INTELLIGENCE BRIEFING
Security Command Center
TLP:CLEAR
2026-05-05 08:19 UTC

CVE-2026-0073: Critical Android Zero-Click RCE via Wireless ADB Bypass

CVE VULNERABILITY | **CRITICAL** | CVSS 9.8

SCC Item ID	SCC-CVE-2026-0121
Type	CVE Vulnerability
CVE ID	CVE-2026-0073
Severity	CRITICAL
CVSS Base Score	9.8
Affected Products	Android (multiple versions; specific version range not confirmed from available sources, human verification recommended against Google Android Security Bulletin May 2026)
Published	3 hours ago
Discovery Source	Serper

Executive Summary

A critical zero-click vulnerability in Android (CVE-2026-0073, CVSS 9.8) allows attackers to execute code and obtain a remote shell on affected devices with no user interaction required. The flaw bypasses authentication in Android's Wireless ADB component, meaning any Android device with Wireless ADB exposed on a network reachable by an attacker is at risk. Google is expected to address this in the May 2026 Android Security Bulletin. Organizations with unpatched Android devices, particularly those used for corporate access, MDM enrollment, or device management, should treat this as an immediate patching priority pending confirmation of affected version ranges from the official bulletin.

Technical Analysis

CVE-2026-0073 is a critical zero-click RCE vulnerability in the Android Wireless ADB (Android Debug Bridge) subsystem. Exploitation requires no user interaction and no authentication; a remote attacker with network access to the ADB port (default TCP 5555) can bypass the authentication handshake and obtain a remote shell with the privileges available to the ADB daemon. Root cause maps to CWE-287 (Improper Authentication) and CWE-94 (Improper Control of Code Generation). MITRE ATT&CK techniques include T1133 (External Remote Services), T1210 (Exploitation of Remote Services), and T1059 (Command and Scripting Interpreter). A related CVE, CVE-2025-48593, has been reported in secondary sources as a separate critical zero-click Android RCE; the relationship has not been confirmed from authoritative sources and warrants verification against primary sources. CVSS base score: 9.8. CVSS vector is pending NVD publication. Specific affected version ranges are

not confirmed from available sources; human verification against the official Google Android Security Bulletin (May 2026) is required before scoping impact. EPSS and KEV data are not yet populated in available records. Five of seven sources are Tier 3 (secondary reporting). Two T1 sources are listed (NVD and Google Advisory), but the NVD entry may not yet be published with full details, and the Google source URL requires verification. Authoritative confirmation is pending full population of the NVD entry and direct access to the Google Android Security Bulletin (May 2026). Recommendation: This item should be marked as 'secondary reporting pending authoritative confirmation' in any dashboard or distribution system until NVD publishes the CVE entry with CVSS vector and affected version range, and the official Google Android Security Bulletin is publicly accessible.

Action Checklist

1. Step 1: Containment. Immediately audit your device inventory for Android devices with Wireless ADB enabled (TCP port 5555 open). Block inbound connections to TCP 5555 at network perimeter firewalls and MDM-enforced host-based rules. Prioritize corporate-managed Android devices, shared kiosks, and any device exposed to internal or external networks. Do not wait for patch deployment to complete this step.
2. Step 2: Detection. Query MDM/EMM platforms (Intune, Jamf, SOTI, etc.) for Android devices with Developer Options or Wireless ADB enabled. Inspect network flow logs and firewall logs for inbound or lateral TCP connections to port 5555 targeting Android device IPs. Look for unexpected ADB session initiation events in device logs. No confirmed IOCs (hashes, IPs, domains) are available from current sources; monitor for anomalous shell execution or process spawning originating from the ADB daemon process.
3. Step 3: Eradication. Apply the May 2026 Android Security Bulletin patch once released by Google and OEM vendors. If the bulletin is not yet available, proceed immediately with Step 1 (containment) to block port 5555 and Step 2 (detection). Check the official Google Android Security Bulletin page regularly for patch availability. On devices where patching is not yet possible, disable Wireless ADB via MDM configuration profile or manually in Developer Options. Confirm the specific affected version range against the official Google bulletin before marking devices as remediated.
4. Step 4: Recovery. After patching, re-audit MDM compliance reports to confirm patch level across the fleet. Re-scan network segments for any residual TCP 5555 exposure. On any device that had Wireless ADB reachable from untrusted networks prior to patching, treat the device as potentially compromised: review for unauthorized application installation, unexpected account additions, or anomalous data exfiltration indicators before returning to full trust.
5. Step 5: Post-Incident. Review MDM policy to enforce Wireless ADB disabled by default across all managed Android devices. Evaluate whether network segmentation adequately isolates mobile device management VLANs. Add TCP 5555 inbound connection attempts to ongoing SOC alerting rules. Document the control gap: Wireless ADB should not be reachable from any untrusted segment on production or corporate devices.

IR / Forensic Enrichment

Triage Priority	IMMEDIATE

Escalation Criteria	Escalate to CISO, legal, and (if applicable) privacy officer immediately if any Android device with confirmed or suspected TCP 5555 exposure prior to containment stores, processes, or transmits PII, PHI, or payment card data — zero-click RCE with no user interaction required means any such device must be treated as breached for purposes of regulatory breach notification timelines (HIPAA 60-day, GDPR 72-hour) until forensic triage rules out data access.
Recovery Notes	Any Android device confirmed to have had TCP 5555 reachable from an untrusted network segment before containment must complete the ADB triage sequence (unauthorized app inventory, account enumeration, ADB key comparison) before being returned to service — patch application alone does not address attacker persistence artifacts such as injected ADB authorization keys or sideloaded APKs. Monitor MDM compliance telemetry and network flows for TCP 5555 activity for a minimum of 30 days post-patch to detect any devices that were missed in the initial inventory sweep or that re-enable Developer Options. If factory reset is required on compromised devices, verify corporate re-enrollment via MDM and re-apply all configuration profiles before the device is used with corporate credentials or data.
Forensic Artifacts	/data/misc/adb/adb_keys on each Android device — this file stores RSA public keys authorized for ADB access; an attacker who exploited CVE-2026-0073's authentication bypass may have written their own key here to establish persistent post-exploitation ADB access independent of the original vulnerability | Android logcat output filtered to 'adbd' process tag ('adb shell logcat -d -s adbd') — the ADB daemon logs connection attempts, authentication events, and session establishment; a successful zero-click exploit would appear as an unauthenticated connection acceptance with no corresponding user authorization prompt | Network flow records (NetFlow/IPFIX/firewall syslog) for TCP port 5555 traffic over the 30-day window preceding discovery, filtered to flows where the destination IP belongs to an Android device — session duration and transferred bytes distinguish reconnaissance (short, low-byte) from active exploitation and data exfiltration (sustained sessions with significant byte counts) | Android package manager output ('adb shell pm list packages -i -U') capturing installer source and UID for all installed applications — applications installed via ADB shell will show installer as 'null' or 'com.android.shell' rather than the MDM enrollment agent or Google Play, flagging attacker-deployed APKs dropped during an active shell session | MDM/EMM audit log showing device compliance state history, specifically the Developer Options and ADB enabled/disabled status timestamps — this establishes the window during which each device was vulnerable and exposed, which is essential for scoping potential data access for breach notification determinations

Per-Action IR Details

Step 1: Containment — Immediately audit your device inventory for Android devices with Wireless ADB enabled (TCP port 5555 open). Block inbound connections to TCP 5555 at network perimeter firewalls and MDM-enforced host-based rules. Prioritize corporate-managed Android devices, shared kiosks, and any device exposed to internal or external networks. Do not wait for patch deployment to complete this step.

NIST Phase: Containment

Reference: NIST 800-61r3 §3.3 — Containment Strategy: isolate affected systems to prevent spread before eradication is possible

Controls: NIST IR-4 (Incident Handling), NIST SC-7 (Boundary Protection), CIS 4.4 (Implement and Manage a Firewall on Servers), CIS 4.5 (Implement and Manage a Firewall on End-User Devices), CIS 12.2 — Manage Network Infrastructure (IG2/IG3: establish and maintain network architecture diagrams and segment high-risk assets)

Compensating: For teams without enterprise MDM: run a network-wide Nmap scan — 'nmap -p 5555 --open -T4 ' — to identify all hosts responding on TCP 5555. Immediately add a deny rule on the perimeter firewall (iptables: 'iptables -I FORWARD -p tcp --dport 5555 -j DROP') and on any Linux-based network gateway. For Windows-based network

appliances, add an inbound block rule via PowerShell: 'New-NetFirewallRule -DisplayName "Block ADB" -Direction Inbound -Protocol TCP -LocalPort 5555 -Action Block'. Document each identified device by MAC and IP before blocking.

Evidence: Before blocking TCP 5555 at the perimeter, capture full packet captures of any active sessions on port 5555 using Wireshark or tcpdump ('tcpdump -i -w adb_capture.pcap port 5555') to preserve evidence of any in-progress ADB session, including client IP, session initiation handshake, and any ADB protocol commands already transmitted. Also snapshot current Android device network state from MDM compliance dashboard showing ADB-enabled status per device before any MDM policy push is applied.

Step 2: Detection — Query MDM/EMM platforms (Intune, Jamf, SOTI, etc.) for Android devices with Developer Options or Wireless ADB enabled. Inspect network flow logs and firewall logs for inbound or lateral TCP connections to port 5555 targeting Android device IPs. Look for unexpected ADB session initiation events in device logs. No confirmed IOCs (hashes, IPs, domains) are available from current sources — monitor for anomalous shell execution or process spawning originating from the ADB daemon process.

NIST Phase: Detection Analysis

Reference: NIST 800-61r3 §3.2 — Detection and Analysis: correlate indicators across log sources, characterize scope of activity, and determine whether adverse events meet incident criteria

Controls: NIST SI-4 (System Monitoring), NIST AU-6 (Audit Record Review, Analysis, and Reporting), NIST AU-2 (Event Logging), NIST IR-5 (Incident Monitoring), CIS 8.2 (Collect Audit Logs), MITRE ATT&CK T1059 (Command and Scripting Interpreter) — post-exploitation shell spawned via ADB daemon, MITRE ATT&CK T1190 (Exploit Public-Facing Application) — unauthenticated ADB access over network

Compensating: Without SIEM: query firewall syslog files with grep for port 5555 connections to Android device IP ranges: 'grep ":5555" /var/log/firewall.log | awk '{print $1,$2,$5,$6}' | sort | uniq -c | sort -rn'. For Android device-level detection on rooted or enterprise-managed devices, use adb shell to check running processes: 'adb -s :5555 shell ps -A | grep adbd' and review '/data/misc/adb/adb_keys' for unauthorized public keys added by an attacker. Use osquery on any Linux-based management host to query network connections: 'SELECT * FROM process_open_sockets WHERE remote_port = 5555;'.

Evidence: Collect: (1) Android device ADB authorization key store at '/data/misc/adb/adb_keys' — any attacker who successfully exploited CVE-2026-0073 may have injected their RSA public key here to maintain persistent ADB access; (2) Android logcat output filtered for 'adbd' process entries ('adb shell logcat -d -s adbd') showing authentication bypass events or unexpected connection acceptances; (3) network flow logs (NetFlow/IPFIX) for the 30-day lookback window filtered to TCP 5555 flows destined to Android device IPs, noting source IPs, session duration, and byte counts indicative of data transfer; (4) MDM/EMM audit logs showing any device enrollment changes, app installations, or policy overrides timestamped around detected port 5555 activity.

Step 3: Eradication — Apply the May 2026 Android Security Bulletin patch as distributed by Google and OEM vendors. Verify patch application via MDM compliance policy reporting. On devices where patching is not yet possible, disable Wireless ADB via MDM configuration profile or manually in Developer Options. Confirm the specific affected version range against the official Google bulletin before marking devices as remediated.

NIST Phase: Eradication

Reference: NIST 800-61r3 §3.4 — Eradication: remove the vulnerability and any artifacts of exploitation from affected systems, verify eradication before recovery

Controls: NIST SI-2 (Flaw Remediation), NIST SI-7 (Software, Firmware, and Information Integrity), NIST CM-3 (Configuration Change Control), CIS 7.3 (Perform Automated Operating System Patch Management), CIS 7.4 (Perform Automated Application Patch Management), CIS 7.1 (Establish and Maintain a Vulnerability Management Process), CIS 7.2 (Establish and Maintain a Remediation Process)

Compensating: For teams without automated patch management: use Android Debug Bridge in batch mode to verify the Security Patch Level on each device before declaring remediation complete — 'adb -s :5555 shell getprop ro.build.version.security_patch' — and confirm the returned date is 2026-05-01 or later (matching the May 2026 bulletin). For devices that cannot be patched, push a MDM restriction profile disabling Developer Options entirely (Android Enterprise: 'DeveloperOptionsDisabled = true' in the managed configuration payload). Maintain a spreadsheet

tracking each device's serial number, current patch level, ADB status, and remediation date as a manual compliance record.

Evidence: Before patching, preserve: (1) a full device property dump ('adb shell getprop > device_props__.txt') capturing current Android version, security patch level, and build fingerprint to establish the pre-patch baseline for the incident record; (2) contents of '/data/misc/adb/adb_keys' to document any attacker-injected ADB authorization keys that must be removed as part of eradication — do not skip this step even if patching proceeds, as the patch does not remove injected keys; (3) list of all installed packages ('adb shell pm list packages -i -U') to identify any applications installed by an attacker via the ADB shell prior to patch application.

Step 4: Recovery — After patching, re-audit MDM compliance reports to confirm patch level across the fleet. Re-scan network segments for any residual TCP 5555 exposure. On any device that had Wireless ADB reachable from untrusted networks prior to patching, treat the device as potentially compromised: review for unauthorized application installation, unexpected account additions, or anomalous data exfiltration indicators before returning to full trust.

NIST Phase: Recovery

Reference: NIST 800-61r3 §3.5 — Recovery: restore systems to normal operation only after verifying integrity; devices with prior untrusted ADB exposure require compromise assessment before trust restoration

Controls: NIST IR-4 (Incident Handling), NIST CP-10 (System Recovery and Reconstitution), NIST SI-7 (Software, Firmware, and Information Integrity), NIST AU-6 (Audit Record Review, Analysis, and Reporting), CIS 1.1 (Establish and Maintain Detailed Enterprise Asset Inventory), CIS 5.1 (Establish and Maintain an Inventory of Accounts), CIS 5.3 (Disable Dormant Accounts)

Compensating: Without EDR on Android: for each device with confirmed prior TCP 5555 exposure, run the following manual triage sequence via ADB — (1) 'adb shell pm list packages -i -U' to identify apps installed by UID 2000 (shell) or with unknown installers, which would indicate attacker-installed APKs; (2) 'adb shell dumpsys account' to enumerate added accounts not matching corporate provisioning; (3) 'adb shell settings get global always_finish_activities' and related developer settings to detect attacker-modified runtime configurations; (4) compare current '/data/misc/adb/adb_keys' against the pre-incident baseline captured in Step 3 and remove any unauthorized entries. If any of these checks are positive, factory reset the device and re-enroll from MDM before returning to service.

Evidence: Collect post-patch validation evidence: (1) MDM compliance report export (PDF or CSV) showing Android Security Patch Level = 2026-05-01 or later for each device, timestamped, to serve as the remediation close-out record; (2) second Nmap scan output ('nmap -p 5555 --open > post_patch_scan_.txt') confirming TCP 5555 is no longer responding on any device IP; (3) for devices flagged as potentially compromised, the full ADB triage output from the compensating control commands above, preserved per device as named text files for the incident record.

Step 5: Post-Incident — Review MDM policy to enforce Wireless ADB disabled by default across all managed Android devices. Evaluate whether network segmentation adequately isolates mobile device management VLANs. Add TCP 5555 inbound connection attempts to ongoing SOC alerting rules. Document the control gap — Wireless ADB should not be reachable from any untrusted segment on production or corporate devices.

NIST Phase: Post Incident

Reference: NIST 800-61r3 §4 — Post-Incident Activity: conduct lessons-learned review, update policies and detection rules, and share intelligence to prevent recurrence

Controls: NIST IR-4 (Incident Handling), NIST IR-8 (Incident Response Plan), NIST SI-5 (Security Alerts, Advisories, and Directives), NIST CM-3 (Configuration Change Control), NIST SC-7 (Boundary Protection), CIS 4.2 (Establish and Maintain a Secure Configuration Process for Network Infrastructure), CIS 7.1 (Establish and Maintain a Vulnerability Management Process), CIS 8.2 (Collect Audit Logs)

Compensating: Without a commercial SIEM: create a persistent Sigma rule translated to your available log tool targeting TCP 5555 inbound attempts — for Sysmon-monitored Linux gateways, add a cron-scheduled logwatch script: 'grep ":5555" /var/log/firewall.log | mail -s "ADB Port 5555 Alert" soc@company.com'. For the MDM policy gap, export your current Android Enterprise managed configuration baseline to a version-controlled repository (Git) so future deviations from the ADB-disabled policy are auditable. Draft a one-page lessons-learned memo documenting: (a) number of devices found with ADB enabled, (b) network segments where TCP 5555 was reachable, (c) time-to-contain

from initial advisory to firewall block, and (d) MDM policy gap — Wireless ADB was not explicitly prohibited in the prior baseline configuration.

Evidence: Post-incident documentation package should include: (1) the original MDM compliance report showing ADB-enabled devices at time of discovery (baseline evidence of control gap); (2) firewall rule change records showing the date/time TCP 5555 block was implemented, to calculate time-to-contain for the lessons-learned metric; (3) the updated MDM configuration profile with 'DeveloperOptionsDisabled = true' and its deployment confirmation report, as evidence the structural gap was closed; (4) the updated network segmentation diagram or firewall ACL showing mobile device VLANs are now isolated from untrusted segments on TCP 5555.

Detection Guidance

Primary detection focus is network-level and MDM-level. Once the May 2026 Security Bulletin is published, query MDM/EMM for devices reporting Android patch levels below the patched version. Until the bulletin is available, focus on network detection: inbound or east-west TCP SYN to port 5555 targeting known Android device IPs. In firewall logs, alert on source = any untrusted IP, destination port = 5555, destination = mobile device subnet. In SIEM, create rules for behavioral indicators post-exploitation: unexpected process spawning under the ADB daemon, new APK installation events not originating from managed app distribution, unauthorized user account creation, or shell command execution chains. No confirmed IOCs are available from current Tier 3 sources. Exploitation signatures specific to this CVE have not been published in available reporting; detection relies on behavioral and network indicators until NVD and official bulletin details are available.

Framework Mappings

MITRE-ATTACK

- **T1133** — External Remote Services
- **T1210** — Exploitation of Remote Services
- **T1059** — Command and Scripting Interpreter

NIST-800-53R5

- **AC-17** — Remote Access
- **AC-20** — Use of External Systems
- **IA-2** — Identification and Authentication (Organizational Users)
- **IA-5** — Authenticator Management
- **SC-7** — Boundary Protection
- **AC-6** — Least Privilege
- **SI-2** — Flaw Remediation
- **CM-7** — Least Functionality
- **SI-3** — Malicious Code Protection
- **SI-4** — System Monitoring
- **SI-7** — Software, Firmware, and Information Integrity
- **SI-10** — Information Input Validation
- **IA-8** — Identification and Authentication (Non-Organizational Users)

OWASP-TOP10-2021

- **A03:2021** — Injection
- **A07:2021** — Identification and Authentication Failures

CIS-V8

- **16.10** — Apply Secure Design Principles in Application Architectures
- **6.3** — Require MFA for Externally-Exposed Applications
- **6.4** — Require MFA for Remote Network Access
- **6.5** — Require MFA for Administrative Access

SOC2-TSC

- **CC6.1** — The entity implements logical access security software, infrastructure, and architectures over protected information assets

HIPAA-SECURITY

- **164.312(d)** — Person or Entity Authentication

ISO-27001-2022

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

MITRE ATT&CK Mapping

Technique ID	Technique Name	Tactic
T1133	External Remote Services	Persistence
T1210	Exploitation of Remote Services	Lateral-Movement
T1059	Command and Scripting Interpreter	Execution

Sources

Source	URL	Tier
	https://gbhackers.com/critical-android-zero-click-vulnerability/	T3
CVE-2025-48593: Critical Zero-Click Vulnerability in Android ...	https://socprime.com/blog/cve-2025-48593-vulnerability-in-android/	T3
Critical Android Zero-Click Vulnerability Grants Remote Shell Access	https://cybersecuritynews.com/android-zero-click-vulnerability/	T3

Source	URL	Tier
Critical Android 0-Click Vulnerability Enables Remote Code Execution	https://www.linkedin.com/pulse/critical-android-0-click-vulnerabili...	T3
Critical Zero-Click Android Flaw Grants Remote Shell Access ...	https://securityonline.info/android-critical-zero-click-rce-cve-202...	T3
NVD	https://nvd.nist.gov/vuln/detail/CVE-2026-0073	T1
Google Security Advisory	https://chromereleases.googleblog.com/	T1

Generated 2026-05-05 08:19 UTC by TJS Security Command Center