Janicab Series: Attibution and IoCs

malwarology.com/2022/05/janicab-series-attibution-and-iocs/

May 31, 2022

2022-05-31

Malware Analysis , Janicab

In late April 2022, I was requested to analyze a software artifact. It was an instance of Janicab, a software with infostealing and spying capabilities known since 2013. Differently to other analyses I do as part of my job, in this particular case I can disclose parts of it with you readers. I'm addressing those parts in a post series. Based on this specific sample, here I'going to discuss a bit about the attribution. Furthermore, I'm going to collect the Indicators of Compromise (IoCs) related to thi specific infection chain. If you want to know more about the various stages of the infection, I recommend you reading the previous posts of this series: first part, second part, and third part.

Attribution

Given the body of knowledge consisting of the artifacts involved in the infection chain, provided by the previous sections, I am now in a good position to briefly discuss why I believe to have dealt with a Janicab instance. Janicab was first disclosed in 2013 by F-Secure Labs. The name Janicab appears in their first publication on this topic within the signature created by the analysts for the antivirus product: `Backdoor:Python/Janicab.A` . That publication is about a malware targeting Mac operating systems. Despite of its briefness, the article is long enough for us to observe a distinctive technique adopted by the malware: the use of C2 published on social media like YouTube. In that primordial case, the C2 url was directly published as a YouTube video description. I discuss a very similar technique, regarding an artifact belonging to this infection chain, in this post. F-Secure Lab published another article about Janicab in 2015. The similarities between the sample discussed in this report and that one addressed by F-Secure Lab post are manifold. Most of the similarities are about the techniques:

- Use of a LNK file with hidden target arguments as a first link of the infection chain.
- C2 ip address obtained starting from a numeric seed posted in a YouTube comment to a video. In the case discussed by F-Secure analysts, the comment pattern was slightly different: `our (.*)th psy anniversary` .
- Same conversion function from the C2 numeric seed to the C2 ip address.
- Same C2 resources and requests parameters.

An interesting and relevant post concerning Janicab was published by Securelist (Kaspersky) in 2020. In this publication, the analysts claim that Janicab is operated by the same group as Powersing and Evilnum malware. The claim is supported by several observations such as:

- Distribution via LNK files embedding other artifacts.
- C2 obtained from dead drop resolvers with regular expression matched on public posts.
- Partial code overlap and/or code similarities.

Although I don't have access to a reliable source of information concerning Janicab victimology, the claim made by Kaspersky analysts provides for some potentially interesting leads. By including Evilnum and Powersing operations and targets, they hypothesize that the group behind Janicab acts as a mercenary outfit mostly involved in intelligence operations. The main targets seem to be law firms and fintech companies.

Indicators of Compromise

Indicator	Type	Artifact
7057bcfa5d994af8829819762643e8ae	MD5	SMTP-error.txt.lnk
b2aaa5c7b64231dbf25c0fac70eb9d7118468b2f	SHA1	SMTP-error.txt.lnk
e4a000e5d39ca4915cbe2f0dd4dcd17fc9a6f0b059634b37d39c18f40cb2773f	SHA256	SMTP-error.txt.lnk
fb4a625c222ef53201e224b48d3f3f28	MD5	cab.cab
9e145251a4fd70c3de7d0b397115ed49c669dc87	SHA1	cab.cab
5c5d2aab69939c6a6037f2e93de32d8ffe8cbcf602578e89784038e141f0b515	SHA256	cab.cab
2fec5b88e18705db18310a52e495c6aa	MD5	2.vbe
031e1981c18a55015abc3eac4ef1162e4bfd0fa8	SHA1	2.vbe
e4210de7e526bdb7661d7631edc4f84a66eb361935f4b7412e63074ca76f4b40	SHA256	2.vbe
a927e643f42ee4f979c03346e9142bfb	MD5	.vbe
a08e881bb1d73764becffc49930b4093ba1dc8a7	SHA1	.vbe
1a55fdf465ec4a4565a12fc44d48308545884f4cfa545c524529792dcdac81b4	SHA256	.vbe
d627882fd4311454646e6f653e2ae0cc	MD5	k.dll
dfa7f4b0647170712a5b7ff3d7ee03c5ef2d7f2d	SHA1	k.dll
192f058c4d756b9e4f3779b8dd880064cadbf5f8bb43529599b7f4a29c4770cf	SHA256	k.dll
3b91704b9d500f33019d3d2bb43f3d46	MD5	SMTP-error.txt
dc3bf7b3ff83a12a5e8120f800d067cc9adde46d	SHA1	SMTP-error.txt
7e4df228c9d9c84fcac9474798d71f053cb217336e784dde84d5cb1242f19575	SHA256	SMTP-error.txt
d822313bbed34ac72451d3174ec06937	MD5	replacer.py
b0c20bb39a559d378f989161365ebd826000dff7	SHA1	replacer.py
6b3e2feeb3fafe32586f547296028fcaabd32fdae0cddf18afbf68523ce0d7ff	SHA256	replacer.py
e0c0c90742083433b2adbbb13f9286e6	MD5	MicrosoftMicrosoft Sync Services.lnk
de0e5b035d214b47c722b9fc985d58145f2b3e18	SHA1	MicrosoftMicrosoft Sync Services.lnk
2b7dd592b5a3c756ff109d83707ac36717fb577d19369dbb0e30c4f9cc01a8a2	SHA256	MicrosoftMicrosoft Sync Services.lnk
2b3d0c7fd1f3a7abd6d016f7eaa1c0d2	MD5	runOnce.reg
4810dc9dead5d4ec82e147363d70d5cc5feb0083	SHA1	runOnce.reg
5e989c4940741407f04bb7a630c0a41af8738dd377a395936a3652308ca1f68f	SHA256	runOnce.reg

Indicator	Type	Artifact
3cafc122e092ba0d0ef446882ebdf07a	MD5	vista.reg
0d9138fad68568d6cb139735b18d0de85c8ad311	SHA1	vista.reg
646f87d1fdc1b63d558b739aca164e24812ac668c9016185b985ec5f8816c22c	SHA256	vista.reg

This post closes the series about Janicab. As always, if you want to share comments or feedbacks (rigorously in broken Italian or broken English) do not esitate to drop me a message at **admin[@]malwarology.com**.