INTELLIGENCE BRIEFING
Security Command Center
TLP:CLEAR
2026-05-30 06:23 UTC

ChatGPhish: ChatGPT Markdown Renderer Exploited for Prompt Injection and Phishing Redirection

SECURITY ANALYSIS | HIGH | CVSS 7.5

SCC Item ID	SCC-STY-2026-0161
Type	Security Analysis
Severity	HIGH
CVSS Base Score	7.5
Affected Products	OpenAI ChatGPT (chatgpt.com), web interface, Markdown rendering layer
Published	2026-05-29T14:07:12
Discovery Source	Rss

Executive Summary

Permiso Security disclosed ChatGPhish, a technique that weaponizes ChatGPT's Markdown rendering layer to inject malicious links and images into AI-generated summaries, redirecting users to attacker-controlled phishing infrastructure without compromising OpenAI's backend. Because users implicitly trust content that appears to originate from an AI assistant, this attack surface turns ChatGPT into an unwitting phishing delivery mechanism with no visible indicators of compromise in the conversational interface. For enterprises that have standardized on ChatGPT for research, summarization, or workflow automation, this signals that AI-generated output can no longer be treated as inherently safe, the AI's credibility becomes the attacker's most valuable asset.

Technical Analysis

ChatGPhish, disclosed by Permiso Security, exploits the interaction between ChatGPT's web interface Markdown renderer and the model's summarization behavior. The attack chain begins with an adversary publishing or hosting web content containing adversarially crafted Markdown, specifically, embedded URLs and image tags designed to pass through the model's summarization pipeline intact. When a ChatGPT user asks the model to summarize or interact with that content, the model faithfully incorporates the Markdown into its response. The chatgpt.com renderer then processes and renders that output, resolving image sources and displaying hyperlinks without sufficient validation against the model's own origin context. The rendered result presents weaponized links and pixel-tracking image tags as though they are part of the AI's native response, visually indistinguishable from legitimate AI output.

The attack maps cleanly to four CWEs: CWE-79 (Cross-Site Scripting via improper output neutralization), CWE-116 (improper encoding or escaping of output), CWE-20 (improper input validation at the content ingestion layer), and CWE-601 (open redirect enabling phishing redirection). No backend compromise is required. The attacker only needs to control the source content the model is asked to summarize.

From an ATT&CK perspective, the technique chains T1598.003 (Spearphishing Link via Service) and T1566.002 (Spearphishing via Link) with T1189 (Drive-by Compromise) and T1204.001 (User Execution: Malicious Link). T1056.003 (Web Portal Capture) and T1071.001 (Application Layer Protocol: Web Protocols) apply to the downstream credential harvesting and exfiltration phases. T1190 (Exploit Public-Facing Application) accurately characterizes the abuse of chatgpt.com's rendering layer as the initial access vector.

The defensive gap this exploits is architectural: enterprises that deployed ChatGPT as a trusted productivity tool built no controls around its output pipeline. Users are conditioned to trust AI-generated summaries as neutral and authoritative, which eliminates the skepticism that might otherwise cause them to hover over a link before clicking. The technique also bypasses conventional phishing defenses, email gateways, link scanners, and DMARC controls, because the delivery mechanism is a web browser session with a trusted AI service. Source quality from Permiso Security is credible; the disclosure methodology and CWE mapping indicate a structured research process. The referenced OpenAI security patches indicate active effort to address rendering-layer vulnerabilities as a class.

Action Checklist

1. Step 1: Assess exposure, inventory all enterprise use of ChatGPT's web interface (chatgpt.com), including Teams/Slack integrations, browser-based use by analysts and knowledge workers, and any workflows where ChatGPT is asked to summarize external URLs or web content.
2. Step 2: Review controls, verify that web proxy policies (aligned with NIST AC-4, Information Flow Enforcement) block or alert on outbound requests to newly registered or uncategorized domains initiated from browser sessions; confirm endpoint DLP tools flag credential submission to unexpected domains.
3. Step 3: Update threat model, add 'AI-output phishing redirection' as an explicit attack vector in your threat register; map to T1598.003, T1566.002, and T1189; flag ChatGPT summarization of untrusted external content as a high-risk use pattern requiring user awareness guidance.
4. Step 4: Issue user awareness guidance, brief all ChatGPT users, particularly analysts who summarize external research, that AI-rendered links can be adversarially crafted; instruct users to verify destination URLs before clicking any link in an AI-generated response, regardless of how authoritative the context appears (supports NIST AT-2, Literacy Training and Awareness).
5. Step 5: Enforce MFA and session controls on downstream targets, ensure that any sensitive application a user might reach via a phishing redirect enforces MFA (CIS 6.3, CIS 6.4) and session anomaly detection (NIST AC-12, Session Termination; NIST SI-4, System Monitoring); a successful redirect does not guarantee credential capture if MFA is enforced.
6. Step 6: Monitor developments, track OpenAI's security advisories and the Permiso Security disclosure for patch confirmation; review OpenAI's vulnerability disclosure process (https://help.openai.com/en/articles/6653653-how-to-report-security-vulnerabilities-to-openai) for patch status updates and any evidence of active exploitation in the wild.

IR / Forensic Enrichment

Triage Priority	URGENT
Escalation Criteria	Escalate immediately to incident commander and legal/privacy counsel if proxy logs or IdP authentication records show a user followed a ChatGPT-rendered link to an attacker-controlled domain and subsequently submitted credentials or if any downstream sensitive application shows an authentication event from an anomalous ASN or device fingerprint coinciding with a chatgpt.com browser session, as this constitutes probable credential compromise and may trigger breach notification obligations under GDPR, CCPA, or HIPAA depending on the data classification of the accessed application.
Recovery Notes	Because ChatGPhish does not compromise OpenAI's backend and leaves no persistent malware on the chatgpt.com platform itself, recovery focuses entirely on downstream impact: any user account that may have submitted credentials to a phishing domain must be treated as compromised — force password reset and session revocation on all associated applications before reinstating access. Monitor IdP and SaaS authentication logs for the 30 days following awareness guidance issuance, specifically watching for impossible travel, new device registrations, or privilege escalation events that could indicate a delayed attacker use of harvested credentials. Recovery is complete when OpenAI confirms a fix to the Markdown rendering layer that prevents injection of arbitrary external links; until that confirmation, the compensating controls from Steps 2 and 5 remain active and the threat register entry stays open.
Forensic Artifacts	Web proxy or NGFW egress logs: filter for HTTP 301/302 redirect chains where 'Referer: https://chatgpt.com' and the destination domain was registered within the past 90 days — this is the direct network fingerprint of a ChatGPhish redirect event | Browser history databases: Chrome '%LOCALAPPDATA%\Google\Chrome\User Data\Default\History' and Firefox '%APPDATA%\Mozilla\Firefox\Profiles*.default\places.sqlite' — query for URLs visited immediately after chatgpt.com sessions to reconstruct whether a user followed a Markdown-injected link to an attacker domain | IdP sign-in logs (Azure AD, Okta, Google Workspace): extract authentication events occurring within 5 minutes of a chatgpt.com browser session, filtering for new device fingerprints, new ASNs, or geolocations inconsistent with the user's 30-day baseline — these indicate post-redirect credential submission and successful authentication to a targeted application | DNS resolver query logs: search for resolution of attacker-controlled domains from analyst workstations during chatgpt.com session timeframes — DNS queries for the redirect destination domain will appear even if the user did not complete the credential submission, confirming that the Markdown-injected link was rendered and the beacon fired | Browser network capture or SSL inspection logs: if SSL inspection is deployed, capture the full request-response chain for sessions originating from chatgpt.com — look for image src or anchor href attributes in ChatGPT response bodies pointing to external domains, as these are the Markdown injection payloads; the image beacon GET request to an attacker domain will appear as a zero-user-interaction outbound request within milliseconds of the ChatGPT response rendering

Per-Action IR Details

Step 1: Assess exposure — inventory all enterprise use of ChatGPT's web interface (chatgpt.com), including Teams/Slack integrations, browser-based use by analysts and knowledge workers, and any workflows where ChatGPT is asked to summarize external URLs or web content.

NIST Phase: Preparation

Reference: NIST 800-61r3 §2 — Preparation: establishing visibility into systems and workflows that constitute the attack surface prior to incident declaration

Controls: NIST AC-20 (Use of External Systems) — governs authorized use of third-party AI platforms like chatgpt.com, NIST CM-8 (System Component Inventory) — requires asset inventory inclusive of SaaS and browser-based tools, CIS 1.1 (Establish and Maintain Detailed Enterprise Asset Inventory) — extend scope to include SaaS endpoints including chatgpt.com browser sessions, CIS 2.1 (Establish and Maintain a Software Inventory) — catalog browser extensions and integrations that relay content to or from ChatGPT

Compensating: Run 'Get-MpThreatDetection' or review proxy logs manually if no SIEM is available. Query DNS resolver logs or firewall logs for outbound connections to chatgpt.com; use a simple grep or PowerShell: 'Select-String -Path -Pattern "chatgpt\.com"' to enumerate active users. For Teams/Slack integrations, pull webhook configuration lists from admin consoles directly — no tooling required. Document all workflows where analysts paste external URLs into ChatGPT for summarization, as these are the primary ChatGPhish delivery paths.

Evidence: Before scoping begins, preserve proxy/firewall egress logs showing outbound sessions to chatgpt.com for the prior 30 days — these establish baseline usage patterns and will later be differenced against post-incident traffic to identify anomalous redirect chains. Capture browser history exports from analyst workstations (Chrome: '%LOCALAPPDATA%\Google\Chrome\User Data\Default\History'; Firefox: '%APPDATA%\Mozilla\Firefox\Profiles*.default\places.sqlite') to reconstruct which users interacted with ChatGPT-rendered content containing external links.

Step 2: Review controls — verify that web proxy policies (aligned with NIST AC-4, Information Flow Enforcement) block or alert on outbound requests to newly registered or uncategorized domains initiated from browser sessions; confirm endpoint DLP tools flag credential submission to unexpected domains.

NIST Phase: Preparation

Reference: NIST 800-61r3 §2 — Preparation: validating that detection and prevention controls are in place before exploitation is confirmed

Controls: NIST AC-4 (Information Flow Enforcement) — enforce proxy policy rules that intercept browser-initiated outbound flows to uncategorized or newly registered domains, specifically those reachable via links rendered inside chatgpt.com sessions, NIST SI-4 (System Monitoring) — validate that proxy or NGFW alerting fires on click-through redirects from chatgpt.com to non-allowlisted external domains, NIST SC-7 (Boundary Protection) — confirm that egress filtering captures redirects originating from browser sessions on the chatgpt.com origin, CIS 4.4 (Implement and Manage a Firewall on Servers) — verify server-side firewall rules deny outbound connections to uncategorized domains for hosts where analysts operate, CIS 7.1 (Establish and Maintain a Vulnerability Management Process) — include ChatGPhish-class AI-output redirection in the documented threat surface reviewed during vulnerability management cycles

Compensating: Without a commercial proxy with domain-age categorization, deploy Pi-hole or a local Squid proxy configured with a blocklist seeded from domains registered within the last 30 days using the Newly Observed Domains feed from Bambenek Consulting (free) or the WHOIS-based feeds from abuse.ch. Write a daily cron job that pulls the Quad9 newly registered domains blocklist and reloads Squid: 'curl -s https://raw.githubusercontent.com/nicowillis/newly-registered-domains/main/domains.txt >> /etc/squid/blocklist.txt && squid -k reconfigure'. For DLP, use browser Group Policy (Chrome ADMX) to block form submission on domains not in a pre-approved allowlist.

Evidence: Pull web proxy logs filtered for HTTP 301/302 redirect chains where the referrer header is 'chatgpt.com' and the destination is a domain registered within the past 90 days — this is the precise network artifact ChatGPhish redirection produces. Capture SSL inspection logs if available, as the malicious Markdown-injected image beacon or link may exfiltrate session context via URL parameters on the redirect. Preserve DLP alert logs showing credential-field submissions to non-allowlisted domains initiated from browser processes active during chatgpt.com sessions.

Step 3: Update threat model — add 'AI-output phishing redirection' as an explicit attack vector in your threat register; map to T1598.003, T1566.002, and T1189; flag ChatGPT summarization of untrusted external content as a high-risk use pattern requiring user awareness guidance.

NIST Phase: Preparation

Reference: NIST 800-61r3 §2 — Preparation: updating the threat model and detection baseline to reflect newly disclosed attack techniques before active exploitation is observed

Controls: NIST RA-3 (Risk Assessment) — formally document ChatGPhish as a threat scenario with likelihood and impact ratings in the organizational risk register, NIST PM-16 (Threat Awareness Program) — incorporate ChatGPhish and the MITRE ATT&CK mappings (T1598.003, T1566.002, T1189) into the enterprise threat awareness program, NIST SI-5 (Security Alerts, Advisories, and Directives) — process the Permiso Security ChatGPhish disclosure as a formal security advisory requiring documented response, CIS 7.1 (Establish and Maintain a Vulnerability Management Process) — add AI-output phishing redirection as a tracked attack pattern with assigned remediation ownership

Compensating: Create a Sigma rule targeting proxy logs for the ChatGPhish redirect pattern: match on 'cs-referer' containing 'chatgpt.com' AND 'cs-host' matching newly registered domain regex (TLD + domain registered <90 days). Publish the rule to your detection backlog. For MITRE mapping, use the ATT&CK Navigator (free, browser-based) to annotate T1598.003 (Spearphishing Link via Service), T1566.002 (Spearphishing Link), and T1189 (Drive-by Compromise) with a ChatGPhish-specific note. Document the threat register update in a shared wiki or ticketing system with the Permiso disclosure URL as the source reference.

Evidence: Before updating the threat model, retrieve the Permiso Security ChatGPhish disclosure and any associated IOCs (domain patterns, Markdown injection syntax, image beacon URL structures) and preserve them as source evidence for the threat register entry. Screenshot or archive the disclosure page in case it is updated or retracted. No host-based forensic artifacts exist at this stage since this is pre-incident threat modeling.

Step 4: Issue user awareness guidance — brief all ChatGPT users, particularly analysts who summarize external research, that AI-rendered links can be adversarially crafted; instruct users to verify destination URLs before clicking any link in an AI-generated response, regardless of how authoritative the context appears (supports NIST AT-2, Literacy Training and Awareness).

NIST Phase: Preparation

Reference: NIST 800-61r3 §2 — Preparation: training and awareness activities that reduce the human-layer attack surface specific to this technique

Controls: NIST AT-2 (Literacy Training and Awareness) — issue ChatGPhish-specific awareness communication covering the Markdown injection mechanism, the trust exploitation model, and URL verification behavior, NIST AT-3 (Role-Based Training) — provide targeted briefing to analyst and knowledge worker roles who routinely use ChatGPT to summarize external URLs, as these users represent the highest-exposure population for ChatGPhish delivery, NIST AC-20 (Use of External Systems) — update acceptable use policy for chatgpt.com to explicitly prohibit summarizing untrusted or attacker-influenced external content without URL pre-verification, CIS 6.1 (Establish an Access Granting Process) — as part of access provisioning for ChatGPT use, include acknowledgment of AI-output link verification requirements

Compensating: Draft a one-page advisory using the ChatGPhish technique description: explain that a malicious prompt injected into a webpage that an analyst asks ChatGPT to summarize can cause ChatGPT to render attacker-controlled links that appear contextually legitimate. Distribute via email and pin in Slack/Teams channels used by the analyst population. Include a demonstration screenshot (hypothetical, labeled as such) showing how a benign-looking AI summary can contain a hyperlinked word pointing to an attacker domain. No tooling required — human communication is the control here.

Evidence: Capture pre-briefing baseline by reviewing any existing phishing simulation reports or security awareness training completion records — these establish whether the analyst population had prior exposure to AI-delivery phishing concepts. After briefing, log distribution records (email send receipts, Slack post timestamps, Teams channel post) as evidence that notification was issued. If any user reports having already clicked a suspicious link from a ChatGPT response, treat that report as an incident trigger and escalate immediately to detection_analysis phase.

Step 5: Enforce MFA and session controls on downstream targets — ensure that any sensitive application a user might reach via a phishing redirect enforces MFA (CIS 6.3, CIS 6.4) and session anomaly detection (NIST AC-12, Session Termination; NIST SI-4, System Monitoring); a successful redirect does not guarantee credential capture if MFA is enforced.

NIST Phase: Containment

Reference: NIST 800-61r3 §3.3 — Containment Strategy: implementing controls that limit the impact of a successful ChatGPhish redirect even when prevention fails

Controls: NIST AC-12 (Session Termination) — configure session timeout and re-authentication requirements on sensitive downstream applications so that sessions established via a phishing redirect have a reduced window for attacker exploitation, NIST SI-4 (System Monitoring) — enable login anomaly detection (impossible travel, new device, new ASN) on identity providers and sensitive SaaS applications that a redirected user might authenticate to, NIST IA-2 (Identification and Authentication — Organizational Users) — enforce phishing-resistant MFA (FIDO2/WebAuthn preferred) on all applications reachable via the redirect chain, as SMS-based MFA is bypassable via real-time phishing proxies, CIS 6.3 (Require MFA for Externally-Exposed Applications) — audit all externally-exposed applications for MFA enforcement, prioritizing those accessible to analyst populations who use ChatGPT, CIS 6.4 (Require MFA for Remote Network Access) — verify VPN and remote access gateways enforce MFA, as a ChatGPhish redirect could target VPN credential harvesting pages, CIS 6.5 (Require MFA for Administrative Access) — confirm privileged accounts used by analysts who operate ChatGPT enforce MFA unconditionally

Compensating: For teams without a commercial IdP with anomaly detection, enable Azure AD or Okta free-tier sign-in risk policies if available, or implement manual review of authentication logs daily. Use Microsoft Authenticator or Google Authenticator (free) for TOTP-based MFA if FIDO2 is unavailable. For session monitoring without EDR, write a PowerShell script that queries Azure AD sign-in logs via Graph API and alerts on logins from ASNs not seen in the prior 30 days: 'Get-MgAuditLogSignIn -Filter "riskState eq riskyAndConfirmed"'. For non-Azure environments, deploy Authelia (open source) as an MFA reverse proxy in front of sensitive internal applications.

Evidence: Before enforcing session controls, pull the current authentication logs from all sensitive downstream applications (IdP logs, VPN authentication logs, SaaS SSO logs) and establish a 30-day baseline of normal login ASNs, geolocations, user agents, and session durations. This baseline is the forensic reference point — any post-redirect authentication that deviates from it (new ASN, new device fingerprint, session duration mismatch) becomes a high-confidence indicator of successful ChatGPhish credential capture. Preserve these baseline exports before making any configuration changes that might alter log retention.

Step 6: Monitor developments — track OpenAI's security advisories and the Permiso Security disclosure for patch confirmation, additional technical indicators, and any evidence of active exploitation in the wild; review OpenAI's vulnerability disclosure process (https://help.openai.com/en/articles/6653653-how-to-report-security-vulnerabilities-to-openai) for patch status updates.

NIST Phase: Post Incident

Reference: NIST 800-61r3 §4 — Post-Incident Activity: intelligence sharing, lessons learned, and sustained monitoring for evolving threat indicators after initial response actions are taken

Controls: NIST SI-5 (Security Alerts, Advisories, and Directives) — establish a tracked watch item for OpenAI security advisories and Permiso Security updates related to ChatGPhish; assign an owner and review cadence, NIST RA-10 (Threat Intelligence) — integrate ChatGPhish IOCs (Markdown injection patterns, redirect domain characteristics, image beacon URL structures) into the threat intelligence workflow for continuous update, NIST IR-6 (Incident Reporting) — if active exploitation evidence is identified internally, initiate formal incident reporting per organizational policy and applicable regulatory requirements, CIS 7.1 (Establish and Maintain a Vulnerability Management Process) — include ChatGPhish patch status as a tracked open item in the vulnerability management process, with a defined escalation path if OpenAI does not issue a remediation within a defined SLA window, CIS 7.2 (Establish and Maintain a Remediation Process) — document the current unpatched status of the ChatGPT Markdown renderer vulnerability and assign a risk acceptance or compensating control record until a vendor fix is confirmed

Compensating: Subscribe to Permiso Security's blog RSS feed and OpenAI's security changelog via a free RSS reader (Feedly free tier). Set a Google Alert for 'ChatGPhish' and 'ChatGPT Markdown injection' to catch third-party reporting. Create a weekly 15-minute calendar block for a designated team member to check patch status and update the threat register entry. If active exploitation IOCs (specific phishing domains, Markdown payload syntax) are published by Permiso or the community, import them immediately into Pi-hole or Squid blocklists using the compensating control described in Step 2. Note: the OpenAI URL provided in the original step should be validated by the reader before use, as URL accuracy cannot be confirmed from training data alone.

Evidence: Maintain a running log of all ChatGPhish-related intelligence updates with timestamps, source, and disposition (actioned/noted/no change required) — this log constitutes the post-incident monitoring record required under NIST 800-61r3 §4. If Permiso publishes additional technical indicators (specific Markdown injection payloads,

image beacon domain patterns, redirect chain signatures), archive them immediately and cross-reference against proxy and browser history logs from the prior 30 days to determine whether any enterprise users were exposed before awareness guidance was issued.

Detection Guidance

There are no network-layer or endpoint IOCs native to this technique during the AI interaction phase, the attack occurs within a legitimate HTTPS session to chatgpt.com. Detection must focus on the downstream effects of successful phishing redirection.

Log sources and hunt priorities:

1. Web proxy / DNS logs: Hunt for outbound connections to newly registered domains (less than 30 days old), domains with high entropy names, or domains resolving to hosting infrastructure inconsistent with the claimed service, particularly those initiated from browser sessions immediately following ChatGPT interactions. NIST AU-2 (Event Logging) and AU-6 (Audit Record Review) apply here.

2. Credential submission anomalies: Alert on users submitting credentials to domains not in the enterprise approved application inventory. This supports detection of T1056.003 (Web Portal Capture) downstream of the redirect.

3. Image beacon / tracking pixel detection: Some next-generation proxy solutions can detect single-pixel image loads from off-network, uncategorized domains, a technique used by attackers to confirm target reachability before serving phishing content. Enable logging on these events if your proxy supports it.

4. User-reported suspicious AI output: Establish a low-friction internal reporting channel for users who observe unexpected links or images in ChatGPT responses. Behavioral telemetry from users is currently the most reliable early warning signal for this technique.

5. Browser isolation: Organizations with browser isolation technologies applied to ChatGPT sessions can prevent rendered Markdown from resolving external URLs in the user's native browser context, consider this a near-term compensating control.

D3FEND countermeasures applicable: D3-PBWSAM (Proxy-based Web Server Access Mediation) to intercept and inspect outbound redirects; D3-UAP (User Account Permissions) to restrict access to sensitive applications from unmanaged browser contexts; D3-LAM (Local Account Monitoring) to detect anomalous authentication events on downstream systems following a potential redirect.

Indicators of Compromise

Type	Value	Context	Confidence
TOOL	`Pending — refer to Permiso Security's ChatGPhish disclosure for published indicators`	Permiso Security's disclosure is expected to include adversarially crafted Markdown payload patterns and attacker-controlled domain characteristics; actual IOC values were not present in the source material available for this analysis	LOW

Framework Mappings

MITRE-ATTACK

- **T1059** — Command and Scripting Interpreter
- **T1204.001** — Malicious Link
- **T1598.003** — Spearphishing Link
- **T1190** — Exploit Public-Facing Application
- **T1189** — Drive-by Compromise
- **T1566.002** — Spearphishing Link
- **T1566** — Phishing
- **T1056.003** — Web Portal Capture
- **T1071.001** — Web Protocols

NIST-800-53R5

- **CM-7** — Least Functionality
- **SI-3** — Malicious Code Protection
- **SI-4** — System Monitoring
- **SI-7** — Software, Firmware, and Information Integrity
- **CA-8** — Penetration Testing
- **RA-5** — Vulnerability Monitoring and Scanning
- **SC-7** — Boundary Protection
- **SI-2** — Flaw Remediation
- **AT-2** — Literacy Training and Awareness
- **SI-8** — Spam Protection
- **CA-7** — Continuous Monitoring
- **SI-10** — Information Input Validation

OWASP-TOP10-2021

- **A03:2021** — Injection

CIS-V8

- **16.10** — Apply Secure Design Principles in Application Architectures
- **6.3** — Require MFA for Externally-Exposed Applications
- **14.2** — Train Workforce Members to Recognize Social Engineering Attacks

ISO-27001-2022

- **A.8.28** — Secure coding
- **A.8.26** — Application security requirements
- **A.8.8** — Management of technical vulnerabilities
- **A.5.34** — Privacy and protection of personal information

HIPAA-SECURITY

- **164.312(d)** — Person or Entity Authentication
- **164.308(a)(5)(i)** — Security Awareness and Training

SOC2-TSC

- **CC6.1** — Logical access security software, infrastructure, and architectures

MITRE ATT&CK Mapping

Technique ID	Technique Name	Tactic
T1059	Command and Scripting Interpreter	Execution
T1204.001	Malicious Link	Execution
T1598.003	Spearphishing Link	Reconnaissance
T1190	Exploit Public-Facing Application	Initial-Access
T1189	Drive-by Compromise	Initial-Access
T1566.002	Spearphishing Link	Initial-Access
T1566	Phishing	Initial-Access
T1056.003	Web Portal Capture	Collection
T1071.001	Web Protocols	Command-And-Control

Sources

Source	URL	Tier
Security News	https://thehackernews.com/2026/05/chatgphish-vulnerability-turns-ch...	T3
How to Report Security Vulnerabilities to OpenAI	https://help.openai.com/en/articles/6653653-how-to-report-security-...	T1
Security vulnerability in chatGPT : r/OpenAI - Reddit	https://www.reddit.com/r/OpenAI/comments/1plp2bj/security_vulnerabi...	T3
OpenAI Patches ChatGPT Data Exfiltration Flaw and Codex GitHub ...	https://thehackernews.com/2026/03/openai-patches-chatgpt-data.html	T3
ChatGPT Security Risks: All You Need to Know - SentinelOne	https://www.sentinelone.com/cybersecurity-101/data-and-ai/chatgpt-s...	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 2026-05-30 06:23 UTC by TJS Security Command Center