INTELLIGENCE BRIEFING
Security Command Center
TLP:CLEAR
2026-04-27 05:57 UTC

CVE-2026-40050: Critical Path Traversal in CrowdStrike LogScale Self-Hosted Enables Unauthenticated File Access

CVE VULNERABILITY | **CRITICAL** | CVSS 9.1

SCC Item ID	SCC-CVE-2026-0079
Type	CVE Vulnerability
CVE ID	CVE-2026-40050
Severity	CRITICAL
CVSS Base Score	9.1
EPSS Score	0.0027 (50th percentile)
Affected Products	CrowdStrike LogScale (self-hosted deployments); specific affected versions not confirmed in available sources
Published	17 hours ago
Discovery Source	Serper

Executive Summary

A critical path traversal vulnerability has been reported in CrowdStrike LogScale self-hosted deployments that could allow unauthenticated remote attackers to read arbitrary files from the underlying host system. Organizations running LogScale on-premises may be exposed; no credentials are required to exploit this flaw if confirmed. Because LogScale commonly ingests security telemetry and log data, unauthorized file access could expose sensitive operational data, credentials stored on the host, or other log pipeline configurations. **Verification pending:** Patch availability, affected version ranges, and full technical details require confirmation against CrowdStrike's official advisory and NVD.

Technical Analysis

IMPORTANT, Verification Status: Specific affected version ranges, CVSS vector string, and patched version details have NOT been independently verified against NVD or CrowdStrike's official advisory in this session. All sources currently available are T3 tier (secondary/tertiary). Operators must validate all version and patch specifics against CrowdStrike's official advisory and the NVD record at https://nvd.nist.gov/vuln/detail/CVE-2026-40050 before taking remediation action.

CVE-2026-40050 is reported as a path traversal vulnerability (CWE-22: Improper Limitation of a Pathname to a Restricted Directory) in CrowdStrike LogScale self-hosted editions. The flaw is reported to permit unauthenticated remote attackers to traverse directory boundaries and access arbitrary files on the host operating system via a crafted HTTP request. MITRE ATT&CK mapping: T1190 (Exploit Public-Facing Application) for initial access; T1083 (File and Directory Discovery) for post-exploitation file enumeration. CVSS base score is reported at 9.1 (Critical) in secondary sources; EPSS score is 0.00265 (~50th percentile), indicating limited active exploitation reported at time of writing. CISA KEV: not listed. CWE-22 classification is applied by vulnerability type inference and requires confirmation against the NVD record.

Action Checklist

1. Step 1: Containment. Immediately restrict network access to your LogScale self-hosted instance. If LogScale is internet-facing, place it behind a Web Application Firewall (WAF) or VPN and block unauthenticated external access to the LogScale HTTP/API port until patch availability is confirmed and applied. Confirm affected version scope against CrowdStrike's official advisory before assuming you are or are not exposed.
2. Step 2: Detection. Review web server and application access logs on the LogScale host for requests containing path traversal sequences (e.g., '../', '%2e%2e%2f', '%252e%252e%252f', URL-encoded variants). Look for GET or POST requests targeting file paths outside the expected LogScale application directory. Check for unexpected file access patterns in host-level audit logs (e.g., auditd on Linux) referencing sensitive paths such as /etc/passwd, /etc/shadow, SSH key directories, or application config files.
3. Step 3: Eradication (Pending Patch Confirmation). Confirm whether CrowdStrike has released a patch for CVE-2026-40050 by checking CrowdStrike's official security advisory and the NVD record. Once patch availability is confirmed, obtain the vendor-provided patch, review release notes for affected and fixed versions, test in a staging environment, and deploy to production only after validation.
4. Step 4: Recovery. After patching is confirmed complete, validate that path traversal test requests (using safe, non-destructive probe payloads in a staging environment) no longer return out-of-bounds file content. Re-enable external access only after patch verification is complete. Monitor LogScale access logs and host file integrity monitoring (FIM) alerts for at least 72 hours post-remediation for signs of prior or continued exploitation.
5. Step 5: Post-Incident. Audit whether sensitive files were accessible on the LogScale host (credentials, API keys, TLS certificates, configuration files containing downstream system access). Review network segmentation controls for SIEM and log pipeline infrastructure, as these systems frequently hold credentials and access to broad internal telemetry. Evaluate whether LogScale hosts follow least-privilege file system permissions to limit the blast radius of future path traversal class vulnerabilities.

IR / Forensic Enrichment

Triage Priority	IMMEDIATE

Escalation Criteria	Escalate to CISO and legal/privacy counsel immediately if auditd or HTTP access log analysis confirms successful traversal reads (HTTP 200-OK with non-zero response body) of credential files, TLS private keys, API tokens, or any file containing PII/PHI — as these conditions likely trigger breach notification obligations under applicable regulatory frameworks (e.g., GDPR Article 33, HIPAA §164.412, state breach notification laws) and indicate that downstream systems accessible via stolen credentials may also be compromised.
Recovery Notes	Before re-enabling external access to LogScale, confirm three conditions: the installed version matches the CrowdStrike-confirmed fixed release (verified by version endpoint and binary hash), a staged traversal probe returns HTTP 4xx rather than file content, and all credentials readable by the LogScale process user have been rotated or confirmed unexposed. Maintain elevated monitoring of LogScale HTTP access logs and auditd for a minimum of 72 hours post-patch, with specific alerting on any URI containing `..%2F`, `%252e`, or `%c0%af` sequences and on any `openat` syscalls by the LogScale JVM to paths outside the application install directory. If the pre-patch exploitation window spanned more than 24 hours with internet-facing exposure, treat downstream systems whose credentials were stored on the LogScale host as potentially compromised and initiate credential rotation and session invalidation for those systems in parallel with LogScale recovery.
Forensic Artifacts	LogScale HTTP/API access logs: Entries with URI paths containing raw or URL-encoded path traversal sequences ('../', '%2e%2e%2f', '%252e%252e%252f', '%c0%af') paired with HTTP 200 status codes and non-zero response Content-Length values — these confirm successful unauthenticated file reads via CVE-2026-40050. | Linux auditd syscall logs (/var/log/audit/audit.log): `syscall=openat` records where `comm` matches the LogScale JVM process (typically 'java') and the `name` field resolves to file paths outside the LogScale application and data directories (e.g., /etc/passwd, /etc/shadow, /root/.ssh/id_rsa, /opt/logscale/config/) during the exploitation window. | OS-level file access timestamps: `stat` output for sensitive host files (/etc/passwd, /etc/shadow, ~/.ssh/authorized_keys, TLS certificate and key files, application .env and config files) — `atime` updates during the period of exploitation-pattern HTTP traffic confirm the attacker successfully read those files via the path traversal. | Network connection state snapshots: `ss -tnp` or `netstat -antp` output captured at containment time, showing source IPs with established or TIME_WAIT connections to the LogScale HTTP/API port — these represent potential attacker sessions active at the time of discovery and are critical for threat actor attribution and lateral movement assessment. | Pre-patch LogScale binary and JAR SHA-256 hashes: Cryptographic hashes of all LogScale JAR files and the primary application binary collected before patching, used to confirm the vulnerable version was in production during the exposure window and to satisfy chain-of-custody requirements if regulatory breach notification or law enforcement referral is required.

Per-Action IR Details

Step 1: Containment — Immediately restrict network access to your LogScale self-hosted instance. If LogScale is internet-facing, place it behind a WAF or VPN and block unauthenticated external access to the LogScale HTTP/API port until patching is complete. Confirm affected version scope against CrowdStrike's official advisory before assuming you are or are not exposed.

NIST Phase: Containment

Reference: NIST 800-61r3 §3.3 — Containment Strategy

Controls: NIST IR-4 (Incident Handling), NIST SC-7 (Boundary Protection), CIS 4.4 (Implement and Manage a Firewall on Servers), CIS 4.5 (Implement and Manage a Firewall on End-User Devices), CIS 6.3 (Require MFA for Externally-Exposed Applications)

Compensating: Immediately apply a host-based firewall rule to restrict LogScale's HTTP/API port (default 8080 or 443) to trusted internal CIDR ranges only: `sudo ufw deny from any to any port 8080` (Linux/UFW) or `netsh advfirewall firewall add rule name='Block LogScale External' protocol=TCP dir=in localport=8080 action=block remoteip=0.0.0.0/0` (Windows). For teams without a WAF, deploy an nginx reverse proxy with IP allowlisting as a temporary chokepoint. Verify the block is effective using `nmap -p 8080 ` from an external vantage point.

Evidence: Before restricting access, capture a full snapshot of active network connections to the LogScale host (`ss -tnp` or `netstat -antp` on Linux) to identify any currently active or recently established sessions from unexpected source IPs — potential attacker persistence via an open session. Export the LogScale HTTP/API access logs covering the 30 days prior to containment to preserve pre-patch exploitation evidence before log rotation discards them.

Step 2: Detection — Review web server and application access logs on the LogScale host for requests containing path traversal sequences (e.g., '../', '%2e%2e%2f', '%252e%252e%252f', URL-encoded variants). Look for GET or POST requests targeting file paths outside the expected LogScale application directory. Check for unexpected file access patterns in host-level audit logs (e.g., auditd on Linux) referencing sensitive paths such as /etc/passwd, /etc/shadow, SSH key directories, or application config files.

NIST Phase: Detection Analysis

Reference: NIST 800-61r3 §3.2 — Detection and Analysis

Controls: NIST SI-4 (System Monitoring), NIST AU-6 (Audit Record Review, Analysis, and Reporting), NIST AU-3 (Content of Audit Records), CIS 8.2 (Collect Audit Logs), CIS 7.1 (Establish and Maintain a Vulnerability Management Process)

Compensating: Run the following grep against LogScale's HTTP access log to surface path traversal attempts specific to CVE-2026-40050: `grep -iE '(\.\./|%2e%2e%2f|%252e%252e%252f|%c0%af|%c1%9c)' /var/log/logscale/access.log | awk '{print $1, $7, $9}' > traversal_hits.txt`. Cross-reference hits against auditd syscall logs using `ausearch -sc open,openat,read --start today` filtered for file paths outside the LogScale install directory (e.g., `/opt/logscale`). For host file access, deploy the free Sigma rule `proc_creation_lnx_auditd_file_read_sensitive_files.yml` against auditd logs to flag reads of `/etc/passwd`, `/etc/shadow`, and `~/.ssh/authorized_keys` attributable to the LogScale JVM process.

Evidence: Preserve the following before any log rotation: (1) LogScale HTTP/API access logs with full URI paths, source IPs, HTTP status codes, and response body sizes — large 200-OK responses to traversal-pattern URIs indicate successful file read. (2) Linux auditd logs (`/var/log/audit/audit.log`) filtered for `syscall=open` or `syscall=openat` where the process name matches the LogScale JVM (e.g., `java`) and the file path resolves outside `/opt/logscale` or the configured data directory. (3) OS-level file access timestamps (`stat /etc/passwd /etc/shadow`) — anomalous `atime` (access time) updates during the suspected exploitation window confirm file read activity.

Step 3: Eradication — Apply the patch released by CrowdStrike for CVE-2026-40050. The specific patched version number must be confirmed against CrowdStrike's official advisory — do not rely on secondary sources for the target version. After patching, verify the installed LogScale version matches the vendor-confirmed fixed release.

NIST Phase: Eradication

Reference: NIST 800-61r3 §3.4 — Eradication

Controls: NIST SI-2 (Flaw Remediation), NIST SI-7 (Software, Firmware, and Information Integrity), NIST CM-6 (Configuration Settings), CIS 7.3 (Perform Automated Operating System Patch Management), CIS 7.4 (Perform Automated Application Patch Management), CIS 7.2 (Establish and Maintain a Remediation Process)

Compensating: After applying the CrowdStrike-supplied LogScale update package, verify the installed binary version matches the vendor-confirmed fixed release using LogScale's built-in version endpoint: `curl -s http://localhost:8080/api/v1/status | python3 -m json.tool | grep version`. Compute a SHA-256 hash of the updated LogScale JAR or binary and compare against the checksum published in CrowdStrike's advisory: `sha256sum /opt/logscale/lib/logscale-*.jar`. If CrowdStrike publishes a YARA rule for the vulnerable version's binary signature, run it against the post-patch binary to confirm the vulnerable code path is absent: `yara -r crowdstrike_cve_2026_40050_vuln.yar /opt/logscale/`.

Evidence: Before applying the patch, collect a full package manifest of the current LogScale installation (`rpm -qa | grep logscale` or `dpkg -l | grep logscale`) and hash all LogScale JAR files (`find /opt/logscale -name '*.jar' -exec sha256sum {} \;`) to document the pre-patch state for forensic comparison and regulatory chain-of-custody records. If exploitation is suspected, take a memory snapshot of the running LogScale JVM process (`sudo gcore $(pgrep -f logscale)`) before shutdown to preserve in-memory evidence of any active exploit payloads or injected code.

Step 4: Recovery — After patching, validate that path traversal test requests (using safe, non-destructive probe payloads in a staging environment) no longer return out-of-bounds file content. Re-enable external access only after patch verification is complete. Monitor LogScale access logs and host file integrity monitoring (FIM) alerts for at least 72 hours post-remediation for signs of prior or continued exploitation.

NIST Phase: Recovery

Reference: NIST 800-61r3 §3.5 — Recovery

Controls: NIST SI-6 (Security and Privacy Function Verification), NIST SI-7 (Software, Firmware, and Information Integrity), NIST IR-5 (Incident Monitoring), NIST AU-6 (Audit Record Review, Analysis, and Reporting), CIS 7.1 (Establish and Maintain a Vulnerability Management Process)

Compensating: In a staging environment mirroring the production LogScale version, send a safe traversal probe using curl and confirm the patched instance returns HTTP 400 or 403 rather than file content: `curl -v 'http://staging-logscale:8080/api/v1/files/..%2F..%2Fetc%2Fpasswd'`. For FIM on the production host without a commercial agent, deploy AIDE (free, Linux): initialize a baseline database post-patch (`aide --init`) and schedule hourly checks (`aide --check`) against `/etc`, `~/.ssh`, LogScale config directories, and any paths that appeared in exploitation-pattern log hits. Alert on any unexpected `atime` or `mtime` changes to credential files.

Evidence: During the 72-hour post-recovery monitoring window, preserve hourly snapshots of LogScale HTTP access logs and auditd output. Specifically retain any HTTP 4xx responses to traversal-pattern URIs — a sudden drop from pre-patch 200-OK to post-patch 400/403 for the same URI patterns confirms the patch is functioning. If AIDE is deployed, export the change report (`aide --check > /var/log/aide/post_patch_check_$(date +%Y%m%d%H%M).log`) for each check cycle as forensic evidence of system integrity during recovery.

Step 5: Post-Incident — Audit whether sensitive files were accessible on the LogScale host (credentials, API keys, TLS certificates, configuration files containing downstream system access). Review network segmentation controls for SIEM and log pipeline infrastructure — these systems frequently hold credentials and access to broad internal telemetry. Evaluate whether LogScale hosts follow least-privilege file system permissions to limit the blast radius of future path traversal class vulnerabilities.

NIST Phase: Post Incident

Reference: NIST 800-61r3 §4 — Post-Incident Activity

Controls: NIST IR-4 (Incident Handling), NIST IR-8 (Incident Response Plan), NIST AC-6 (Least Privilege), NIST RA-3 (Risk Assessment), NIST AU-11 (Audit Record Retention), CIS 3.2 (Establish and Maintain a Data Inventory), CIS 5.4 (Restrict Administrator Privileges to Dedicated Administrator Accounts), CIS 4.2 (Establish and Maintain a Secure Configuration Process for Network Infrastructure)

Compensating: Enumerate all sensitive files readable by the LogScale process user account to quantify blast radius: `sudo -u logscale find / -readable -type f \( -name '*.key' -o -name '*.pem' -o -name '*.conf' -o -name 'id_rsa' -o -name '.env' \) 2>/dev/null > logscale_readable_sensitive_files.txt`. For each file listed, assess whether the credential or secret it contains grants access to downstream systems (SIEM backends, cloud providers, AD/LDAP). Rotate any credential or API key confirmed readable by the LogScale process user, prioritizing those granting broad internal access. Harden file permissions post-incident: `chmod 640 /opt/logscale/config/*` and ensure the LogScale service account has no read access to `/etc/shadow` or SSH private key directories.

Evidence: Compile the final forensic package: (1) The traversal_hits.txt output from Step 2 grep, annotated with HTTP response sizes to identify which file reads returned substantive content vs. empty responses. (2) The auditd syscall log excerpt showing all `openat` calls by the LogScale JVM process to paths outside the application directory during the exploitation window. (3) The AIDE or `stat`-based atime analysis for `/etc/passwd`, `/etc/shadow`, TLS cert paths, and any `.env` or config files on the host — atime updates during the window confirm attacker file read. (4) Network flow logs or `ss` snapshots showing source IPs that sent traversal-pattern requests, for threat intelligence enrichment and

potential abuse reporting.

Detection Guidance

Focus detection on web access logs and host-level audit logs for the LogScale server. Query for HTTP requests containing path traversal indicators: '../', '..%2f', '%2e%2e/', '%252e%252e%252f', and Unicode-encoded variants. Filter for requests that reference operating system file paths (/etc/, /var/, /home/, /root/, /proc/) rather than LogScale application endpoints. On Linux hosts, auditd rules targeting open() and read() syscalls on sensitive files (/etc/passwd, /etc/shadow, ~/.ssh/authorized_keys, application .env or config files) can surface exploitation attempts. Absence of authentication headers in requests that trigger file reads is a secondary indicator. At time of writing and based on available secondary sources, no confirmed public IOCs (IPs, hashes, domains) have been reported for active exploitation; monitor threat intelligence feeds and CISA alerts for updates. Detection should focus on behavioral patterns rather than static IOCs.

Framework Mappings

MITRE-ATTACK

- **T1190** — Exploit Public-Facing Application
- **T1083** — File and Directory Discovery

NIST-800-53R5

- **CA-8** — Penetration Testing
- **RA-5** — Vulnerability Monitoring and Scanning
- **SC-7** — Boundary Protection
- **SI-2** — Flaw Remediation
- **SI-7** — Software, Firmware, and Information Integrity
- **AC-3** — Access Enforcement
- **SI-10** — Information Input Validation

OWASP-TOP10-2021

- **A01:2021** — Broken Access Control

CIS-V8

- **16.10** — Apply Secure Design Principles in Application Architectures
- **16.12** — Implement Code-Level Security Checks
- **7.3** — Perform Automated Operating System Patch Management
- **7.4** — Perform Automated Application Patch Management

ISO-27001-2022

- **A.8.8** — Management of technical vulnerabilities

MITRE ATT&CK Mapping

Technique ID	Technique Name	Tactic
T1190	Exploit Public-Facing Application	Initial-Access
T1083	File and Directory Discovery	Discovery

Sources

Source	URL	Tier
	https://securityaffairs.com/191343/hacking/critical-bug-in-crowdstr...	T3
Critical bug in CrowdStrike LogScale let attackers access files - Reddit	https://www.reddit.com/r/InfoSecNews/comments/1swjmae/critical_bug_...	T3
Critical bug in CrowdStrike LogScale let attackers access files	https://x.com/Cyber_O51NT/status/2048564925484659068	T3
Critical bug in CrowdStrike LogScale let attackers access files	https://www.linkedin.com/posts/the-cyber-security-hub_critical-bug-...	T3
CrowdStrike Issues Critical Alert: LogScale Vulnerability Allows ...	https://securityonline.info/crowdstrike-logscale-vulnerability-cve-...	T3
NVD	https://nvd.nist.gov/vuln/detail/CVE-2026-40050	T1

Generated 2026-04-27 05:57 UTC by TJS Security Command Center