

Eimeria: five layers from RAR5 to RunPE

 derp.ca/research/eimeria-multi-stage-loader

Kirk

May 8, 2026



Eimeria wraps five layers of obfuscation around a skeleton crew: one RAT, one C2 endpoint, and enough staging to make a DarkGate loader jealous. The delivery chain starts as a RAR5 archive on Dedik Services Ltd infrastructure (UK-registered ASN, host geolocated to Frankfurt). It unfolds through a signed carrier DLL side-load, a hidden AES engine inside a fake zlib DLL, an IExpress self-extracting archive, Autolt process hollowing, and a final .NET C2 beacon.

No existing YARA rules covered this chain. Triage assigned the family label Eimeria on May 8, 2026.

Sample overview

Field	Value
Archive SHA256	c872cd101d9c2a773f08558dde7b716161cf977d4aa99c2347c0269423434f8c
Archive type	RAR5

Field	Value
Submission	2026-05-08 on Triage (260508-n6j eqagv2w)
Triage score	10/10 (dsclock.exe, zlibwapi.dll)
VirusTotal	1/74 (TrendMicro-HouseCall: Trojan.Win32.Gen.TL0101DN26ZN)
Family label	eimeria (Triage-assigned)
C2 endpoint	ws://94.26.90.139:3006
Provider	Dedik Services Ltd (AS207043)
C2 VT	11/92 malicious
IPinfo geo	Frankfurt am Main, Hesse, DE (50.1155, 8.6842)
Liveness	Confirmed at intake (nc -vz succeeded)

Individual file VT hits:

- dsclock.exe (signed carrier): 0/70
- zlibwapi.dll (hidden loader): 4/68 (Bkav, APEX, TrendMicro, Cynet)
- msbuilder64.dll (encrypted payload): 0/61

Layer 0: the RAR5 bundle

The original submission is a RAR5 archive. bsdtar extracted four files under jjez/:

File	Size	Type	Triage score
dsclock.exe	1,679,312	PE32 GUI x86, signed	10
libcurl.dll	362,496	PE32 DLL x86	3
msbuilder64.dll	4,652,720	Data (max-entropy)	1
zlibwapi.dll	93,696	PE32 DLL x86	10

RAR entry timestamps are 2026-04-20 17:37. The .exe + .dll layout under a generic jjez/ directory signals DLL side-loading. libcurl.dll is a legitimate dependency for dsclock.exe network functions. Not malicious, but necessary for the carrier to look real.

Layer 1: the signed carrier and the zlib DLL that was not

dsclock.exe is the carrier. It has been on VirusTotal since 2022 with 0/70 detections and shows no malicious behaviour on its own. It is Authenticode-signed by Duality Software Co. Ltd. (Saint Petersburg, Russia) -- a legitimate Russian software certificate with 0% detection across 143 known files -- and was compiled with a PDB path of 0:\Projects\dsclock\Release\DSClock.x86.pdb. It imports libcurl functions like a normal clock utility would.

The attack vector is DLL side-loading. Both files sit together in jjez/. When dsclock.exe runs, Windows loads zlibwapi.dll from the same directory before checking system paths. Same technique as dropping a malicious version.dll next to a signed executable.

zlibwapi.dll is the first surprise. It exports the usual zlib and minizip entry points, but hidden in its .text section is an AES-CBC decryption engine with its own SBOX table at offset 0xdf80 and RCON values at 0xe080. Capa reports PE header parsing, PE section enumeration, CreateProcess, BCryptGenRandom, file I/O, and Base64. None of that belongs in a zlib wrapper. It is not a compression library. It is a deception library.

The AES decrypts the third file in the bundle: msbuilder64.dll.

AES detail		Value
Algorithm	AES-128-CBC	
IV	First 16 bytes of msbuilder64.dll	
SBOX location	0xdf80 (confirmation: starts 637c777b)	
RCON location	0xe080	
Key material	Not stored as plaintext; runtime-derived or obfuscated	

Layer 2: AES-CBC and the IExpress surprise

msbuilder64.dll is 4,652,720 bytes of what the file command calls data. The byte distribution is flat: every value 0x00-0xFF appears with roughly equal frequency. The first 16 bytes are the AES-CBC IV, prepended to the ciphertext.

Decryption (cross-referenced against Triage execution trace) reveals a PE32+ executable of the same size. Capa identifies it as an IExpress self-extracting archive. Capsule summary: the big encrypted blob was hiding a setup executable.

The embedded CAB file sits at IExpress stub offset 0x2a830: 4,478,592 bytes containing 26 files.

Layer 3: IExpress spills 26 decoy files

The IExpress archive extracts 19 .potm decoys and seven extensionless binary blobs:

Decoys: Bus, Centre, Code, Conference, Cost, Development, Dream, Fuel, Plant, Process, Reference, Relationship, Sight, Speed, St, Station, Supply, Technology, Unit

Blobs: Beach, Election, Light, Name, Owner, Prince, Resistance

These are PowerPoint-shaped rocks. There are no slides, no macros, no presentations -- all 26 are encrypted binary blobs dressed in a .potm costume. The names appear to be randomly chosen dictionary words.

At runtime, two additional files appear on disk: Deal.exe (1,107,552 bytes) and bMgXiqSim (3,967,774 bytes). Deal.exe is the Autolt-compiled RunPE loader. bMgXiqSim is a concatenated bundle of the IExpress contents, repackaged in the same encrypted format.

Layer 4: Autolt, RC4, LZNT1, and process hollowing

Deal.exe is the business end. Compiled with Autolt3, it embeds both the Autolt runtime and a 24,773-line compiled pcode script. Capa flags Autolt and warns that detected capabilities may be runtime noise. The pcode disagrees.

The Autolt script implements a full RunPE (process hollowing) loader:

1. **Restore ntdll** -- loads a clean copy of ntdll.dll from the filesystem, overwriting userland EDR hooks before any injection runs.
2. **RC4 decrypt** -- decrypts a 2,398,823-byte hex blob using key wNDRKtWS12MEvmD4jr3ZyvqQTviBYboE5Ce.
3. **LZNT1 decompress** -- decompresses the RC4 output via ntdll!RtlDecompressBuffer (LZNT1 format 2, big-endian block headers with 2-bit signature).
4. **Select injection target** -- enumerates running processes and selects from explorer.exe, svchost.exe, or taskhostw.exe.
5. **Hollow and inject** -- CreateProcess suspended, NtUnmapViewOfSection, VirtualAllocEx, WriteProcessMemory, fix imports, apply relocations, SetThreadContext, ResumeThread.

6. **Architecture detection** -- handles x86 and x64 targets differently, and detects .NET assemblies for alternate handling.

The final injected payload is an 8,355-byte PE with MZ and PE headers present. The section table needs fixup, but the content is a .NET assembly that connects to `ws://94.26.90.139:3006`.

Anti-analysis

The Autolt loader runs ntdll restoration (detailed in Layer 4 step 1) alongside these additional checks:

Check	Mechanism
Natural delay	28-second delay before any malicious behaviour
Anti-emulation	Pi calculation: 1,048,576 iterations of <code>sum(1/n^2)</code> , checks <code>sqrt(sum*6) > 3.1415</code> . Fails under emulation.
Sleep check	Compares real vs emulated sleep duration
Stress test	CPU/memory stress test to detect thin VPS
Memory purge	Calls <code>EmptyWorkingSet</code> to reduce memory scanning surface
Secure zeroing	<code>RtlZeroMemory</code> on sensitive buffers after use

Persistence

Mechanism	Detail
Run key	<code>HKCU\...\Run\ReportFootballHost_EXX</code>
File copy	Autolt runtime -> <code>AppData\Local\Material\ReportFootballHost\KitchenTaylor.exe</code>
Script copy	Pcode -> <code>AppData\Local\Material\ReportFootballHost\HorseLiterature.a3x</code>

Mechanism	Detail
Task Scheduler	Material_ReportFootballHost_Startup (logon trigger)
Directory	AppData\Local\Material\ReportFootballHost (hidden + system attributes)

C2 protocol

Detail	Value
Endpoint	ws://94.26.90.139:3006
Provider	Dedik Services Ltd (AS207043)
Transport	TCP, WebSocket (ws:// schema from config extraction)
Liveness	Confirmed at intake, retry pattern in sandbox (6+ flows, 260 bytes TX, 0 RX)
DNS	None observed -- hardcoded IP

The C2 was live on 2026-05-12 20:19 UTC via nc -vz 94.26.90.139 3006. Triage sandbox reached it but received no response bytes. The server either blocks sandbox IPs or did not recognize the request shape.

Lineage

Eimeria is a Triage-assigned label, not a known public family. The closest comparable family is **DarkGate**, which also uses Autolt + RC4 + RunPE hollowing with ntdll restoration. The architectural differences are significant:

- Eimeria uses five delivery layers (RAR5 -> signed EXE -> zlib DLL -> IExpress -> Autolt). DarkGate typically ships as a single Autolt script.
- The signed carrier uses a legitimate Russian software certificate (Duality Software Co. Ltd.).
- The zlibwapi.dll disguise (AES hidden inside a legitimate zlib DLL) is novel.
- WebSocket C2 instead of DarkGate HTTP/HTTPS.
- RAR5 initial container is uncommon for this style of loader.

Someone built this. They knew DarkGate's playbook but wrote their own. The signed certificate, multi-layer staging, and WebSocket C2 point to a deliberate build, not a commodity toolkit.

IOC summary

Network

Type	Value	Context
IP:Port	94.26.90.139:3006	C2 WebSocket endpoint

Hashes

File	SHA256
RAR5 archive	c872cd101d9c2a773f08558dde7b716161cf977d4aa99c2347c0269423434f8c
dsclock.exe	62fdad7df8fd7bc2b211c2de06c002831b36987b48a943758432f25006661578
zlibwapi.dll	53abc3c2f3e919ecd84724439b4d4fb679857316c6af91987e6db1dde9e8a198
msbuilder64.dll	e155acf50ab0dad1a80f0a67d396d0ad5691fc9e314e4efd1da1dd3180c9632f
GxNWZFTx.exe (IExpress)	84fdf804149920cb474a030479fda1d5c9a5939388353054169ec692b8f75d3a
Deal.exe (Autolt)	5d69a932a077fee044b193c28e84564143f5c7e51079ab48e88fef74ab0b77b7

Host

Indicator	Value
Persistence path	%LOCALAPPDATA%\Material\ReportFootballHost\KitchenTaylor.exe
Pcode path	%LOCALAPPDATA%\Material\ReportFootballHost\HorseLiterature.a3x
Run key	HKCU\...\Run\ReportFootballHost_EXX

Indicator	Value
Scheduled task	Material_ReportFootballHost_Startup
RC4 key	wNDRKtWS12MEvmD4jr3ZyvqQTviBYboE5Ce
Compression	LZNT1 (ntdll!RtlDecompressBuffer, format 2)
AES mode	AES-128-CBC with prepended IV

Behavioural

Technique	Detail
Initial access	RAR5 archive with DLL side-load bundle
Defence evasion	Ntdll restoration from disk (EDR hook bypass)
Anti-analysis	Pi calculation, 28s delay, sleep check, stress test, memory purge
Execution	IExpress -> Autolt compiled pcode
Defense evasion / execution	Process hollowing via RunPE
Persistence	Run key + Task Scheduler + shortcut
C2	WebSocket to bare IP on port 3006
Injection targets	explorer.exe, svchost.exe, taskhostw.exe
Payload	.NET assembly (8,355-byte C2 beacon)

If you operate a threat intelligence platform with sample access or have seen similar delivery chains, reach out. Multi-stage loader chains like this are harder to track than single-stage payloads, and every sample helps.

Kirk

I like the internet. Want to get in touch?