WildCard: The APT Behind SysJoker Targets Critical Sectors in Israel

: 11/27/2023

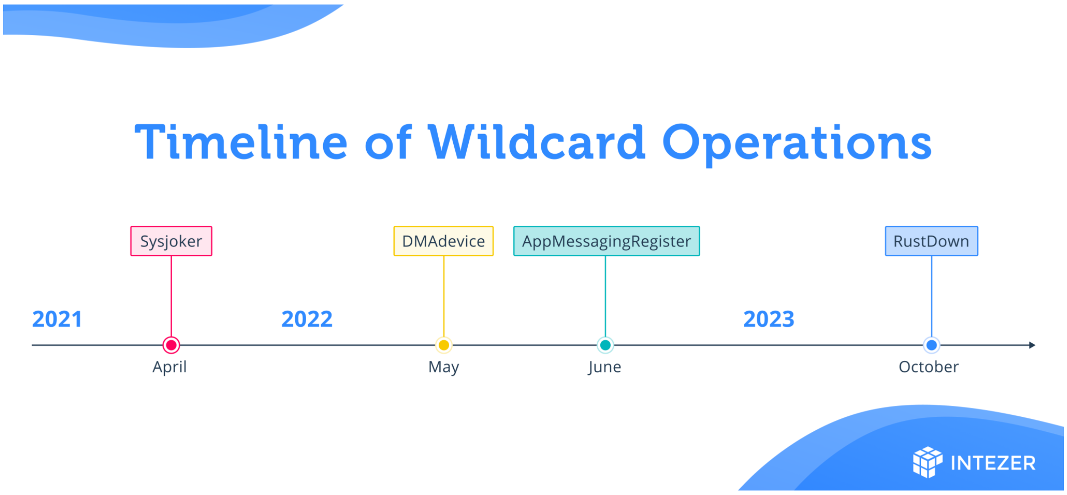
Written by Nicole Fishbein - 27 November 2023

Our research team has identified a new APT group, dubbed "WildCard," initially detected through its use of the SysJoker malware, which targeted Israel's educational sector in 2021. WildCard has since expanded its reach, creating sophisticated malware variants disguised as legitimate software, and a recently developed malware called 'RustDown,' written in Rust for potential operational advantages. Connections to Operation ElectricPowder indicate WildCard's advanced capabilities with a focus on critical sectors within Israel. While we've begun to understand WildCard's tactics and methods, their precise identity is still enigmatic, demanding deeper analysis and collaboration within the infosec community.

The current war between Israel and Hamas has brought increased interest in a variety of threats targeting Israel. Of course, this includes the usual suspects like Iranian, Hezbollah, and Hamas-affiliated groups that consistently target Israeli organizations and are likely to increase their operational tempo to match the current conflict. **We believe the shadow of a previously unidentified threat actor has slipped below the threshold and deserves greater attention.** Its first sighting began with our discovery of the SysJoker malware targeting the educational sector in Israel in 2021. Since then, the group behind SysJoker has evolved its tooling and targeting in important ways.

As we continued to track this threat cluster, we found previously undiscovered 2022 variants masquerading as 'DMAdevice' and 'AppMessagingRegistrar' software, both also written in C++. They share code and behavior patterns with our original discovery of SysJoker for Windows. Then in October 2023, we noticed a new malware written in Rust that shares behavioral traits with SysJoker. The developers refer to the malware as 'RustDown'. The original version of SysJoker was used to target Windows, macOS, and Linux machines, the migration to Rust might be an attempt to simplify multi-platform targeting in addition to making it harder to analyze.

We've also uncovered possible connections with ClearSky's Operation ElectricPowder. If proven, we see an actor displaying worrying capabilities and intent primarily targeting different critical sectors in Israel. To better describe the threat actor that ties these 3-4 different sets of activity together, we are clustering these sets of activity under the name **WildCard**. At this time, we can better describe WildCard's TTPs across multiple operations and variants, but attribution remains elusive.



Timeline of WildCard operations.

Technical Analysis

The Original SysJoker Malware

In January 2022, we published our discovery of SysJoker, an unattributed multi-platform backdoor leveraged against an educational institute in Israel. SysJoker masqueraded as a system update and generated its command-and-control by decoding a string retrieved from a text file hosted on GDrive. This dead drop resolver method is a consistent theme in the WildCard threat actor's future operations, along with naming their malware after legitimate components. Note that the development of C++ multi-platform backdoors is rare in the Middle East and aroused further suspicion about the nature of the unidentified malware developers.

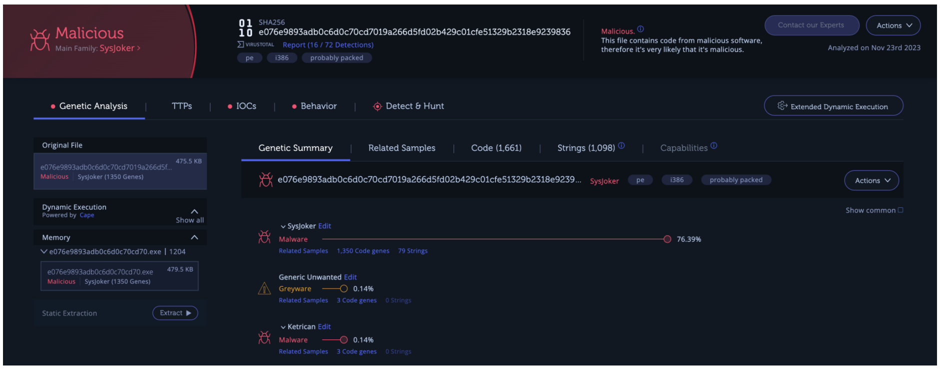
Previously Undiscovered Variants Appear

After our publication, the WildCard threat actor continued to evolve their malware, re-implementing some of the malware's behaviors to avoid detection and adding new capabilities. We found three samples of a malware variant written in C++. Two named DMAdevice.exe and one named AppMessagingRegistrar.exe. These variants were compiled five months after our original publication.

Hash	Compilation Timestamp	Filename
e076e9893adb0c6d0c70cd7019a266d5fd02b429c01cfe51329b2318e9239836	19 May 2022 18:07:42	DMAdevice.exe
6c8471e8c37e0a3d608184147f89d81d62f9442541a04d15d9ead0b3e0862d95	19 May 2022 18:05:18	DMAdevice.exe
67ddd2af9a8ca3f92bda17bd990e0f3c4ab1d9bea47333fe31205eede8ecc706	19 Jun 2022 20:20:06	AppMessagingRegistrar.exe

DMAdevice

Using Intezer Analyze we were able to identify code reuse between these samples and the original Windows SysJoker samples.



Malicious
Main Family: SysJoker

SHA256 e076e9893adb0c6d0c70cd7019a266d5fd02b429c01cfe51329b2318e9239836
VIRUSTOTAL Report (16 / 72 Detections)
pe i386 probably packed

Malicious.
This file contains code from malicious software, therefore it's very likely that it's malicious.

Analyzed on Nov 23rd 2023

Original File
e076e9893adb0c6d0c70cd7019a266d5f... 475.5 KB
Malicious | SysJoker (1350 Genes)

Dynamic Execution
Powered by Cape

Memory
e076e9893adb0c6d0c70cd70.exe | 1204
e076e9893adb0c6d0c70cd70.exe 479.5 KB
Malicious | SysJoker (1350 Genes)

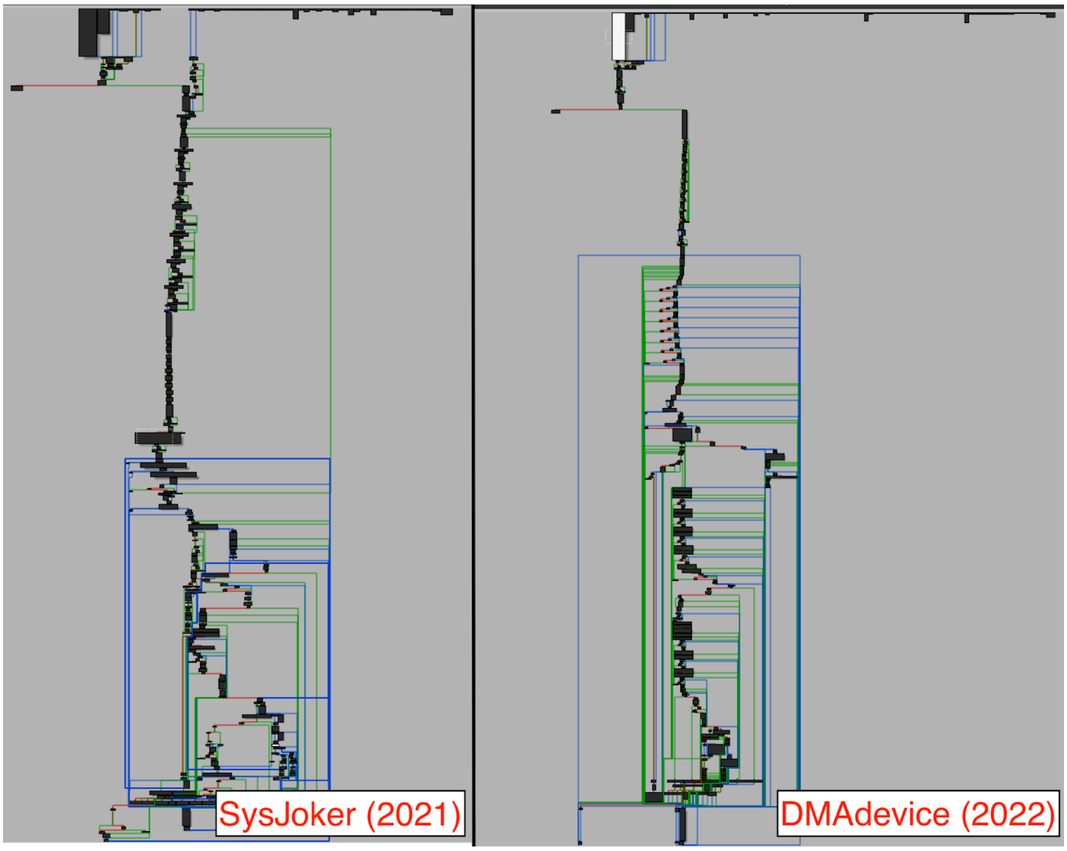
e076e9893adb0c6d0c70cd7019a266d5fd02b429c01cfe51329b2318e9239... SysJoker pe i386 probably packed

SysJoker
Malware 76.39%
Related Samples 1,350 Code genes 79 Strings

Generic Unwanted
Greyware 0.14%
Related Samples 3 Code genes 0 Strings

Ketrican
Malware 0.14%
Related Samples 3 Code genes 0 Strings

The structure of the main methods is largely similar, with some differences.



Comparison between SysJoker and the DMAdevice variant.

Besides the shared code, the two DMAdevice variants share a unique string with SysJoker, a custom alphabet:

0123456789ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghilmnopqrstuvmxyz

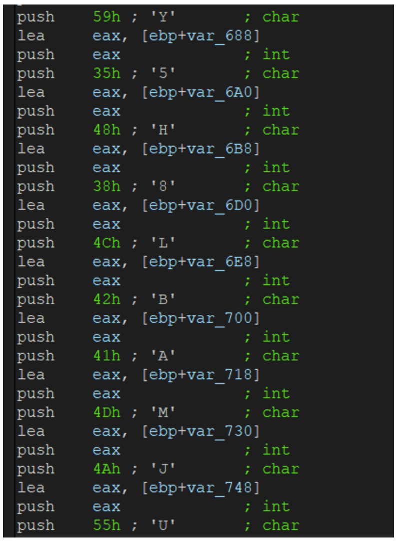
Note the missing ‘jk’ from the lowercase portion of the custom alphabet. This is likely a minor slip on the part of the developers but one that has consistently carried over into these newer variants.

Previous versions of SysJoker used GDrive as a dead drop resolver. The retrieved file content is base64 decoded before it is decrypted using a hardcoded RSA key as an XOR key. The decrypted data is the address of the intended C2 server.

The DMAdevice variant implements similar behavior but instead abuses OneDrive as its dead drop resolver. Meaning that the threat actors retained the usage of popular benign publicly-available services, unlikely to be blocked across the network while keeping the ability to rotate the C2 as needed. The use of OneDrive as a dead drop resolver continues into the RustDown variant.

Interestingly, they have decided to remove the use of the RSA key but keep the same scheme, replacing the key with a different string. In this case, after the stack string is built up, the used XOR key is:

QQL8VJUJMABL8H5YNRC9QNEOHA4I3QDAVWP5RY9L0HCGWZ4T7GTYQTCQTHTTN8RV6BMKT3AICZHOFQS8MTT



```
push    59h ; 'Y'       ; char
lea     eax, [ebp+var_688]
push    eax             ; int
push    35h ; '5'       ; char
lea     eax, [ebp+var_6A0]
push    eax             ; int
push    48h ; 'H'       ; char
lea     eax, [ebp+var_6B8]
push    eax             ; int
push    38h ; '8'       ; char
lea     eax, [ebp+var_6D0]
push    eax             ; int
push    4Ch ; 'L'       ; char
lea     eax, [ebp+var_6E8]
push    eax             ; int
push    42h ; 'B'       ; char
lea     eax, [ebp+var_700]
push    eax             ; int
push    41h ; 'A'       ; char
lea     eax, [ebp+var_718]
push    eax             ; int
push    4Dh ; 'M'       ; char
lea     eax, [ebp+var_730]
push    eax             ; int
push    4Ah ; 'J'       ; char
lea     eax, [ebp+var_748]
push    eax             ; int
push    55h ; 'U'       ; char
```

The XOR string being built on the stack.

The hardcoded User Agent string also changes, as follows:

```
Mozilla/5.0 (X11; CrOS x86_64 8172.45.0) AppleWebKit/537.36 (KHTML, like Gecko)
Chrome/51.0.2704.64 Safari/537.36
```

AppMessagingRegistrar

This variant was compiled after the DMAdevice version. While it also shares code with SysJoker, this variant implements different capabilities. For example, it uses multiple XOR keys to decode strings. This variant also uses different url paths:

- api/update
- /api/register
- /api/library
- /api/requests

However, similar to the other WildCard malware, it also uses OneDrive as a dead drop resolver. Also similar, AppMessagingRegistrar is downloaded from a server inside a ZIP file and is executed by a DLL file. The DLL file masquerades as Brave Browser.

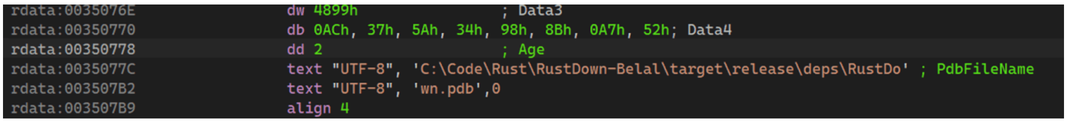
Company: Brave Browser
Product: Brave Browser (4.0.1.5)

Metadate of AppMessagingRegistrar executable.

RustDown: WildCard learns Rust

In October 2023, we discovered a new malware written in Rust. The sample is a 32-bit Windows executable masquerading as a PHP framework component. While the codebase is new, the malware consistently shares TTPs used by the WildCard threat actor in both SysJoker and its variants. The name of the malware is derived from the developers, as evidenced by a leftover PDB path:

```
– C:\Code\Rust\RustDown-Belal\target\release\deps\RustDown.pdb
```



```
rdata:0035076E                dw 4899h                ; Data3
rdata:00350770                db 0ACh, 37h, 5Ah, 34h, 98h, 8Bh, 0A7h, 52h; Data4
rdata:00350778                dd 2                    ; Age
rdata:0035077C                text "UTF-8", 'C:\Code\Rust\RustDown-Belal\target\release\deps\RustDo' ; PdbFileName
rdata:003507B2                text "UTF-8", 'wn.pdb',0
rdata:003507B9                align 4
```

RustDown PDB file path.

According to the PDB file the malware developers refer to this component as RustDown. Additionally, the term "*Belal*" in the folder path may be a transliteration of the common Arabic first name 'Bilal'. We treat this as a *low-confidence* indicator towards the identity of one of the WildCard developers.

Hash	Compilation Timestamp	Filename
d4095f8b2fd0e6deb605baa1530c32336298afd026afc0f41030fa43371e3e72	7 Aug 2023 10:43:32	php-cgi.exe

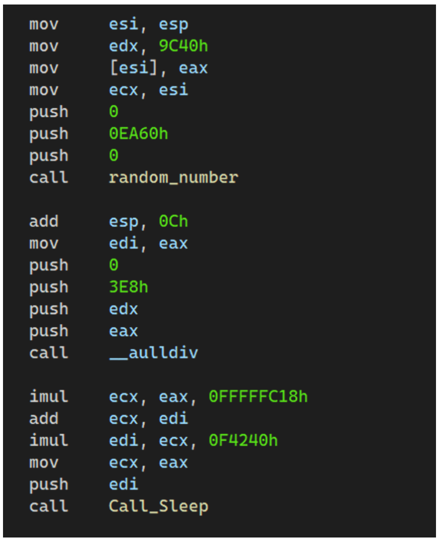
RustDown is intended to look like a legitimate PHP executable named php-cgi. PHP-CGI stands for PHP Common Gateway Interface. Providing an important tool that allows PHP to interact with a web server.

Company: The PHP Group
Product: PHP (7.4.19)

Metadata of RustDown executable.

As the name suggests, RustDown is a backdoor written in Rust and compiled for Windows operating systems. It uses OneDrive as a dead drop resolver.

RustDown implements multiple calls to the Sleep API using randomly chosen time durations, as seen in SysJoker.



```
mov     esi, esp
mov     edx, 9C40h
mov     [esi], eax
mov     ecx, esi
push    0
push    0EA60h
push    0
call    random_number

add     esp, 0Ch
mov     edi, eax
push    0
push    3E8h
push    edx
push    eax
call    __aulldiv

imul    ecx, eax, 0FFFFFC18h
add     ecx, edi
imul    edi, ecx, 0F4240h
mov     ecx, eax
push    edi
call    Call_Sleep
```

Function in RustDown that implements the Sleep functionality.

Next, the backdoor copies the executable to another location and sets up persistence by using a PowerShell command. Both the path and the PowerShell command are obfuscated in an attempt to evade detection. After decrypting the strings, we see that the malware copies itself to the following location, keeping with the theme of the legitimate PHP CGI tool:

> C:\ProgramData\php-7.4.19-Win32-vc15-x64\php-cgi.exe

Next, the malware decodes the PowerShell command that sets the registry value for persistence:

> “powershell” -Command “$reg=
> [WMIClass]’ROOT\DEFAULT:StdRegProv’;$results=$reg.SetStringValue(‘&H80000001′,’Software\Microsoft\Windows\Current\
> ‘php-cgi’, ‘C:\ProgramData\php-7.4.19-Win32-vc15-x64\php-cgi.exe’);”

The general mechanism for interacting with the Current User Hive involves using the identifier ‘&H80000001’ as described here. However, the specific command string it uses appears to be unique and relates to a separate campaign, referred to as Operation Electric Powder described further below.

Obfuscation

As mentioned, the malware encrypts its own strings in two different ways. The bulk of the remaining unobfuscated strings are artifacts of the static compilation of Rust dependencies linked within the binary:

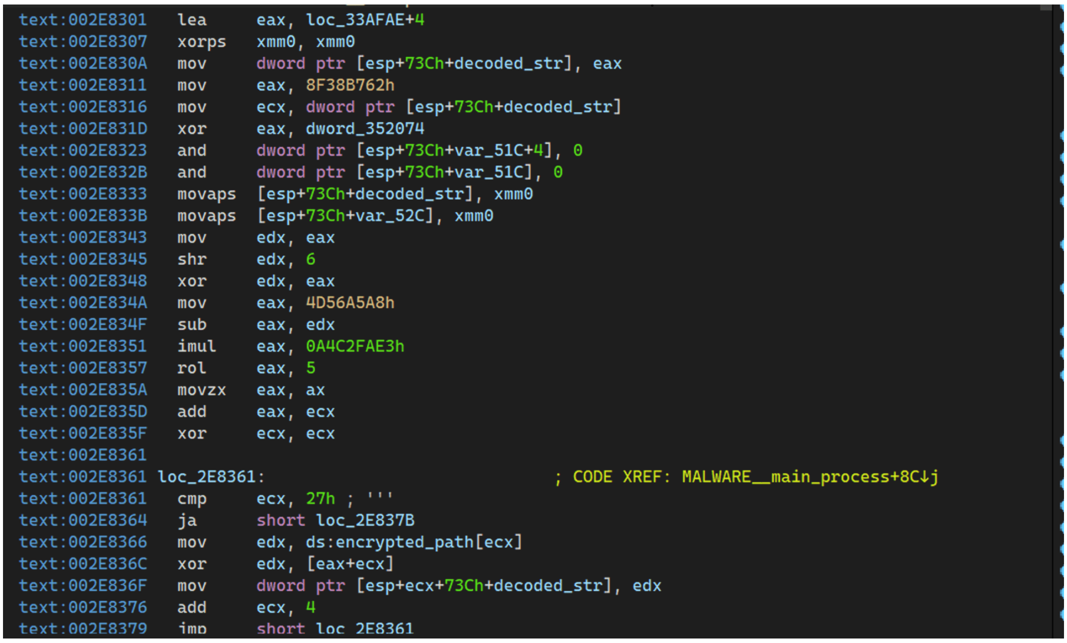
- base64-0.13
- curl-0.4.35
- rand-0.8.3
- rand_chacha-0.3.0
- rand_core-0.6.2
- rustc-demangle-0.1.21
- serde_json-1.0.64
- Whoami-1.1.1

The first type of obfuscation has the following scheme: first, decode the string with a standard Base64 scheme, unlike the first variant of SysJoker, and then decrypt the result with the following XOR key:

> QQL8VJUJMABL8H5YNRC9QNEOHA4I3QDAVWP5RY9L0HCGWZ4T7GTYQTCQTHTTN8RV6BMKT3AICZHOFQS8MTT

The same XOR key was used by the DMAdevice variant of SysJoker.

The malware decrypts additional strings using a XOR cipher, where each string is processed against a distinct key stream. The specific key for each string is determined by using fixed offsets from a table embedded within the malware’s code, combined with a calculation involving hardcoded numerical values and bitwise operations.



```
text:002E8301   lea     eax, loc_33AFAE+4
text:002E8307   xorps   xmm0, xmm0
text:002E830A   mov     dword ptr [esp+73Ch+decoded_str], eax
text:002E8311   mov     eax, 8F38B762h
text:002E8316   mov     ecx, dword ptr [esp+73Ch+decoded_str]
text:002E831D   xor     eax, dword_352074
text:002E8323   and     dword ptr [esp+73Ch+var_51C+4], 0
text:002E832B   and     dword ptr [esp+73Ch+var_51C], 0
text:002E8333   movaps  [esp+73Ch+decoded_str], xmm0
text:002E833B   movaps  [esp+73Ch+var_52C], xmm0
text:002E8343   mov     edx, eax
text:002E8345   shr     edx, 6
text:002E8348   xor     edx, eax
text:002E834A   mov     eax, 4D56A5A8h
text:002E834F   sub     eax, edx
text:002E8351   imul    eax, 0A4C2FAE3h
text:002E8357   rol     eax, 5
text:002E835A   movzx   eax, ax
text:002E835D   add     eax, ecx
text:002E835F   xor     ecx, ecx
text:002E8361
text:002E8361 loc_2E8361:                             ; CODE XREF: MALWARE__main_process+8C↓j
text:002E8361   cmp     ecx, 27h ; '''
text:002E8364   ja      short loc_2E837B
text:002E8366   mov     edx, ds:encrypted_path[ecx]
text:002E836C   xor     edx, [eax+ecx]
text:002E836F   mov     dword ptr [esp+ecx+73Ch+decoded_str], edx
text:002E8376   add     ecx, 4
text:002E8379   jmp     short loc_2E8361
```

Decryption of strings using unique key stream.

Communication With the C2

The communication with the C2 starts with the decoding of a dead drop resolver. This is performed using the first decoding method and the hardcoded XOR key and Base64. The backdoor sends an HTTP Get request to the following resolved URL:

> https://onedrive[.]live.com/download?
> resid=16E2AEE4B7A8BBB1%21112&authkey=!AED7TeCJaC7JNVQ

The backdoor uses a custom user agent. This is similar to the earlier version of SysJoker, which also communicated using a specific, hardcoded user agent:

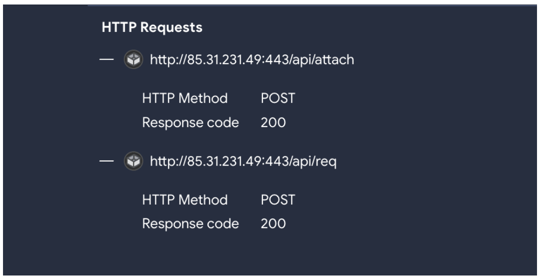
> Mozilla/5.0 (Windows NT 10.0; WOW64) AppleWebKit/537.36 (KHTML, like Gecko)
> Chrome/92.0.4515.159 Safari/537

We were able to get a response from the resolver:

> KnM5Sjpob2glNTY8AmcaYXt8cAh/fHZ+ZnUNcwdld2Mr

The response is encoded using Base64 and XOR-ed with the same key that was used in the previous step. The decrypted result is the IP address of the C2: {“url”:”http://85.31.231[.]49:443″}. During our investigation, we did not find other C2 domains that were served by this OneDrive link.

Next, the malware communicates with the C2 using the HTTP protocol. The URL is formatted in the following way: <C2 domain>/api/<command>. In RustDown, we identified two commands attach and req.



HTTP Requests

— http://85.31.231.49:443/api/attach

HTTP Method POST
Response code 200

— http://85.31.231.49:443/api/req

HTTP Method POST
Response code 200

VirusTotal behavior analysis of RustDown showing the connection method to the C2.

Like the original version of Sysjoker, the RustDown will decode the C2 and send the collected user’s information to the C2’s /api/attach path as an initial handshake. The information sent over has the following structure:

> “ip”:”[Local IP Address]”
> “serial”:”[Host Name]_[Serial Number]_[Username]”
> “name”:”[Username]”
> “os”:”[Operating System Version]”
> “user_token”:”[User Token]”

This differs from the fields used in SysJoker which included an unused ‘av’ field.

```
“sn”: “[Serial Number]”
“us”: “[Username]”
“os”: “[Operating System Version]”
“av”: *Unused
“ip”: “[Local IP Address]”
```

RustDown will send requests to the C2’s /api/req path after registration, similar to older versions. The response from the C2 is a JSON detailing an array of tasks to perform. These instructions include actions along with specific URLs to download a zip archive containing executables and save it at C:\\ProgramData\\php-Win32-lib with the filename specified in the JSON. To unzip the payload, RustDown decryptes another PowerShell command.

Connections to Operation ElectricPowder

In the process of our investigation, we found an interesting set of connections between the newer SysJoker variants (particularly the ‘DMA Device’) and components of Operation ElectricPowder. The latter was an attack that targeted the Israeli Electric Corporation (IEC) in 2016-2017. In both cases, the following specific string is deobfuscated during execution and used to establish persistence.

```
powershell\” -Command \”$reg=
[WMIClass]’ROOT\\DEFAULT:StdRegProv’;$results=$reg.SetStringValue(‘&H80000001′,’Software\\Microsoft\\Windows\\Curre
<process>, <path>
```

The general mechanism to resolve the Current User hive is described here. However, the way the command string is implemented appears to be limited to these malware sets, suggesting a developmental connection across nearly four years.

Additionally, once we discovered RustDown, we found that it also dynamically resolves this PowerShell command string and uses it to achieve persistence. This further strengthens the hypothesis that Operation ElectricPowder may have been the earliest appearance of the WildCard threat actor.

Infection vectors

In our publication of SysJoker, we suspected that the threat actors used an infected npm package to deliver SysJoker. With the discovery of new versions, we see that this pattern of masquerading as legitimate software continues among all of the components of WildCard.

Now with the DMAdevice, AppMessagingRegistrar variant, and Rustdown, we see a pattern of using legitimate services to masquerade the malware. We can assume WildCard uses phishing campaigns to convince victims to download their malware.

As mentioned, part of WildCard’s operations share behavioral patterns with Operation ElectricPowder. This malware also masqueraded as legitimate software and used an elaborate and diverse phishing campaign, including decoy news sites and Facebook profiles.

If the connection between the two operations is solid, it supports WildCard’s investment in extensive social engineering campaigns to reach their targets. The early malware of Operation Electric Powder was poorly disguised as legitimate Microsoft components. SysJoker variants were more elaborately disguised as benign applications or web development components with names reminiscent of TypeScript projects. The newest iteration follows in that web development tool theme by disguising RustDown as a PHP CGI component. At this time, we have not discovered the latest infection vector but feel that these TTPs suggest possible targeting of developer communities in Israel with trojanized applications.

The detection of SysJoker traces back to a 2021 incident at an Israeli educational institution. After analyzing the malware, we identified behavioral patterns akin to those of another malware variant that previously targeted Israeli infrastructure. This similarity points to a deliberate pattern of victim targeting shared between the two types of malware.

Network infrastructure

Another interesting TTP connecting different WildCard operations is the abuse of benign web services as dead drop resolvers or C2 hosting. First-stage components consistently reach out to services like GDrive or OneDrive to receive text that is decoded into the address of the intended C2. The threat actor has used a number of hosting providers to host their C2 infrastructure, most recently Hostinger. During analysis, we found that the C2 is possibly geofenced to respond only to IP addresses from Israel, further supporting our sense of WildCard’s targeting.

Conclusion

As we continue to monitor the threat landscape surrounding the ongoing Israeli-Hamas war, it’s important to emphasize the existence of non-traditional threat actors like WildCard that have slipped below the radar. At the time of our initial discovery of SysJoker, we were missing the necessary components to bring this threat actor into view fully. As additional variants were discovered, we found connections to a notable earlier campaign targeting electric power generation in Israel. More recently, the WildCard developers have undertaken the popular move from C++ to Rust. Despite having to start their project over in Rust, RustDown also shows the same specific traits as newer SysJoker variants and older ElectricPowder components. Clustering these different sets of activities showcases an APT group consistently targeting Israeli critical sectors like education, IT infrastructure, and possibly electric power generation active to this day.

I would like to extend my sincere gratitude to Juan Andres Guerrero-Saade and Ryan Robinson for their contributions to the development and refinement of this blog.

IOCs

Rustdown

d4095f8b2fd0e6deb605baa1530c32336298afd026afc0f41030fa43371e3e72

DMAdevice (SysJoker May 2022 Variant)

e076e9893adb0c6d0c70cd7019a266d5fd02b429c01cfe51329b2318e9239836

6c8471e8c37e0a3d608184147f89d81d62f9442541a04d15d9ead0b3e0862d95

AppMessagingRegistrar (SysJoker June 2022 Variant)

67ddd2af9a8ca3f92bda17bd990e0f3c4ab1d9bea47333fe31205eede8ecc706

SysJoker Downloader

96dc31cf0f9e7e59b4e00627f9c7f7a8cac3b8f4338b27d713b0aaf6abacfe6f

Dead Drop Resolver URL

https://onedrive.live[.]com/download?resid=16E2AEE4B7A8BBB1%21112&authkey=!AED7TeCJaC7JNVQ (RustDown)

https://onedrive.live[.]com/download?cid=F6A7DCE38A4B8570&resid=F6A7DCE38A4B8570%21115&authkey=AKcf8zLcDneJZHw (DMAdevice.exe)

https://onedrive[.]live.com/download?cid=3014636895E3FE3B&resid=3014636895E3FE3B%21106&authkey=AD4OGrVz9h17Jzo (AppMessagingRegistrar.exe)

C2

85.31.231[.]49:443 (Rustdown)

sharing-u-file[.]com (DMAdevice.exe)
audiosound-visual[.]com (AppMessagingRegistrar.exe)filestorage-short[.]org (SysJoker Downloader)

Nicole Fishbein

Nicole is a malware analyst and reverse engineer. Prior to Intezer she was an embedded researcher in the Israel Defense Forces (IDF) Intelligence Corps.