirms neither the exfiltration of keys nor the capture of traffic [1][4].

For operators, the first lesson is that patching in February does not show a controller is clean. Cisco fixed CVE-2026-20127 in February 2026, but Talos traces exploitation of the flaw to at least 2023, and Lumen reports Volt Typhoon exploitation by late 2025 [2][14][5]. Controllers compromised before the patch may still host implants that survive the upgrade. Key rotation should therefore be treated as a consequence of remediating a compromise, not as a fix for it. Keys distributed through a compromised controller remain exposed to the implant, so rotating fabric keying material is effective only after the control plane has been rebuilt from trusted images.

Defenders should hunt for the specific artifacts Talos documented. These include modified runit scripts under `/etc/sv/sshd/` and `/etc/sv/aaamgr/`, unexpected `libcommon.so` and `libdssl.so` libraries, the file `/var/tmp/vdaemon.bin`, and `Accepted publickey for vmanage-admin` entries from unfamiliar addresses in `/var/log/auth.log` [1][3]. Organizations in the space and satellite, defense and aerospace, and policy research sectors should treat this as a priority, because Talos identifies these sectors as UAT-11473 targets [1]. Talos also assesses with high confidence that UAT-11473's infrastructure overlaps considerably with "Silent Train," a covert network of compromised devices that Lumen attributes to Volt Typhoon [1][5]. Talos does not state that UAT-11473 and Volt Typhoon are the same actor. The link rests on shared infrastructure, and covert relay networks of this kind can serve more than one operator [3].

# Background

Cisco Catalyst SD-WAN separates the network into three control components and a fleet of edge routers. The Manager (formerly vManage) provides orchestration and configuration. The Validator (formerly vBond) handles initial authentication and orchestration of new devices joining the fabric. The Controller (formerly vSmart) runs OMP and acts as the route and policy reflector for the overlay. In Cisco's default data-plane key exchange, edge routers do not negotiate keys pairwise through IKE (although IKE is used in some configurations, such as tunnels to third-party devices). Instead, each edge generates an AES-256 key for each WAN transport and advertises it to the Controller in an OMP update, and the Controller re-advertises it to the other edges, which use it to build IPsec tunnels [8]. This centralized exchange avoids the overhead of pairwise IKE negotiations across large fabrics. The same design means the Controller sees, in the course of normal operation, the keying material that protects every site-to-site tunnel it serves.

CVE-2026-20127 is an improper-authentication flaw (CWE-287) in the peering authentication mechanism of the Controller, Manager, and Validator. Cisco rates it CVSS 10.0, and it affects these components regardless of configuration, including on-premises deployments and Cisco's hosted, managed, and FedRAMP environments [2]. An unauthenticated remote attacker can log in as an internal, high-privileged non-root account and reach NETCONF to manipulate fabric configuration [2]. Cisco first published the advisory on February 25, 2026. It states that no workaround addresses the vulnerability, though restricting ports 22 and 830 to known controllers and trusted addresses is offered as a temporary mitigation for on-premises systems [2]. The same day, CISA issued Emergency Directive ED 26-03. It required federal civilian agencies to inventory affected systems, collect forensic artifacts, patch by 5:00 PM ET on February 27, 2026, and hunt for compromise, and to rebuild the control plane from patched images if root compromise was found [6].

The February disclosure was not the beginning of the story. When Cisco disclosed the flaw, Talos tracked its exploitation under the cluster UAT-8616, whose activity it traced to at least 2023 [14]. Lumen's Black Lotus Labs separately reports that Volt Typhoon was exploiting CVE-2026-20127 by late 2025 and continued after disclosure [5]. Lumen describes Silent Train as a framework of rapidly rotating chains of compromised devices, used to support at least four confirmed zero-day campaigns against management and control-plane systems, including other SD-WAN platforms [5]. Talos's October report adds a new actor designation, UAT-11473, and connects it to Silent Train through shared infrastructure rather than through a claim of common identity [1][3]. It describes the activity as historical but does not say when it occurred or whether any harvested keys were successfully exfiltrated [1][4]. Public reporting does not explain how UAT-11473 relates to UAT-8616 [3].

# Security Analysis

## How KeyDrift Works

KeyDrift is a modular framework, and its components divide the work of persistence, tasking, and collection. The table below summarizes the components that Talos documented and that The Hacker News reported in further detail [1][3].

| Component                                    | Form                        | Function                                                                                                                                                                                                                                     |
|----------------------------------------------|-----------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| conf_sshd                                    | Loader script               | Launched by a trojanized runit <code>run</code> script for SSH; writes <code>vdaemon</code> process IDs to <code>/tmp/system-vdaemon</code> before starting the real SSH service                                                             |
| rsshd                                        | ptrace-based injector       | Injects the PolyFlame module into each running <code>vdaemon</code> process                                                                                                                                                                  |
| PolyFlame<br>( <code>libdssl.so</code> )     | Memory-resident main module | TinyShell-like backdoor (command execution, file read/write) or intermediate C2 relay; tasked in-band over SD-WAN control traffic with no hardcoded server addresses recovered                                                               |
| LittleFilch<br>( <code>libcommon.so</code> ) | Collection module           | Hooks OpenSSL <code>SSL_read</code> in <code>vdaemon</code> ; extracts edge transport IPs, peer system-IPs, control-session identifiers, and 32-byte assessed AES-256 key material; RSA-encrypts output to <code>/var/tmp/vdaemon.bin</code> |
| aaamgrd                                      | Packed loader on Validators | Masquerades as a service under <code>/etc/sv/aaamgr/</code> ; decrypts and runs PolyFlame; deletes writable-overlay PAM configs for sshd and login                                                                                           |

The design choices suggest detailed familiarity with the platform's internals. Hooking `SSL_read` inside `vdaemon` lets LittleFilch read OMP messages in plaintext after the DTLS/TLS control session has decrypted them, so the implant never needs to break any cryptography itself [1]. LittleFilch applies filters on buffer size, leading byte, and three marker patterns. Talos reports that it targets a preset list of edge routers rather than the whole fabric, which is consistent with selective intelligence collection [1][3].

Collected data can be sent over UDP in a DTLS 1.2-style record to an address read from `/var/tmp/vdaemon.conf`, a file that is deleted after reading and that Talos could not recover [1]. PolyFlame's tasking also travels inside SD-WAN control traffic. Its execution is time-gated against a value supplied by the operator, which in our assessment would likely complicate sandbox analysis [1]. The actor timestamped its files by copying metadata from legitimate reference files, and it remounted `/rootfs.rw` as executable to support persistence [1].

## Why Key Material Matters

Edge-device intrusions are commonly framed around access: a foothold, credentials, or a pivot into the internal network. KeyDrift retains those capabilities, since PolyFlame is a full backdoor and relay and LittleFilch also collects addressing and session data. It also suggests an additional objective beyond access. Stealing data-plane keys could let an actor positioned anywhere along the WAN path, such as a carrier, an upstream transit network, or a satellite downlink, decrypt inter-site traffic passively, without generating activity on enterprise endpoints. That is our inference from the published technical details. Talos itself frames decryption as a potential capability rather than a confirmed outcome [1][4]. Some secondary coverage goes further and asserts that the theft enables decryption of sensitive inter-site traffic [9]. Defenders should read such claims as plausible worst cases rather than established facts.

The key lifecycle shapes how much a stolen key is worth. A third-party technical reference describing Cisco's scheme says that each key has a 24-hour lifetime and that edges regenerate keys every 12 hours, with old and new keys overlapping during transition [8]. These figures describe default behavior; rekey intervals are configurable, so operators should check the values in their own deployments. Under any timer setting, a single stolen key opens only a bounded window of traffic. A persistent implant on the Controller, however, sees every new key as it is distributed, which turns the controller into a continuous key feed rather than a one-time leak. It also means that, against an adversary on the controller, the centralized exchange offers no forward secrecy. Unlike a per-tunnel Diffie-Hellman exchange, the material needed to decrypt traffic passes through a single point that the adversary already controls. This is our analytic assessment of the architecture, not a statement from Cisco or Talos. It suggests that any ciphertext an adversary may have recorded during the compromise window should be treated as potentially exposed even after remediation.

Cisco's optional IPsec pairwise keys feature, which generates a distinct session key for each pair of edge transports, is disabled by default [8]. Pairwise keying limits how much traffic a single stolen key exposes, which makes it a reasonable hardening measure to evaluate. Public reporting does not establish whether pairwise keying would defeat LittleFilch's collection logic. Because pairwise material is still exchanged through the control plane, operators should not assume it does.

# The Remediation Gap

A central operational point is the gap between patch status and compromise status. Patching CVE-2026-20127 closes the entry point but, by itself, does not remove KeyDrift. Restricting ports 22 and 830, which Cisco offers only as a temporary mitigation until upgrade, likewise cannot remove an implant already in place [2][3]. According to The Hacker News, Cisco's February remediation guidance for the flaw separately warns of "existing exploits which are still in place" from before an upgrade [3]. Public reporting does not say whether upgrading overwrites the trojanized runit scripts or the writable overlay where KeyDrift's files live [3]. Organizations that patched promptly in February but did not hunt may therefore still have compromised controllers.

ED 26-03 already recognized this risk. For systems showing root compromise, it requires a clean rebuild of Manager, Controller, and Validator instances from patched images and migration of edges, rather than in-place repair [6]. Talos's report did not include file hashes, IP addresses, or domains [3]. Detection therefore depends on host-level artifact review and network signatures rather than simple IOC matching.

# Recommendations

## Immediate Actions

Every organization running Catalyst SD-WAN should first confirm that all Controllers, Managers, and Validators run a fixed release for CVE-2026-20127. The first fixed releases by train are 20.9.8.2, 20.12.5.3 or 20.12.6.1, 20.15.4.2, and 20.18.2.1; organizations on end-of-maintenance trains must migrate to a supported release [2][3]. Patch verification should be followed immediately by a compromise assessment on each control component. That assessment should cover the persistence and collection artifacts Talos documented, summarized in the table below [1][3].

| Artifact                               | Where to look                | Notes                                                                                         |
|----------------------------------------|------------------------------|-----------------------------------------------------------------------------------------------|
| Altered runit <code>run</code> scripts | <code>/etc/sv/sshd/</code>   | Trojanized script launches <code>conf_sshd</code> before the real SSH service                 |
| Unexpected service directory           | <code>/etc/sv/aaamgr/</code> | Cisco's legitimate aaamgr is a Python package and does not normally run as a compiled service |

| Artifact                  | Where to look                                                                                                   | Notes                                                                                                                                           |
|---------------------------|-----------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------|
| Implant files             | <code>conf_sshd</code> , <code>rsshd</code> ,<br><code>libdssl.so</code> , <code>libcommon.so</code>            | Loader, injector, PolyFlame, and LittleFilch respectively                                                                                       |
| Staging and output files  | <code>/tmp/system-vdaemon</code> ,<br><code>/var/tmp/vdaemon.bin</code> ,<br><code>/var/tmp/vdaemon.conf</code> | <code>vdaemon.conf</code> is deleted after reading and may be absent                                                                            |
| Injected libraries        | Libraries mapped into running <code>vdaemon</code> processes                                                    | Any library not expected for <code>vdaemon</code> warrants investigation                                                                        |
| Missing PAM configuration | Writable overlay on Validators                                                                                  | Lower-confidence indicator; Talos's description is ambiguous about whether some PAM changes are legitimate, so corroborate with other artifacts |

Beyond the file system, operators should review `/var/log/auth.log` for `Accepted publickey for vmanage-admin` entries from unknown addresses and run `request admin-tech` on each component before engaging Cisco TAC [3]. Network detection should complement host review. Talos's Snort coverage includes SIDs 65938 and 65958 for the vulnerability, and Snort 2 SIDs 67387–67416 and Snort 3 rules 7:44–7:73 (expressed in GID:SID notation) for KeyDrift. The Snort 2 rules are disabled by default for Subscriber rule set users and must be enabled explicitly [1][3]. Until assessment is complete, access to ports 22 and 830 should be restricted to known controllers and trusted management addresses [2].

## Short-Term Mitigations

If any artifact is found, organizations should follow the ED 26-03 model even outside the federal sector. That means preserving forensic evidence, deploying fresh control components from patched images, migrating edges to the rebuilt control plane, and only then rotating keying material [6]. Rotation should cover data-plane IPsec keys, control-plane certificates and DTLS/TLS credentials, the `vmanage-admin` and other administrative SSH keys, and any secrets stored on the controllers. Rotating fabric keys before the rebuild would expose the new keys to any active implant. CSA's [Key Management Lifecycle Best Practices](#) provides a framework for the revocation and replacement steps involved [15].

Once the control plane is trusted again, attention should turn to potential exposure. Organizations should determine which inter-site traffic flowed through the fabric during the possible compromise window, assess its confidentiality lifetime, and notify data owners where long-lived sensitive information may have been exposed. Hardening should follow Cisco's Catalyst SD-WAN Hardening Guide, which covers control-component access restrictions such as limiting NETCONF to the Manager, as well as forwarding logs to remote syslog [7]. Operators should also evaluate enabling pairwise keys and forward control-component logs off-box to a central collector, as ED 26-03 requires of federal agencies, so that on-device log tampering cannot erase evidence [6].

## Strategic Considerations

KeyDrift fits a broader pattern. Lumen's reporting describes repeated targeting of centralized management and control-plane systems, which offer fabric-wide reach [5]. Security architects should classify SD-WAN controllers, orchestrators, and similar key-distribution points as Tier 0 assets, on a par with identity providers and certificate authorities, with matching monitoring, access, and integrity controls. Where traffic is sensitive enough that passive interception by a state-level adversary is unacceptable, organizations should consider application-layer or end-to-end encryption that does not depend on the WAN fabric's keys. This reduces the impact of any single control-plane compromise.

Procurement and vendor assurance processes offer a further lever. Organizations should ask SD-WAN providers for runtime integrity verification of control-plane processes, read-only or attested system partitions, and documented guidance on post-compromise key rotation; initial public reporting did not address post-compromise key rotation [3]. Finally, the harvest-now-decrypt-later concern usually associated with quantum computing applies here in a more immediate form. An adversary that holds stolen symmetric keys and has also captured the corresponding ciphertext on-path could decrypt that traffic today, without any cryptanalytic advance. That possibility strengthens the case for treating cryptographic key lifecycle and key-distribution trust as a governance-level issue.

## CSA Resource Alignment

This note extends CSA's recent run of analyses on Cisco SD-WAN control-plane exploitation. CSA's research note on [Cisco SD-WAN CVE-2026-20245](#) (June 2026) documented a root-escalation zero-day in which attackers likely gained their initial foothold through earlier authentication bypasses, including CVE-2026-20127 [10]. KeyDrift illustrates one way that such control-plane access can be used once established, although public reporting does not connect UAT-11473 to the activity described in that note. It supports that note's emphasis on treating the SD-WAN management plane as a single point of fabric-

wide compromise. CSA's October 2026 note on [Cisco SD-WAN Manager CVE-2026-76504](#) advised organizations to presume compromise of any exposed unpatched Manager and to rotate credentials [11]. The present findings sharpen that advice. On controllers, rotation is effective only after a clean rebuild, and it must extend to data-plane key material, not just administrative credentials.

The key-harvesting dimension connects to CSA's [AI Cryptanalysis and the Cryptographic Assumption Horizon](#) (October 2026). That note identifies data with long confidentiality lifetimes as most exposed to harvest-now-decrypt-later collection and recommends crypto-agility practices such as rehearsing emergency algorithm changes [12]. KeyDrift shows that such collection does not have to wait for a cryptanalytic breakthrough when an adversary can take keys directly from the distribution point. Extending those emergency-change rehearsals to include fabric-wide key replacement would directly shorten recovery from a controller compromise. The procedures in CSA's [Key Management Lifecycle Best Practices](#) for key revocation, replacement, and destruction apply directly to that recovery [15].

For control mapping, CSA's [AI Controls Matrix \(AICM\) v1.1](#) provides the relevant domains [13]. Its Cryptography, Encryption and Key Management (CEK) controls cover key lifecycle, rotation, and revocation after suspected compromise. Its Infrastructure and Virtualization Security (IVS) and Threat and Vulnerability Management (TVM) domains address hardening, patching, and compromise assessment of network control components. Its Logging and Monitoring (LOG) domain supports the off-box log retention that ED 26-03 and Cisco's hardening guidance call for. Organizations running AI workloads across multi-site SD-WAN fabrics face particular exposure. Training data, model weights, and inference traffic moving between sites are often large, high-value, and long-lived in sensitivity, so they fall squarely within the exposure described here.

# References

- [1] Cisco Talos. "[Inside KeyDrift: Analyzing historical UAT-11473 activity on Cisco SD-WAN.](#)" Cisco Talos Intelligence Blog, October 28, 2026.
- [2] Cisco. "[Cisco Catalyst SD-WAN Controller Authentication Bypass Vulnerability \(cisco-sa-sdwan-rpa-EHchtZk\).](#)" Cisco Security Advisory, first published February 25, 2026; updated October 28, 2026.
- [3] The Hacker News. "[Hackers Exploited Cisco SD-WAN Flaw to Plant Malware Built to Steal Encryption Keys.](#)" The Hacker News, October 29, 2026.
- [4] Eduard Kovacs. "[Cisco SD-WAN Zero-Day Exploitation Tied to Volt Typhoon's 'Silent Train' Network.](#)" SecurityWeek, October 29, 2026.
- [5] Lumen Black Lotus Labs. "[Silent Train: Uncovering Volt Typhoon's Operational Framework for Zero-Day Exploitation and Strategic Access.](#)" Lumen, October 28, 2026.
- [6] CISA. "[ED 26-03: Mitigate Vulnerabilities in Cisco SD-WAN Systems.](#)" Cybersecurity and Infrastructure Security Agency, February 25, 2026; updated March 11, 2026.
- [7] Cisco. "[Cisco Catalyst SD-WAN Hardening Guide.](#)" Cisco Security, 2026.
- [8] NetworkAcademy.io. "[Data Plane Encryption \(Cisco SD-WAN\).](#)" NetworkAcademy.io, accessed October 30, 2026.
- [9] Aviatrix Threat Research Center. "[Cisco SD-WAN CVE-2026-20127: KeyDrift Malware Steals Encryption Keys.](#)" Aviatrix, October 2026.
- [10] Cloud Security Alliance. "[Cisco SD-WAN CVE-2026-20245 Zero-Day: Root Access Pre-Disclosure Exploitation.](#)" CSA AI Safety Initiative, June 27, 2026.
- [11] Cloud Security Alliance. "[Cisco SD-WAN Manager CVE-2026-76504: Authentication Bypass Exploited.](#)" CSA AI Safety Initiative, October 2, 2026.
- [12] Cloud Security Alliance. "[AI Cryptanalysis and the Cryptographic Assumption Horizon.](#)" CSA AI Safety Initiative, October 20, 2026.
- [13] Cloud Security Alliance. "[AI Controls Matrix \(AICM\) v1.1.](#)" Cloud Security Alliance, 2025.

- [14] Cisco Talos. "[Active exploitation of Cisco Catalyst SD-WAN by UAT-8616](#)." Cisco Talos Intelligence Blog, February 25, 2026.
- [15] Cloud Security Alliance. "[Key Management Lifecycle Best Practices](#)." Cloud Security Alliance.