INTELLIGENCE BRIEFING
Security Command Center
TLP:CLEAR
2026-03-29 18:41 UTC

Simultaneous Exploitation Wave: Chrome Zero-Days, AWS Supply Chain Attack, and Dual Router Botnets (Week of March 16, 2026)

THREAT CAMPAIGN | **CRITICAL** | CVSS 9.5

SCC Item ID	SCC-CAM-2026-0039
Type	Threat Campaign
Severity	CRITICAL
CVSS Base Score	9.5
Affected Products	Google Chrome (multiple versions, specific CVEs not confirmed in source data); AWS environments (via CI/CD supply chain vector); Routers (multiple unspecified vendors)
Published	2026-03-17

Executive Summary

The week of March 16, 2026 brought reports of simultaneous exploitation across three enterprise attack surfaces: two Google Chrome zero-days with active exploitation claims, a reported supply chain attack affecting AWS environments via poisoned CI/CD pipelines, and two reported concurrent router botnet campaigns targeting network edge devices. **Note: This assessment is based on provisional source reporting and has not been independently verified by CISA, Google, AWS, or affected router vendors. Treat as a high-confidence alert on the possibility of these simultaneous attacks, but confirm details with authoritative sources before finalizing incident response scope.** Organizations running unpatched Chrome, cloud-connected CI/CD infrastructure, or unmanaged edge routers should prioritize verification and implement the recommended immediate actions. If any of these attack chains is confirmed in your environment, escalate to incident response immediately.

Technical Analysis

Four distinct attack chains are reported for this period. (1) Chrome Zero-Days: Two reported in-the-wild zero-days affecting Google Chrome. Vulnerability classes described as consistent with CWE-416 (Use-After-Free) and CWE-787 (Out-of-Bounds Write), historically exploited in Chrome's renderer process and V8 JavaScript engine to achieve arbitrary code execution from a malicious web page (MITRE T1189, T1203). Specific CVE identifiers were not confirmed in available source data and are not assigned here. CIS Advisory 2026-014 flags arbitrary code execution risk. Affected versions not precisely bounded in available data; treat all Chrome versions prior to the March 2026 patch cycle as potentially exposed until vendor confirmation. (2) AWS

Supply Chain Attack: Reported CI/CD pipeline compromise leading to claimed full AWS environment takeover within 72 hours. Reported attack vector described as consistent with CWE-94 (Code Injection) into pipeline configuration and CWE-522 (Insufficiently Protected Credentials) for AWS credential harvesting (MITRE T1195.001, T1195.002, T1072, T1552.001, T1078, T1087.004). If confirmed, successful execution would grant persistent cloud access, lateral movement, and data exfiltration capability. (3) Dual Router Botnets: Two simultaneous botnet campaigns reported to target routers from unspecified vendors. Reported exploitation pattern described as consistent with CWE-284 (Improper Access Control) on edge device management interfaces (MITRE T1583.005, T1584.005, T1498, T1199). If active, compromised routers may be staged for DDoS, traffic interception, or as pivot infrastructure. Specific CVEs and vendor identifiers not confirmed in source data. **Source Confidence Assessment:** All CVSS, EPSS, and KEV fields in source data are unverified or unpopulated. Severity rating (critical) is assigned based on reported impact scope, not on independently verified threat data. Confidence on severity rating is medium pending corroboration from CISA, Google, AWS, and affected router vendors. Source URLs require human validation per session URL policy.

Action Checklist

1. Step 1, Immediate (Chrome): Force-update Google Chrome to the latest stable channel release across all managed endpoints. Do not wait for scheduled patch cycles. Verify update compliance via endpoint management tooling within 24 hours.
2. Step 2, Immediate (CI/CD): Audit all CI/CD pipeline configurations for unauthorized changes, injected steps, or unfamiliar third-party actions. Rotate all AWS credentials and tokens that were accessible to CI/CD pipeline environments. Revoke and reissue; do not simply regenerate alongside existing credentials.
3. Step 3, Immediate (Routers): Inventory all edge routers. Identify devices running end-of-life firmware or without recent vendor security patches. Disable remote management interfaces not operationally required. Segment routers from internal management networks where not already done.
4. Step 4, Detection: Search endpoint logs for Chrome process anomalies (unexpected child processes, renderer crashes at volume, unusual network connections from chrome.exe or chrome). Review CI/CD pipeline execution logs for unexpected steps, external calls, or credential access events. Check AWS CloudTrail for unusual IAM activity, especially new role assumptions, policy changes, or API calls from unfamiliar source IPs. Monitor router syslog and SNMP for configuration changes, new admin accounts, or anomalous outbound traffic.
5. Step 5, Assessment: Determine whether any Chrome instances accessed untrusted or high-risk URLs during the exploitation window. Assess blast radius of CI/CD pipeline exposure, map which AWS accounts, roles, and resources were reachable from the pipeline. Confirm router firmware patch levels against vendor advisories as they become available.
6. Step 6, Communication: Notify security leadership and cloud infrastructure owners of the CI/CD/AWS risk. If pipeline compromise is confirmed or suspected, treat as an active incident and engage IR process. Brief network operations on router botnet activity and expected remediation timeline.
7. Step 7, Long-term: Enforce least-privilege scoping for all CI/CD pipeline IAM roles. Implement OIDC-based short-lived credential issuance for cloud pipelines rather than static secrets. Establish a recurring edge device firmware review cadence. Evaluate browser isolation or enterprise browser controls for high-risk user populations.

IR / Forensic Enrichment

Triage Priority	IMMEDIATE
Escalation Criteria	Escalate to external IR firm or law enforcement if CloudTrail shows confirmed data exfiltration (API calls to S3 GetObject on sensitive buckets from unattributed source IP, or CloudTrail events deleted), if router botnet activity persists after firmware patching and disabling remote management, or if CI/CD pipeline was used to deploy malicious code to production systems.
Recovery Notes	Post-containment: conduct full forensic analysis of Chrome crash dumps and renderer logs to confirm exploitation vector and payload (work with Google CERT if available). For CI/CD: review all commits pushed by pipeline during compromise window for backdoors or modified deployment configs; consider rebuilding affected deployments from pre-compromise tags. For routers: after patching and configuration restoration, establish 24/7 network monitoring of edge device traffic for 30 days with alerting on anomalous outbound connections, BGP route hijacks, or DNS poisoning attempts; consider segmenting router management onto dedicated OOB network with hardware firewall rules.
Forensic Artifacts	Windows Event Log Security (Event 4688, 4689) and Sysmon Event Log (1-3, 7, 22) | Chrome User Data directory: History SQLite, Cache, Downloads, Extensions | AWS CloudTrail JSON logs: AssumeRole, API calls, credential usage, IP source geolocation | CI/CD platform execution logs and pipeline config version history (git commits with diffs) | Router running-config TFTP backup, syslog buffer (90-day retention if available), SNMP trap logs, serial console logs

Per-Action IR Details

Step 1, Immediate (Chrome): Force-update Google Chrome to the latest stable channel release across all managed endpoints. Do not wait for scheduled patch cycles. Verify update compliance via endpoint management tooling within 24 hours.

NIST Phase: Containment

Reference: NIST 800-61r3 §3.2.3 (Containment)

Controls: NIST 800-53 SI-2 (Flaw Remediation), CIS Controls v8 3.11 (Address Unauthorized Software), CIS Controls v8 2.4 (Address Unauthorized Digital Certificates)

Compensating: Organizations without MDM: deploy batch update via Group Policy (Windows: `gpupdate /force` + scheduled task triggering `google-chrome.exe --update`); for macOS: use Managed Client Preferences or script via LaunchDaemon; for Linux: push via package manager automation (apt-get, yum). Verify via registry check (Windows: `reg query HKLM\Software\Google\Chrome\Binaries` /v pv) or `google-chrome --version` command remotely via SSH or similar.

Evidence: Before updating: capture Chrome version registry/config (`HKEY_LOCAL_MACHINE\SOFTWARE\Google\Chrome\Binaries`, version.txt), process memory dumps of running chrome.exe instances (use procdump.exe or gcore), browser history/downloads from `%LOCALAPPDATA%\Google\Chrome\User Data\Default\History` and `Downloads` database, and DNS/HTTP proxy logs of Chrome network activity for 48 hours pre-patch to detect exploitation attempts.

Step 2, Immediate (CI/CD): Audit all CI/CD pipeline configurations for unauthorized changes, injected steps, or unfamiliar third-party actions. Rotate all AWS credentials and tokens that were accessible to CI/CD pipeline environments. Revoke and reissue; do not simply regenerate alongside existing credentials.

NIST Phase: Containment

Reference: NIST 800-61r3 §3.2.2 (Short-term Containment)

Controls: NIST 800-53 AC-2 (Account Management), NIST 800-53 IA-2 (Authentication), NIST 800-53 AC-3 (Access Control), CIS Controls v8 1.4 (Maintain Inventory of Service Accounts), CIS Controls v8 5.2 (Verify Accuracy of Access Control Lists)

Compensating: Export pipeline YAML/JSON configs to version control history review: `git log --all --full-history -- [pipeline-config-path]` or equivalent in GitLab/GitHub Actions. Manually diff against baseline or known-good snapshot. For AWS credential rotation without DevOps platform: use AWS CLI to revoke via `aws iam delete-access-key --access-key-id [KEY_ID]` and issue new keys directly (capture old keys for forensic analysis before deletion). Document all CI/CD service accounts in a spreadsheet and cross-check against AWS IAM user list (`aws iam list-users`).

Evidence: Before rotation: export full CI/CD pipeline execution history (all logs, job outputs, environment variable snapshots if accessible), AWS CloudTrail events for all CI/CD-assumed roles (IAM role assumption, API calls, credential usage) for minimum 30 days prior, git commit/push logs and patch diffs for pipeline config repos, and record of all AWS access keys and their creation/last-used dates (`aws iam get-access-key-last-used`).

Step 3, Immediate (Routers): Inventory all edge routers. Identify devices running end-of-life firmware or without recent vendor security patches. Disable remote management interfaces not operationally required. Segment routers from internal management networks where not already done.

NIST Phase: Containment

Reference: NIST 800-61r3 §3.2.1 (Containment Strategy)

Controls: NIST 800-53 IA-4 (Identifier Management), NIST 800-53 AC-3 (Access Control), NIST 800-53 CM-2 (Baseline Configuration), CIS Controls v8 1.1 (Inventory and Control of Enterprise Assets), CIS Controls v8 5.1 (Establish and Maintain an Access Control Baseline)

Compensating: Manual inventory: SNMP walk (`snmpwalk -v 2c -c [community-string] [router-ip] .1.3.6.1.2.1.1`) to enumerate devices and firmware versions, or telnet/SSH direct query (`show version` for Cisco, `display version` for Huawei, equivalent per vendor). Check firmware EOL status against vendor support matrix (often in CSV on vendor site). Disable telnet/SSH/HTTP management by accessing serial console or out-of-band management interface directly; do not route disablement commands through network if device is suspected compromised. Use vendor-neutral tools like netcat or nmap to audit open management ports: `nmap -p 22,23,80,443,9000 [router-subnet]`.

Evidence: Before disabling: capture full running config via TFTP/SCP (`copy running-config tftp://[server]/backup`), syslog dump for 90 days if available, SNMP trap history, netflow/sflow records showing all outbound traffic from router, serial console logs, any authentication logs (tacacs/radius queries if available), and baseline network diagram showing current router-to-management-network connectivity.

Step 4, Detection: Search endpoint logs for Chrome process anomalies (unexpected child processes, renderer crashes at volume, unusual network connections from chrome.exe or chrome). Review CI/CD pipeline execution logs for unexpected steps, external calls, or credential access events. Check AWS CloudTrail for unusual IAM activity, especially new role assumptions, policy changes, or API calls from unfamiliar source IPs. Monitor router syslog and SNMP for configuration changes, new admin accounts, or anomalous outbound traffic.

NIST Phase: Detection Analysis

Reference: NIST 800-61r3 §3.1.3 (Analysis)

Controls: NIST 800-53 SI-4 (Information System Monitoring), NIST 800-53 AU-2 (Audit Events), NIST 800-53 CA-7 (Continuous Monitoring), CIS Controls v8 8.5 (Implement SIEM or Log Aggregation), CIS Controls v8 8.8 (Collect Detailed Audit Logs)

Compensating: Chrome: parse Windows Event Log 4688 (process creation) and 4689 (termination) for chrome.exe parent/child relationships via `wevtutil qe Security "/q:*[EventData[Data[@Name='ParentImage']='C:\Program Files\Google\Chrome\Application\chrome.exe']]" /f:text`; cross-ref network connections via `netstat -anob` snapshots or Sysmon Event 3 logs. CI/CD: grep pipeline logs for keywords (`password`, `secret`, `token`, `curl`, `wget`, `aws configure`, `assume-role`) and compare job step YAML against version control baseline. AWS CloudTrail: query via AWS CLI (`aws cloudtrail lookup-events --lookup-attributes AttributeKey=ResourceType,AttributeValue=AWS::IAM::Role --max-results 50`) or download raw JSON and parse with

jq for `AssumeRole` events from non-standard source IPs. Router: SSH to device and run `show log` or `display logbuffer` (vendor-specific); grep for `configuration changed`, `user added`, `login failed`, and parse syslog to syslog server if deployed.

Evidence: Windows Event Logs (Security: 4688, 4689, 4648; System: 1000-1002 for crashes), Sysmon Event Log 1-3 (process creation, network connection), Chrome user data directory (Default/History, Default/Cache), AWS CloudTrail JSON logs (all events from compromise window +/- 7 days), CI/CD pipeline execution artifacts (logs, artifacts, environment snapshots), router running config snapshot, router syslog buffer (last 1000 messages), network capture on router management interface (PCAP, 5-minute sample).

Step 5, Assessment: Determine whether any Chrome instances accessed untrusted or high-risk URLs during the exploitation window. Assess blast radius of CI/CD pipeline exposure, map which AWS accounts, roles, and resources were reachable from the pipeline. Confirm router firmware patch levels against vendor advisories as they become available.

NIST Phase: Detection Analysis

Reference: NIST 800-61r3 §3.1.4 (Incident Categorization and Prioritization)

Controls: NIST 800-53 RA-5 (Vulnerability Scanning), NIST 800-53 RA-3 (Risk Assessment), CIS Controls v8 15.1 (Identify and Inventory Unmanaged Devices)

Compensating: Chrome URL assessment: export browser history from `%LOCALAPPDATA%\Google\Chrome\User Data\Default\History` SQLite database using free tool sqlite3 CLI (`sqlite3 History "SELECT url, last_visit_time FROM urls WHERE last_visit_time > [unix_timestamp]"`) and cross-ref against URL reputation feeds (VirusTotal hash-based lookup via free tier, or internal proxy logs if available). CI/CD blast radius: manually trace AWS IAM role trust policy (`aws iam get-role-policy --role-name [CI-CD-role] --policy-name trust` for trust relationship, then `aws iam list-role-policies --role-name [CI-CD-role]` for attached permissions). Create dependency map: for each permission, query `aws ec2 describe-instances`, `aws rds describe-db-instances`, `aws s3 ls` scoped to role. Router firmware: download latest security advisories from vendor website (Cisco, Juniper, etc.) and manually run `show version` on each device, compare against published EOL date and patch levels in advisory matrix.

Evidence: Chrome History SQLite database (with timestamps), browser cache directory for downloaded files, DNS resolution logs (internal DNS or upstream resolver logs), proxy logs showing http/https destinations and response codes, AWS IAM role trust policies and permission policies (export as JSON), AWS resource inventory (EC2, RDS, S3 bucket list with access settings), router show version output and configuration snapshots, vendor security advisory PDFs with patch version matrix.

Step 6, Communication: Notify security leadership and cloud infrastructure owners of the CI/CD/AWS risk. If pipeline compromise is confirmed or suspected, treat as an active incident and engage IR process. Brief network operations on router botnet activity and expected remediation timeline.

NIST Phase: Containment

Reference: NIST 800-61r3 §2.3.1 (Notification and Escalation)

Controls: NIST 800-53 IR-6 (Incident Reporting), CIS Controls v8 17.1 (Designate Leadership Accountable for Incident Response)

Compensating: No tooling dependency — this is a procedural/human step. Create incident ticket in any tracking system (Jira, ServiceNow, email thread with audit trail). Prepare one-page summary: threat name, affected systems (Chrome versions, CI/CD platform, router models), initial evidence (suspicious URLs, unauthorized CloudTrail events, router config changes), containment actions taken, and next steps (forensics, eradication timeline, communication). Route escalation: security leadership → cloud ops lead (for AWS); network ops lead (for routers); development lead (for CI/CD). Document attendees, decisions, and action owners in incident log.

Evidence: Incident ticket with initial summary, attendee list, decisions recorded, assigned owners and due dates, baseline forensic evidence summary (counts of suspicious events, resource names, timelines), initial assessment of severity (did compromise occur or only exploit attempt?), estimated time to remediation.

Step 7, Long-term: Enforce least-privilege scoping for all CI/CD pipeline IAM roles. Implement OIDC-based short-lived credential issuance for cloud pipelines rather than static secrets. Establish a recurring edge

device firmware review cadence. Evaluate browser isolation or enterprise browser controls for high-risk user populations.

NIST Phase: Recovery

Reference: NIST 800-61r3 §3.4 (Post-Incident Activities)

Controls: NIST 800-53 AC-2 (Account Management), NIST 800-53 IA-2 (Authentication), NIST 800-53 CM-2 (Baseline Configuration), NIST 800-53 SI-2 (Flaw Remediation), CIS Controls v8 4.1 (Establish and Maintain a Software Asset Inventory), CIS Controls v8 5.4 (Restrict Administrator Privileges to Dedicated Administrator Accounts), CIS Controls v8 6.1 (Establish and Maintain an Application Allowlist)

Compensating: OIDC credential issuance: if OIDC provider unavailable, implement time-limited IAM credentials via STS AssumeRole with max session duration capped at job runtime (e.g., 900 seconds) and attach role policy that explicitly denies iam:*, sts:*, ec2:ModifyInstanceAttribute. Use AWS Secrets Manager (free tier available) instead of hardcoded secrets in pipeline configs. Router firmware review: add calendar reminder (quarterly minimum) to pull latest vendor firmware matrix and compare against inventory; document in change control ticket before each patch cycle. Browser isolation: evaluate Citrix HDX or Menlo Security isolation (enterprise), or free alternatives like Firefox Multi-Account Containers extension + mandatory HTTPS-only mode + disable downloads outside managed location.

Evidence: Revised CI/CD IAM role policies (exported as JSON), OIDC configuration documentation, router firmware baseline snapshot with version numbers and EOL dates, browser control policy documentation (screenshots of Group Policy settings or MDM profile settings).

Detection Guidance

Chrome: Look for high-frequency renderer process crashes (Event ID 1000/1001 on Windows), unexpected child process spawning from Chrome (e.g., cmd.exe, powershell.exe, wscript.exe as children of chrome.exe), and outbound connections to uncommon destinations initiated by the renderer process. EDR process tree telemetry is the primary detection surface. CI/CD and AWS: Query AWS CloudTrail for CreateUser, AttachUserPolicy, AssumeRole, and GetSecretValue events from pipeline-associated IAM identities, especially outside normal pipeline execution windows. Look for new IAM entities created within 72 hours of pipeline runs. In CI/CD logs, flag any pipeline job that calls external URLs not in an approved allowlist, accesses environment variables containing credential patterns, or modifies pipeline definition files. Routers: Monitor for SNMP traps or syslog entries reflecting configuration changes without corresponding change tickets. Watch for new management-plane login events, especially from external IPs. Outbound traffic spikes from router management IPs to external hosts may indicate botnet command-and-control activity. DNS queries from router management interfaces to non-infrastructure domains are a secondary indicator. No confirmed IOCs (hashes, IPs, domains) were available in source data for this event; detection must rely on behavioral and telemetry-based methods until IOCs are published by authoritative sources.

Indicators of Compromise

Type	Value	Context	Confidence
NOTE	`No confirmed IOCs available`	Specific indicators of compromise — IP addresses, domains, file hashes, or URLs associated with the Chrome zero-days, CI/CD supply chain attack, or router botnets — were not present in the source data for this event. Monitor CISA, Google Project Zero, and vendor security advisories for IOC publication as attribution and forensic analysis mature.	LOW

Framework Mappings

MITRE-ATTACK

- **T1552.001** — Credentials In Files
- **T1072** — Software Deployment Tools
- **T1059** — Command and Scripting Interpreter
- **T1203** — Exploitation for Client Execution
- **T1087.004** — Cloud Account
- **T1078** — Valid Accounts
- **T1195.001** — Compromise Software Dependencies and Development Tools
- **T1189** — Drive-by Compromise
- **T1195** — Supply Chain Compromise
- **T1583.005** — Botnet
- **T1584.005** — Botnet
- **T1195.002** — Compromise Software Supply Chain
- **T1498** — Network Denial of Service
- **T1199** — Trusted Relationship

NIST-800-53R5

- **CM-7** — Least Functionality
- **SI-3** — Malicious Code Protection
- **SI-4** — System Monitoring
- **SI-7** — Software, Firmware, and Information Integrity
- **SC-7** — Boundary Protection
- **SI-2** — Flaw Remediation
- **AC-2** — Account Management
- **AC-6** — Least Privilege
- **IA-2** — Identification and Authentication (Organizational Users)
- **IA-5** — Authenticator Management

- **SA-9** — External System Services
- **SR-2** — Supply Chain Risk Management Plan
- **SR-3** — Supply Chain Controls and Processes
- **SI-10** — Information Input Validation
- **SI-16** — Memory Protection
- **AC-3** — Access Enforcement
- **IR-5** — Incident Monitoring

OWASP-TOP10-2021

- **A04:2021** — Insecure Design
- **A07:2021** — Identification and Authentication Failures
- **A03:2021** — Injection
- **A01:2021** — Broken Access Control

CIS-V8

- **5.2**
- **16.10**
- **6.1**
- **6.2**
- **6.3** — Require MFA for Externally-Exposed Applications
- **15.1** — Establish and Maintain an Inventory of Service Providers

HIPAA-SECURITY

- **164.308(a)(5)(ii)(D)** — Password Management
- **164.312(a)(1)** — Access Control
- **164.312(d)** — Person or Entity Authentication

SOC2-TSC

- **CC6.1** — The entity implements logical access security software, infrastructure, and architectures over protected information assets
- **CC9.2** — Manages risks associated with vendors and business partners

ISO-27001-2022

- **A.8.8** — Management of technical vulnerabilities
- **A.5.21** — Managing information security in the ICT supply chain
- **A.5.23** — Information security for use of cloud services

NIST-CSF-2

- **GV.SC-01** — Cybersecurity supply chain risk management program
- **DE.AE-08** — Incidents are declared when adverse events meet the defined incident criteria

MITRE ATT&CK Mapping

Technique ID	Technique Name	Tactic
T1552.001	Credentials In Files	Credential-Access
T1072	Software Deployment Tools	Execution
T1059	Command and Scripting Interpreter	Execution
T1203	Exploitation for Client Execution	Execution
T1087.004	Cloud Account	Discovery
T1078	Valid Accounts	Defense-Evasion
T1195.001	Compromise Software Dependencies and Development Tools	Initial-Access
T1189	Drive-by Compromise	Initial-Access
T1195	Supply Chain Compromise	Initial-Access
T1583.005	Botnet	Resource-Development
T1584.005	Botnet	Resource-Development
T1195.002	Compromise Software Supply Chain	Initial-Access
T1498	Network Denial of Service	Impact
T1199	Trusted Relationship	Initial-Access

Sources

Source	URL	Tier
Security News	https://thehackernews.com/2026/03/weekly-recap-chrome-0-days-router...	T3
Google Chrome's Security Update Decision—New Browser Danger ...	https://www.forbes.com/sites/daveywinder/2026/02/25/chrome-145-secu...	T3
A Vulnerability in Google Chrome Could Allow for Arbitrary Code ...	https://www.cisecurity.org/advisory/a-vulnerability-in-google-chrom...	T3
Google Chrome Multiple Vulnerabilities	https://www.hkcert.org/security-bulletin/google-chrome-multiple-vul...	T3
Vulnerability findings | Security Command Center	https://docs.cloud.google.com/security-command-center/docs/concepts...	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 2026-03-29 18:41 UTC by TJS Security Command Center