INTELLIGENCE BRIEFING
Security Command Center
TLP:CLEAR
2026-04-28 06:34 UTC

CVE-2026-31673: In the Linux kernel, the following vulnerability has been resolved: af_unix: read UNIX_DIAG_VFS dat...

CVE VULNERABILITY | HIGH | CVSS 7.8

SCC Item ID	SCC-CVE-2026-0084
Type	CVE Vulnerability
CVE ID	CVE-2026-31673
Severity	HIGH
CVSS Base Score	7.8
EPSS Score	0.0002 (5th percentile)
Affected Products	Linux kernel (af_unix subsystem); specific version range not confirmed in available data
Published	2026-04-25T09:16:00.423
Discovery Source	Nvd

Executive Summary

CVE-2026-31673 is a race condition in the Linux kernel's UNIX domain socket subsystem that can allow a local attacker to trigger a use-after-free condition, potentially enabling privilege escalation to root. Organizations running Linux-based servers, containers, or workstations on unpatched kernel versions are exposed. The business risk is unauthorized elevation of access on any affected system, which could undermine host-level security controls across on-premises and cloud infrastructure.

Technical Analysis

CVE-2026-31673 (CWE-362: Race Condition, CWE-416: Use-After-Free) affects the af_unix subsystem of the Linux kernel. During UNIX domain socket diagnostic lookups via UNIX_DIAG_VFS netlink requests, the kernel holds a reference to the socket but not to the associated VFS path object (u->path). Concurrently, unix_release_sock() can clear u->path under unix_state_lock() and drop the path reference after releasing the lock. This creates a use-after-free or invalid memory access window when reading inode and device numbers for the UNIX_DIAG_VFS netlink attribute. CVSS base score: 7.8 (High). EPSS: 0.018% (4.6th percentile), indicating minimal probability of near-term exploitation. MITRE ATT&CK mapping: T1068 (Exploitation for Privilege Escalation). Not listed in CISA KEV. The upstream fix reads VFS path data while holding unix_state_lock(), then emits the netlink attribute after releasing the lock, closing the race window. Specific affected kernel version ranges are not confirmed in available data; consult the upstream Linux kernel changelog

and your distribution vendor advisory for version scope. Source: NVD (nvd.nist.gov/vuln/detail/CVE-2026-31673, T1).

Action Checklist

1. Containment: Identify all Linux hosts (servers, containers, VMs, workstations) in your environment running unpatched kernel versions. Prioritize systems where untrusted local users or container workloads can execute code, as exploitation requires local access.
2. Detection: Review audit logs and kernel logs (dmesg, /var/log/kern.log) for anomalous UNIX socket diagnostic activity or unexpected privilege changes. Monitor for T1068-related behavioral indicators: unexpected setuid execution, process privilege changes, or anomalous netlink socket activity from non-root processes. No public exploit or IOC signatures confirmed as of this item's data.
3. Eradication: Apply the kernel patch addressing CVE-2026-31673 from your Linux distribution vendor. Check your vendor's security advisory portal (e.g., Red Hat, Ubuntu, SUSE, Debian) for availability and version scope before patching. Confirm the fix is present in your running kernel version before closing the finding.
4. Recovery: After patching, reboot affected systems to load the updated kernel. Validate the running kernel version matches the patched release. Review any privileged access granted during the exposure window and rotate credentials if privilege escalation activity is suspected.
5. Post-Incident: Evaluate your kernel patch cadence for Linux hosts. If patch deployment lagged, assess whether a vulnerability management SLA for High-severity kernel CVEs is defined and enforced. Consider enabling Linux Audit Framework rules to monitor netlink socket usage and privilege escalation attempts as a standing detective control.

IR / Forensic Enrichment

Triage Priority	URGENT
Escalation Criteria	Escalate to CISO and initiate full incident response if auditd or dmesg evidence confirms a non-root process successfully executed setuid syscalls or gained UID 0 on any host during the CVE-2026-31673 exposure window, or if any system hosts PII, PHI, PCI-scoped data, or is reachable by multi-tenant container workloads where lateral movement from a compromised host could breach tenant isolation.
Recovery Notes	After loading the patched kernel, monitor auditd logs for residual privilege escalation attempts for a minimum of 72 hours, as a successful pre-patch exploitation may have installed a persistence mechanism (backdoor user, SUID binary, or LD_PRELOAD hook) that survives the kernel update. Validate integrity of SUID binaries and `/etc/passwd` against a trusted baseline or configuration management snapshot. For any host where exploitation is suspected but not confirmed, treat the system as compromised, rebuild from a known-good image, and perform post-incident forensic review of the preserved memory and log artifacts before returning to production.

Forensic Artifacts	dmesg output and /var/log/kern.log: look for KASAN use-after-free reports, kernel BUG stack traces, or slab corruption messages referencing 'af_unix', 'unix_diag', or 'UNIX_DIAG_VFS' — these are the direct kernel-level indicators of CVE-2026-31673 exploitation | auditd logs (ausearch output or /var/log/audit/audit.log): syscall records for AF_UNIX socket(2) calls (a0=1) from non-root UIDs, followed by setuid/setreuid/setresuid syscalls resulting in UID transition to 0, which would indicate successful privilege escalation via the use-after-free condition | LiME memory image (acquired before reboot): heap analysis of the kernel slab allocator may reveal freed unix_sock structures still referenced by a racing netlink diagnostic path, providing forensic confirmation of the race condition trigger | /var/log/auth.log or /var/log/secure: unexpected sudo grants, su sessions, or PAM authentication events from non-privileged users timed within the exploitation window that would indicate post-exploitation lateral movement or persistence after gaining root via CVE-2026-31673 | /proc/net/unix snapshot and lsof -U output captured before patching: documents the state of all UNIX domain socket file descriptors at the time of suspected exploitation, enabling reconstruction of which processes held references to the potentially freed unix_sock object

Per-Action IR Details

Containment — Identify all Linux hosts (servers, containers, VMs, workstations) in your environment running unpatched kernel versions. Prioritize systems where untrusted local users or container workloads can execute code, as exploitation requires local access.

NIST Phase: Containment

Reference: NIST 800-61r3 §3.3 — Containment Strategy: isolate affected assets and prioritize based on exploitation prerequisites (local access required for CVE-2026-31673 use-after-free via af_unix).

Controls: NIST IR-4 (Incident Handling), NIST CM-8 (System Component Inventory), CIS 1.1 (Establish and Maintain Detailed Enterprise Asset Inventory), CIS 2.2 (Ensure Authorized Software is Currently Supported)

Compensating: Run `uname -r` on each host or deploy a one-liner survey via SSH: `for h in $(cat hosts.txt); do ssh $h 'echo "$(hostname): $(uname -r)"'; done`. For container environments, enumerate running container base images with `docker inspect --format '{{.Config.Image}}' $(docker ps -q)` and cross-reference kernel version with the host. Use osquery with `SELECT version FROM kernel_info;` for bulk enumeration across the fleet. Maintain a spreadsheet of host/kernel-version/patch-status updated before each containment decision.

Evidence: Before taking containment action, snapshot the current kernel version and loaded modules on each host: `uname -r`, `lsmod | grep unix`, and `cat /proc/net/unix` to baseline UNIX domain socket state. Capture running process list with `ps auxf` and open file descriptors with `lsof -U` (UNIX domain sockets) to establish a pre-patch baseline for comparison if exploitation is later suspected.

Detection — Review audit logs and kernel logs (dmesg, /var/log/kern.log) for anomalous UNIX socket diagnostic activity or unexpected privilege changes. Monitor for T1068-related behavioral indicators: unexpected setuid execution, process privilege changes, or anomalous netlink socket activity from non-root processes. No public exploit or IOC signatures confirmed as of this item's data.

NIST Phase: Detection Analysis

Reference: NIST 800-61r3 §3.2 — Detection and Analysis: correlate kernel-level log sources and behavioral indicators consistent with local privilege escalation via use-after-free in the af_unix subsystem (MITRE T1068 — Exploitation for Privilege Escalation).

Controls: NIST SI-4 (System Monitoring), NIST AU-2 (Event Logging), NIST AU-6 (Audit Record Review, Analysis, and Reporting), NIST IR-5 (Incident Monitoring), CIS 8.2 (Collect Audit Logs)

Compensating: Enable Linux Audit Framework (auditd) with rules targeting privilege escalation indicators specific to this vector: `auditctl -a always,exit -F arch=b64 -S setuid -S setgid -k priv_esc` and `auditctl -a always,exit -F arch=b64 -S socket -F a0=1 -k unix_socket` (AF_UNIX = 1). Search dmesg for kernel BUG or use-after-free messages: `dmesg | grep -iE 'use.after.free|BUG|KASAN|af_unix|unix_diag'`. For netlink socket abuse from non-root processes, run: `ss -xp

| awk '$NF !~ /pid=1,/ && $1=="u_str"'` and flag unexpected non-root PIDs. Deploy the Sigma rule for Linux privilege escalation (sigma/rules/linux/process_creation/proc_creation_lnx_setuid_execution.yml) if log forwarding is available.

Evidence: Preserve dmesg output immediately (`dmesg > /tmp/dmesg_$(date +%s).txt`) before any reboot — kernel panic traces, KASAN reports, or use-after-free stack traces referencing `unix_diag` or `af_unix` will be lost on reboot. Collect `/var/log/kern.log` and `/var/log/syslog` entries timestamped around any suspected exploitation window. Pull auditd logs for syscalls: `ausearch -sc socket -sc setuid -sc setreuid --start today` to identify non-root processes opening AF_UNIX sockets or gaining elevated UID.

Eradication — Apply the kernel patch addressing CVE-2026-31673 from your Linux distribution vendor (Red Hat, Ubuntu, SUSE, Debian, etc.) as soon as it is available. Confirm the fix is present in your running kernel version before closing the finding. Reference: Red Hat advisory at access.redhat.com/security/cve/cve-2026-31673.

NIST Phase: Eradication

Reference: NIST 800-61r3 §3.4 — Eradication: remove the vulnerable kernel version from all affected hosts and verify the af_unix use-after-free fix is present in the running kernel before returning systems to production.

Controls: NIST SI-2 (Flaw Remediation), NIST SI-7 (Software, Firmware, and Information Integrity), NIST CM-8 (System Component Inventory), CIS 7.3 (Perform Automated Operating System Patch Management), CIS 7.2 (Establish and Maintain a Remediation Process)

Compensating: For Red Hat/CentOS/Rocky: `yum update kernel` then verify with `rpm -q --changelog kernel | grep CVE-2026-31673`. For Ubuntu/Debian: `apt-get update && apt-get install --only-upgrade linux-image-$(uname -r)` then verify with `apt-cache show linux-image-$(uname -r) | grep CVE-2026-31673`. For unmanaged hosts without vendor patch availability, apply the upstream kernel commit that resolves the af_unix UNIX_DIAG_VFS race condition as a temporary measure — build and stage the patched kernel in a test environment first. Validate patch presence before closing: `grep -r 'CVE-2026-31673' /usr/share/doc/linux*/changelog.gz 2>/dev/null || zgrep 'CVE-2026-31673' /usr/share/doc/linux-image-*/changelog.Debian.gz`.

Evidence: Before applying the patch, capture the exact running kernel version and build metadata: `uname -a`, `cat /proc/version`, and `rpm -qa kernel` or `dpkg -l linux-image*`. Document the installed kernel package hash for chain-of-custody if exploitation is suspected: `rpm -V kernel` (RPM) or `dpkg --verify linux-image-$(uname -r)` (Debian). If a reboot has not yet occurred since suspected exploitation, preserve a full memory image using LiME (Linux Memory Extractor) before patching, as heap artifacts from the use-after-free condition in the af_unix slab allocator may be recoverable.

Recovery — After patching, reboot affected systems to load the updated kernel. Validate the running kernel version matches the patched release. Review any privileged access granted during the exposure window and rotate credentials if privilege escalation activity is suspected.

NIST Phase: Recovery

Reference: NIST 800-61r3 §3.5 — Recovery: restore systems to known-good state by loading the patched kernel, validate integrity of the running environment, and remediate any access granted under potentially compromised privilege state.

Controls: NIST IR-4 (Incident Handling), NIST CP-10 (System Recovery and Reconstitution), NIST AC-2 (Account Management), NIST AU-11 (Audit Record Retention), CIS 5.1 (Establish and Maintain an Inventory of Accounts), CIS 6.2 (Establish an Access Revoking Process)

Compensating: After reboot, confirm the patched kernel is active: `uname -r` output must match the patched version from the vendor advisory. Audit for new privileged accounts or group membership changes created during the exposure window: `grep -E '(sudo|wheel|adm)' /etc/group` and `awk -F: '$3==0' /etc/passwd` to identify any UID-0 accounts beyond root. Check for new SUID binaries installed post-exposure: `find / -perm -4000 -newer /var/log/dpkg.log 2>/dev/null` (Debian) or `find / -perm -4000 -newer /var/log/yum.log 2>/dev/null` (RPM). For containers, rebuild affected images from a verified base image rather than patching in place.

Evidence: Before rebooting, capture the final pre-patch system state: running process tree (`ps auxf > /tmp/procs_prepatch.txt`), active network connections (`ss -tulnp > /tmp/netstat_prepatch.txt`), and current user session list (`who`, `last`, `lastb`). Post-reboot, compare `/etc/passwd`, `/etc/shadow` (hash only), and `/etc/sudoers`

modification timestamps against known-good backups or configuration management baselines to detect persistence mechanisms installed by a successful privilege escalation exploiting CVE-2026-31673.

Post-Incident — Evaluate your kernel patch cadence for Linux hosts. If patch deployment lagged, assess whether a vulnerability management SLA for High-severity kernel CVEs is defined and enforced. Consider enabling Linux Audit Framework rules to monitor netlink socket usage and privilege escalation attempts as a standing detective control.

NIST Phase: Post Incident

Reference: NIST 800-61r3 §4 — Post-Incident Activity: conduct lessons-learned review focused on kernel patch SLA gaps, update detection capability to monitor af_unix and netlink abuse as a standing control, and share findings to improve organizational response posture.

Controls: NIST IR-4 (Incident Handling), NIST IR-8 (Incident Response Plan), NIST SI-2 (Flaw Remediation), NIST SI-5 (Security Alerts, Advisories, and Directives), NIST AU-2 (Event Logging), CIS 7.1 (Establish and Maintain a Vulnerability Management Process), CIS 7.2 (Establish and Maintain a Remediation Process), CIS 8.2 (Collect Audit Logs)

Compensating: Implement persistent auditd rules for af_unix and netlink privilege escalation detection by adding to `/etc/audit/rules.d/priv_esc.rules`: `-a always,exit -F arch=b64 -S socket -F a0=1 -F key=unix_socket_monitor` and `-a always,exit -F arch=b64 -S setuid -S setreuid -S setresuid -F auid>=1000 -F key=priv_esc_attempt`. Use `augenrules --load` to activate without reboot. Schedule a monthly kernel version audit cron job: `0 6 1 * * root uname -r >> /var/log/kernel_version_audit.log`. Subscribe to vendor security mailing lists (RHSA, USN, DSA) and define a written SLA: High-severity kernel CVEs (CVSS ≥ 7.0, local privilege escalation) should be patched within 30 days of vendor advisory, with emergency patching within 72 hours if active exploitation is confirmed.

Evidence: Collect and preserve the full audit trail from the exposure window — auditd logs, dmesg captures, and `/var/log/auth.log` or `/var/log/secure` entries — for the lessons-learned review and potential regulatory documentation. Document the time delta between CVE-2026-31673 publication date and patch deployment across all affected hosts as a metric for SLA assessment. Retain this data per your log retention policy (NIST AU-11) and incident documentation requirements (NIST IR-5).

Detection Guidance

No public exploit code or active campaign IOCs are associated with CVE-2026-31673 as of this item's data (EPSS: 0.018%, 4.6th percentile). Detection focus should be behavioral. Monitor kernel logs (dmesg, journalctl -k) for kernel oops, null pointer dereferences, or use-after-free crash messages referencing af_unix or unix_diag. Use Linux Audit Framework (auditd) to log netlink socket calls (SOCK_RAW, NETLINK_SOCK_DIAG) from non-root processes. Alert on unexpected privilege escalation events: monitor for processes gaining elevated capabilities (cap_setuid, cap_setgid) without a corresponding authorized sudo or su event. In containerized environments, review container runtime logs for anomalous socket diagnostic calls. No attack infrastructure, public exploit, or malware samples are confirmed associated with this CVE as of the data collection date.

Framework Mappings

MITRE-ATTACK

- **T1068** — Exploitation for Privilege Escalation

NIST-800-53R5

- **AC-6** — Least Privilege

- **SC-7** — Boundary Protection
- **SI-2** — Flaw Remediation
- **SI-16** — Memory Protection

CIS-V8

- **16.10** — Apply Secure Design Principles in Application Architectures

ISO-27001-2022

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

MITRE ATT&CK Mapping

Technique ID	Technique Name	Tactic
T1068	Exploitation for Privilege Escalation	Privilege-Escalation

Sources

Source	URL	Tier
nvd	https://nvd.nist.gov/vuln/detail/CVE-2026-31673	T1
(consolidated)	https://nvd.nist.gov/vuln/detail/CVE-2026-31674	T1
(consolidated)	https://nvd.nist.gov/vuln/detail/CVE-2026-31675	T1
(consolidated)	https://nvd.nist.gov/vuln/detail/CVE-2026-31676	T1
(consolidated)	https://nvd.nist.gov/vuln/detail/CVE-2026-31678	T1
(consolidated)	https://nvd.nist.gov/vuln/detail/CVE-2026-31679	T1
(consolidated)	https://nvd.nist.gov/vuln/detail/CVE-2026-31680	T1
(consolidated)	https://nvd.nist.gov/vuln/detail/CVE-2026-31682	T1
(consolidated)	https://nvd.nist.gov/vuln/detail/CVE-2026-31683	T1
(consolidated)	https://nvd.nist.gov/vuln/detail/CVE-2026-31685	T1
CVE-2026-31673 - Red Hat Customer Portal	https://access.redhat.com/security/cve/cve-2026-31673	T3

DISCLAIMER

This intelligence report is produced by Tech Jacks Solutions Security Command Center (SCC) for informational purposes only. It does not constitute professional security advice, legal counsel, or an incident response engagement. The information herein is derived from publicly available sources and AI-assisted analysis; while every effort is made to ensure accuracy, Tech Jacks Solutions makes no warranties regarding completeness or timeliness. Organizations should conduct their own validation and consult qualified security professionals before taking action based on this report. Tech Jacks Solutions is not liable for any damages resulting from the use of this information.

Generated 2026-04-28 06:34 UTC by TJS Security Command Center