INTELLIGENCE BRIEFING
Security Command Center
TLP:CLEAR
2026-03-29 18:33 UTC

Android March 2026 Patch Rollup, Zero-Day Actively Exploited (CVE-2026-0006 CVSS 9.8) [SCC-2026-0003]

CVE VULNERABILITY | **CRITICAL** | CVSS 9.8

SCC Item ID	SCC-CVE-2026-0003
Type	CVE Vulnerability
CVE ID	CVE-2026-0006
Severity	CRITICAL
CVSS Base Score	9.8
Affected Products	Android 16 and earlier — unpatched against SPL 2026-03-05
Published	20260305

Executive Summary

Google's March 2026 Android security update patches a zero-click remote code execution vulnerability in Android's System and Media Codecs components (CVE-2026-0006, CVSS 9.8) that is actively being exploited in the wild. The update also addresses 128 additional vulnerabilities. All managed Android devices should receive this update immediately via MDM.

Technical Analysis

CVE-2026-0006 is a zero-click RCE in Android 16 System/Media Codecs. Zero-click means no user interaction is required, receiving a specially crafted media file is sufficient to trigger execution. CVSS base score: 9.8 (CRITICAL). The vulnerability is confirmed as actively exploited as of the March 2026 bulletin publication date. The fix is delivered as Android Security Patch Level (SPL) 2026-03-05. Organizations should push this update to all managed Android devices via MDM as a priority action. Devices on older Android versions that are no longer receiving security updates remain permanently vulnerable. The patch rollup addresses 128 total CVEs across Android 16, Google Play system components, and Pixel-specific firmware.

Action Checklist

1. 1. DETECT: Run MDM compliance report, identify all managed Android devices and their current Security Patch Level (SPL). Flag any device not on SPL 2026-03-05 or later.

2. 2. DETECT: For BYOD environments, query enrolled devices via MDM for current Android version and patch level.

3. 3. MITIGATE: Push March 2026 Android security update (SPL 2026-03-05) to all managed devices via MDM immediately.

4. 4. CONTAIN: For devices that cannot receive the patch (EOL devices, delayed OEM rollout): restrict access to corporate email, VPN, and sensitive applications until patched.

5. 5. VERIFY: Re-run MDM compliance report 48 hours post-push. Confirm all managed devices show SPL 2026-03-05 or later.

6. 6. MONITOR: Flag any managed device that has not accepted the update within 72 hours for manual follow-up.

IR / Forensic Enrichment

Triage Priority	IMMEDIATE
Escalation Criteria	Escalate to CISO and executive management if any device with unpatched CVE-2026-0006 exposure is detected accessing sensitive data or connected to critical infrastructure after 24 hours of discovery; coordinate external IR firm engagement if forensic evidence of exploitation (suspicious processes, network connections, file modifications) is found on any device.
Recovery Notes	Post-containment, verify all devices are on SPL 2026-03-05 or later and restore full access (VPN, email, apps) within 4 hours of patch confirmation to minimize user friction and reduce likelihood of BYOD bypass attempts. Conduct 48-hour post-patch monitoring for anomalous behavior (app crashes, excessive battery drain, unusual network traffic) indicating compatibility issues or residual exploitation artifacts; prepare rollback plan (previous SPL version) if critical incompatibilities emerge. Document final patch adoption rate and user non-compliance patterns for future critical update planning.
Forensic Artifacts	Android /data/system/packages.xml (installed package manifest with timestamps; indicates which system patches were applied) | Android /system/etc/security/cacerts/ and /data/misc/keychain/ (certificate store; modified timestamps indicate patch application) | /var/log/syslog and dmesg kernel logs (system update process, potential errors) | MDM server audit logs (policy deployment timestamps, device acknowledgment logs, update delivery status) | Network flow logs / proxy logs (inbound/outbound traffic from Android devices 24-48 hours pre/post-patch; baseline for anomaly detection)

Per-Action IR Details

1. DETECT: Run MDM compliance report, identify all managed Android devices and their current Security Patch Level (SPL). Flag any device not on SPL 2026-03-05 or later.

NIST Phase: Detection Analysis

Reference: NIST 800-61r3 §3.1.1 (Detection and Analysis — System and Network Auditing)

Controls: NIST 800-53 SI-4 (Information System Monitoring and Alerting), NIST 800-53 CM-3 (Configuration Change Control), CIS 6.1 (Establish and Maintain a Data Discovery and Classification Process)

Compensating: If MDM lacks native SPL reporting: export enrolled device list (CSV via MDM admin console); cross-reference against publicly available Android version-to-SPL mapping tables (Android Security & Privacy Year in Review); manually categorize by patch date. Use adb (Android Debug Bridge) over USB to query `getprop ro.build.version.security_patch` on each device if direct MDM reporting unavailable. Document findings in spreadsheet with device IMEI, last-known location, and ownership type (corporate vs. BYOD).

Evidence: Before running report, capture: (1) MDM enrollment roster with device identifiers and enrollment timestamps; (2) baseline SPL snapshot from 2026-02-28 or earlier for comparison; (3) MDM server logs for the past 7 days (device check-in timestamps, policy delivery confirmations). If CVE-2026-0006 exploitation is suspected on any device, preserve device forensic image before MDM wipe/reset actions.

2. DETECT: For BYOD environments, query enrolled devices via MDM for current Android version and patch level.

NIST Phase: Detection Analysis

Reference: NIST 800-61r3 §3.1.2 (Identifying Affected Assets and Scope)

Controls: NIST 800-53 IA-4 (Identifier Management), NIST 800-53 CM-8 (Information System Component Inventory), CIS 2.1 (Maintain Inventory of All Assets Connected to the Network)

Compensating: Query MDM database directly via SQL or API if GUI reports are unavailable: filter enrolled devices by ownership_type='BYOD' and extract android_version, spL_date fields. Cross-reference against BYOD enrollment agreements and acceptable-use policy (AUP) to identify which BYOD devices have security update enforcement clauses. Send automated MDM ping to all BYOD devices and log response times; non-responsive devices may indicate disconnection or uninstall. For manual verification, send MDM-initiated device information request and set 24-hour deadline for user response.

Evidence: Before querying: (1) snapshot current MDM-enrolled BYOD roster with enrollment dates and last check-in timestamps; (2) extract any existing telemetry on BYOD device patch-compliance history (30-day lookback); (3) preserve MDM audit logs showing which BYOD devices have previously failed security policy compliance.

3. MITIGATE: Push March 2026 Android security update (SPL 2026-03-05) to all managed devices via MDM immediately.

NIST Phase: Containment

Reference: NIST 800-61r3 §3.2.1 (Containment Strategy) and §3.3 (Eradication)

Controls: NIST 800-53 SI-2 (Software, Firmware, and Information System Component Flaws), NIST 800-53 CM-5 (Access Restrictions for Change), CIS 3.12 (Address Unauthorized Software)

Compensating: For MDM-free environments: distribute update via email notification with direct links to carrier or manufacturer OTA (over-the-air) update page; require users to manually initiate updates within 48 hours and provide screenshot confirmation. Alternatively, publish internal wiki with step-by-step OTA update instructions and monitor help desk tickets for users who report update failures. For organizations with Android Enterprise (formerly Android for Work): use Managed Google Play to push system update notifications; for non-Enterprise deployments, coordinate directly with device manufacturers (Samsung Knox, Motorola, etc.) to enable expedited rollout in your carrier/region.

Evidence: Before pushing update: (1) verify update package integrity (SHA-256 hash against Google's official March 2026 release notes); (2) capture MDM server configuration baseline, including current policy version and device group assignments; (3) enable MDM audit logging for all update delivery events; (4) if any device is known to be compromised or exhibits suspicious behavior, preserve forensic image before update (updates may overwrite evidence of exploitation).

4. CONTAIN: For devices that cannot receive the patch (EOL devices, delayed OEM rollout): restrict access to corporate email, VPN, and sensitive applications until patched.

NIST Phase: Containment

Reference: NIST 800-61r3 §3.2.2 (Containment Strategy — Short-term and Long-term)

Controls: NIST 800-53 AC-3 (Access Enforcement), NIST 800-53 AC-4 (Information Flow Enforcement), CIS 5.3 (Configure Data Access Control Lists)

Compensating: If MDM cannot enforce conditional access: (1) manually disable VPN profiles for non-compliant devices via MDM (mark as 'suspended pending patch'); (2) configure email gateway (Exchange Online, Gmail Admin) to reject IMAP/POP3 logins from known Android devices with outdated SPL (use device fingerprinting via User-Agent header or device-specific certificate pinning); (3) block non-compliant devices at network edge using MAC address filtering on Wi-Fi controller (requires manual MAC-to-device mapping); (4) notify users via SMS/push notification that device access is restricted and provide escalation contact for exceptions.

Evidence: Before applying access restrictions: (1) preserve current access logs for all affected devices (past 7 days of VPN logins, email access, app activity); (2) document baseline of which users/devices currently access sensitive data; (3) capture MDM policy configuration before changes (for audit and rollback capability); (4) if EOL devices contain sensitive data, perform forensic acquisition before fully severing access (May be needed for incident investigation if compromise is discovered later).

5. VERIFY: Re-run MDM compliance report 48 hours post-push. Confirm all managed devices show SPL 2026-03-05 or later.

NIST Phase: Recovery

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

Controls: NIST 800-53 CA-7 (Continuous Monitoring), NIST 800-53 SI-4 (Information System Monitoring), CIS 8.1 (Establish and Maintain a Data-Driven Security Awareness and Training Program)

Compensating: If MDM lacks automated verification: manually poll each device using adb (`adb shell getprop ro.build.version.security_patch`) or request users to navigate to Settings > About Phone > Android Version / Security Patch Level and email screenshots within 48 hours. Parse responses manually and build compliance matrix. For BYOD, accept user attestation signed via email as proof of compliance; maintain list of attestations for audit trail.

Evidence: Before verification: (1) export and archive the pre-update compliance report (from Step 1) for change comparison; (2) establish timestamp baseline for update deployment (document exact time MDM pushed update to each device); (3) ensure MDM server time is synchronized (NTP) to avoid clock-skew errors in delivery/receipt logs; (4) preserve MDM delivery logs for 90 days minimum (proof of update completion for regulatory audits).

6. MONITOR: Flag any managed device that has not accepted the update within 72 hours for manual follow-up.

NIST Phase: Post Incident

Reference: NIST 800-61r3 §3.4.1 (Post-Incident Activities) and §4 (Lessons Learned)

Controls: NIST 800-53 IR-6 (Incident Reporting), NIST 800-53 CA-7 (Continuous Monitoring), CIS 6.2 (Establish and Maintain a Data Inventory)

Compensating: Create manual alert spreadsheet: log device identifiers, ownership, user contact info, and update status as of each 24-hour checkpoint (24h, 48h, 72h post-push). At 72h, auto-generate escalation list. Assign security team members to contact non-compliant device owners via phone/SMS (for BYOD) or IT ticketing system (for corporate). Document reason for non-compliance (device offline, user rejected, network connectivity issues) and assign remediation deadline. For repeat non-compliant users, escalate to management; mark devices for deprovisioning after 30 days of non-compliance.

Evidence: Before flagging: (1) preserve MDM device check-in logs for the full 72-hour window; (2) cross-reference non-responsive devices against network logs to determine if connectivity loss is cause; (3) maintain historical compliance database (rolling 6-month record of patch adoption rates per device cohort) for trend analysis; (4) if a non-compliant device later shows signs of compromise (suspicious app behavior, data exfiltration), retrieve and preserve forensic image immediately (evidence of exploitation attempts).

Detection Guidance

MDM telemetry is the primary detection surface. Query your MDM console (Intune, Jamf, VMware Workspace ONE, etc.) for: SecurityPatchLevel < 2026-03-05. Filter by enrolled Android devices. For EDR-enrolled Android devices: watch for unusual media processing behavior or unexpected outbound connections originating from media framework processes.

Indicators of Compromise

Type	Value	Context	Confidence
FILE_PATH	`/system/lib/libandroid_runtime.so`	Core Android runtime library commonly targeted in privilege escalation zero-days; monitor for unexpected modifications or replacements	MEDIUM
FILE_PATH	`/data/local/tmp/`	Common staging directory for Android exploits to drop and execute malicious payloads; suspicious when files are written by unprivileged processes, system daemons, or apps lacking `WRITE_EXTERNAL_STORAGE` permissions, or when followed by execution attempts via `Runtime.exec()` or `ProcessBuilder` - legitimate use is restricted to app-specific cache operations by the owning application and rarely involves executable binaries or persistence mechanisms.	MEDIUM
FILE_PATH	`/proc/self/mem`	Frequently accessed during memory corruption exploits on Android; anomalous access patterns may indicate exploitation attempts	MEDIUM
FILE_PATH	`/system/bin/sh`	Shell binary spawned as child of unexpected system processes may indicate successful privilege escalation via zero-day exploitation	MEDIUM
FILE_PATH	`/data/data//cache/`	Application cache directories used to stage exploit payloads prior to execution in Android zero-day attack chains	MEDIUM
FILE_PATH	`/system/app/`	System application directory; unauthorized writes or new APK installation here indicate successful privilege escalation	MEDIUM
FILE_PATH	`/system/priv-app/`	Privileged application directory targeted by attackers after gaining elevated access; monitor for unexpected additions	MEDIUM
FILE_PATH	`/dev/binder`	Android Binder IPC driver frequently exploited in kernel-level vulnerabilities; anomalous interactions may signal exploitation	MEDIUM
REGISTRY_KEY	`N/A - Android platform does not use Windows registry`	Not applicable for Android OS threat; Android uses SQLite databases and flat configuration files instead	MEDIUM

Type	Value	Context	Confidence
FILE_PATH	`/sdcard/Download/`	External storage download directory commonly used as initial drop location for exploit APKs or payloads delivered via phishing	MEDIUM
URL	`hxxp://[attacker-infrastructure]/payload.apk`	Generic pattern for malicious APK delivery infrastructure used in Android zero-day exploit chains; monitor proxy logs for APK downloads from non-Play Store sources	MEDIUM
FILE_PATH	`/system/framework/services.jar`	Core Android framework component; tampering or unexpected modification may indicate persistent implant installation post-exploitation	MEDIUM
FILE_PATH	`/data/system/packages.xml`	Package manager database; modifications outside of legitimate install flows may indicate stealthy APK installation post-exploitation	MEDIUM
FILE_PATH	`/proc/kallsyms`	Kernel symbol table accessed by exploit code to defeat KASLR; anomalous read access from userspace applications is suspicious	MEDIUM
FILE_PATH	`/sys/kernel/debug/`	Kernel debug filesystem; access from non-privileged processes is a strong indicator of privilege escalation attempt or kernel exploit activity	MEDIUM

Framework Mappings

MITRE-ATTACK

- **T1203** — Exploitation for Client Execution
- **T1404** — Exploitation for Privilege Escalation

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
T1203	Exploitation for Client Execution	Execution
T1404		

Sources

Source	URL	Tier
Android Security Bulletin, March 2026	https://source.android.com/docs/security/bulletin/2026/2026-03-01	T3
NVD	https://nvd.nist.gov/vuln/detail/CVE-2026-0006	T1

Generated 2026-03-29 18:33 UTC by TJS Security Command Center