Spectre (SPC) v9 Campaigns and Updates

By Jason Reaves

Published: 2024-06-19 · Archived: 2026-04-05 16:11:19 UTC

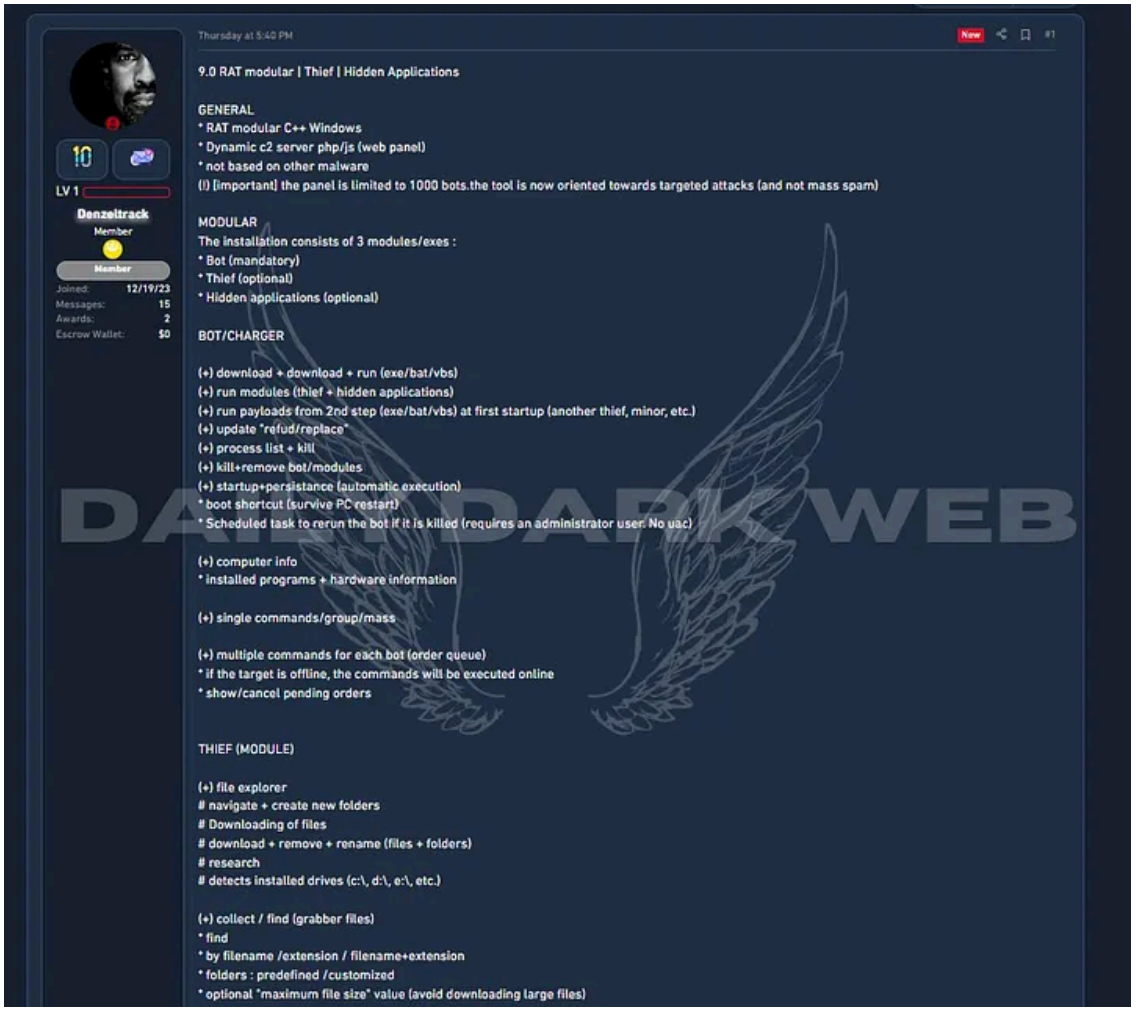
By Jason Reaves and Joshua Platt

Spectre RAT was previously discussed a few years ago[1] in an excellent overview by Yoroi but recently has resurfaced in campaigns being distributed on livechat-files[.com[3] using code signing certificates.

One noteworthy trend with their code signing certificates was their ability to stay undetected for far longer than some of the traditional mass spam campaigns where the certs and AV detections were generally corrected by the next day.

The advert for Spectre RAT v9 confirms that it is primarily designed for targeted attacks:

> Press enter or click to view image in full size



Campaign

```
First Submission:  2024-05-29 18:05:16 UTC
Compilation TimeStamp: 2024-05-27 14:30:30 UTC
SHA-256: f90d1716de7244f368a81d2b9d247c2b6213447aee6da606267edceef0cc1377
Code Signing Certificate
Name: Xi'an Jiashi Xinnuo Information Technology Co., Ltd.
Issuer: Certum Extended Validation Code Signing 2021 CA
Valid From: 2024-05-10 05:35:18
Valid To: 2025-05-10 05:35:17
Valid Usage Code Signing
Algorithm: sha256RSA
Thumbprint: C2016ABA9447FCB75B03F158B31EAC7D76262377
Thumbprint: MD5 ACD454260943CF6CD1357DF75DB109D0
Thumbprint:
SHA256 0777CE1ACD929ED7A1DF146BEA6126DAADA3EE564A4D57CAF924B4BEADFC8FB3
Serial Number 34 1D FC 31 CA 4B DB B1 82 4E 25 4B CD 5B 59 E0
IP: 91.92.240[.]40
Domain: serowakrasolaristic[.]xyz
```

The following files were also signed with the same code signing certificate:

```
First Submission: 2024-06-03 16:16:36 UTC
Compilation TimeStamp: 2024-05-27 14:51:43 UTC
SHA-256: 84499164a4848a100a22361f38d36ddaea66d01d2e68580271692f9a6fc2a570
IP: 91.92.240[.]40
First Submission: 2024-06-04 00:28:31 UTC
Compilation TimeStamp: 2024-05-01 16:54:39 UTC
SHA-256: aed440f54dc3f39d5eff26ff4eee34f991750bff7b2b7031260cd2cdd43339dd
```

Using the cloud file hosting domain *cdn.livechat-files[.]com* as a pivot point, we were quickly able to track back an initial launch date of May 15 2024, with the initial redirect domain being *cdn-namecheap[.]com.* The file details associated with the first sighting of this campaign are listed below:

```
First Submission: 2024-05-11 03:22:14 UTC
Compilation TimeStamp: 2024-05-01 16:58:45 UTC
SHA-256:  37c495acbd56aa54755e1a69c5f0bd4edfe758c1b627ca8185196378f3314f45
Code Signing Certificate
Name: JauiInderte Agiletron Information Technology Co., Ltd.
Issuer: GlobalSign GCC R45 EV CodeSigning CA 2020
Valid From: 2024-01-31 01:42:53
Valid To: 2025-01-31 01:42:53
Valid Usage Code Signing
Algorithm: sha256RSA
Thumbprint: D0C7D82E733D076804E5DFF6FB93069D2F9CB192
Thumbprint: MD5 0BD0D08DAEABFD4B060DD4486EE7A068
Thumbprint: SHA256 0846ECB892A26A8804A58C9122FFB7BEA31A47387A2452765B50058890F88ABA
Serial Number: 0C 6D 55 B6 A1 9A C5 AD 30 52 EF 24
```

The following files were also signed with the same code signing certificate:

```
First Submission: 2024-05-19 17:24:50 UTC
Compilation TimeStamp: 2024-05-02 18:22:05 UTC
SHA-256: 94827a4ab543972eacee8e610ec94d8469de43fe8dc0302015f1c587b158025d
IP: 91.92.240[.]40
Domain: serowakrasolaristic[.]xyz

   First Submission: 2024-05-23 17:19:24 UTC
Compilation TimeStamp: 2024-05-14 10:33:22 UTC
SHA-256: 8ce3bc41fb200cf7ba41f6b0d9dc976126dc3a4271a1e3b5725c80f3bd031738
IP: 91.92.255[.]73
Domain: holosymmetryspecscollunbeatable[.]xyz

First Submission: 2024-02-03 07:50:41 UTC
Compilation TimeStamp: 1992-06-19 22:22:17 UTC
```

```
SHA-256: 500670f00b1e99426a3f5a49634475b69e3bca76442f7ad6db3b082fd094aecb
IP: 80.79.4[.]144

First Submission: 2024-02-05 19:51:53 UTC
Compilation TimeStamp: 1992-06-19 22:22:17 UTC
SHA-256: b79199586df6a084fe73ec610858f2965b835c06a0761f44e771b6f8c247067e
IP: 80.79.4[.]144
```

After observing the two month gap between signed files, we noted a similar but slightly different hosting mechanism used to deliver the file from early February. While the hosting platform was the same, the distribution domain instead utilized *cdn-staging.livechat-files[.]com*. This led to another signed SpectreRAT sample, which aligned with the previously uncovered campaigns and pushed the timeline back to early January 2024. The code signing certificate also appeared to follow the same sequence as the previous samples.

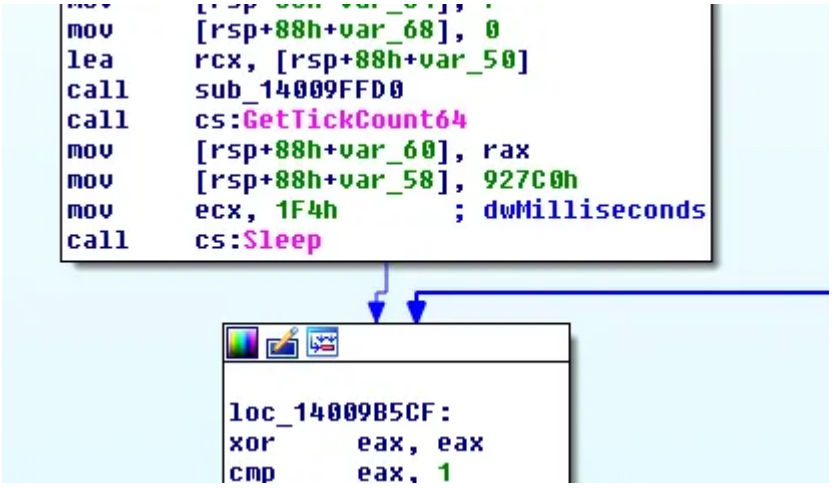
```
First Submission: 2024-01-10 05:56:57 UTC
Compilation TimeStamp: 2024-01-03 12:38:59 UTC
SHA-256: 9bee19ac1946bc15dd7de3027d0b9ede2e92beaa246fb21d65e6faf817682106
Code Signing Certificate
Name: Mutiix QuansumKeep Information Technologies Co., Ltd.
Issuer: GlobalSign GCC R45 EV CodeSigning CA 2020
Valid From: 2024-01-03 08:19:05
Valid To: 2025-01-03 08:19:05
Valid Usage Code Signing
Algorithm: sha256RSA
Thumbprint: 8282D32D753A4E0BBA8057D7D6835F103B8D6530
Thumbprint: MD5 4D85FD3EEC6CCF4C907113E62DB0E4F2
Thumbprint: SHA256 4E3A1FB1BE71D954173003EDB79A06CD17F9AC8319BA3115BE277CDAB0A3BF92
Serial Number: 4A 6C E4 49 DE 5C 97 48 35 DE 71 64
IP: 91.92.241[.]187
Domain: dystopianoverbiassperple[.]com
```

Spectre

The crypter leverages timing checks mixed with GetTickCount and Sleep wrapped around a block of function calls, the idea here is that in virtual machines some functionality takes drastically shorter to accomplish than it does on a real machine. In this case the actions being leveraged are allocating memory on the heap and then freeing it. To make it look more innocuous, they are also getting the foreground window name and copying it into newly allocated memory off the heap while converting it to ascii.

Setup:

> Press enter or click to view image in full size



End of the loop after the heap manipulation:

This isn't a new technique, it was previously leveraged by a crypter being used by Locky[2].

Get Jason Reaves's stories in your inbox

Join Medium for free to get updates from this writer.

Remember me for faster sign in

The crypter also leverages TIMEOUT calls which are packaged into the unpacking routines:

The crypter will alsomove itself if it is not running as a hardcoded filename before restarting:

```
"C:\Windows\System32\cmd.exe" /c ping localhost -n 6 > nul & del "C:\Users\user\Desktop\mal.exe" & "
```

Once unpacked, the Spectre sample has a basic string encoding setup as a simple single byte XOR. However, they also rebuild the data before decoding it, making it slightly harder to properly signature on and decode all the relevant strings. One needs to rebuild them first based on the way they are loaded during the rebuild process.

Relevant decoded strings:

```
OzEsMTIsMDYwLDYy
cWVwZ3djaXBhcW1uaXJrcXRrYSx4e3I=
YWF7ZmFwZmFlcGN2Z2R3cWVxYWNzYWlgZWxzLHhxeg==

04-29
lyqi.dll
wlmxz
```

```
F44BE522-0833-28F5-5508
eygkp
wsbic

chgj.php
jtez.php
pefb.zip
pefb_nonir.zip
roed.zip
roed_x64.zip
xofq.exe
eyrd=
&tucy=
&pvwz=
&ykam=
&byul=
&dcfl=
&oghd=
&vhup
&pthq=
&yhtz=
&dybj=
&klne=
&jlgo=
&aicj=
&qube=
&wjba=
&wrja=
?myqg=
ehmn
aej
g
/v
down/
\\Microsoft\\Windows\\Start Menu\\Programs\\Startup\\
nircmdc.exe
zip.exe
/c ping localhost -n 6 > nul &
/c ping localhost -n 10 > nul &
cout

http://
true
false
void
.asd
&
@
```

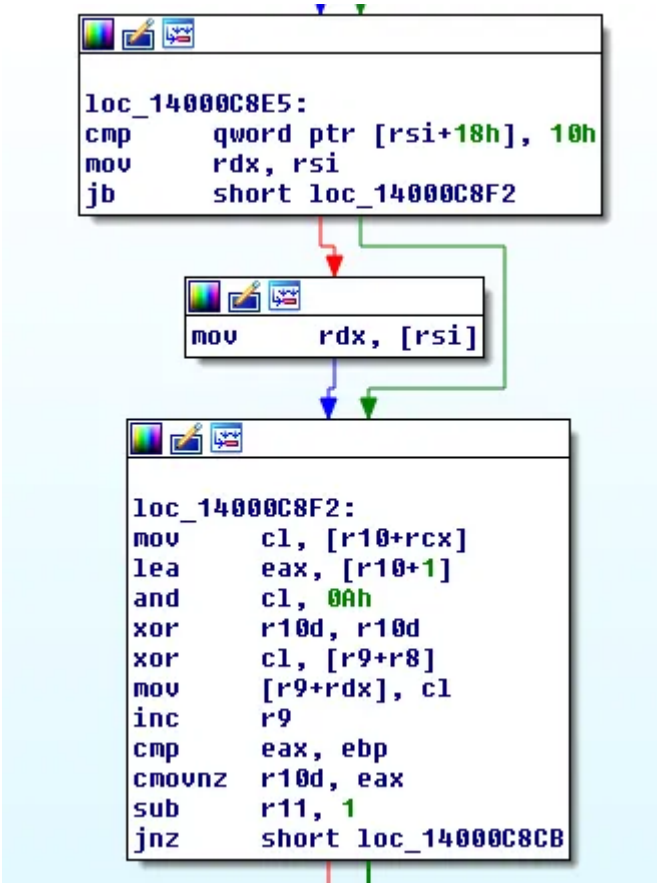
```
[@]
|
~
[|]
[*]
.png
.exe
.lnk
.vbs
.txt
.7z
.bak
*

 --headless=old --disable-gpu --remote-debugging -port=0
 MyTasks\\
OnyxGraphicsKit
```

Most of the main functionality resides in function tables which are called in sequence. Dummy or placeholder functions can be found in many of the tables:

```
dq offset sub_140020150
dq offset Dummy_140025B60
dq offset sub_140020150
dq offset sub_140029908
dq offset sub_14002D790
dq offset sub_14002D904
dq offset sub_140025544
dq offset sub_140025544
dq offset sub_140025B5C
dq offset Dummy_140025B60
dq offset nullsub_1
dq offset sub_14000BB10 ;
                        ;
dq offset sub_14000D08C
dq offset sub_14000D30C
dq offset sub_14001FFD0
dq offset sub_140020154
dq offset Dummy_140025B60
```

The only component that gets additional encoding is the C2 addresses which are hardcoded in the binary, C2 decoding involves a hardcoded string used as a key for the following:

Press enter or click to view image in full size



A demonstration of this decoding using Python is provided below:

```
>>> def decode(c2):
...   a = bytearray(base64.b64decode(c2))
...   key = bytearray('61C8EB3FE72795B6DBF7A787D5020913')
...   for i in range(len(a)):
...     temp = key[i]
...     temp = (temp & 0xa)
...     a[i] ^= temp
...   return(a)
...
>>>
>>> decode('cWVwZ3djaXBhcW1uaXJrcXRrYSx4e3I=')
bytearray(b'serowakrasolaristic[.xyz')
>>> decode('YWF7ZmFwZmFlcGN2Z2R3cWVxYWNzYWlgZWxzLHhxeg==')
bytearray(b'caynardceratodusescascabels[.xyz')
>>> decode('OzEsMTIsMDYwLDYy')
bytearray(b'91.92.240[.40')
```

Debug string:

```
C:\DEV\SPC\DEV\v9\
```

IOCs

IPs:

```
179.43.142[.]145
179.43.142[.]190
193.233.185[.]133
193.233.191[.]162
209.182.227[.]122
213.139.205[.]131
185.225.74[.]131
91.92.255[.]73
91.92.247[.]196
94.156.69[.]212
94.156.64[.]35
91.92.250[.]157
91.92.240[.]40
91.92.244[.]110
91.92.243[.]158
91.92.255[.]84
91.92.244[.]110
94.156.65[.]162
91.92.241[.]187
```

Domains:

```
holosymmetryspecscollunbeatable[.]xyz
gonorhynchidaeanalgesidaefascinatedly[.]xyz
cyanoauricharesstealthful[.]xyz
expansivenessburnishesitel[.]xyz
serowakrasolaristic[.[xyz
symphoniesreinflatablexerodermatic[.]com
pandemoniumpleurolysishummus[.]xyz
electivesprotagonmillenary[.]xyz
chairermisassayssebate[.]xyz
impersuasiblyredeliveranceunspleened[.]com
ponticcyclersrecubate[.]com
sappedisomorphousnonappreciativeness[.]com
evanescingunsatanicallychrysal[.]com
pharyngologicalpseudoanginaperpetrable[.]com
dystopianoverbiassperple[.]com
cdn.livechat-files[.]com
cdn-staging.livechat-files[.]com
```

References

1: https://yoroi.company/en/research/spectre-v4-0-the-speed-of-malware-threats-after-the-pandemics/

2: https://github.com/sysopfb/VM_Timing_Detect

3: https://urlscan.io/search/#filename%3A%22.scr%22%20AND%20domain%3Alivechat-files.com

4: https://x.com/DailyDarkWeb/status/1740825011932573712
