INTELLIGENCE BRIEFING
Security Command Center
TLP:CLEAR
2026-06-09 19:31 UTC

Google Chrome Critical Sandbox Escape Vulnerabilities in ANGLE and Network Components (CVE-2026-10881, CVE-2026-10882)

CVE VULNERABILITY | **CRITICAL** | CVSS 9.6

SCC Item ID	SCC-CVE-2026-0279
Type	CVE Vulnerability
CVE ID	CVE-2026-10881, CVE-2026-10882
Severity	CRITICAL
CVSS Base Score	9.6
EPSS Score	0.0008 (24th percentile)
Affected Products	Google Chrome (ANGLE graphics abstraction layer, Network component), specific version range unverified
Published	2026-06-08
Discovery Source	Gemini

Executive Summary

Google has patched two critical vulnerabilities in Chrome, CVE-2026-10881 (out-of-bounds memory access in the ANGLE graphics layer) and CVE-2026-10882 (use-after-free in the Network component), both rated CVSS 9.6. A remote attacker who delivers malicious web content can exploit either flaw to escape Chrome's browser sandbox, gaining code execution on the underlying host. Any organization or individual running an unpatched version of Chrome faces direct risk; the broad install base of Chrome across enterprise endpoints amplifies organizational exposure.

Technical Analysis

CVE-2026-10881 is an out-of-bounds read/write (CWE-787, CWE-125) in Chrome's ANGLE (Almost Native Graphics Layer Engine) graphics abstraction layer. CVE-2026-10882 is a use-after-free (CWE-416) in Chrome's Network component. Both vulnerabilities carry a CVSS base score of 9.6 and enable remote code execution via malicious web content, bypassing Chrome's sandbox isolation. MITRE ATT&CK techniques T1203 (Exploitation for Client Execution) and T1211 (Exploitation for Defense Evasion) are relevant. No CISA KEV listing is confirmed at time of writing; EPSS score is 0.0008 (23rd percentile), suggesting low observed exploitation activity so far. Specific affected version range must be verified against the official Google Chrome Releases blog (https://chromereleases.googleblog.com/). No confirmed IOCs or threat actor attribution are available. Patch to

the latest stable Chrome release immediately.

Action Checklist

1. Step 1: Identification & Containment, Identify all enterprise endpoints running Google Chrome. Use asset inventory (CIS 1.1, Establish and Maintain Detailed Enterprise Asset Inventory) to enumerate affected hosts. Block Chrome from launching on unpatched systems via endpoint policy if immediate patching is not feasible. Verify the current patched version against the Google Chrome Releases blog (https://chromereleases.googleblog.com/), specific patched version number must be validated there before deployment.
2. Step 2: Detection, Query endpoint telemetry and EDR for Chrome processes spawning unexpected child processes or shell processes (e.g., cmd.exe, powershell.exe, bash as child of chrome.exe). Review web proxy and DNS logs for connections to newly registered or low-reputation domains preceding anomalous Chrome behavior. Enable audit logging on endpoints per NIST AU-2 (Event Logging) and CIS 8.2 (Collect Audit Logs). No confirmed IOC hashes or C2 infrastructure are available at this time; behavioral detection is the primary signal.
3. Step 3: Eradication, Update all Chrome installations to the patched stable release confirmed via the Google Chrome Releases blog. Enforce automated application patch management per CIS 7.4 (Perform Automated Application Patch Management). Where auto-update is disabled by policy, push the update via endpoint management tooling. Verify update completion across all inventoried assets.
4. Step 4: Recovery, After patching, confirm Chrome version on all endpoints matches the patched release. Monitor endpoint telemetry for 72 hours post-patch for any residual anomalous process spawning that could indicate pre-patch compromise. Review NIST IR-5 (Incident Monitoring); document any hosts where exploitation cannot be ruled out and treat them as potentially compromised pending forensic review.
5. Step 5: Post-Incident, Review application patch SLA compliance for browser software. If auto-update was disabled on any endpoints, document the gap and enforce CIS 7.4 (Perform Automated Application Patch Management) and CIS 2.2 (Ensure Authorized Software is Currently Supported). Evaluate whether browser isolation (e.g., remote browser isolation) is warranted for high-risk user populations such as finance and executive staff.

IR / Forensic Enrichment

Triage Priority	IMMEDIATE
Escalation Criteria	Escalate to senior IR leadership, legal, and privacy counsel immediately if any endpoint shows chrome.exe spawning a shell process (cmd.exe, powershell.exe) or exhibits outbound C2 connections following Chrome activity, as this indicates successful sandbox escape with potential host-level compromise and may trigger breach notification obligations under GDPR, CCPA, or HIPAA if the affected host processes PII or PHI.

Recovery Notes	After patching, verify the Chrome version on every inventoried endpoint against the confirmed patched release on the Google Chrome Releases blog — do not rely on self-reported auto-update status. Any host where exploitation cannot be ruled out (i.e., unpatched Chrome was active and anomalous child processes or network activity were observed) should be reimaged rather than cleaned in place, given that a successful sandbox escape from either CVE-2026-10881 or CVE-2026-10882 grants the attacker host-level code execution with full persistence capability. Continue behavioral monitoring for chrome.exe process lineage anomalies for a minimum of 72 hours post-patch, as attacker tooling installed pre-patch may persist independently of Chrome after the vulnerability is remediated.
Forensic Artifacts	Sysmon Event ID 1 (Process Create) logs: ParentImage=chrome.exe with Image=cmd.exe, powershell.exe, or any non-Chrome binary — direct behavioral signature of sandbox escape from CVE-2026-10881 or CVE-2026-10882 resulting in host-level code execution | Chrome crash reports and memory dumps at '%LOCALAPPDATA%\Google\Chrome\User Data\Crashpad\reports\' — ANGLE out-of-bounds memory access (CVE-2026-10881) and Network component use-after-free (CVE-2026-10882) may generate crash telemetry or minidumps that preserve exploit shellcode or heap spray artifacts in memory | Web proxy and DNS logs filtered for the unpatched Chrome user-agent string accessing low-reputation or newly registered domains in the session window immediately preceding anomalous process activity — identifies the malicious web content delivery vector for these browser-based exploits | Chrome SQLite databases at '%LOCALAPPDATA%\Google\Chrome\User Data\Default\History' (visited URLs) and '%LOCALAPPDATA%\Google\Chrome\User Data\Default\Network Action Predictor' — reconstruct the browsing session that delivered the malicious content exploiting CVE-2026-10881 or CVE-2026-10882 | Registry key HKLM\SOFTWARE\Google\Update\Clients\{8A69D345-D564-463c-AFF1-A69D9E530F96} (value: 'pv') and HKLM\SOFTWARE\Policies\Google\Chrome\UpdateDefault — confirms vulnerable Chrome version present at time of incident and whether auto-update suppression by policy extended the exposure window

Per-Action IR Details

Step 1: Containment — Identify all enterprise endpoints running Google Chrome. Use asset inventory (CIS 1.1 — Establish and Maintain Detailed Enterprise Asset Inventory) to enumerate affected hosts. Block Chrome from launching on unpatched systems via endpoint policy if immediate patching is not feasible. Verify the current patched version against the Google Chrome Releases blog (https://chromereleases.googleblog.com/) — specific patched version number could not be confirmed from available sources and must be validated there before deployment.

NIST Phase: Containment

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

Controls: CIS 1.1 (IG1/IG2/IG3) — Establish and Maintain Detailed Enterprise Asset Inventory, NIST IR-4 (Incident Handling), NIST CM-8 (System Component Inventory)

Compensating: Run osquery with 'SELECT name, version, install_location FROM programs WHERE name LIKE "%Chrome%";' across all endpoints to enumerate installed Chrome versions without SIEM. Where osquery is unavailable, use a PowerShell one-liner: 'Get-ItemProperty HKLM:\Software\Google\Update\Clients* | Select-Object name,pv' to pull installed Chrome version from the registry. To block Chrome launch on unpatched Windows hosts without EDR, deploy a Software Restriction Policy or AppLocker rule targeting chrome.exe by path using Group Policy — achievable by a 2-person team in under one hour across a flat AD environment.

Evidence: Before blocking or touching affected hosts, capture: (1) the Chrome version string from registry key HKLM\SOFTWARE\Google\Update\Clients\{8A69D345-D564-463c-AFF1-A69D9E530F96} (value: 'pv') to confirm vulnerable version; (2) Windows Security Event Log Event ID 4688 (Process Creation) showing chrome.exe and any child processes it spawned at the time of suspected exploitation — ANGLE and Network component exploits resulting

in sandbox escape would produce chrome.exe spawning unexpected children such as cmd.exe, powershell.exe, or a low-privilege shell; (3) a full process tree snapshot via Sysmon Event ID 1 (Process Create) if Sysmon is deployed, capturing ParentImage and CommandLine fields for all chrome.exe descendants.

Step 2: Detection — Query endpoint telemetry and EDR for Chrome processes spawning unexpected child processes or shell processes (e.g., cmd.exe, powershell.exe, bash as child of chrome.exe). Review web proxy and DNS logs for connections to newly registered or low-reputation domains preceding anomalous Chrome behavior. Enable audit logging on endpoints per NIST AU-2 (Event Logging) and CIS 8.2 (Collect Audit Logs). No confirmed IOC hashes or C2 infrastructure are available at this time — behavioral detection is the primary signal.

NIST Phase: Detection Analysis

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

Controls: NIST AU-2 (Event Logging), NIST AU-6 (Audit Record Review, Analysis, And Reporting), NIST IR-5 (Incident Monitoring), CIS 8.2 (IG1/IG2/IG3) — Collect Audit Logs

Compensating: Deploy Sysmon with SwiftOnSecurity's base config (https://github.com/SwiftOnSecurity/sysmon-config — search-retrieved, recommend human validation) and hunt for Sysmon Event ID 1 entries where ParentImage ends in 'chrome.exe' and Image ends in 'cmd.exe', 'powershell.exe', 'wscript.exe', or 'mshta.exe' — this is the direct behavioral signature of a Chrome sandbox escape via either CVE-2026-10881 or CVE-2026-10882. Use the Sigma rule category 'proc_creation_win_susp_chrome_child_process' as a detection template adapted for your log source. For DNS, parse DNS debug logs or run Wireshark captures on endpoints of concern, filtering for DNS queries from chrome.exe's PID to identify C2 beaconing that would follow successful exploitation via the Network component (CVE-2026-10882).

Evidence: Preserve before analysis: (1) Sysmon Event ID 1 (Process Create) and Event ID 10 (Process Access) logs from all endpoints where chrome.exe was active in the detection window — CVE-2026-10882 use-after-free in the Network component may produce anomalous IPC or socket handle access detectable via Event ID 10; (2) Chrome's internal log at '%LOCALAPPDATA%\Google\Chrome\User Data\chrome_debug.log' if verbose logging was enabled — may capture crash or memory fault telemetry tied to ANGLE (CVE-2026-10881) or Network component (CVE-2026-10882) exploitation; (3) web proxy logs filtered for the user agent string of the unpatched Chrome version accessing domains with low Alexa/Umbrella rank or newly registered (<30 days) domains in the 30-minute window before any anomalous process spawn, as the exploit is delivered via malicious web content.

Step 3: Eradication — Update all Chrome installations to the patched stable release confirmed via the Google Chrome Releases blog. Enforce automated application patch management per CIS 7.4 (Perform Automated Application Patch Management). Where auto-update is disabled by policy, push the update via endpoint management tooling. Verify update completion across all inventoried assets.

NIST Phase: Eradication

Reference: NIST 800-61r3 §3.4 — Eradication

Controls: NIST SI-2 (Flaw Remediation), CIS 7.4 (IG1/IG2/IG3) — Perform Automated Application Patch Management, CIS 2.2 (IG1/IG2/IG3) — Ensure Authorized Software is Currently Supported, NIST CM-6 (Configuration Settings)

Compensating: For teams without enterprise patch management, use Google's enterprise MSI installer distributed via a shared network path and a PowerShell deployment script executed via PsExec or a scheduled task pushed through Group Policy: 'Start-Process msiexec.exe -ArgumentList "/i \\server\share\ChromeEnterprise.msi /qn" -Wait'. Post-deployment, re-run the osquery or PowerShell version check from Step 1 to verify the 'pv' registry value reflects the patched version confirmed on the Google Chrome Releases blog. Document every host where the update could not be verified — those hosts must be treated as potentially compromised and triaged per Step 4.

Evidence: Before pushing the patch, snapshot: (1) the current chrome.exe binary hash (SHA-256 via 'Get-FileHash') from '%ProgramFiles%\Google\Chrome\Application\chrome.exe' on each affected host to establish a pre-patch baseline and confirm the vulnerable binary version for incident records; (2) Chrome's component updater log at '%LOCALAPPDATA%\Google\Chrome\User Data\BrowserMetrics' and '%LOCALAPPDATA%\Google\Chrome\User Data\CrashpadMetrics' for any crash telemetry that may indicate prior exploitation attempts against ANGLE or the

Network component; (3) a registry export of HKCU\Software\Google\Chrome and HKLM\SOFTWARE\Policies\Google\Chrome to document whether auto-update was disabled by policy, which is relevant to both root cause analysis and CIS 7.4 compliance documentation.

Step 4: Recovery — After patching, confirm Chrome version on all endpoints matches the patched release. Monitor endpoint telemetry for 72 hours post-patch for any residual anomalous process spawning that could indicate pre-patch compromise. Review NIST IR-5 (Incident Monitoring) — document any hosts where exploitation cannot be ruled out and treat them as potentially compromised pending forensic review.

NIST Phase: Recovery

Reference: NIST 800-61r3 §3.5 — Recovery

Controls: NIST IR-5 (Incident Monitoring), NIST IR-4 (Incident Handling), NIST AU-6 (Audit Record Review, Analysis, And Reporting), CIS 7.1 (IG1/IG2/IG3) — Establish and Maintain a Vulnerability Management Process

Compensating: Without EDR, configure Sysmon to log to Windows Event Forwarding (WEF) and collect centrally. During the 72-hour monitoring window, run a scheduled PowerShell script every 30 minutes on high-risk hosts (those where Chrome was unpatched longest or accessed high-risk web content) that queries Event ID 4688 for any process whose ParentProcessName is chrome.exe and whose NewProcessName is not a known Chrome helper binary — alert via email or a Slack webhook on any hit. For hosts flagged as potentially compromised, capture a full memory image using WinPmem (free, open source) before reimaging, to preserve forensic evidence of any post-exploitation activity from CVE-2026-10881 or CVE-2026-10882.

Evidence: Preserve before clearing any host: (1) full memory image using WinPmem on any host where chrome.exe spawned anomalous children — in-memory artifacts of ANGLE out-of-bounds exploitation (CVE-2026-10881) or Network component use-after-free (CVE-2026-10882) are most reliably captured from volatile memory before reboot or patch; (2) Windows Security Event Log (Event IDs 4624, 4625, 4648, 4688) covering the 48-hour window preceding patch deployment to identify any lateral movement or credential access following a sandbox escape; (3) Chrome's '%LOCALAPPDATA%\Google\Chrome\User Data\Default\History' and 'Network Action Predictor' SQLite databases to reconstruct which URLs were visited immediately before any anomalous behavior, tying the exploit delivery vector to a specific malicious domain.

Step 5: Post-Incident — Review application patch SLA compliance for browser software across the enterprise. If auto-update was disabled on any endpoints, document the gap and enforce CIS 7.4 (Perform Automated Application Patch Management) and CIS 2.2 (Ensure Authorized Software is Currently Supported). Evaluate whether browser isolation (e.g., remote browser isolation) is warranted for high-risk user populations such as finance and executive staff, given the repeated pattern of critical browser sandbox escapes.

NIST Phase: Post Incident

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

Controls: CIS 7.4 (IG1/IG2/IG3) — Perform Automated Application Patch Management, CIS 2.2 (IG1/IG2/IG3) — Ensure Authorized Software is Currently Supported, CIS 7.2 (IG1/IG2/IG3) — Establish and Maintain a Remediation Process, NIST IR-8 (Incident Response Plan), NIST CM-7 (Least Functionality)

Compensating: Document the lessons-learned findings in a structured post-incident report referencing the specific Chrome auto-update policy registry key (HKLM\SOFTWARE\Policies\Google\Chrome\UpdateDefault) — a value of '0' means updates were disabled by policy, which must be remediated. For browser isolation on a zero-budget: configure Chrome's built-in Site Isolation feature ('--site-per-process' is default in modern Chrome but verify via chrome://policy) and enforce Chrome's enterprise policy 'RendererCodeIntegrityEnabled' = true via Group Policy to raise the cost of future sandbox escapes targeting the renderer. Add a recurring osquery scheduled query to the weekly asset review to alert on any Chrome installation where the version lags more than 14 days behind the current stable release.

Evidence: Collect and retain for post-incident review: (1) the registry export of Chrome update policy keys from all endpoints where auto-update was found disabled, as evidence of the configuration gap that extended exposure to CVE-2026-10881 and CVE-2026-10882; (2) web proxy logs aggregated across the incident window to identify whether any users accessed the same suspected malicious delivery domain — relevant if a targeted campaign (spear-phishing link or watering hole) was the delivery mechanism for either CVE; (3) the full timeline of Chrome version deployment across the asset inventory (from patch release date on the Google Chrome Releases blog to confirmed deployment per

endpoint) to measure SLA compliance and support process improvement.

Detection Guidance

Primary behavioral indicator: Chrome renderer or GPU process (chrome.exe --type=renderer, --type=gpu-process on Windows) spawning unexpected system processes. Alert on chrome.exe as parent of cmd.exe, powershell.exe, wscript.exe, mshta.exe, or any process not in a known-good child process baseline. On Linux/macOS, alert on Chrome spawning bash, sh, or curl as children. Secondary indicator: unusual outbound network connections originating from Chrome processes to non-browser-telemetry destinations shortly after loading an unknown or low-reputation URL. Log sources: EDR process telemetry, Windows Security Event Log (Event ID 4688 with process creation auditing enabled, supports NIST AU-2 and AU-3), endpoint AV/EDR behavioral alerts. No confirmed CVE-specific IOC hashes, domains, or IPs are available. Detection relies on behavioral heuristics until exploitation is observed in the wild. EPSS score of 0.0008 indicates low current exploitation activity. Monitor CISA KEV (https://www.cisa.gov/known-exploited-vulnerabilities) for addition of CVE-2026-10881 or CVE-2026-10882; if either appears on KEV, escalate to 24-hour patch SLA and increase detection sensitivity.

Framework Mappings

MITRE-ATTACK

- **T1211** — Exploitation for Defense Evasion
- **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
- **SI-16** — Memory Protection

CIS-V8

- **16.10** — Apply Secure Design Principles in Application Architectures
- **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
- **A.5.34** — Privacy and protection of personal information
- **A.5.23** — Information security for use of cloud services

MITRE ATT&CK Mapping

Technique ID	Technique Name	Tactic
T1211	Exploitation for Defense Evasion	Defense-Evasion
T1203	Exploitation for Client Execution	Execution

Sources

Source	URL	Tier
CVE-2026-10881 Detail - NVD	https://nvd.nist.gov/vuln/detail/CVE-2026-10881	T1
CVE-2026-10881 - Vulnerability Details - OpenCVE	https://app.opencve.io/cve/CVE-2026-10881	T3
CVE-2026-10882 - Red Hat Customer Portal	https://access.redhat.com/security/cve/cve-2026-10882	T3
CVE-2026-0881: Firefox Privilege Escalation Vulnerability	https://www.sentinelone.com/vulnerability-database/cve-2026-0881/	T3
Microsoft Edge Multiple Vulnerabilities	https://www.hkcert.org/security-bulletin/microsoft-edge-multiple-vu...	T3
NVD	https://nvd.nist.gov/vuln/detail/CVE-2026-10881, CVE-2026-10882	T1
Google Security Advisory	https://chromereleases.googleblog.com/	T1

Generated 2026-06-09 19:31 UTC by TJS Security Command Center