INTELLIGENCE BRIEFING
Security Command Center
TLP:CLEAR
2026-07-02 07:13 UTC

VEIL#DROP Abuses Google's Blogger Infrastructure to Deliver PureLogs Stealer via Fileless, Polymorphic Loader Chain

THREAT CAMPAIGN | **HIGH** | CVSS 5.0

SCC Item ID	SCC-CAM-2026-0612
Type	Threat Campaign
Severity	HIGH
CVSS Base Score	5.0
Affected Products	Windows systems (WSH/PowerShell execution environments); abused infrastructure: Google Blogger; abused signed binaries: regsvcs.exe, installutil.exe, msbuild.exe, aspnet_compiler.exe
Published	2026-07-01T13:18:50
Discovery Source	Rss

Executive Summary

A multi-stage attack campaign tracked as VEIL#DROP uses disguised JavaScript files to deliver PureLogs, a credential-stealing malware, through a chain that abuses Google's Blogger platform and Microsoft-signed Windows system tools to evade detection. Windows-based organizations are at risk; the attack targets locally stored credentials, browser-saved passwords, and sensitive application data. Because the campaign routes through trusted infrastructure and avoids writing files to disk, conventional perimeter defenses and antivirus signatures provide limited protection. The primary business risk is undetected credential theft leading to account takeover or further network compromise.

Technical Analysis

VEIL#DROP is a fileless, polymorphic attack chain targeting Windows systems with PowerShell and Windows Script Host (WSH) execution enabled. The initial vector is a disguised JavaScript file (T1566.001, T1059.007) that invokes a multi-stage PowerShell loader (T1059.001). Intermediate payloads are staged via Google Blogger URLs (T1102.001), exploiting the platform's trusted reputation to bypass URL reputation filtering. The chain employs dynamic URL construction and runtime script mutation (T1027.009, T1620) to defeat signature detection, and uses reflective .NET assembly loading (T1620) to execute in memory without touching disk. LOLBin fallback execution cycles through regsvcs.exe (T1218.009), installutil.exe (T1218.004), msbuild.exe (T1127.001), and aspnet_compiler.exe (T1218.008), all Microsoft-signed binaries, to resist application

allowlisting controls (T1553). Indicator removal and anti-forensics behaviors are mapped to T1070.004. The terminal payload is PureLogs, a .NET-based information stealer targeting browser credentials, saved passwords, and application secrets (T1555). Masquerading techniques (T1036.007) and exfiltration over C2 (T1041) complete the chain. Relevant weaknesses: CWE-116 (improper encoding/escaping) and CWE-693 (protection mechanism failure). No CVE is associated. No vendor patch applies; mitigation is behavioral and configuration-based. Core campaign claims derive from Securonix research as reported by The Hacker News. Attribution is unconfirmed. Treat technical specifics as medium-confidence pending independent corroboration from additional vendors.

Action Checklist

1. Step 1: Containment. Block outbound connections from scripting hosts (wscript.exe, cscript.exe, powershell.exe, pwsh.exe) to *.blogspot.com and *.blogger.com at the proxy or firewall layer. This disrupts the staging infrastructure VEIL#DROP relies on per Securonix reporting. Prioritize endpoints where WSH and PowerShell are permitted to initiate outbound HTTP/HTTPS.
2. Step 2: Detection. Query EDR and SIEM for LOLBin process chains: parent-child relationships where powershell.exe, wscript.exe, or cscript.exe spawns regsvcs.exe, installutil.exe, msbuild.exe, or aspnet_compiler.exe. Alert on outbound HTTP/HTTPS from these binaries to any Blogger/Blogspot URL. Monitor Windows event logs for in-memory .NET assembly load events (Event ID 4104 for PowerShell script block logging; Sysmon Event ID 7 for image loads from non-standard paths). Enable NIST AU-2 event logging and AU-12 audit record generation across scripting and .NET execution environments. Reference CIS 8.2 for audit log collection baseline.
3. Step 3: Eradication. No vendor patch applies; this campaign abuses legitimate infrastructure and signed binaries. Restrict WSH execution via Group Policy (disable wscript.exe and cscript.exe for standard users). Enforce PowerShell Constrained Language Mode and script block logging (NIST CM controls; CIS 4.6). Apply application control rules (AppLocker or Windows Defender Application Control) to block execution of regsvcs.exe, installutil.exe, msbuild.exe, and aspnet_compiler.exe by non-administrative processes. Implement least-privilege user account permissions to restrict LOLBin execution contexts. Rotate credentials for any accounts on endpoints where indicators were observed.
4. Step 4: Recovery. Validate that script block logging and process creation auditing are generating events before restoring normal operations. Confirm outbound proxy blocks on Blogger domains are active and logging. Re-image endpoints where PureLogs execution is confirmed or suspected; do not rely on in-place cleanup given fileless execution. After re-imaging, force password resets for all accounts that were active on affected systems. Enforce multi-factor authentication on recovered accounts. Review SIEM for any lateral movement or credential use anomalies in the 30 days prior to detection.
5. Step 5: Post-Incident. This campaign exposes gaps in three control areas: (1) LOLBin execution controls, assess whether application control policies (NIST AC-3, Access Enforcement; CIS 4.6) adequately restrict signed binary abuse; (2) Egress filtering, evaluate whether scripting processes are permitted to initiate outbound network connections without restriction; (3) Credential exposure posture, review whether browser-saved credentials and application secrets are protected (NIST AC-6, Least Privilege; CIS 3.3, Configure Data Access Control Lists). Document findings and update incident playbooks to include LOLBin chain detection as a standing hunt hypothesis.

Detection Guidance

Primary detection surface is behavioral, not signature-based. Key signals: (1) Process chain anomalies, wscript.exe, cscript.exe, or powershell.exe spawning regsvcs.exe, installutil.exe, msbuild.exe, or aspnet_compiler.exe. Query Sysmon Event ID 1 (process creation) and Windows Security Event ID 4688 for these parent-child relationships. (2) Outbound network from scripting hosts, proxy or firewall logs showing HTTP/HTTPS requests to *.blogspot.com or *.blogger.com originating from powershell.exe, wscript.exe, or any LOLBin listed above. (3) In-memory .NET loading, PowerShell Event ID 4104 (script block logging) capturing Assembly.Load or Reflection.Assembly patterns; Sysmon Event ID 7 (image loaded) for .NET assemblies loading from unusual paths or with no path. (4) Polymorphic script mutation, repeated but structurally varied PowerShell invocations on the same host within a short window. Enable NIST AU-2 event logging and AU-6 audit record review. Ensure PowerShell script block logging and module logging are active per CIS 8.2. Local account monitoring and system file analysis are applicable countermeasures. Note: specific IOC hashes and static Blogger URLs are not independently corroborated from multiple sources; behavioral rules provide more durable detection than static indicators for this campaign pending additional vendor confirmation.

Indicators of Compromise

Type	Value	Context	Confidence
DOMAIN	`*.blogspot.com`	Google Blogger subdomains used as payload staging infrastructure per Securonix reporting; outbound connections from scripting hosts to this domain pattern are anomalous	MEDIUM
DOMAIN	`*.blogger.com`	Google Blogger infrastructure abused for intermediate payload delivery; same detection logic as blogspot.com	MEDIUM

Framework Mappings

MITRE-ATTACK

- **T1027.010** — Command Obfuscation
- **T1102.001** — Dead Drop Resolver
- **T1127.001** — MSBuild
- **T1059.007** — JavaScript
- **T1070.004** — File Deletion
- **T1218.008** — Odbcconf
- **T1566.001** — Spearphishing Attachment
- **T1059.001** — PowerShell
- **T1036.007** — Double File Extension
- **T1218.004** — InstallUtil
- **T1555** — Credentials from Password Stores
- **T1553** — Subvert Trust Controls

- **T1218.009** — Regsvcs/Regasm
- **T1620** — Reflective Code Loading
- **T1027.009** — Embedded Payloads
- **T1041** — Exfiltration Over C2 Channel

NIST-800-53R5

- **AT-2** — Literacy Training and Awareness
- **SC-7** — Boundary Protection
- **SI-3** — Malicious Code Protection
- **SI-4** — System Monitoring
- **SI-8** — Spam Protection
- **CM-7** — Least Functionality
- **SI-7** — Software, Firmware, and Information Integrity
- **CA-7** — Continuous Monitoring

ISO-27001-2022

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

CIS-V8

- **8.2** — Collect Audit Logs

NIST-CSF-2

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

MITRE ATT&CK Mapping

Technique ID	Technique Name	Tactic
T1027.010	Command Obfuscation	Defense-Evasion
T1102.001	Dead Drop Resolver	Command-And-Control
T1127.001	MSBuild	Defense-Evasion
T1059.007	JavaScript	Execution
T1070.004	File Deletion	Defense-Evasion
T1218.008	Odbcconf	Defense-Evasion
T1566.001	Spearphishing Attachment	Initial-Access
T1059.001	PowerShell	Execution
T1036.007	Double File Extension	Defense-Evasion
T1218.004	InstallUtil	Defense-Evasion

Technique ID	Technique Name	Tactic
T1555	Credentials from Password Stores	Credential-Access
T1553	Subvert Trust Controls	Defense-Evasion
T1218.009	Regsvcs/Regasm	Defense-Evasion
T1620	Reflective Code Loading	Defense-Evasion
T1027.009	Embedded Payloads	Defense-Evasion
T1041	Exfiltration Over C2 Channel	Exfiltration

Sources

Source	URL	Tier
Security News	https://thehackernews.com/2026/07/veildrop-malware-chain-uses-blogg...	T2
MSBuild Abuse Helps Attackers Launch Stealthy Fileless Windows ...	https://cyberpress.org/msbuild-fuels-fileless-intrusions/	T3
Microsoft Security Response Center Blog	https://www.microsoft.com/en-us/msrc/blog	T1
Windows powershell.exe getting detected as Trojan or Malware	https://forums.malwarebytes.com/topic/301554-windows-powershellexe-...	T1

Generated 2026-07-02 07:13 UTC by TJS Security Command Center