INTELLIGENCE BRIEFING
Security Command Center
TLP:CLEAR
2026-04-05 13:24 UTC

Google Chrome Zero-Day Vulnerability Exploited in Active Attacks (2026)

CVE VULNERABILITY | HIGH | CVSS 8.8

SCC Item ID	SCC-CVE-2026-0028
Type	CVE Vulnerability
Severity	HIGH
CVSS Base Score	8.8
Affected Products	Google Chrome (multiple versions, pre-patch; affects approximately 3.5 billion users across platforms)
Published	2 hours ago
Discovery Source	Serper

Executive Summary

Google has patched a high-severity zero-day vulnerability in Chrome that attackers are actively exploiting through malicious webpages (per Bleeping Computer reporting). This is the fourth Chrome zero-day exploited in attacks in 2026, affecting all major platforms. Chrome has approximately 3.5 billion users globally; actual exposure scope depends on patch adoption rates. Any organization running unpatched Chrome browsers faces direct risk of code execution on endpoints; immediate patching is the required response.

Technical Analysis

Google released an emergency Chrome update addressing a zero-day vulnerability confirmed as exploited in the wild. The attack vector is malicious webpages, consistent with MITRE ATT&CK T1189 (Drive-by Compromise) and T1203 (Exploitation for Client Execution), indicating a renderer or browser engine flaw that enables arbitrary code execution without requiring additional user interaction beyond visiting a compromised or attacker-controlled page. A CVSS base score of 8.8 (High) has been assigned. A specific CVE identifier and CWE classification were not available in source material at time of writing; NVD and the Chrome Releases blog should be consulted for the authoritative CVE record. No CISA KEV listing was confirmed in the available data, though active exploitation makes KEV addition likely. EPSS scoring data was not available. This is the fourth Chrome zero-day exploited in attacks in 2026 per Bleeping Computer reporting. No specific threat actor attribution is confirmed. Affected scope covers Chrome versions prior to the patched release across Windows, macOS, and Linux. The Stable channel update details are published at https://chromereleases.googleblog.com, consult that source directly for the patched version number, as the specific build was not confirmed in available

sources.

Action Checklist

1. Step 1: Containment. Immediately identify all endpoints running unpatched Chrome across your environment using endpoint management tooling (Intune, SCCM, Jamf, or equivalent). Prioritize internet-facing workstations, developer machines, and privileged-access endpoints. Consider temporarily blocking access to uncategorized or suspicious web destinations via proxy or DNS filtering (if available in your environment) until patching is confirmed.
2. Step 2: Detection. Query endpoint detection logs for unusual Chrome renderer subprocess activity, unexpected child processes spawned from chrome.exe or Google Chrome Helper, and lateral movement or persistence artifacts originating from browser processes. Review web proxy logs for visits to newly registered or low-reputation domains immediately before any anomalous endpoint behavior. No confirmed IOCs were available in source material; monitor Chrome Releases blog and CISA KEV for published indicators.
3. Step 3: Eradication. Deploy the patched Chrome version to all endpoints via your software distribution platform. Verify the installed version matches or exceeds the patched build published at https://chromereleases.googleblog.com. For environments where Chrome auto-update is disabled, push the update manually and confirm version compliance. Remove or isolate any endpoint where post-exploitation activity is suspected pending forensic review.
4. Step 4: Recovery. After patching, validate Chrome version compliance across the fleet using endpoint inventory reports. Re-enable any web access controls relaxed during containment. Monitor EDR telemetry for 24-48 hours post-patch for residual anomalous browser subprocess behavior. Confirm no persistence mechanisms were established on endpoints that may have been exposed prior to patching.
5. Step 5: Post-Incident. Assess whether Chrome auto-update policies are enforced across the environment; gaps here allowed exposure windows for all four 2026 Chrome zero-days. Review browser isolation controls and consider evaluating enhanced browser security configurations per CIS Benchmark for Chrome. Evaluate whether privilege levels on endpoints running Chrome align with least-privilege principles to limit code execution impact. Document time-to-patch metrics for this event against your patching SLA targets.

IR / Forensic Enrichment

Triage Priority	IMMEDIATE
Escalation Criteria	Escalate to CISO and legal counsel immediately if Sysmon or proxy log analysis confirms any endpoint visited the malicious delivery domain or if post-exploitation artifacts (anomalous Chrome child processes, dropped executables, or new persistence mechanisms) are found on endpoints with access to PII, PHI, PCI-scoped systems, or privileged credentials, as these conditions may trigger breach notification obligations under HIPAA, PCI DSS, or applicable state breach notification laws.

Recovery Notes	After fleet-wide patch verification, maintain elevated monitoring of Chrome subprocess activity via Sysmon Event ID 1 for a minimum of 72 hours, specifically watching for cmd.exe, powershell.exe, or any scripting interpreter spawned with chrome.exe or Google Chrome Helper as the parent process, which would indicate a pre-patch compromise that survived the update. Verify Chrome extension inventories on all endpoints that accessed suspicious domains during the exposure window, as malicious extensions installed via post-exploitation are not removed by a Chrome version update. Confirm that any DNS or proxy blocking rules applied during containment are formally reviewed before removal to avoid re-exposing the environment to domains that served the exploit.
Forensic Artifacts	Chrome User Data SQLite databases — specifically `%LocalAppData%\Google\Chrome\User Data\Default\History` and `Web Data` — to recover the URL of the malicious delivery webpage, which is the primary evidence of exploitation delivery for this browser-based zero-day | Sysmon Event ID 1 (Process Create) logs filtered on ParentImage containing chrome.exe or Google Chrome Helper with child processes that are not legitimate Chrome subprocess types (renderer, gpu-process, utility, crashpad_handler) — abnormal child processes are the direct forensic signature of a successful renderer sandbox escape following exploitation | Volatile memory dump (WinPmem or equivalent) from endpoints with suspected compromise, to recover in-memory shellcode or injected code from the Chrome renderer process that would not be written to disk and would not survive a reboot or patch deployment | Web proxy or DNS debug logs covering the 60-minute window prior to any detected anomalous Chrome subprocess activity, to identify the specific domain serving the exploit — this is the network-layer artifact of the malicious webpage delivery vector described in this advisory | Windows Scheduled Tasks export and `HKCU\Software\Microsoft\Windows\CurrentVersion\Run` registry hive snapshot from any endpoint with confirmed or suspected exploitation, to detect attacker-established persistence mechanisms that may have been deployed after a successful renderer escape and privilege escalation on the endpoint

Per-Action IR Details

Step 1: Containment — Immediately identify all endpoints running unpatched Chrome across your environment using endpoint management tooling (Intune, SCCM, Jamf, or equivalent). Prioritize internet-facing workstations, developer machines, and privileged-access endpoints. Consider temporarily blocking access to uncategorized or suspicious web destinations via proxy or DNS filtering until patching is confirmed.

NIST Phase: Containment

Reference: NIST 800-61r3 §3.3 — Containment, Eradication, and Recovery: Containment Strategy

Controls: NIST IR-4 (Incident Handling), NIST CM-7 (Least Functionality), NIST SC-7 (Boundary Protection), CIS 1.1 (Establish and Maintain Detailed Enterprise Asset Inventory), CIS 4.4 (Implement and Manage a Firewall on Servers), CIS 4.5 (Implement and Manage a Firewall on End-User Devices)

Compensating: Run the following PowerShell one-liner on Windows endpoints to enumerate installed Chrome version across the domain via remote registry or WMI: `Get-WmiObject -Class Win32_Product -Filter "Name like '%Chrome%'" | Select-Object Name, Version, PSComputerName`. For macOS endpoints managed without Jamf, use: `find /Applications -name 'Google Chrome.app' -exec defaults read {}/Contents/Info.plist CFBundleShortVersionString \;`. For DNS-layer blocking of uncategorized domains without a commercial proxy, deploy Pi-hole with a blocklist targeting newly registered domains (NRDs); the Hagezi NRD blocklist is a maintained free option. Document all unpatched endpoints in a tracking spreadsheet before proceeding.

Evidence: Before isolating or patching any endpoint, capture: (1) Chrome's current installed version from `HKLM\SOFTWARE\Google\Update\Clients\{8A69D345-D564-463C-AFF1-A69D9E530F96}\pv` (Windows registry); (2) Chrome process tree snapshot via Sysmon Event ID 1 (Process Create) showing chrome.exe and all child

renderer/GPU/utility processes; (3) Network connection state from `netstat -ano` or Sysmon Event ID 3 (Network Connection) filtered on chrome.exe PIDs to capture any active C2 or exfiltration connections established before containment; (4) Web proxy or DNS resolver logs for the 24-hour window preceding detection showing domains visited by the endpoint.

Step 2: Detection — Query endpoint detection logs for unusual Chrome renderer subprocess activity, unexpected child processes spawned from chrome.exe or Google Chrome Helper, and lateral movement or persistence artifacts originating from browser processes. Review web proxy logs for visits to newly registered or low-reputation domains immediately before any anomalous endpoint behavior. No confirmed IOCs were available in source material; monitor Chrome Releases blog and CISA KEV for published indicators.

NIST Phase: Detection Analysis

Reference: NIST 800-61r3 §3.2 — Detection and Analysis: Signs of an Incident and Incident Analysis

Controls: NIST IR-5 (Incident Monitoring), NIST SI-4 (System Monitoring), NIST AU-6 (Audit Record Review, Analysis, and Reporting), NIST AU-12 (Audit Record Generation), CIS 8.2 (Collect Audit Logs), CIS 7.1 (Establish and Maintain a Vulnerability Management Process)

Compensating: Deploy Sysmon with the SwiftOnSecurity or Olaf Hartong configuration to capture process creation (Event ID 1), network connections (Event ID 3), and file creation (Event ID 11) for all Chrome-related processes. Use this PowerShell query to hunt for anomalous Chrome child processes in Windows Event Logs (requires Sysmon): `Get-WinEvent -LogName 'Microsoft-Windows-Sysmon/Operational' | Where-Object {$_.Message -match 'chrome.exe' -and $_.Id -eq 1} | Select-Object TimeCreated, Message | Where-Object {$_.Message -notmatch 'renderer|gpu-process|utility|crashpad|chrome.exe'}`. On Linux/macOS, use osquery with: `SELECT pid, ppid, name, path, cmdline FROM processes WHERE name NOT IN ('chrome','Google Chrome Helper','crashpad_handler') AND ppid IN (SELECT pid FROM processes WHERE name LIKE '%chrome%');`. Apply the Sigma rule 'proc_creation_win_browser_suspicious_child_process' from the SigmaHQ repository against collected Sysmon logs using the free sigmac converter.

Evidence: Capture before any remediation: (1) Sysmon Event ID 1 logs filtered on ParentImage containing 'chrome.exe' or 'Google Chrome Helper' where the child Image is cmd.exe, powershell.exe, wscript.exe, mshta.exe, or any shell/interpreter — these represent post-exploitation command execution spawned from Chrome's compromised renderer sandbox; (2) Sysmon Event ID 3 (Network Connection) from chrome.exe PIDs showing outbound connections to non-Google IP ranges, particularly on uncommon ports, which may indicate C2 beacon activity established after renderer escape; (3) Web proxy or DNS query logs (Windows DNS debug log at `%SystemRoot%\System32\dns\dns.log` or equivalent) for domains visited within 60 minutes before anomalous process activity — the delivery vector for this zero-day is a malicious webpage, so the triggering domain is a critical artifact; (4) Windows Prefetch files (`%SystemRoot%\Prefetch\`) and Sysmon Event ID 11 (File Create) for any executables written to `%TEMP%`, `%AppData%\Roaming`, or Chrome's user-data directory immediately after browser activity, indicating a dropped payload from renderer code execution.

Step 3: Eradication — Deploy the patched Chrome version to all endpoints via your software distribution platform. Verify the installed version matches or exceeds the patched build published at https://chromereleases.googleblog.com. For environments where Chrome auto-update is disabled, push the update manually and confirm version compliance. Remove or isolate any endpoint where post-exploitation activity is suspected pending forensic review.

NIST Phase: Eradication

Reference: NIST 800-61r3 §3.3 — Containment, Eradication, and Recovery: Eradication

Controls: NIST SI-2 (Flaw Remediation), NIST CM-3 (Configuration Change Control), NIST IR-4 (Incident Handling), CIS 7.3 (Perform Automated Operating System Patch Management), CIS 7.4 (Perform Automated Application Patch Management), CIS 2.2 (Ensure Authorized Software is Currently Supported)

Compensating: For environments without SCCM/Intune/Jamf, use the Chrome standalone installer MSI from https://chromeenterprise.google/intl/en_us/browser/download/ and deploy via Group Policy Software Installation or a simple PowerShell script executed via PsExec across the domain: `Start-Process msiexec.exe -ArgumentList '/i

C:\Installers\chrome_installer.msi /quiet /norestart' -Wait`. After deployment, verify compliance with: `Get-ItemProperty 'HKLM:\SOFTWARE\Google\Update\Clients\{8A69D345-D564-463C-AFF1-A69D9E530F96}' | Select-Object pv`. For endpoints with suspected post-exploitation activity, do not patch in place — image the drive first using dd or FTK Imager Lite (free) before reimaging, to preserve forensic evidence.

Evidence: Before patching suspected-compromised endpoints, preserve: (1) Full volatile memory dump using WinPmem (free, open source) to capture any in-memory shellcode or injected code from Chrome renderer exploitation that will not survive a reboot or patch — memory is the primary artifact for renderer sandbox escapes; (2) Copy of Chrome's user-data directory (`%LocalAppData%\Google\Chrome\User Data\Default\`) including Cache, History, Visited Links, and Web Data SQLite databases, which may contain the URL of the malicious delivery page; (3) Any files written to disk by Chrome processes in the 2-hour window before isolation, using Sysmon Event ID 11 or the Windows Master File Table (MFT) parsed with tools like MFTECmd (free, Eric Zimmermann tools) to identify attacker-dropped artifacts prior to eradication.

Step 4: Recovery — After patching, validate Chrome version compliance across the fleet using endpoint inventory reports. Re-enable any web access controls relaxed during containment. Monitor EDR telemetry for 24-48 hours post-patch for residual anomalous browser subprocess behavior. Confirm no persistence mechanisms were established on endpoints that may have been exposed prior to patching.

NIST Phase: Recovery

Reference: NIST 800-61r3 §3.3 — Containment, Eradication, and Recovery: Recovery

Controls: NIST IR-4 (Incident Handling), NIST SI-7 (Software, Firmware, and Information Integrity), NIST CM-3 (Configuration Change Control), NIST AU-6 (Audit Record Review, Analysis, and Reporting), CIS 1.1 (Establish and Maintain Detailed Enterprise Asset Inventory), CIS 7.2 (Establish and Maintain a Remediation Process)

Compensating: Without EDR, use Autoruns (free, Sysinternals) on endpoints that were potentially exposed to verify no persistence mechanisms — specifically check: `HKCU\Software\Microsoft\Windows\CurrentVersion\Run`, `HKLM\Software\Microsoft\Windows\CurrentVersion\Run`, Scheduled Tasks, and Chrome extension directories (`%LocalAppData%\Google\Chrome\User Data\Default\Extensions\`) for any extensions installed or modified during the exposure window. Run the following osquery to check for scheduled tasks created during the exposure window: `SELECT name, action, path, enabled, last_run_time FROM scheduled_tasks WHERE last_run_time > strftime('%s','now','-7 days');`. For version compliance validation without SCCM, use a PowerShell script via PSRemoting to poll all domain endpoints and export a CSV of Chrome versions for manual review.

Evidence: Before re-enabling web access controls and closing the incident: (1) Capture current state of Chrome extension inventory from `%LocalAppData%\Google\Chrome\User Data\Default\Extensions\` and compare against a known-good baseline — malicious extensions are a common post-exploitation persistence mechanism for browser-based attacks; (2) Export Windows Scheduled Tasks (`schtasks /query /fo CSV /v > tasks_post_recovery.csv`) and compare against a pre-incident baseline to identify any attacker-created persistence; (3) Review `HKCU\Software\Microsoft\Windows\CurrentVersion\Run` and `HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Run` registry hives on all endpoints that accessed high-risk domains during the exposure window, to confirm no post-exploitation persistence survived patching.

Step 5: Post-Incident — Assess whether Chrome auto-update policies are enforced across the environment; gaps here allowed exposure windows for all four 2026 Chrome zero-days. Review browser isolation controls and consider evaluating enhanced browser security configurations per CIS Benchmark for Chrome. Evaluate whether privilege levels on endpoints running Chrome align with least-privilege principles to limit code execution impact. Document time-to-patch metrics for this event against your patching SLA targets.

NIST Phase: Post Incident

Reference: NIST 800-61r3 §4 — Post-Incident Activity: Lessons Learned and Evidence Retention

Controls: NIST IR-4 (Incident Handling), NIST IR-8 (Incident Response Plan), NIST SI-2 (Flaw Remediation), NIST RA-3 (Risk Assessment), 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: Enforce Chrome auto-update via free Google Chrome Enterprise policy templates (ADMX): set `AutoUpdateCheckPeriodMinutes` to enforce update checks and disable `UpdateDefault` policy value '0' (which blocks

updates) across the fleet via Group Policy. Validate the policy is applied with `chrome://policy` in the browser or via: `Get-ItemProperty 'HKLM:\SOFTWARE\Policies\Google\Update' | Select-Object AutoUpdateCheckPeriodMinutes, UpdateDefault`. For least-privilege enforcement without a PAM tool, audit local administrator group membership on all endpoints using: `Get-LocalGroupMember -Group 'Administrators' | Export-Csv admins_audit.csv`. Document time-to-patch for this event (exposure window from Chrome Releases blog post date to fleet-wide patch confirmation) and compare against your defined SLA — the recurrence of four Chrome zero-days in 2026 makes this metric a board-level risk indicator.

Evidence: For the lessons-learned record and to support any regulatory notification assessment: (1) Retain web proxy/DNS logs covering the full exposure window (from the date of active exploitation confirmation by Google to fleet-wide patch completion) — these establish whether any organizational endpoints visited the delivery infrastructure, which is required for breach scope determination; (2) Preserve the Chrome version inventory snapshots taken at the start of containment and at post-patch compliance validation to document the exposure window duration for each endpoint; (3) Retain all Sysmon and EDR logs from the exposure window per your AU-11 (Audit Record Retention) policy — minimum 90 days recommended for a CVSS 8.8 actively exploited vulnerability to support any delayed forensic investigation or regulatory inquiry.

Detection Guidance

No confirmed IOCs (IPs, domains, hashes) were available in the source material at time of writing. Behavioral detection is the primary available approach. Monitor for: (1) Chrome renderer processes (chrome.exe, Google Chrome Helper) spawning unexpected child processes or writing to atypical filesystem locations; (2) network connections initiated by Chrome renderer subprocesses to non-browser destinations; (3) new scheduled tasks, registry run keys, or startup entries created in the timeframe of Chrome activity. In EDR platforms, scope process tree queries to chrome.exe or Google Chrome Helper parent processes with anomalous children. In SIEM, correlate browser process anomalies with web proxy logs to identify candidate malicious pages visited. Consult the Chrome Releases blog and CISA KEV for any IOCs published with the official advisory. Source material for this item was T3 tier (news reporting); treat all behavioral indicators as hypothetical pending authoritative Google or CISA publication.

Indicators of Compromise

Type	Value	Context	Confidence
URL	`https://chromereleases.googleblog.com`	Authoritative Google source for patched version number and official advisory details — consult for CVE ID and build confirmation	HIGH

Framework Mappings

MITRE-ATTACK

- **T1189** — Drive-by Compromise
- **T1203** — Exploitation for Client Execution

NIST-800-53R5

- **SC-7** — Boundary Protection

- **SI-2** — Flaw Remediation
- **SI-3** — Malicious Code Protection
- **SI-4** — System Monitoring
- **IR-5** — Incident Monitoring

CIS-V8

- **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

NIST-CSF-2

- **DE.AE-08** — Incidents are declared when adverse events meet the defined incident criteria

MITRE ATT&CK Mapping

Technique ID	Technique Name	Tactic
T1189	Drive-by Compromise	Initial-Access
T1203	Exploitation for Client Execution	Execution

Sources

Source	URL	Tier
	https://jang.com.pk/en/63068-chrome-security-alert-google-fixes-cri...	T3
Google fixes fourth Chrome zero-day exploited in attacks in 2026	https://www.bleepingcomputer.com/news/security/google-fixes-fourth-...	T3
Google Zero-Day Alert For 3.5 Billion Chrome Users—Attacks ...	https://www.forbes.com/sites/daveywinder/2026/03/15/google-zero-day...	T3
Google Releases Critical Chrome Security Update to Address Zero ...	https://www.infosecurity-magazine.com/news/google-chrome-security-u...	T3
Update Chrome now: Zero-day bug allows code execution via ...	https://www.malwarebytes.com/blog/news/2026/02/update-chrome-now-ze...	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 2026-04-05 13:24 UTC by TJS Security Command Center