INTELLIGENCE BRIEFING
Security Command Center
TLP:CLEAR
2026-06-04 14:10 UTC

SEO-Poisoned Fake Open-Source Tool Sites Deliver Remus Stealer and AnimateClipper via TDS Infrastructure

THREAT CAMPAIGN | **HIGH** | CVSS 7.5

SCC Item ID	SCC-CAM-2026-0409
Type	Threat Campaign
Severity	HIGH
CVSS Base Score	7.5
Affected Products	Developers and security analysts downloading Ghidra, dnSpy, or SpiderFoot via search; cryptocurrency wallet browser extensions; 2FA tools; password managers; 20+ blockchain ecosystems
Published	2026-06-04T05:51:28
Discovery Source	Rss

Executive Summary

Financially motivated operators are running a large-scale campaign using fake websites that impersonate trusted open-source security tools, Ghidra, dnSpy, and SpiderFoot, and rank them at the top of Google search results through SEO manipulation. Developers and security analysts who download from these sites receive credential-stealing malware (Remus Stealer), a cryptocurrency clipboard hijacker targeting 20+ blockchain networks (AnimateClipper), or a loader (SessionGate) selected by a profiling layer that filters victims before payload delivery. Organizations with developers, security engineers, or analysts who search for and download open-source tooling are directly exposed; secondary risk extends to any systems those users access, including internal credentials, session tokens, cryptocurrency assets, and password manager contents.

Technical Analysis

Active since September 2025 and weaponized for direct malware delivery in January 2026, this campaign operates typosquatted or lookalike domains impersonating Ghidra, dnSpy, and SpiderFoot. SEO poisoning elevates these sites above legitimate results. The download flow routes through a CloudFront-hosted JavaScript intermediary that forwards requests to a Traffic Distribution System (TDS). The TDS profiles visitors, likely via browser fingerprinting, geolocation, and referrer analysis, and gates payload delivery, frustrating automated sandbox analysis and bulk detection. Three payloads are in rotation: Remus Stealer (credential and browser

session theft targeting saved passwords, cookies, and tokens), AnimateClipper (clipboard hijacker supporting 20+ blockchain address formats, silently replacing copied wallet addresses), and SessionGate (loader, likely for secondary-stage deployment). Relevant CWEs: CWE-693 (Protection Mechanism Failure, TDS-based sandbox evasion), CWE-494 (Download of Code Without Integrity Check, no signature verification on downloaded binaries), CWE-345 (Insufficient Verification of Data Authenticity, SEO-manipulated source trust). MITRE ATT&CK techniques include T1608.001 (Stage Capabilities: Upload Malware), T1036.005 (Masquerading: Match Legitimate Name or Location), T1614.001 (System Location Discovery, TDS geofencing), T1539 (Steal Web Session Cookie), T1555.003 (Credentials from Web Browsers), T1176 (Browser Extensions, wallet extension targeting), T1071.001 (Application Layer Protocol: Web Protocols, C2 over HTTP/S), and T1027 (Obfuscated Files or Information). No CVE is assigned. No vendor patch applies, this is a distribution infrastructure threat, not a software vulnerability. Mitigation is behavioral and procedural.

Action Checklist

1. Containment: Block known malicious domains impersonating Ghidra (ghidra.re and variants), dnSpy, and SpiderFoot at DNS and web proxy layers immediately. Alert the security team to any recent downloads of these tools from non-official sources (official sources: ghidra.re via NSA GitHub, github.com/dnSpy/dnSpy, github.com/smicallef/spiderfoot). Isolate any endpoint where a suspicious installer was executed since September 2025. (Reference: NIST IR-4.1, Detection and Analysis; NIST AC-3, Access Enforcement)
2. Detection: Search EDR and proxy logs for downloads of installer binaries from domains other than the verified GitHub repositories for Ghidra, dnSpy, and SpiderFoot. Query SIEM for clipboard-monitoring process activity (T1115), anomalous browser data access by unsigned processes (T1555.003), and outbound connections to CloudFront distributions initiated by newly-dropped executables. Review DNS logs for resolution of lookalike domains containing 'ghidra', 'dnspy', or 'spiderfoot' outside github.com and official project domains. Flag any process spawned from a user's Downloads folder that attempts to read browser profile directories. No CVE-specific event IDs apply; detection is behavior-based. (Reference: NIST AU-2, Event Logging; NIST AU-6, Audit Record Review, Analysis, and Reporting)
3. Eradication: If compromise is confirmed, terminate and quarantine the malicious process, remove the dropped binary, and revoke all browser sessions on the affected endpoint (force sign-out of all browser profiles). Rotate all credentials stored in the browser password manager on that device. For any user who interacts with cryptocurrency wallets, treat all wallet addresses copied on the compromised host as potentially swapped; verify on-chain before any transaction. Remove any browser extensions installed after the suspected compromise date and audit remaining extensions against the known-good baseline. (Reference: NIST IA-4, Identifier Management; NIST AC-2, Account Management; CIS 2.1, Maintain a Software Inventory)
4. Recovery: After credential rotation and session revocation, re-image the affected endpoint or perform a verified clean restore. Re-download security tools exclusively from official GitHub repositories, verifying SHA-256 hashes published in official release notes before execution. Confirm no persistence mechanisms remain by reviewing startup configurations and system initialization. Monitor the affected user's accounts for 30 days for anomalous access patterns. Re-enable endpoint only after EDR confirms clean state. (Reference: NIST IR-4.2, Containment; NIST SI-7, Software, Firmware, and Information Integrity)
5. Post-Incident: This campaign exploited the absence of verified download sources and developer trust in search engine results. Implement a policy requiring all open-source tool downloads to be sourced exclusively from pinned, verified GitHub release pages with hash verification (CWE-494 gap closure).

Enforce application allowlisting or developer workstation controls to block unsigned executables from running from user download directories. Conduct awareness training specifically targeting developer and security analyst populations on SEO poisoning and supply chain download risks. (Reference: CIS 2.3, Address Unauthorized Software; NIST SI-7, Software, Firmware, and Information Integrity; NIST AC-3, Access Enforcement)

IR / Forensic Enrichment

Triage Priority	IMMEDIATE
Escalation Criteria	Escalate immediately to CISO and legal counsel if any confirmed compromise involves a host used for cryptocurrency transactions (AnimateClipper address-swapping scope), if browser-stored credentials include access to production systems or CI/CD pipelines (Remus Stealer blast radius into software supply chain), or if the affected user population includes contractors or third-party developers with privileged repository access that could extend this SEO-poisoning campaign into an internal supply chain incident requiring breach notification assessment.
Recovery Notes	After re-imaging, enforce a minimum 30-day enhanced monitoring period on the affected user's corporate accounts, OAuth tokens, and any downstream systems accessible via credentials that were stored in the compromised browser profile — Remus Stealer exfiltrates browser-saved passwords and session cookies, meaning attacker access to SaaS platforms, VPNs, and code repositories may persist beyond endpoint remediation if session tokens were not fully revoked. For any user who copied a cryptocurrency wallet address on the compromised host during the exposure window, conduct a full on-chain review of all transactions across the 20+ blockchain ecosystems targeted by AnimateClipper before declaring recovery complete. Verify re-downloaded Ghidra, dnSpy, and SpiderFoot binaries against SHA-256 hashes published in the official GitHub release assets — do not trust hashes published on any site other than the official GitHub release page for each project.

Forensic Artifacts	Browser SQLite databases on compromised developer workstations: Chrome Login Data (%LOCALAPPDATA%\Google\Chrome\User Data\Default\Login Data), Cookies, and Web Data files — Remus Stealer targets these specifically for credential and session token harvesting; copy with a forensic tool before any browser cleanup to preserve original timestamps and record carving potential. | Windows Clipboard history store (%LOCALAPPDATA%\Microsoft\Windows\Clipboard\) and Sysmon Event ID 1 records showing processes with clipboard API calls — AnimateClipper's core payload mechanism is cryptocurrency address substitution via clipboard intercept (T1115), and these artifacts establish which wallet addresses were potentially swapped and the time window of active clipboard monitoring. | Network proxy or firewall logs showing HTTP 302 redirect chains from SEO-ranked lookalike domains (hostnames matching 'ghidra', 'dnspy', or 'spiderfoot' outside github.com) to *.cloudfront.net endpoints — this documents the TDS infrastructure used by SessionGate to profile and route victims, and the CloudFront distribution IDs in these logs are the primary network IOCs for this campaign. | Sysmon Event ID 11 (FileCreate) and Event ID 1 (ProcessCreate) records for executables created in %USERPROFILE%\Downloads, %TEMP%, or %APPDATA%\Roaming between September 2025 and the investigation date, specifically any process spawned from those paths that subsequently accesses browser profile directories or makes outbound connections — this covers the full Remus Stealer and AnimateClipper initial execution and persistence chain. | Windows Registry export of HKCU\Software\Microsoft\Windows\CurrentVersion\Run and HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Run, plus Task Scheduler XML exports from %SYSTEMROOT%\System32\Tasks\ — the SessionGate loader establishes persistence before delivering the final payload stage, and these locations are the expected persistence artifacts to differentiate a SessionGate-only infection from a fully staged Remus Stealer or AnimateClipper compromise.

Per-Action IR Details

Containment — Block known malicious domains impersonating Ghidra (ghidra.re and variants), dnSpy, and SpiderFoot at DNS and web proxy layers immediately. Alert the security team to any recent downloads of these tools from non-official sources (official sources: ghidra.re via NSA GitHub, github.com/dnSpy/dnSpy, github.com/smicallef/spiderfoot). Isolate any endpoint where a suspicious installer was executed since September 2025.

NIST Phase: Containment

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

Controls: NIST AC-4 (Information Flow Enforcement), CIS 4.4 (Implement and Manage a Firewall on Servers), CIS 4.5 (Implement and Manage a Firewall on End-User Devices)

Compensating: Use Pi-hole or Windows DNS Server RPZ (Response Policy Zones) to blackhole lookalike domains matching regex patterns for 'ghidra', 'dnspy', and 'spiderfoot' outside github.com. On Windows endpoints without EDR, run: `Get-DnsClientCache | Where-Object {$_.Entry -match 'ghidra|dnspy|spiderfoot'} | Export-Csv dns_cache_snapshot.csv` on all developer workstations before flushing. Isolate suspected hosts by disabling the NIC via Device Manager or `netsh interface set interface "Ethernet" admin=disable` to prevent exfiltration of already-harvested browser credentials while preserving volatile state.

Evidence: Before blocking or isolating: (1) Export the full DNS client cache from suspect endpoints (`ipconfig /displaydns > dns_cache_$(hostname).txt`) to capture lookalike domain resolutions made prior to containment. (2) Pull web proxy or firewall logs filtered for HTTP GET/POST requests to non-GitHub domains containing 'ghidra', 'dnspy', or 'spiderfoot' in the hostname, particularly any delivering .exe, .msi, or .zip files since September 2025. (3) Collect the browser download history file from each suspect user profile (Chrome: `%LOCALAPPDATA%\Google\Chrome\User Data\Default\History`; Firefox: `%APPDATA%\Mozilla\Firefox\Profiles\*.default\places.sqlite`) — AnimateClipper and Remus Stealer arrive as fake installers, so the download URL in browser history is primary evidence of which TDS-served domain delivered the payload.

Detection — Search EDR and proxy logs for downloads of installer binaries from domains other than the verified GitHub repositories for Ghidra, dnSpy, and SpiderFoot. Query SIEM for clipboard-monitoring process activity (T1115), anomalous browser data access by unsigned processes (T1555.003), and outbound connections to CloudFront distributions initiated by newly-dropped executables. Review DNS logs for resolution of lookalike domains containing 'ghidra', 'dnspy', or 'spiderfoot' outside github.com and official project domains. Flag any process spawned from a user's Downloads folder that attempts to read browser profile directories. No CVE-specific event IDs apply — detection is behavior-based. Reference NIST AU-2 (Event Logging) and AU-6 (Audit Record Review, Analysis, and Reporting) for log source coverage requirements.

> **NIST Phase:** Detection Analysis
>
> **Reference:** NIST 800-61r3 §3.2 — Detection and Analysis
>
> **Controls:** NIST AU-2 (Event Logging), NIST AU-6 (Audit Record Review, Analysis, And Reporting), NIST AU-12 (Audit Record Generation)
>
> **Compensating:** Deploy Sysmon with SwiftOnSecurity config (minimum) and enable Event ID 1 (Process Create), Event ID 3 (Network Connect), and Event ID 11 (FileCreate). Hunt with PowerShell: `Get-WinEvent -LogName 'Microsoft-Windows-Sysmon/Operational' | Where-Object {$_.Message -match 'Downloads' -and $_.Message -match 'Chrome|Firefox|Edge|Brave'} | Select-Object TimeCreated, Message` to catch processes spawned from %USERPROFILE%\Downloads accessing browser profile paths. For clipboard hijacking (AnimateClipper/T1115), run: `Get-WinEvent -LogName Security | Where-Object {$_.Id -eq 4688 -and $_.Message -match 'clip'}` and augment with Sysmon EID 1 filtering on processes calling `OpenClipboard` via command-line pattern analysis. Use osquery to query `SELECT pid, name, path FROM processes WHERE path LIKE '%Downloads%'` across the fleet.
>
> **Evidence:** Before concluding detection scope: (1) Sysmon Event ID 1 records for processes originating from %USERPROFILE%\Downloads with parent process explorer.exe or a browser process — this is the initial execution path for both Remus Stealer and AnimateClipper fake installers. (2) Sysmon Event ID 3 (Network Connection) showing newly-created executables from Downloads initiating outbound HTTPS to *.cloudfront.net — this is the SessionGate TDS callback pattern. (3) Windows Security Event ID 4663 (Object Access) on browser profile directories (%LOCALAPPDATA%\Google\Chrome\User Data\Default\Login Data, Cookies, Web Data) by any process not matching the browser's own signed executable — this is T1555.003 credential harvesting by Remus Stealer. (4) Sysmon Event ID 13 (Registry Value Set) under HKCU\Software\Microsoft\Windows\CurrentVersion\Run or HKLM equivalents for persistence drops by SessionGate loader. (5) Network proxy logs showing HTTP 302 redirect chains originating from SEO-ranked lookalike domains before payload delivery — this is the TDS profiling fingerprint.

Eradication — If compromise is confirmed: terminate and quarantine the malicious process, remove the dropped binary, and revoke all browser sessions on the affected endpoint (force sign-out of all browser profiles). Rotate all credentials stored in the browser password manager on that device. For any user who interacts with cryptocurrency wallets, treat all wallet addresses copied on the compromised host as potentially swapped — verify on-chain before any transaction. Remove any browser extensions installed after the suspected compromise date and audit remaining extensions against the known-good baseline per CIS 2.1 (Maintain a Software Inventory). Reference D3-CRO (Credential Rotation) and D3-UAP (User Account Permissions) to scope credential reset and access restriction.

> **NIST Phase:** Eradication
>
> **Reference:** NIST 800-61r3 §3.4 — Eradication
>
> **Controls:** NIST AC-2 (Account Management), NIST AC-6 (Least Privilege), CIS 2.1 (Establish and Maintain a Software Inventory), CIS 5.1 (Establish and Maintain an Inventory of Accounts), CIS 6.2 (Establish an Access Revoking Process)
>
> **Compensating:** Export the full list of installed browser extensions before removal for evidence: Chrome — `Get-ChildItem 'C:\Users\*\AppData\Local\Google\Chrome\User Data\Default\Extensions\'`; Firefox — `Get-Content 'C:\Users\*\AppData\Roaming\Mozilla\Firefox\Profiles\*\extensions.json'`. Diff against a known-good snapshot or compare install timestamps against the suspected compromise date using `(Get-Item $path).CreationTime`. For

credential rotation without a PAM tool, generate a prioritized list of all saved passwords by exporting Chrome's Login Data SQLite file (copy first — do not open in place): `sqlite3 Login\ Data .dump | grep origin_url` to enumerate every site requiring rotation. Immediately revoke all active Google/Microsoft OAuth tokens via the respective account security dashboards to kill existing sessions derived from stolen browser cookies.

Evidence: Before eradication actions: (1) Acquire a full memory image of the compromised host using WinPmem (free) — Remus Stealer harvests credentials from browser memory and may hold decrypted credential material in process heap that will be lost after process termination. (2) Copy (do not delete) the malicious binary from %USERPROFILE%\Downloads or %APPDATA%\Roaming or %TEMP% to a write-protected evidence share — hash with SHA-256 immediately: `Get-FileHash -Algorithm SHA256 `. (3) Export the browser extension directories for any extensions installed post-compromise date, specifically examining manifest.json for `clipboardRead`, `clipboardWrite`, or `nativeMessaging` permissions that AnimateClipper's browser component would require. (4) Snapshot the Windows Clipboard history if enabled (Win+V) before clearing — AnimateClipper may leave artifacts of swapped wallet addresses in clipboard history stored at `%LOCALAPPDATA%\Microsoft\Windows\Clipboard\`.

Recovery — After credential rotation and session revocation, re-image the affected endpoint or perform a verified clean restore. Re-download security tools exclusively from official GitHub repositories, verifying SHA-256 hashes published in official release notes before execution. Confirm no persistence mechanisms remain by reviewing startup configurations per D3-SICA (System Init Config Analysis). Monitor the affected user's accounts for 30 days for anomalous access patterns. Re-enable endpoint only after EDR confirms clean state. Reference NIST IR controls for post-incident validation steps.

NIST Phase: Recovery

Reference: NIST 800-61r3 §3.5 — Recovery

Controls: CIS 2.2 (Ensure Authorized Software is Currently Supported), CIS 4.6 (Securely Manage Enterprise Assets and Software), CIS 7.3 (Perform Automated Operating System Patch Management), CIS 7.4 (Perform Automated Application Patch Management)

Compensating: Before re-imaging, run Autoruns (Sysinternals, free) with VirusTotal integration enabled — export results as XML for the evidence record: `autorunsc.exe -a * -ct -h -s -vt > autoruns_$(hostname)_$(Get-Date -Format yyyyMMdd).csv`. After re-image, script the verified re-download of Ghidra, dnSpy, and SpiderFoot via PowerShell: fetch the SHA-256 hash from the official GitHub release page, download the binary, and compare with `(Get-FileHash -Algorithm SHA256).Hash` before allowing execution. Implement a 30-day PowerShell script running nightly to check for anomalous new scheduled tasks or Run key entries: `Get-ScheduledTask | Where-Object {$_.TaskPath -notmatch 'Microsoft'} | Export-Csv scheduled_tasks_$(Get-Date -Format yyyyMMdd).csv`.

Evidence: Before re-imaging (critical — evidence lost after wipe): (1) Full disk image using FTK Imager Lite (free) or `dd` — do not skip this for any host where cryptocurrency wallet interaction occurred, as on-chain transaction records alone may be insufficient to reconstruct the full scope of AnimateClipper address swapping. (2) Export Windows Registry hives for persistence analysis: `reg export HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Run run_hklm.reg` and HKCU equivalent — SessionGate loader persistence will appear here or in scheduled tasks. (3) Copy all user-accessible browser profile directories to evidence storage before re-image, preserving timestamps. (4) Document all cryptocurrency wallet addresses found in clipboard history, browser autofill (Web Data SQLite), and any locally stored wallet configuration files — these are needed to assess AnimateClipper's transaction interception scope across the 20+ targeted blockchain ecosystems.

Post-Incident — This campaign exploited the absence of verified download sources and developer trust in search engine results. Implement a policy requiring all open-source tool downloads to be sourced exclusively from pinned, verified GitHub release pages with hash verification (CWE-494 gap closure). Enforce application allowlisting or developer workstation controls to block unsigned executables from running from user download directories (NIST AC-3, Access Enforcement). Conduct awareness training specifically targeting developer and security analyst populations on SEO poisoning and supply chain download risks. Reference CIS 2.3 (Address Unauthorized Software) and NIST SI-7 (Software, Firmware, and Information Integrity) — note: SI-7 is not in the verified knowledge base extract above, so citing it here as a framework-level reference only; verify the specific control text in SP 800-53 Rev. 5 before including in formal documentation.

NIST Phase: Post Incident

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

Controls: NIST AC-3 (Access Enforcement), CIS 2.3 (Address Unauthorized Software), CIS 7.1 (Establish and Maintain a Vulnerability Management Process), CIS 7.2 (Establish and Maintain a Remediation Process), CIS 4.7 (Manage Default Accounts on Enterprise Assets and Software)

Compensating: Implement application allowlisting using Windows Software Restriction Policies (SRP) or AppLocker (available in Windows Pro/Enterprise without additional cost) to block execution of unsigned binaries from %USERPROFILE%\Downloads, %TEMP%, and %APPDATA% — publish a GPO rule: deny execution for path `%USERPROFILE%\Downloads\*.exe` for all non-admin users. For developer populations who legitimately need to run downloaded tools, create a documented exception workflow requiring SHA-256 verification against the official GitHub release asset checksums before an admin-approved path exclusion is granted. Deploy a free YARA rule scanning Downloads directories nightly using YARA standalone: write rules matching known Remus Stealer and AnimateClipper string patterns (e.g., clipboard hook API sequences, hardcoded CloudFront C2 patterns) sourced from public threat intel reports.

Evidence: Post-incident documentation to retain for lessons learned and future detection improvement: (1) Full TDS redirect chain HTTP logs showing the SEO-poisoned domain → CloudFront TDS → payload delivery URL sequence — this documents the specific profiling criteria (OS, browser, geolocation) that SessionGate used to select between Remus Stealer and AnimateClipper payloads for your environment. (2) Timeline correlation of Google Search query (if recoverable from browser history) to download event to first malicious process execution — this establishes dwell time and validates whether existing detections would have caught earlier stages. (3) Final inventory of all credentials confirmed rotated and all cryptocurrency wallet addresses confirmed uncompromised on-chain — required for breach notification scoping if PII or financial assets were affected. (4) Comparison of lookalike domain registration dates against your DNS query logs to determine whether earlier detection was possible and to tune future DNS monitoring thresholds for typosquatting patterns targeting security tool names.

Detection Guidance

Priority log sources: DNS resolver logs, web proxy logs, EDR telemetry, and browser history on developer and analyst workstations. Behavioral indicators to query: (1) DNS or proxy resolution of domains containing 'ghidra', 'dnspy', or 'spiderfoot' that do not resolve to github.com, ghidra.sre (NSA), or the project's official domain, flag for immediate review. (2) Executable files dropped to user download directories that subsequently spawn processes reading Chrome, Firefox, or Edge profile directories (Credentials path: AppData\Local\Google\Chrome\User Data). (3) Clipboard-monitoring behavior from unsigned or recently-dropped binaries, look for ReadProcessMemory or GetClipboardData API calls from processes without verified signatures. (4) Outbound HTTPS connections from newly-created processes to CloudFront (*.cloudfront.net) where the parent process is a recently-downloaded installer, not a known application. (5) Any browser extension installed after September 2025 that requests permissions for 'clipboardRead', 'tabs', and 'storage' simultaneously on developer workstations, cross-reference against CIS 2.1 software inventory baseline. TDS evasion note: the profiling layer may suppress payload delivery against sandbox IPs and known security vendor IP ranges, so dynamic analysis in cloud sandboxes may return clean results; prioritize endpoint behavioral detection over sandbox detonation for this campaign. (Reference: NIST AU-2, Event Logging; NIST AU-6, Audit Record Review, Analysis, and Reporting)

Indicators of Compromise

Type	Value	Context	Confidence
DOMAIN	`Lookalike domains impersonating ghidra, dnspy, spiderfoot — specific domains not confirmed in available T3 sources`	SEO-poisoned fake download sites; exact domains not published in available sources — monitor DNS logs for typosquats of these tool names	LOW
URL	`CloudFront-hosted JavaScript redirect layer — specific URL not confirmed in available T3 sources`	Intermediate redirect layer forwarding download requests to TDS; pattern: CloudFront distribution URL triggered on download button click from fake site	LOW
HASH	`Remus Stealer, AnimateClipper, SessionGate binaries — specific hashes not confirmed in available T3 sources`	No verified file hashes published in available source material; treat any unsigned installer from non-GitHub source for these tools as suspect	LOW

Framework Mappings

MITRE-ATTACK

- **T1195.002** — Compromise Software Supply Chain
- **T1555** — Credentials from Password Stores
- **T1608.001** — Upload Malware
- **T1036.005** — Match Legitimate Resource Name or Location
- **T1553** — Subvert Trust Controls
- **T1614.001** — System Language Discovery
- **T1539** — Steal Web Session Cookie
- **T1176** — Software Extensions
- **T1583.001** — Domains
- **T1555.003** — Credentials from Web Browsers
- **T1071.001** — Web Protocols
- **T1566.002** — Spearphishing Link
- **T1608.006** — SEO Poisoning
- **T1027** — Obfuscated Files or Information
- **T1059.003** — Windows Command Shell

NIST-800-53R5

- **CM-7** — Least Functionality
- **SA-9** — External System Services
- **SR-3** — Supply Chain Controls and Processes
- **SI-7** — Software, Firmware, and Information Integrity

- **AT-2** — Literacy Training and Awareness
- **SC-7** — Boundary Protection
- **SI-3** — Malicious Code Protection
- **SI-4** — System Monitoring
- **SI-8** — Spam Protection
- **CM-3** — Configuration Change Control

OWASP-TOP10-2021

- **A08:2021** — Software and Data Integrity Failures

CIS-V8

- **2.5** — Allowlist Authorized Software
- **2.6** — Allowlist Authorized Libraries
- **6.3** — Require MFA for Externally-Exposed Applications
- **8.2** — Collect Audit Logs

HIPAA-SECURITY

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

SOC2-TSC

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

NIST-CSF-2

- **DE.CM-01** — Networks and network services are monitored

ISO-27001-2022

- **A.5.23** — Information security for use of cloud services

MITRE ATT&CK Mapping

Technique ID	Technique Name	Tactic
T1195.002	Compromise Software Supply Chain	Initial-Access
T1555	Credentials from Password Stores	Credential-Access
T1608.001	Upload Malware	Resource-Development
T1036.005	Match Legitimate Resource Name or Location	Defense-Evasion
T1553	Subvert Trust Controls	Defense-Evasion
T1614.001	System Language Discovery	Discovery
T1539	Steal Web Session Cookie	Credential-Access
T1176	Software Extensions	Persistence

Technique ID	Technique Name	Tactic
T1583.001	Domains	Resource-Development
T1555.003	Credentials from Web Browsers	Credential-Access
T1071.001	Web Protocols	Command-And-Control
T1566.002	Spearphishing Link	Initial-Access
T1608.006	SEO Poisoning	Resource-Development
T1027	Obfuscated Files or Information	Defense-Evasion
T1059.003	Windows Command Shell	Execution

Sources

Source	URL	Tier
Security News	https://thehackernews.com/2026/06/fake-sites-mimicking-open-source-...	T3
PSA: New Zero-Day vulnerability found impacting most password ...	https://www.reddit.com/r/firefox/comments/1mvui40/psa_new_zeroday_v...	T3
Google Issues HIGH-SEVERITY Alert (Crypto Holders Beware)	https://www.youtube.com/watch?v=fsC-EHBwDzI	T3
How browser extensions expose crypto to a fatal design flaw the ...	https://cryptoslate.com/how-browser-extensions-expose-your-crypto-t...	T3
Trust Wallet Browser Exploit: Supply Chain Failure | Jag Foo, CFtP ...	https://www.linkedin.com/posts/jag-foo-cftp-cpp-psp-pci-3bb85175_tr...	T3

Generated 2026-06-04 14:10 UTC by TJS Security Command Center