INTELLIGENCE BRIEFING
Security Command Center
TLP:CLEAR
2026-06-14 06:11 UTC

Splunk Enterprise Critical Pre-Auth RCE (CVE-2026-20253), Exploit Published

CVE VULNERABILITY | CRITICAL | CVSS 9.5

SCC Item ID	SCC-CVE-2026-0299
Type	CVE Vulnerability
CVE ID	CVE-2026-20253
Severity	CRITICAL
CVSS Base Score	9.5
EPSS Score	0.0007 (21th percentile)
Affected Products	Splunk Enterprise 10.0.0-10.0.6, 10.2.0-10.2.3; Splunk Cloud not affected; patched in 10.0.7 and 10.2.4
Published	2026-06-13T09:23:03
Discovery Source	Rss

Executive Summary

A critical unauthenticated remote code execution vulnerability in Splunk Enterprise (CVE-2026-20253, CVSS 9.5) allows any attacker to fully compromise a Splunk server without credentials. Affected versions span Splunk Enterprise 10.0.0-10.0.6 and 10.2.0-10.2.3; a working public exploit was published June 13, 2026, making exploitation accessible to low-skill attackers. Organizations running unpatched Splunk Enterprise instances face immediate risk of full platform compromise, including loss of security visibility, data exfiltration from indexed logs, and potential lateral movement across the enterprise.

Technical Analysis

CVE-2026-20253 is a pre-authentication remote code execution vulnerability in Splunk Enterprise affecting versions 10.0.0-10.0.6 and 10.2.0-10.2.3. The root cause is missing authentication on a PostgreSQL sidecar service endpoint (CWE-306), which is then combined with unrestricted file upload (CWE-434) and code injection (CWE-94) primitives to form a complete unauthenticated exploit chain. An unauthenticated attacker with network access to the exposed endpoint can upload arbitrary files and execute code in the context of the Splunk process. watchTowr Labs published a working proof-of-concept exploit on June 13, 2026, significantly lowering the exploitation barrier. Relevant MITRE ATT&CK techniques include T1190 (Exploit Public-Facing Application), T1059/T1059.006 (Command and Script Interpreter: Python), T1505.003 (Server Software Component: Web Shell), T1222 (File and Directory Permissions Modification), and T1083 (File and Directory Discovery). CVSS

base score is 9.5. Splunk Cloud is not affected. Patches are available in Splunk Enterprise 10.0.7 (for the 10.0.x line) and 10.2.4 (for the 10.2.x line).

Action Checklist

1. Step 1: Containment. Immediately restrict network access to the Splunk Enterprise management and indexer ports (default 8089, 9997, and the PostgreSQL sidecar port) at the firewall or network segmentation layer for all unpatched Splunk Enterprise 10.0.0-10.0.6 and 10.2.0-10.2.3 instances. If internet-facing, take offline or place behind an authenticated proxy until patched. Reference: NIST AC-4 (Information Flow Enforcement), CIS 4.4 (Implement and Manage a Firewall on Servers).
2. Step 2: Detection. Review Splunk internal logs (splunkd.log, audit.log) for unauthenticated requests to the PostgreSQL sidecar endpoint, unexpected file creation events in Splunk app directories, and anomalous process spawning from the Splunk service account. Hunt for MITRE T1190 indicators: external source IPs issuing requests to management ports without a preceding authentication event. Also review for T1505.003 indicators: new or modified files in $SPLUNK_HOME/etc/apps/ or $SPLUNK_HOME/var/run/. Reference: NIST AU-6 (Audit Record Review, Analysis, and Reporting), NIST AU-2 (Event Logging), CIS 8.2 (Collect Audit Logs).
3. Step 3: Eradication. Apply Splunk Enterprise patches: upgrade 10.0.x instances to 10.0.7 and 10.2.x instances to 10.2.4 per Splunk's official advisory. Verify upgrade success by confirming the version string in Splunk Web (Settings > About) or via CLI: splunk version. Remove any unauthorized files or scripts discovered in Step 2. Reference: NIST SI-2 (Flaw Remediation), CIS 7.3 (Perform Automated Operating System Patch Management), CIS 7.4 (Perform Automated Application Patch Management).
4. Step 4: Recovery. After patching, re-enable network access to Splunk services and confirm search, indexing, and forwarding functions are operational. Validate that the PostgreSQL sidecar endpoint now requires authentication. Review Splunk audit logs for any evidence of compromise during the exposure window and confirm log integrity. Reference: NIST AU-9 (Protection of Audit Information), NIST AU-11 (Audit Record Retention).
5. Step 5: Post-Incident. Conduct a lessons-learned review addressing: whether Splunk management ports were unnecessarily internet-exposed (NIST AC-17 Remote Access, CIS 4.4), whether patch management processes detected the June 13 exploit publication and triggered emergency patching (CIS 7.1, CIS 7.2), and whether privileged service accounts running Splunk follow least-privilege principles (NIST AC-6 Least Privilege, CIS 5.4). Document the exposure window for any regulatory reporting obligations.

IR / Forensic Enrichment

Triage Priority	IMMEDIATE
Escalation Criteria	Escalate to CISO, legal counsel, and executive leadership immediately if forensic review of splunkd.log or audit.log reveals any unauthenticated successful request to the PostgreSQL sidecar endpoint during the exposure window, any unauthorized file creation in $SPLUNK_HOME/etc/apps/, or if the compromised Splunk instance indexed data subject to PII, PHI, PCI-DSS, or SEC-regulated environments — as these conditions trigger breach notification obligations and indicate the security monitoring platform itself has been weaponized.

Recovery Notes	After restoring network access to patched Splunk 10.0.7 or 10.2.4 instances, monitor splunkd.log and audit.log continuously for a minimum of 72 hours for re-exploitation attempts against the now-patched endpoint, as the June 13 public exploit will drive automated scanning by opportunistic actors. Verify all Splunk forwarders, search heads, and indexers re-establish authenticated connections and that no forwarder is redirecting data to an unauthorized outputs.conf destination planted during the exposure window. Confirm Splunk search and alerting functions are intact — if this Splunk instance served as the organization's primary SIEM, validate that no detection rules were disabled or deleted by an attacker who leveraged the RCE to blind the SOC.
Forensic Artifacts	$SPLUNK_HOME/var/log/splunk/splunkd.log — primary artifact for unauthenticated requests to the PostgreSQL sidecar endpoint; look for HTTP requests to the sidecar port lacking a valid session_key or authtoken field in the log entry, particularly from external source IPs, during the exposure window between June 13, 2026 (public exploit publication) and containment. | $SPLUNK_HOME/var/log/splunk/audit.log — records Splunk REST API calls and administrative actions; unauthenticated RCE exploitation of CVE-2026-20253 that progresses to persistence would surface here as unauthorized app installs, role modifications, or saved search creation by an unknown or service-account identity. | $SPLUNK_HOME/etc/apps/ and $SPLUNK_HOME/var/run/ directory — the T1505.003 (Server Software Component: Web Shell) post-exploitation path for this vulnerability targets these directories; forensic timeline analysis (via `stat` or `$MFT` on Windows) of files created or modified after the exploit publication date (June 13, 2026) that predate the patch is the primary indicator of a successful implant drop. | Full RAM image of the splunkd process (captured via LiME on Linux or WinPmem on Windows before patching) — the PostgreSQL sidecar RCE is a pre-auth code execution path that may inject shellcode or stage a reverse shell payload entirely in memory; a process memory dump is the only artifact that captures the in-flight exploit payload, C2 callback addresses, or injected threads before splunkd is restarted during upgrade. | OS-level process creation logs for the Splunk service account — on Windows, Sysmon Event ID 1 (Process Creation) filtered on ParentImage matching splunkd.exe; on Linux, auditd records for execve() calls by the splunk UID — these capture any OS command execution (cmd.exe, powershell.exe, bash, sh, wget, curl) spawned by the unauthenticated RCE, which is the definitive indicator of successful exploitation beyond mere vulnerability scanning.

Per-Action IR Details

Step 1: Containment — Immediately restrict network access to the Splunk Enterprise management and indexer ports (default 8089, 9997, and the PostgreSQL sidecar port) at the firewall or network segmentation layer for all unpatched Splunk Enterprise 10.0.0–10.0.6 and 10.2.0–10.2.3 instances. If internet-facing, take offline or place behind an authenticated proxy until patched. Reference: NIST AC-4 (Information Flow Enforcement), CIS 4.4 (Implement and Manage a Firewall on Servers).

NIST Phase: Containment

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

Controls: NIST AC-4 (Information Flow Enforcement), CIS 4.4 (Implement and Manage a Firewall on Servers)

Compensating: On the Splunk host or upstream firewall, apply immediate block rules: `iptables -I INPUT -p tcp --dport 8089 -j DROP && iptables -I INPUT -p tcp --dport 9997 -j DROP` (Linux) or equivalent Windows Firewall rules via `netsh advfirewall firewall add rule name='Block Splunk Mgmt' protocol=TCP dir=in localport=8089,9997 action=block`. Enumerate all internet-facing Splunk instances first using a quick nmap sweep of your IP space: `nmap -p 8089,9997 --open`. A 2-person team can execute ACL changes on a perimeter firewall in parallel with host-level rules.

Evidence: Before isolating any Splunk instance, capture volatile state: run `netstat -ano` (Windows) or `ss -tulnp` (Linux) to record all active connections to ports 8089 and 9997 — document source IPs that may already be mid-exploitation. Capture a list of currently authenticated sessions via the Splunk REST API (`curl -k

https://localhost:8089/services/authentication/httpauth-tokens -u admin:pass`) before firewall rules drop live connections. Record the output of `ps aux | grep splunk` or `Get-Process -Name splunk*` to baseline running child processes spawned by splunkd prior to containment, as the PostgreSQL sidecar RCE may have already launched a shell.

Step 2: Detection — Review Splunk internal logs (splunkd.log, audit.log) for unauthenticated requests to the PostgreSQL sidecar endpoint, unexpected file creation events in Splunk app directories, and anomalous process spawning from the Splunk service account. Hunt for MITRE T1190 indicators: external source IPs issuing requests to management ports without a preceding authentication event. Also review for T1505.003 indicators: new or modified files in $SPLUNK_HOME/etc/apps/ or $SPLUNK_HOME/var/run/. Reference: NIST AU-6 (Audit Record Review, Analysis, and Reporting), NIST AU-2 (Event Logging), CIS 8.2 (Collect Audit Logs).

NIST Phase: Detection Analysis

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

Controls: NIST AU-6 (Audit Record Review, Analysis, And Reporting), NIST AU-2 (Event Logging), CIS 8.2 (Collect Audit Logs)

Compensating: Without a SIEM, grep splunkd.log directly for unauthenticated sidecar hits: `grep -i 'postgres\|sidecar' $SPLUNK_HOME/var/log/splunk/splunkd.log | grep -v 'session_key'` to surface requests lacking a valid session token. For process spawning, deploy Sysmon (Event ID 1 — Process Creation) configured to alert on any child process whose parent image path matches `splunkd.exe` or `/opt/splunk/bin/splunk` spawning cmd.exe, powershell.exe, bash, or sh. Use `find $SPLUNK_HOME/etc/apps/ $SPLUNK_HOME/var/run/ -newer /tmp/baseline_timestamp -type f` (Linux) or `Get-ChildItem -Recurse -Path $env:SPLUNK_HOME\etc\apps\ | Where-Object {$_.LastWriteTime -gt (Get-Date).AddDays(-7)}` (Windows) to identify recently written files consistent with web shell or persistence implant drop.

Evidence: This step is analytical and does not alter live state; however, preserve log files before any patching or reimaging action downstream. Archive $SPLUNK_HOME/var/log/splunk/splunkd.log, audit.log, and web_access.log to a write-protected forensic share immediately. Capture $SPLUNK_HOME/etc/apps/ and $SPLUNK_HOME/var/run/ directory listings with timestamps (`ls -laR` or `Get-ChildItem -Recurse | Select FullName,LastWriteTime,Length`). If the sidecar RCE was triggered, memory may contain injected shellcode in the splunkd process — acquire a full RAM image using WinPmem (Windows) or LiME (Linux) before any remediation step that could terminate the process.

Step 3: Eradication — Apply Splunk Enterprise patches: upgrade 10.0.x instances to 10.0.7 and 10.2.x instances to 10.2.4 per Splunk's official advisory. Verify upgrade success by confirming the version string in Splunk Web (Settings > About) or via CLI: splunk version. Remove any unauthorized files or scripts discovered in Step 2. Reference: NIST SI-2 (no mapped control in provided knowledge base — flag for human review), CIS 7.3 (Perform Automated Operating System Patch Management), CIS 7.4 (Perform Automated Application Patch Management).

NIST Phase: Eradication

Reference: NIST 800-61r3 §3.4 — Eradication

Controls: CIS 7.3 (Perform Automated Operating System Patch Management), CIS 7.4 (Perform Automated Application Patch Management)

Compensating: Download Splunk Enterprise 10.0.7 or 10.2.4 directly from Splunk's official release page (validate SHA-256 hash against the published checksum before installing). Perform the upgrade using Splunk's CLI: `$SPLUNK_HOME/bin/splunk stop && dpkg -i splunk-10.0.7-.deb && $SPLUNK_HOME/bin/splunk start` (Linux .deb) or equivalent MSI on Windows. For each unauthorized file identified in Step 2, compute its SHA-256 hash (`sha256sum ` or `Get-FileHash`) and preserve the hash as evidence before deletion. After removal, run `find $SPLUNK_HOME -name '*.py' -o -name '*.sh' -o -name '*.ps1' | xargs sha256sum > post_eradication_hashes.txt` as a clean-state baseline.

Evidence: CRITICAL — volatile capture must precede patching. Before running the Splunk upgrade, acquire: (1) full memory image of the splunkd process using WinPmem or LiME to preserve any in-memory payload or injected

shellcode from the PostgreSQL sidecar RCE; (2) a forensic copy of any unauthorized files found in $SPLUNK_HOME/etc/apps/ or $SPLUNK_HOME/var/run/ using `dd` or a hash-verified file copy — these are direct evidence of post-exploitation persistence; (3) a running process snapshot (`ps auxf` / `Get-Process`) to document any attacker-spawned child processes that the patch installation or service restart will terminate. Patch installation restarts splunkd and will destroy live process artifacts.

Step 4: Recovery — After patching, re-enable network access to Splunk services and confirm search, indexing, and forwarding functions are operational. Validate that the PostgreSQL sidecar endpoint now requires authentication. Validate that the PostgreSQL sidecar endpoint now requires authentication. Review Splunk audit logs for any evidence of compromise during the exposure window and confirm log integrity. Reference: NIST AU-9 (Protection of Audit Information), NIST AU-11 (Audit Record Retention), D3-SFA (System File Analysis).

NIST Phase: Recovery

Reference: NIST 800-61r3 §3.5 — Recovery

Controls: NIST AU-9 (Protection Of Audit Information), NIST AU-11 (Audit Record Retention)

Compensating: Validate the sidecar endpoint authentication fix without a commercial scanner: issue an unauthenticated curl request to the PostgreSQL sidecar port from a test host (`curl -v -k https://:/`) and confirm a 401 or connection-refused response — a 200 response without credentials indicates the patch did not apply correctly. Confirm search head and indexer connectivity by running `$SPLUNK_HOME/bin/splunk list forward-server` and verifying all forwarders re-register within 15 minutes of firewall rule re-enablement. For log integrity, compare splunkd.log and audit.log SHA-256 hashes against pre-isolation snapshots to detect tampering during the exposure window.

Evidence: Before re-enabling network access, confirm no attacker-planted scheduled tasks or cron jobs persist: enumerate `crontab -l -u splunk` and `Get-ScheduledTask | Where-Object {$_.TaskPath -like '*splunk*'}` — the CVE-2026-20253 RCE could have been used to establish cron/scheduled-task persistence that survives patching. Also verify $SPLUNK_HOME/etc/system/local/ for unauthorized configuration changes (inputs.conf, outputs.conf, authorize.conf) that could redirect log data or grant attacker-controlled accounts elevated roles. These checks do not alter live state but must complete before restoring external access.

Step 5: Post-Incident — Conduct a lessons-learned review addressing: whether Splunk management ports were unnecessarily internet-exposed (NIST AC-17 Remote Access, CIS 4.4), whether patch management processes detected the June 13 exploit publication and triggered emergency patching (CIS 7.1, CIS 7.2), and whether privileged service accounts running Splunk follow least-privilege principles (NIST AC-6 Least Privilege, CIS 5.4). Consider implementing D3-MFA (Multi-factor Authentication) for Splunk administrative access and D3-UAP (User Account Permissions) review for the Splunk service account. Document the exposure window for any regulatory reporting obligations.

NIST Phase: Post Incident

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

Controls: NIST AC-17 (Remote Access), NIST AC-6 (Least Privilege), CIS 4.4 (Implement and Manage a Firewall on Servers), CIS 7.1 (Establish and Maintain a Vulnerability Management Process), CIS 7.2 (Establish and Maintain a Remediation Process), CIS 5.4 (Restrict Administrator Privileges to Dedicated Administrator Accounts)

Compensating: For a 2-person team, structure the lessons-learned review around three bounded questions: (1) Pull firewall logs to produce a definitive list of external IPs that reached port 8089 or 9997 between the affected version's install date and containment — this defines the exposure window for regulatory purposes. (2) Check your vulnerability feed subscription (NVD RSS, CISA KEV email alerts) for the June 13, 2026 CVE-2026-20253 publication date versus the date your team was notified — the gap is your detection latency metric. (3) Run `$SPLUNK_HOME/bin/splunk show config server` to confirm the Splunk service account UID and validate it holds no sudo or local admin rights beyond $SPLUNK_HOME ownership. Document findings in a one-page IR after-action report tied to the specific exposure window.

Evidence: For the lessons-learned record and any regulatory notification, preserve: (1) firewall flow logs or netflow records covering the full exposure window (Splunk Enterprise version install date through containment timestamp)

showing all external connections to ports 8089, 9997, and the PostgreSQL sidecar port — these establish whether exploitation was attempted or successful; (2) the full splunkd.log and audit.log archive from the exposure window with chain-of-custody documentation; (3) a timestamped record of the June 13, 2026 public exploit publication versus your team's first awareness — required for breach notification timelines under regulations such as GDPR Article 33 (72-hour window) or SEC cybersecurity incident disclosure rules if Splunk served as a security monitoring platform for regulated data.

Detection Guidance

Query Splunk's own internal indexes for anomalous pre-authentication activity. Key log sources: $SPLUNK_HOME/var/log/splunk/splunkd.log and $SPLUNK_HOME/var/log/splunk/audit.log. Look for HTTP requests to the PostgreSQL sidecar service port that do not carry a valid session token or authentication header, particularly POST requests with multipart/form-data content type indicating file upload attempts (CWE-434 exploitation pattern). Search for unexpected child processes spawned by splunkd (T1059, T1059.006): index=_internal source=*splunkd.log (python OR subprocess OR exec) | where NOT like(component, "SearchOperator%"). Look for new or modified files in Splunk app directories not corresponding to a known deployment: index=_internal source=*audit.log action=file_write path="*etc/apps*" | stats count by user, path. For network-level detection, alert on any external IP establishing connections to Splunk management port 8089 or the PostgreSQL sidecar port without a corresponding authentication success event. Reference: NIST AU-6, AU-12, AU-2; CIS 8.2.

Framework Mappings

MITRE-ATTACK

- **T1059.006** — Python
- **T1505.003** — Web Shell
- **T1059** — Command and Scripting Interpreter
- **T1222** — File and Directory Permissions Modification
- **T1190** — Exploit Public-Facing Application
- **T1083** — File and Directory Discovery

NIST-800-53R5

- **CM-2** — Baseline Configuration
- **CM-7** — Least Functionality
- **SI-3** — Malicious Code Protection
- **SI-4** — System Monitoring
- **SI-7** — Software, Firmware, and Information Integrity
- **CA-8** — Penetration Testing
- **RA-5** — Vulnerability Monitoring and Scanning
- **SC-7** — Boundary Protection
- **SI-2** — Flaw Remediation
- **SI-10** — Information Input Validation

- **IA-2** — Identification and Authentication (Organizational Users)

OWASP-TOP10-2021

- **A03:2021** — Injection
- **A04:2021** — Insecure Design
- **A07:2021** — Identification and Authentication Failures

CIS-V8

- **16.10** — Apply Secure Design Principles in Application Architectures
- **6.3** — Require MFA for Externally-Exposed Applications
- **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
T1059.006	Python	Execution
T1505.003	Web Shell	Persistence
T1059	Command and Scripting Interpreter	Execution
T1222	File and Directory Permissions Modification	Defense-Evasion
T1190	Exploit Public-Facing Application	Initial-Access
T1083	File and Directory Discovery	Discovery

Sources

Source	URL	Tier
Security News	https://thehackernews.com/2026/06/critical-splunk-enterprise-flaw-l...	T3
CVE-2026-20253 - CVE Record	https://www.cve.org/CVERecord?id=CVE-2026-20253	T3
CVE-2026-45253 Detail - NVD	https://nvd.nist.gov/vuln/detail/CVE-2026-45253	T1
CVE-2026-41253: iTerm2 Remote Code Execution Vulnerability	https://www.sentinelone.com/vulnerability-database/cve-2026-41253/	T3

Source	URL	Tier
Known Exploited Vulnerabilities Catalog | CISA	https://www.cisa.gov/known-exploited-vulnerabilities-catalog	**T1**
NVD	https://nvd.nist.gov/vuln/detail/CVE-2026-20253	**T1**

Generated 2026-06-14 06:11 UTC by TJS Security Command Center