IBM X-Force Threat Analysis: QuirkyLoader - A new malware loader delivering infostealers and RATs

By Raymond Joseph Alfonso

Published: 2025-08-19 · Archived: 2026-04-05 16:43:49 UTC

Since November 2024, IBM X-Force has observed a new loader, QuirkyLoader, being used to deliver additional payloads to infected systems. Some of the well-known malware families that use QuirkyLoader include:

- Agent Tesla
- AsyncRAT
- FormBook
- MassLogger
- Remcos
- Rhadamanthys
- Snake Keylogger

The multi-stage infection begins with an email. The threat actor uses both legitimate email service providers and a self-hosted email server to send emails with a malicious archive attached. This archive contains three key components: a legitimate executable, an encrypted payload and a malicious DLL. The actor uses DLL side-loading, a technique where launching the legitimate executable also loads the malicious DLL. This DLL, in turn, loads, decrypts and injects the final payload into its target process.

Notably, X-Force observed that the threat actor consistently writes the DLL loader module in .NET languages and uses ahead-of-time (AOT) compilation. This process compiles the code into native machine code before execution, making the resulting binary appear as though it were written in C or C++.

Threat type

- Loader

Analysis

Infection chain

The QuirkyLoader infection chain begins when a user opens a malicious archive file attached to a spam email. This archive contains a legitimate executable, an encrypted payload disguised as a DLL and a DLL loader module. In some instances, the archive includes other legitimate DLLs to hide the malicious module.

Executing the legitimate .EXE file starts the infection's subsequent stages. The executable uses DLL side-loading to load the malicious DLL. This DLL then loads, decrypts and injects the final payload into a target process. It

accomplishes this by performing process hollowing on one of the following processes: *AddInProcess32.exe*, *InstallUtil.exe* or *aspnet_wp.exe*.

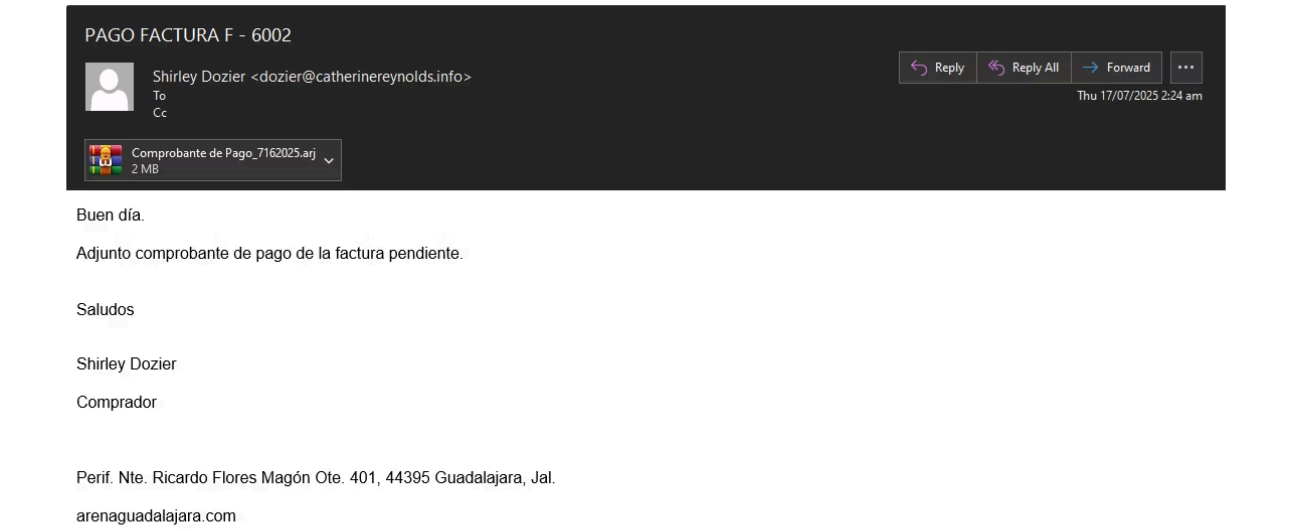


Figure 1: Sample email

PAGO FACTURA F - 6002
Shirley Dozier <dozier@catherinereynolds.info>
To
Cc
Reply
Reply All
Forward
Thu 17/07/2025 2:24 am
Comprobante de Pago_7162025.arj
2 MB
Buen día.
Adjunto comprobante de pago de la factura pendiente.
Saludos
Shirley Dozier
Comprador
Perif. Nte. Ricardo Flores Magón Ote. 401, 44395 Guadalajara, Jal.
arenaguadalajara.com

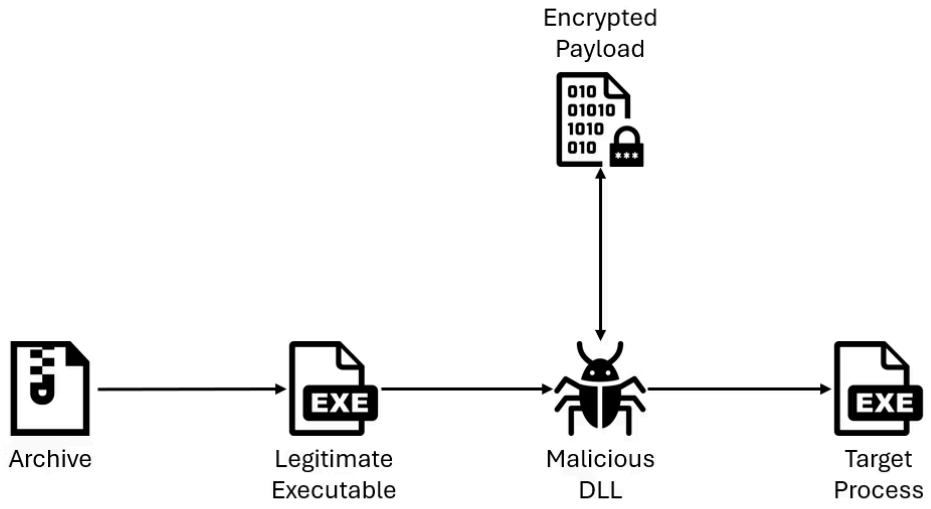
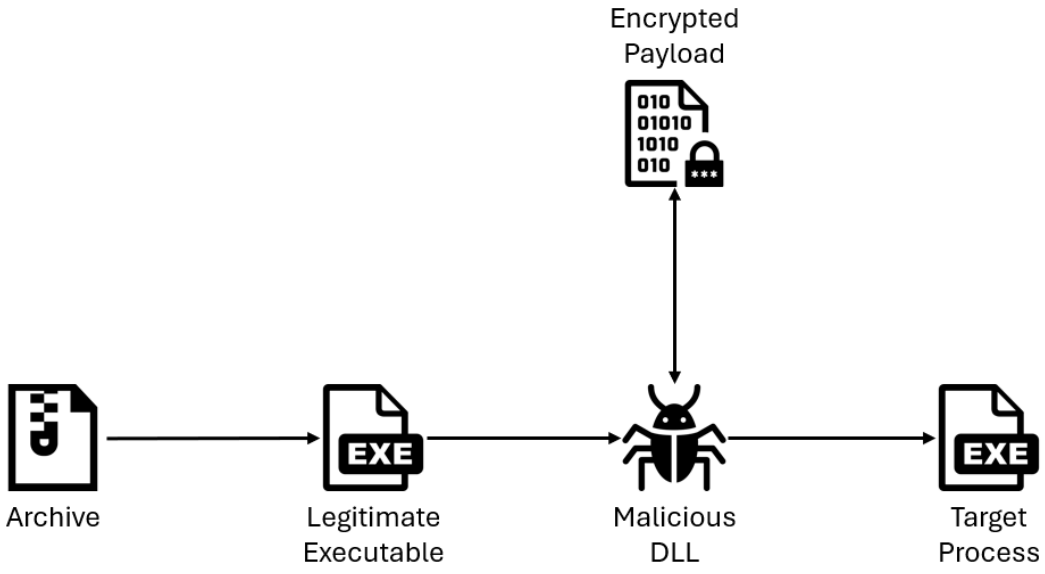


Figure 2: Infection chain



DLL loader module

QuirkyLoader's DLL module is consistently written in C# .NET. It is compiled using Ahead-of-Time (AOT) compilation, which compiles the C# code into Microsoft Intermediate Language (MSIL) first, and then compiles the MSIL into native machine code. This technique bypasses the traditional .NET method of first compiling code into Microsoft Intermediate Language (MSIL) and then using the Common Language Runtime (CLR) to translate it into native code. As a result, the final binary resembles a program written in C or C++.

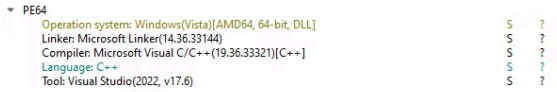


Figure 3: Compiler and language identification for a .NET AOT binary

PE64
Operation system: Windows(Vista)[AMD64, 64-bit, DLL]
Linker: Microsoft Linker(14.36.33144)
Compiler: Microsoft Visual C/C++(19.36.33321)[C++]
Language: C++
Tool: Visual Studio(2022, v17.6)

To load the encrypted payload, the malware calls the Win32 APIs *CreateFileW()* and *ReadFile()*. It then decrypts the buffer containing the payload, typically using a block cipher.

Interestingly, one variant uses the Speck-128 cipher with Counter (CTR) mode to decrypt the payload, a method not commonly used by malware. The Speck cipher works by expanding the master key into several round keys. It uses these round keys along with a nonce to generate a keystream by performing Add-Rotate-XOR (ARX) operations. Finally, the malware XORs the generated keystream against the encrypted data in 16-byte blocks to produce the decrypted payload.

```
__int64 __fastcall SPECK_128_KeyStream(__int64 *Nonce_Lower_Half, __int64 *Nonce_Upper_Half, __int64 Round_Keys) {  __int64 result; // rax  __int64 v4; // r10  LODWORD(result) = 0;  if ( Round_Keys && *(Round_Keys + 8) >= 32 )  {  do  {  *Nonce_Lower_Half = *(Round_Keys + 8LL * result + 16) ^ (*Nonce_Upper_Half + __ROL8__(*Nonce_Lower_Half, 56));  *Nonce_Upper_Half = *Nonce_Lower_Half ^ __ROL8__(*Nonce_Upper_Half, 3);  result = (result + 1);  }  while ( result < 32 );  }  else  {  do  {  v4 = *Nonce_Upper_Half + __ROL8__(*Nonce_Lower_Half, 56);  if ( result >= *(Round_Keys + 8) ) ERR_Mb_15();  *Nonce_Lower_Half = *(Round_Keys + 8LL * result + 16) ^ v4;  *Nonce_Upper_Half = *Nonce_Lower_Half ^ __ROL8__(*Nonce_Upper_Half, 3);  result = (result + 1);  }  while ( result < 32 );  }  return result; }
```

Code block 1 Key Stream Generation of Speck Cipher

To evade detection by security software, the malware dynamically resolves the Win32 APIs required for process hollowing.

First, the malware uses *CreateProcessW()* to launch a process in a suspended state. It then unmaps the memory of the suspended process with *ZwUnmapViewOfSection()* and writes its malicious payload into that memory space using *ZwWriteVirtualMemory()*. After performing these initializations, the malware sets the payload's starting point with *SetThreadContext()* and calls *ResumeThread()* to execute it.

```
GetProcAddress ( 0x00007ff899380000, "CreateProcessW" ) GetProcAddress ( 0x00007ff899380000, "OpenProcess" ) GetProcAddress ( 0x00007ff899380000, "TerminateProcess" ) GetProcAddress ( 0x00007ff899380000, "CloseHandle" ) GetProcAddress ( 0x00007ff899380000, "GetThreadContext" )
```

GetProcAddress (0x00007ff899380000, "Wow64GetThreadContext") GetProcAddress (0x00007ff899380000, "SetThreadContext") GetProcAddress (0x00007ff899380000, "Wow64SetThreadContext") GetProcAddress (0x00007ff899380000, "ResumeThread") GetProcAddress (0x00007ff899380000, "VirtualAllocEx") GetProcAddress (0x00007ff89a6d0000, "ZwUnmapViewOfSection") GetProcAddress (0x00007ff89a6d0000, "ZwWriteVirtualMemory") GetProcAddress (0x00007ff899790000, "memset") GetProcAddress (0x00007ff899380000, "VirtualProtectEx") GetProcAddress (0x00007ff899380000, "FlushInstructionCache") GetProcAddress (0x00007ff899380000, "ReadProcessMemory")

Victimology

While information regarding the geographical distribution of QuirkyLoader's operations has been limited for the past few months, two distinct campaigns were discovered in July 2025 targeting Taiwan and Mexico. The campaign in Taiwan specifically targeted employees of Nusoft Taiwan, a network and internet security research company, and distributed the Snake Keylogger infostealer. In Mexico, the campaign randomly targeted individuals, delivering both the Remcos RAT and AsyncRAT.

Related network infrastructure

IBM X-Force uncovered additional network IOCs related to the domain used to distribute the malspam emails. The investigation started with the domain catherinereynolds[.]info, which resolves to the IP address 157[.]66[.]225[.]11 and hosts a Zimbra web client. Upon closer inspection, it was found that the domain uses an SSL certificate with the common name mail[.]catherinereynolds[.]info. Pivoting from this certificate, the IPs 103[.]75[.]77[.]90 and 161[.]248[.]178[.]212 were discovered to be using the same SSL certificate. X-Force is highly confident that these additional IPs are related because they use similar ISPs, host similar services and share the same common name in their SSL certificates.

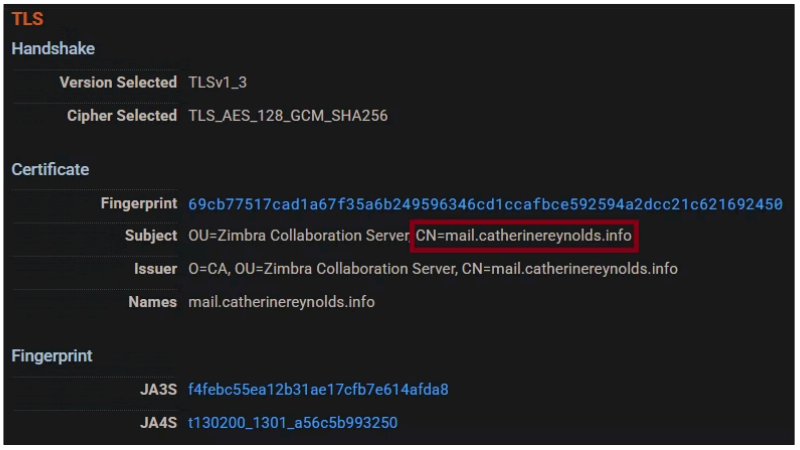
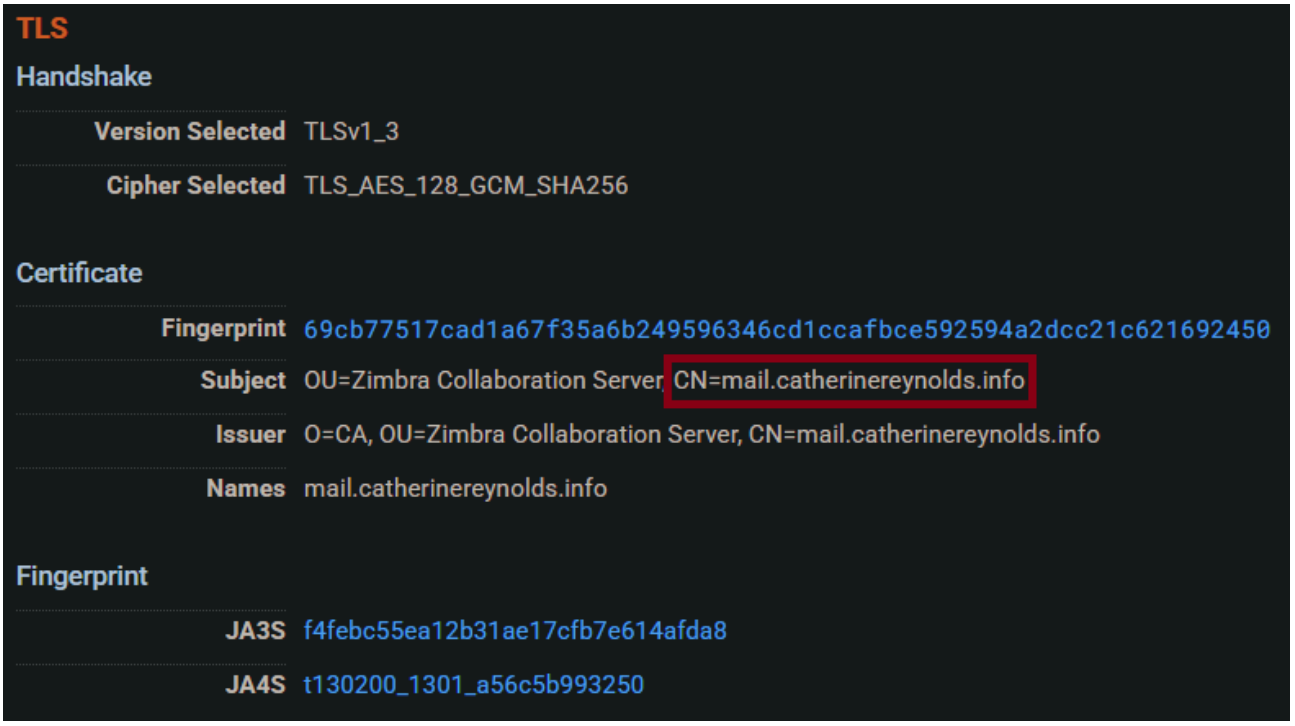


Figure 4: SSL Certificate of catherinereynolds[.]info



Conclusion

QuirkyLoader is a new loader malware that is actively distributing well-known malware families like Agent Tesla, AsyncRAT and Remcos. The threat actor initiates a multi-stage infection using malicious emails containing an archive file. By leveraging DLL side-loading, the malware executes its core DLL module, which is consistently written in .NET and compiled ahead-of-time to disguise its nature. This module then decrypts and injects the final payload, demonstrating a sophisticated method for delivering various malware threats.

Recommendations

- Block messages with executable attachments
- Avoid opening unexpected emails
- Avoid opening files that come from untrusted sources
- Keep security products up-to-date and properly configured
- Since the final payloads are typically infostealers and remote access tools, actively monitor and inspect outbound network traffic
- Closely monitor the behavior of the following legitimate processes, as they are common targets for process hollowing by QuirkyLoader:
-
    - AddInProcess32.exe
    - InstallUtil.exe
    - aspnet_wp.exe

Indicators of compromise

Indicator	Indicator Type	Context
011257eb766f2539828bdd45f8aa4ce3c4048ac2699d988329783290a7b4a0d3	File	QuirkyLoader DLL Module
0ea3a55141405ee0e2dfbf333de01fe93c12cf34555550e4f7bb3fdec2a7673b	File	QuirkyLoader DLL Module
a64a99b8451038f2bbcd322fd729edf5e6ae0eb70a244e342b2f8eff12219d03	File	QuirkyLoader DLL Module
9726e5c7f9800b36b671b064e89784fb10465210198fbbb75816224e85bd1306	File	QuirkyLoader DLL Module
a1994ba84e255eb02a6140cab9fc4dd9a6371a84b1dd631bd649525ac247c111	File	QuirkyLoader DLL Module
d954b235bde6ad02451cab6ee1138790eea569cf8fd0b95de9dc505957c533cd	File	Sample email of QuirkyLoader
5d5b3e3b78aa25664fb2bfdbf061fc1190310f5046d969adab3e7565978b96ff	File	Sample email of QuirkyLoader
6f53c1780b92f3d5affcf095ae0ad803974de6687a4938a2e1c9133bf1081eb6	File	Sample email of QuirkyLoader

ea65cf2d5634a81f37d3241a77f9cd319e45c1b13ffbaf5f8a637b34141292eb	File	Sample email of QuirkyLoader
1b8c6d3268a5706fb41ddfff99c8579ef029333057b911bb4905e24aacc05460	File	Sample email of QuirkyLoader
d0a3a1ee914bcbfcf709d367417f8c85bd0a22d8ede0829a66e5be34e5e53bb9	File	Sample email of QuirkyLoader
b22d878395ac2f2d927b78b16c9f5e9b98e006d6357c98dbe04b3fd78633ddde	File	Sample email of QuirkyLoader
a83aa955608e9463f272adca205c9e1a7cbe9d1ced1e10c9d517b4d1177366f6	File	Sample email of QuirkyLoader
3391b0f865f4c13dcd9f08c6d3e3be844e89fa3afbcd95b5d1a1c5abcacf41f4	File	Sample email of QuirkyLoader
b2fdf10bd28c781ca354475be6db40b8834f33d395f7b5850be43ccace722c13	File	Sample email of QuirkyLoader
bf3093f7453e4d0290511ea6a036cd3a66f456cd4a85b7ec8fbfea6b9c548504	File	Email attachment containing QuirkyLoader

97aee6ca1bc79064d21e1eb7b86e497adb7ece6376f355e47b2ac60f366e843d	File	Email attachment containing QuirkyLoader
b42bc8b2aeec39f25babdcbbdaab806c339e4397debfde2ff1b69dca5081eb44	File	Email attachment containing QuirkyLoader
5aaf02e4348dc6e962ec54d5d31095f055bd7fb1e58317682003552fd6fe25dc	File	Email attachment containing QuirkyLoader
8e0770383c03ce69210798799d543b10de088bac147dce4703f13f79620b68b1	File	Email attachment containing QuirkyLoader
049ef50ec0fac1b99857a6d2beb8134be67ae67ae134f9a3c53699cdaa7c89ac	File	Email attachment containing QuirkyLoader
cba8bb455d577314959602eb15edcaa34d0b164e2ef9d89b08733ed64381c6e0	File	Email attachment containing QuirkyLoader
catherinereynolds[.]info	Domain	Domain used for malspam campaign
mail[.]catherinereynolds[.]info	Domain	Domain used for malspam campaign
157[.]66[.]22[.]11	IPv4	IP address that catherinereynolds[.]info resolves to
103[.]75[.]77[.]90	IPv4	IP address related to QuirkyLoader

161[.]248[.]178[.]212	IPv4	IP address related to QuirkyLoader

IBM X-Force Premier Threat Intelligence is now integrated with OpenCTI, delivering actionable threat intelligence about this threat activity and more. Access insights on threat actors, malware and industry risks. Install the OpenCTI Connector to enhance detection and response, strengthening your cybersecurity with IBM X-Force's expertise. Stay ahead—integrate today.

The latest tech news, backed by expert insights

Stay up to date on the most important—and intriguing—industry trends on AI, automation, data and beyond with the Think newsletter. See the IBM Privacy Statement.

Thank you! You are subscribed.
