INTELLIGENCE BRIEFING
Security Command Center
TLP:CLEAR
2026-05-03 18:28 UTC

FEMITBOT Weaponizes Telegram Mini Apps for Scalable Crypto Fraud and Android Malware Distribution

THREAT CAMPAIGN | **HIGH** | CVSS 7.5

SCC Item ID	SCC-CAM-2026-0265
Type	Threat Campaign
Severity	HIGH
CVSS Base Score	7.5
Affected Products	Telegram Mini Apps (WebView), Android (APK sideloading); impersonated brands include Apple, Coca-Cola, Disney, eBay, IBM, MoonPay, NVIDIA, YouKu, BBC, CineTV, Coreweave, Claro
Published	2026-05-03T10:11:21
Discovery Source	Rss

Executive Summary

FEMITBOT is a fraud-as-a-service platform that exploits Telegram Mini Apps to deliver cryptocurrency investment scams and Android malware across multiple simultaneously impersonated brands, including Apple, Disney, IBM, and others. The platform operates from shared backend infrastructure with a consistent API signature, enabling rapid rebranding and high-volume campaign deployment against Telegram users globally. Organizations face reputational risk if their brand is impersonated, and employees or customers who use Telegram are directly exposed to credential harvesting and device compromise.

Technical Analysis

FEMITBOT abuses Telegram's Mini App feature, which renders web content inside a WebView context within the Telegram client, to serve convincing phishing UIs without requiring victims to leave the app. The platform's backend infrastructure shares a consistent API signature across phishing domains, allowing CTM360 researchers to cluster campaigns and attribute them to a single operator or organized group (source: CTM360 FEMITBOT report). Android malware is distributed via sideloaded APKs delivered through bot interactions, bypassing Google Play Protect (CWE-693: Protection Mechanism Failure). The WebView delivery context enables UI redressing that obscures the phishing origin (CWE-1021: Improper Restriction of Rendered UI Layers or Frames). Relevant MITRE ATT&CK techniques include T1566.003 (Spearphishing via Service,

Telegram bot delivery), T1204.002 (User Execution: Malicious File, APK sideloading), T1036.005 (Masquerading, brand impersonation), T1583.001 (Acquire Infrastructure: Domains), and T1071.001 (Application Layer Protocol: WebView delivery). No CVE is assigned. No patch is available from a single vendor; mitigations are procedural and platform-level. Source quality is moderate (T3 sources; CTM360 is the originating researcher).

Action Checklist

1. Containment, Block Telegram bot interactions and Mini App WebView access on managed Android devices via MDM policy. If employees use Telegram for business, issue an advisory restricting unsolicited bot engagement until further notice.
2. Detection, Query mobile device management (MDM) and endpoint logs for APK installations originating outside Google Play on Android devices. Flag processes spawned from sideloaded APKs. Monitor DNS and proxy logs for domains matching the CTM360-identified FEMITBOT API signature pattern; request the IOC list from CTM360's published report for domain blocklist ingestion.
3. Eradication, Remove any sideloaded APKs identified during detection. Revoke sessions and force credential resets for any accounts accessed on potentially compromised Android devices. Block identified FEMITBOT phishing domains at DNS and proxy layers.
4. Recovery, Confirm APK removal and validate no persistence mechanisms remain on affected devices (review scheduled jobs, accessibility service grants, device admin permissions). Monitor affected accounts for unauthorized access or transaction activity post-remediation.
5. Post-Incident, Review MDM policy enforcement for APK sideloading controls (ensure 'Unknown Sources' is disabled on all managed Android devices). Assess whether brand monitoring is in place to detect impersonation of your organization across Telegram and similar platforms. File a brand abuse report with Telegram if your organization's brand appears in active FEMITBOT campaigns.

IR / Forensic Enrichment

Triage Priority	URGENT
Escalation Criteria	Escalate immediately to legal, communications, and executive leadership if any confirmed employee submitted credentials, wallet addresses, or payment information to a FEMITBOT Mini App, or if your organization's brand is confirmed as an active impersonation target in the campaign, as either condition triggers potential fraud liability, regulatory breach notification obligations (GDPR Art. 33, state privacy laws if PII was harvested), and reputational crisis response requirements.
Recovery Notes	After APK removal and credential revocation, maintain enhanced monitoring of affected accounts and corporate DNS/proxy logs for a minimum of 30 days, as FEMITBOT-distributed Android malware with accessibility service grants may have exfiltrated session tokens or 2FA codes that enable delayed account compromise even after device remediation. Validate that all crypto-related transactions initiated from affected devices or accounts during the compromise window are reviewed with the relevant platform (MoonPay, or any exchange whose credentials were exposed), and file fraud disputes where applicable. Re-run the `adb shell dumpsys accessibility` and device admin checks at 7-day and 30-day intervals to confirm no persistence re-establishment from any overlooked APK component.

Forensic Artifacts	Sideloaded APK file(s) on Android devices: retrieve via `adb pull $(adb shell pm path | cut -d: -f2)`, hash with SHA-256, and submit to VirusTotal — FEMITBOT-distributed APKs would show permissions including ACCESSIBILITY_SERVICE, BIND_DEVICE_ADMIN, RECEIVE_SMS, and READ_CONTACTS consistent with credential harvesting and persistence. | Android accessibility service grants log: `adb shell dumpsys accessibility` output captures any FEMITBOT APK that registered an accessibility service to overlay screens, intercept OTPs, or auto-click crypto investment confirmations within Telegram Mini App WebViews. | Corporate DNS/proxy query logs (30-day window): filter for HTTP POST requests and DNS lookups matching CTM360-identified FEMITBOT backend API hostnames — the shared infrastructure signature across all impersonated brands (Apple, Disney, IBM, etc.) means a single API domain pattern will surface all campaign variants touching your network. | Telegram Mini App WebView network traffic: if captured via `adb shell tcpdump` or corporate proxy SSL inspection, preserve the full HTTP request/response for any Mini App session — specifically the onboarding API calls (wallet registration, referral code submission, KYC-phishing form posts) that constitute the FEMITBOT fraud-as-a-service workflow. | MDM application inventory export with install-source metadata: timestamp-stamped export from your MDM console (Intune, Jamf, Google Workspace) showing package name, version, install source, and install time for all Android apps — any entry with install source other than `com.android.vending` during the campaign window is a primary forensic indicator for this specific FEMITBOT sideloading vector.

Per-Action IR Details

Containment — Block Telegram bot interactions and Mini App WebView access on managed Android devices via MDM policy. If employees use Telegram for business, issue an advisory restricting unsolicited bot engagement until further notice.

NIST Phase: Containment

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

Controls: NIST IR-4 (Incident Handling), NIST SC-7 (Boundary Protection), CIS 4.5 (Implement and Manage a Firewall on End-User Devices), CIS 6.3 (Require MFA for Externally-Exposed Applications)

Compensating: For teams without MDM: push an emergency Group Policy or Android Enterprise config via Google Workspace/Intune free tier to disable 'Unknown Sources' and restrict Telegram WebView rendering. Alternatively, use an MDM-free approach by deploying a DNS-layer block (Pi-hole or Cloudflare Gateway free tier) on all corporate Wi-Fi exit points, targeting *.t.me Mini App subdomains and the CTM360-identified FEMITBOT API signature domains. Issue a written advisory via email/Slack restricting bot engagement; document receipt acknowledgments for compliance trail.

Evidence: Before enforcing MDM policy, capture: current Telegram app version and Mini App permissions from affected devices via MDM inventory export; Android logcat output (`adb logcat -d > device_logcat.txt`) capturing WebView activity and any invoked intents from Telegram; screenshot or screen recording of any Mini App the user interacted with, specifically preserving the WebView URL (chrome://inspect or Android Developer Bridge) to match against FEMITBOT API signature patterns (e.g., consistent endpoint paths like `/api/invest`, `/api/register` observed in CTM360 reporting); MDM enrollment status and last policy sync timestamp to confirm policy gap window.

Detection — Query mobile device management (MDM) and endpoint logs for APK installations originating outside Google Play on Android devices. Flag processes spawned from sideloaded APKs. Monitor DNS and proxy logs for domains matching the CTM360-identified FEMITBOT API signature pattern; request the IOC list from CTM360's published report for domain blocklist ingestion.

NIST Phase: Detection Analysis

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

Controls: NIST SI-4 (System Monitoring), NIST AU-2 (Event Logging), NIST AU-6 (Audit Record Review, Analysis, and Reporting), CIS 8.2 (Collect Audit Logs), CIS 2.3 (Address Unauthorized Software)

Compensating: Without SIEM/EDR: run `adb shell pm list packages -i -3` on each managed Android device to enumerate all third-party APKs and their install sources — flag any package installer value that is NOT `com.android.vending` (Google Play). Cross-reference package names against CTM360 IOC list. For DNS detection without a SIEM, pull Pi-hole or router DHCP/DNS query logs and grep for FEMITBOT-associated TLDs using: `grep -Ei '(moonpay|cinetvapp|claro-invest|youku-earn)' /var/log/pihole.log` (adjust domain fragments to match CTM360 IOC list). Use osquery on any enrolled Linux/Mac MDM-adjacent management hosts to query `SELECT * FROM processes WHERE path LIKE '/data/data/%'` via Android Debug Bridge integration.

Evidence: Capture before analysis: MDM application inventory export showing all installed APKs, install timestamps, and install source for each managed Android device; DNS query logs from corporate resolvers or guest Wi-Fi for the 30-day window prior to detection, filtered for domains matching FEMITBOT infrastructure patterns identified by CTM360 (shared backend API hostnames, consistent URI structures across impersonated brands); proxy/NGFW logs showing HTTP POST requests to FEMITBOT API endpoints — specifically look for JSON payloads consistent with crypto investment onboarding flows (fields like `invite_code`, `wallet_address`, `referral_id`); Android package manager logs (`/data/system/packages.xml` if device is rooted or via MDM forensic agent) showing sideload events with timestamps; Telegram chat logs or forwarded message metadata if the user consented to preservation, capturing the bot username and Mini App launch URL.

Eradication — Remove any sideloaded APKs identified during detection. Revoke sessions and force credential resets for any accounts accessed on potentially compromised Android devices. Block identified FEMITBOT phishing domains at DNS and proxy layers.

NIST Phase: Eradication

Reference: NIST 800-61r3 §3.4 — Eradication

Controls: NIST IR-4 (Incident Handling), NIST SI-2 (Flaw Remediation), NIST SI-3 (Malicious Code Protection), NIST AC-2 (Account Management), CIS 2.3 (Address Unauthorized Software), CIS 5.3 (Disable Dormant Accounts)

Compensating: Without enterprise MDM remote wipe: instruct affected users to manually uninstall flagged APKs via Android Settings > Apps, then verify removal with `adb shell pm list packages -3` re-run — confirm the package no longer appears. For credential revocation without SSO: enumerate all corporate SaaS apps (email, VPN, collaboration tools) and force token invalidation manually per-platform (e.g., Google Workspace Admin Console > User > 'Sign out all sessions'; Microsoft 365 Admin > Revoke refresh tokens via `Revoke-AzureADUserAllRefreshToken`). Deploy FEMITBOT domain blocklist as a Pi-hole blocklist entry or as static DNS RPZ entries on the corporate resolver; use CTM360's published IOC feed or ingest into pfSense/OPNsense alias lists for automated blocking.

Evidence: Before eradication, forensically preserve: a full SHA-256 hash of each sideloaded APK (`adb shell pm path ` then hash the APK file) for submission to VirusTotal and retention as evidence; decompiled APK manifest (`apktool d `) capturing declared permissions — FEMITBOT-distributed malware would likely request ACCESSIBILITY_SERVICE, READ_CONTACTS, RECEIVE_SMS, and BIND_DEVICE_ADMIN, which are high-confidence indicators of credential harvesting and persistence capability; a list of all accounts authenticated on the device during the suspected compromise window, exported from each SaaS platform's access log (e.g., Google Workspace Admin > Reports > Login Activity filtered by device ID or user); network capture (Wireshark or `tcpdump` via ADB) of any C2 beaconing prior to APK removal, specifically HTTPS POST traffic to FEMITBOT backend infrastructure.

Recovery — Confirm APK removal and validate no persistence mechanisms remain on affected devices (review scheduled jobs, accessibility service grants, device admin permissions). Monitor affected accounts for unauthorized access or transaction activity post-remediation.

NIST Phase: Recovery

Reference: NIST 800-61r3 §3.5 — Recovery

Controls: NIST IR-4 (Incident Handling), 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)

Compensating: Without EDR: verify persistence removal on Android using `adb shell settings get secure accessibility_enabled` and `adb shell settings get secure enabled_accessibility_services` — any entry referencing the removed APK's package name indicates incomplete eradication. Check device admin grants via `adb shell dpm

list-owners` and `adb shell dumpsys device_policy`. For scheduled job persistence, inspect `adb shell dumpsys jobscheduler` for any registered jobs from the sideloaded package namespace. Monitor accounts using free SIEM alternatives: forward Google Workspace login audit logs to a self-hosted Graylog or Elastic Stack (free tier) instance and create an alert for logins from new countries or IPs not matching the user's historical baseline in the 30 days post-remediation.

Evidence: Capture for recovery validation: post-remediation output of `adb shell dumpsys accessibility` confirming no residual accessibility service grants from the removed APK; device admin policy dump (`adb shell dumpsys device_policy`) confirming no lingering Device Administrator grants that would allow the APK to resist uninstall; account access logs from all corporate SaaS platforms for the affected user(s) covering the 72-hour window post-credential-reset, specifically flagging any OAuth token grants or API key issuances that may have occurred during the compromise window before revocation; MoonPay or other crypto platform transaction logs if the user entered wallet or payment information into a FEMITBOT Mini App — these should be requested from the user and preserved as potential fraud evidence.

Post-Incident — Review MDM policy enforcement for APK sideloading controls (ensure 'Unknown Sources' is disabled on all managed Android devices). Assess whether brand monitoring is in place to detect impersonation of your organization across Telegram and similar platforms. File a brand abuse report with Telegram if your organization's brand appears in active FEMITBOT campaigns.

NIST Phase: Post Incident

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

Controls: NIST IR-4 (Incident Handling), NIST IR-8 (Incident Response Plan), NIST SI-5 (Security Alerts, Advisories, and Directives), CIS 7.1 (Establish and Maintain a Vulnerability Management Process), CIS 7.2 (Establish and Maintain a Remediation Process), CIS 2.2 (Ensure Authorized Software is Currently Supported)

Compensating: Without a commercial brand monitoring service: set up free Google Alerts for your organization's brand name combined with terms like 'Telegram bot', 'crypto investment', 'earn rewards', and 'Mini App'; supplement with manual weekly searches on Telegram itself using the public search for your brand name + common FEMITBOT lure keywords ('airdrop', 'mining', 'referral bonus'). For MDM policy gap remediation on a budget: use Android Enterprise Zero-Touch Enrollment (free via Google) to enforce `DISALLOW_INSTALL_UNKNOWN_SOURCES` as a device policy across all managed Android enrollments. Document the brand abuse report submission to Telegram (via https://telegram.org/support) with timestamps and case numbers for regulatory/legal record-keeping.

Evidence: For the lessons-learned record, preserve: the full MDM policy configuration export showing the pre-incident state of 'Unknown Sources' enforcement, documenting the policy gap that permitted sideloading; a chronological timeline of FEMITBOT campaign activity affecting your organization's brand (drawn from CTM360 report, Telegram abuse report responses, and internal MDM/DNS log review), formatted per NIST 800-61r3 §4 incident documentation requirements; the IOC list ingested during this incident (FEMITBOT domains, APK hashes, bot usernames) archived in your threat intelligence platform or a versioned flat file for future detection rule updates; documentation of any impersonated brand variants observed (e.g., 'AppleInvestBot', 'DisneyEarnApp') specific to your organization for submission to legal/brand protection teams.

Detection Guidance

Primary detection surface is Android MDM and endpoint telemetry. Look for: (1) APK installations from sources other than Google Play Store on managed devices; (2) apps requesting accessibility service permissions or device administrator rights shortly after install; (3) DNS or proxy requests to domains matching the FEMITBOT API signature, CTM360's report contains the specific API pattern and associated phishing domain list (request directly from ctm360.com/reports). Behavioral indicators include Telegram bot interactions followed by external URL redirects, and WebView sessions rendering pages that request cryptocurrency wallet credentials or seed phrases. For threat hunting, pivot on T1566.003 by reviewing Telegram network traffic for bot-initiated external domain redirects, and on T1583.001 by clustering phishing domains sharing the FEMITBOT backend API

response structure. No public SIEM rule set is confirmed available at this time; detection engineering should build rules against the CTM360-published IOCs once obtained.

Indicators of Compromise

Type	Value	Context	Confidence
DOMAIN	`See CTM360 FEMITBOT report for full domain list`	FEMITBOT phishing domains share a consistent backend API signature; full IOC list is available in the CTM360 published report at ctm360.com/reports/femitbot-telegram-mini-apps-fraud-campaigns	MEDIUM

Framework Mappings

MITRE-ATTACK

- **T1456** — Drive-By Compromise
- **T1566.003** — Spearphishing via Service
- **T1583.001** — Domains
- **T1204.002** — Malicious File
- **T1566** — Phishing
- **T1102.001** — Dead Drop Resolver
- **T1588.002** — Tool
- **T1036.005** — Match Legitimate Resource Name or Location
- **T1071.001** — Web Protocols
- **T1496** — Resource Hijacking
- **T1608.004** — Drive-by Target
- **T1036** — Masquerading
- **T1056** — Input Capture

NIST-800-53R5

- **AT-2** — Literacy Training and Awareness
- **CA-7** — Continuous Monitoring
- **SC-7** — Boundary Protection
- **SI-3** — Malicious Code Protection
- **SI-4** — System Monitoring
- **SI-8** — Spam Protection

CIS-V8

- **14.2** — Train Workforce Members to Recognize Social Engineering Attacks

HIPAA-SECURITY

- **164.308(a)(5)(i)** — Security Awareness and Training

ISO-27001-2022

- **A.5.34** — Privacy and protection of personal information

MITRE ATT&CK Mapping

Technique ID	Technique Name	Tactic
T1456	Drive-By Compromise	Initial-Access
T1566.003	Spearphishing via Service	Initial-Access
T1583.001	Domains	Resource-Development
T1204.002	Malicious File	Execution
T1566	Phishing	Initial-Access
T1102.001	Dead Drop Resolver	Command-And-Control
T1588.002	Tool	Resource-Development
T1036.005	Match Legitimate Resource Name or Location	Defense-Evasion
T1071.001	Web Protocols	Command-And-Control
T1496	Resource Hijacking	Impact
T1608.004	Drive-by Target	Resource-Development
T1036	Masquerading	Defense-Evasion
T1056	Input Capture	Collection

Sources

Source	URL	Tier
Security News	https://www.bleepingcomputer.com/news/security/telegram-mini-apps-a...	T3
Telegram Mini Apps	https://core.telegram.org/bots/webapps	T3
FEMITBOT: Telegram Mini Apps Fraud Report | CTM360	https://www.ctm360.com/reports/femitbot-telegram-mini-apps-fraud-ca...	T3

Generated 2026-05-03 18:28 UTC by TJS Security Command Center