INTELLIGENCE BRIEFING
Security Command Center
TLP:CLEAR
2026-06-02 06:32 UTC

CVE-2026-7459: The Simple History - Track, Log, and Audit WordPress Changes plugin for WordPress is vulnerable to a...

CVE VULNERABILITY | HIGH | CVSS 7.5

SCC Item ID	SCC-CVE-2026-0252
Type	CVE Vulnerability
CVE ID	CVE-2026-7459
Severity	HIGH
CVSS Base Score	7.5
EPSS Score	0.0006 (19th percentile)
Affected Products	Simple History - Track, Log, and Audit WordPress Changes plugin for WordPress, all versions up to and including 5.26.0
Published	2026-05-30T10:16:22.610
Discovery Source	Nvd

Executive Summary

CVE-2026-7459 is a high-severity authenticated account takeover vulnerability in the Simple History WordPress plugin, affecting all versions through 5.26.0. Any WordPress site with a Subscriber-level account and the experimental features option enabled is at risk of full administrator compromise through password-reset token extraction. Exploitation requires no special tools, only a valid low-privilege login and knowledge of the attack path.

Technical Analysis

CVE-2026-7459 affects Simple History - Track, Log, and Audit WordPress Changes plugin versions ≤ 5.26.0 (WordPress). CVSS Base: 7.5 (High). CWEs: CWE-285 (Improper Authorization), CWE-200 (Exposure of Sensitive Information), CWE-640 (Weak Password Recovery Mechanism). MITRE: T1110.001, T1213, T1530, T1078.

Root cause: The REST API endpoints react_to_event() and unreact_to_event() register get_items_permissions_check() as their permission_callback. This check verifies only that the requester is authenticated, it does not enforce the per-logger capability restrictions that Log_Query normally applies. A Subscriber-level user can POST to /wp-json/simple-history/v1/events//react with the _fields=context parameter

to read the full context of any logged event, including SimpleUserLogger entries. SimpleUserLogger records full password-reset email bodies, including reset URLs and reset keys.

Attack chain: (1) Attacker with Subscriber credentials triggers a password reset for an admin account. (2) Attacker enumerates recent event IDs via the reaction endpoint. (3) Attacker reads context.message from the SimpleUserLogger entry to extract the reset key. (4) Attacker completes the password reset, achieving full admin takeover.

Precondition: The WordPress option simple_history_experimental_features_enabled must be set to true. This is disabled by default. Exploitation scope is limited to sites where an administrator has manually enabled experimental features.

Patch status: Users should upgrade beyond 5.26.0. Verify the fixed version in the official plugin changelog at wordpress.org/plugins/simple-history/.

Action Checklist

1. Step 1: Containment. Identify all WordPress installations running Simple History ≤ 5.26.0. Check each site's WordPress options table (or Simple History settings UI) for simple_history_experimental_features_enabled = true. Disable the experimental features option immediately on any site where it is enabled. This removes the exploitable endpoint behavior while patching is arranged. Reference: NIST AC-3 (Access Enforcement).
2. Step 2: Detection. Query WordPress audit logs and web server access logs for POST requests to /wp-json/simple-history/v1/events/*/react or /unreact containing the _fields=context parameter. Flag any such requests originating from Subscriber-level accounts, especially those occurring shortly after a wp-login.php lost password request for an admin account. Review Simple History event logs for anomalous react/unreact activity. Reference: NIST AU-6, CIS 8.2.
3. Step 3: Eradication. Update Simple History to a version after 5.26.0 that includes the fix for CVE-2026-7459 (verify availability in the official plugin changelog at wordpress.org/plugins/simple-history/changelog/). Confirm the update via the WordPress plugin dashboard and verify the installed version number matches or exceeds the patched release. After patching, rotate credentials for any administrator accounts on sites where experimental features were enabled, particularly if Subscriber-level accounts exist. Reference: NIST SI-2 (Flaw Remediation), CIS 7.3, CIS 7.4, D3-CRO (Credential Rotation).
4. Step 4: Recovery. After patching, re-enable experimental features only if operationally required, verifying the fixed version is installed first. Audit all WordPress administrator accounts for unexpected changes: password reset timestamps, email address changes, or unfamiliar login sessions. Re-validate that permission_callback enforcement on REST endpoints is operating correctly by reviewing plugin release notes. Monitor web server logs for continued enumeration attempts against the react/unreact endpoints. Reference: NIST IR-4, AU-6.
5. Step 5: Post-Incident. Review the process for enabling experimental or beta features in production WordPress environments. Implement a change control requirement that experimental features are approved before enabling. Audit REST API exposure: confirm WordPress REST API routes are not broadly accessible without authentication where not required (NIST AC-3, AC-6). Review Subscriber-level account inventory and remove dormant or unnecessary low-privilege accounts (CIS 5.3, CIS 5.4). Add detection rules for REST API enumeration patterns to your WAF or SIEM baseline.

Detection Guidance

Primary indicators: POST requests to /wp-json/simple-history/v1/events//react or /wp-json/simple-history/v1/events//unreact with the query parameter _fields=context. Look for these in web server access logs (Apache/Nginx) and any WAF logs. Example grep command: grep -E '/wp-json/simple-history/v1/events/[0-9]+/(react|unreact).*_fields=context' access.log. Correlate against authentication logs: a Subscriber-level login followed within a short window by a wp-login.php?action=lostpassword request targeting an admin username, then REST API enumeration, is a strong indicator of active exploitation.

Log sources to query:

- Web server access logs: grep for '/wp-json/simple-history/v1/events/' and 'react' with '_fields=context'

- WordPress authentication logs (if logging plugin is in place): look for lost password requests for admin-class accounts

- Simple History's own event log: look for high volumes of react/unreact events from a single low-privilege user

- WP-CLI or database: SELECT option_value FROM wp_options WHERE option_name = 'simple_history_experimental_features_enabled'; value of '1' or 'true' confirms the precondition is met

Behavioral indicators: Sequential numeric event ID enumeration in a short time window from a single source IP, Subscriber-class user reading context fields of events logged by SimpleUserLogger, and admin password reset completion shortly after the enumeration pattern.

No public IOCs (IPs, hashes, domains) are currently associated with active exploitation of this CVE. EPSS score is 0.061% (19th percentile), indicating low observed exploitation pressure at time of publication. CISA KEV: not listed.

Framework Mappings

MITRE-ATTACK

- **T1110.001** — Password Guessing
- **T1213** — Data from Information Repositories
- **T1530** — Data from Cloud Storage
- **T1078** — Valid Accounts

NIST-800-53R5

- **AC-2** — Account Management
- **AC-6** — Least Privilege
- **IA-2** — Identification and Authentication (Organizational Users)
- **IA-5** — Authenticator Management
- **AC-3** — Access Enforcement
- **SC-28** — Protection of Information at Rest

OWASP-TOP10-2021

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

HIPAA-SECURITY

- **164.312(a)(1)** — Access Control

ISO-27001-2022

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

MITRE ATT&CK Mapping

Technique ID	Technique Name	Tactic
T1110.001	Password Guessing	Credential-Access
T1213	Data from Information Repositories	Collection
T1530	Data from Cloud Storage	Collection
T1078	Valid Accounts	Defense-Evasion

Sources

Source	URL	Tier
nvd	https://nvd.nist.gov/vuln/detail/CVE-2026-7459	T1
Track, Log, and Audit WordPress Changes (CVE-2026-7459) | Freshy	https://freshysites.com/security-bulletins/simple-history-track-log...	T3
Simple History WordPress Plugin Account Takeover (CVE-2026-7459)	https://threat-modeling.com/simple-history-wordpress-plugin-account...	T3
CVE-2026-7459: CWE-640 Weak Password Recovery Mechanism ...	https://radar.offseq.com/threat/cve-2026-7459-cwe-640-weak-password...	T3
Known Exploited Vulnerabilities Catalog | CISA	https://www.cisa.gov/known-exploited-vulnerabilities-catalog	T1

Generated 2026-06-02 06:32 UTC by TJS Security Command Center