“Nanocore Malware” detection

Malware github link - https://github.com/InQuest/malware-samples/tree/master/2019-06-Nanocore
Analysis - https://github.com/its-verred/Threat-Intel/tree/main/Yara/Nanocore

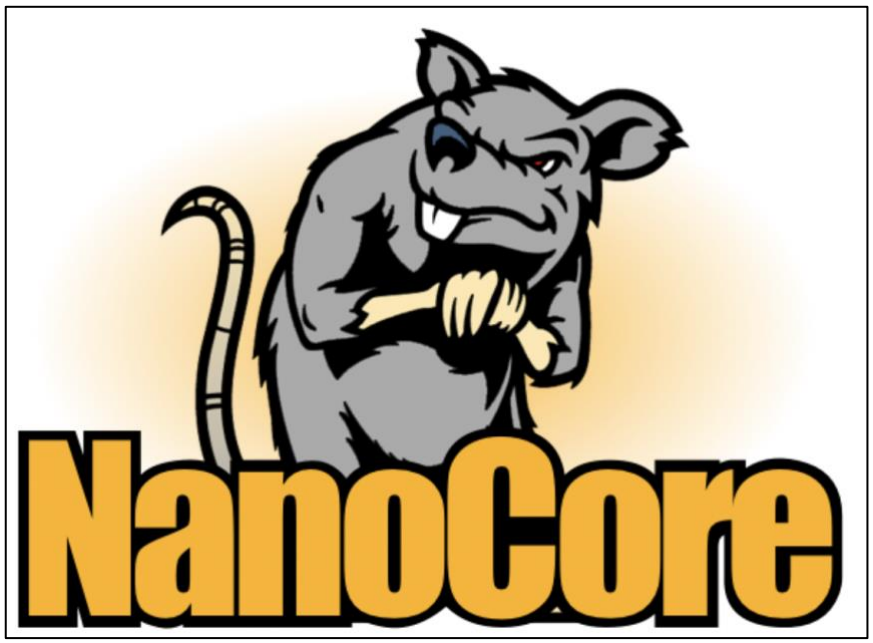


Table of Contents

Remote Access Trojan (RAT) ***2***

Nanocore ***2***

How Does NanoCore RAT Work? **4**

Analysis **5**

Yara rule: **7**

References ***8***

Remote Access Trojan (RAT)

A remote access Trojan (RAT) is a malware program that includes a back door for administrative control over the target computer. RATs are usually downloaded invisibly with a user-requested program -- such as a game -- or sent as an email attachment. Once the host system is compromised, the intruder may use it to distribute RATs to other vulnerable computers and establish a botnet.

Because a RAT enables administrative control, it makes it possible for the intruder to do just about anything on the targeted computer, including:

- Monitoring user behavior through keyloggers or another spyware.
- Accessing confidential information, such as credit card and social security numbers.
- Activating a system's webcam and recording video.
- Taking screenshots.
- Distributing viruses and other malware.
- Formatting drives.
- Deleting, downloading, or altering files and file systems.

Various flavors and versions of these RATs are freely available and easily modified to fit the unique requirements of any given attack. A few examples from a much larger list of popular RATs include Poison-Ivy, JRAT, NjRAT, Orcust-RAT, CyberGate, DarkComet, DreamWare, BlackShades, NetWire, NanoCore.

Nanocore

NanoCore is a high-risk RAT that provides attackers with details on the device name and OS. This information is used to carry out various malicious activities, such as manipulating confidential files, hijacking webcam, and microphone, stealing login credentials and more.

NanoCore RAT comes with a few base plugins and the ability to expand its functionality, so threat actors can develop additional features for other malicious actions. There is already a wide range of NanoCore plugins available online that can be used for cryptocurrency mining, ransomware attacks, and more.

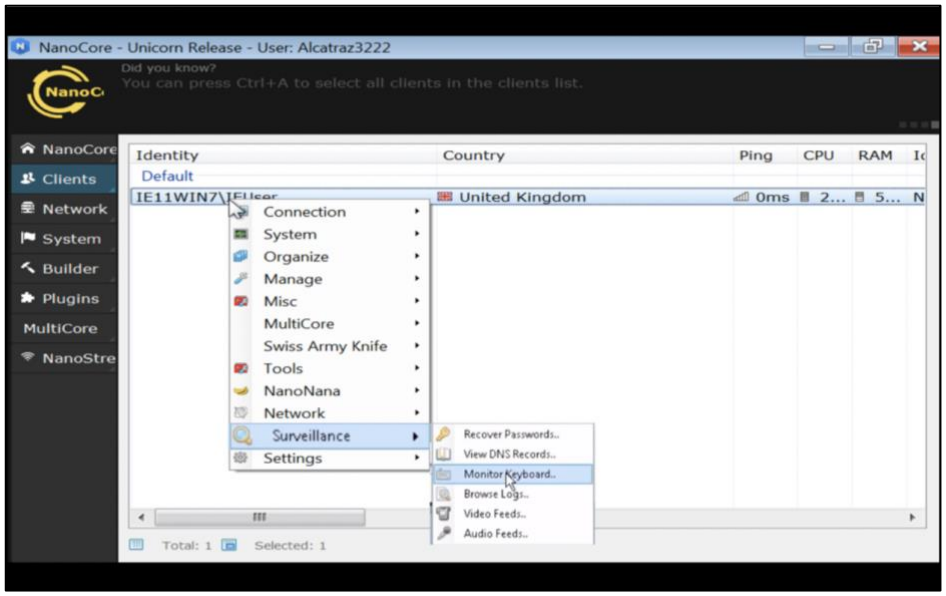


Figure 1. The NanoCore user interface showing surveillance features.

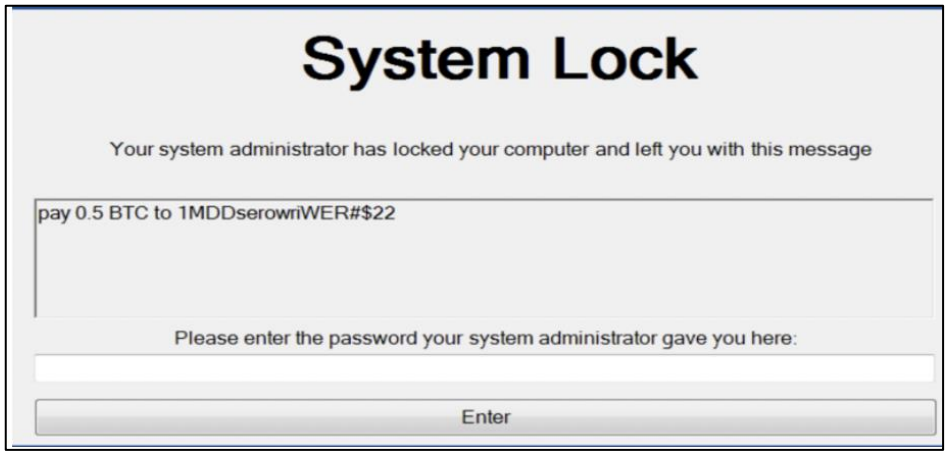


Figure 2. NanoCore ransomware capability.

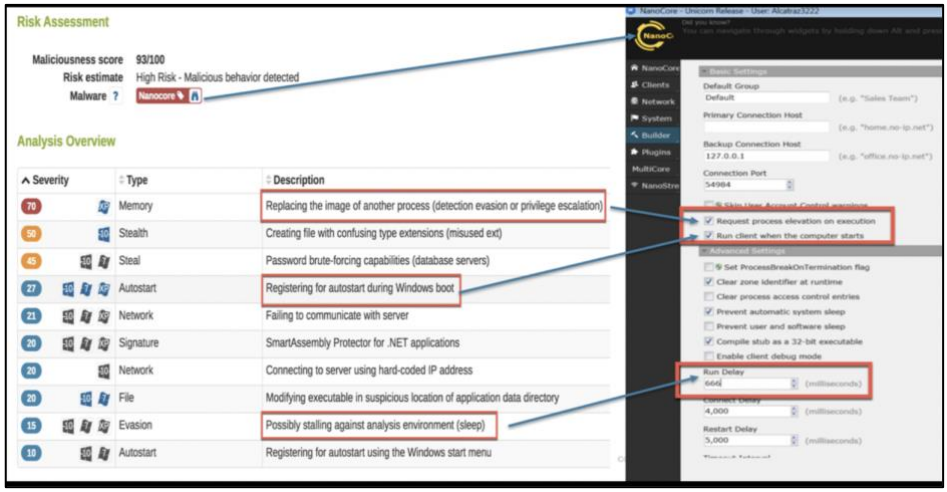


Figure 3. Lastline behavioural analysis and corresponding NanoCore features.

How Does NanoCore RAT Work?

Most malwares are designed for one specific type of attack. However, NanoCore allows hackers to do just about anything they want to once they gain complete, anonymous control over infected devices.

In 2015, targeted emails were sent to energy companies in Asia and the Middle East by spoofing email addresses of a legitimate South Korean oil company. Attached to the email was a malicious RTF file that dropped the NanoCore trojan.
This is the sequence of events that shows how NanoCore was executed, ultimately putting the victims' Office 365 data at risk.

Technique	***Action***
Phishing	*A malicious RTF file email attachment is sent to the victim's Outlook.*
Payload Deployment	*The user clicks the attachment and the trojan is uploaded on the device without any detection.*
Business Email Compromise	*Keyloggers are used to steal Office 365 credentials to gain access to financial information and other business-critical data.*
Ransomware	*Information is moved over to servers owned by hackers. The victims are then asked to a pay fee to get the stolen Office 365 data back.*

Analysis

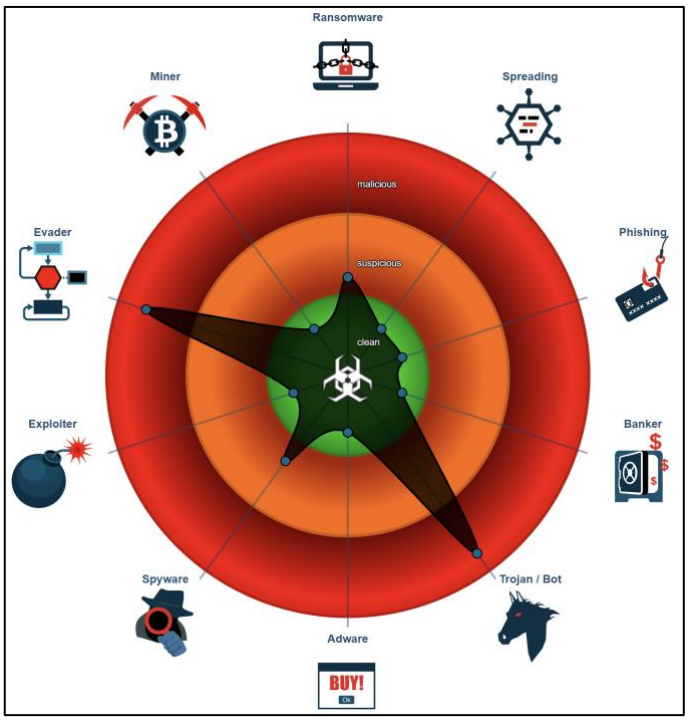


Figure 4. The malware displaying Trojan behaviour.

When executing malware, it's noticed that the malware performs some checks to evade detection.

- *The above figure shows some of the processes checked by malware. This operation is performed by using the classic Win32 API calls "CreateToolhelp32Snapshot" and "Process32Next".*

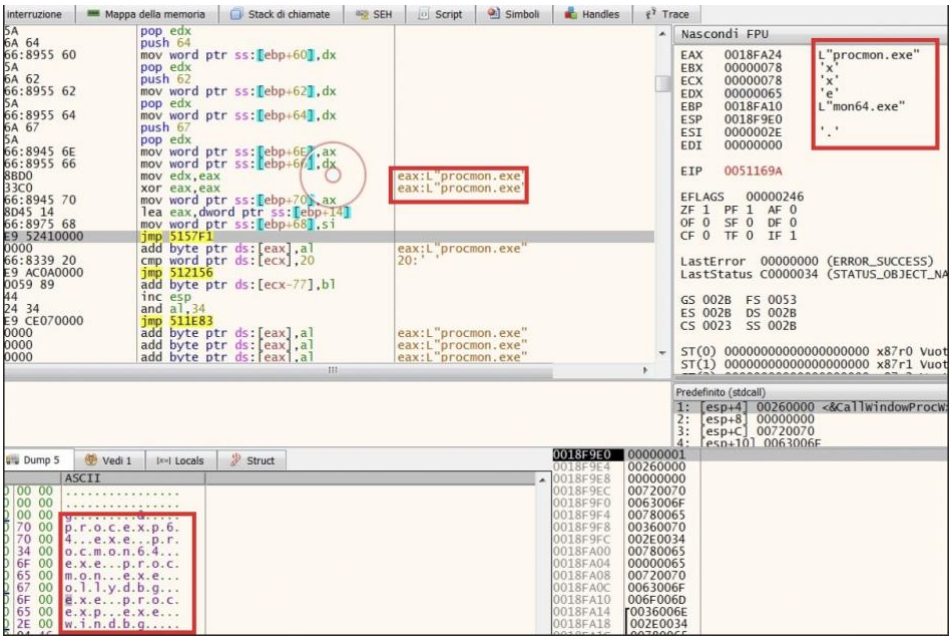


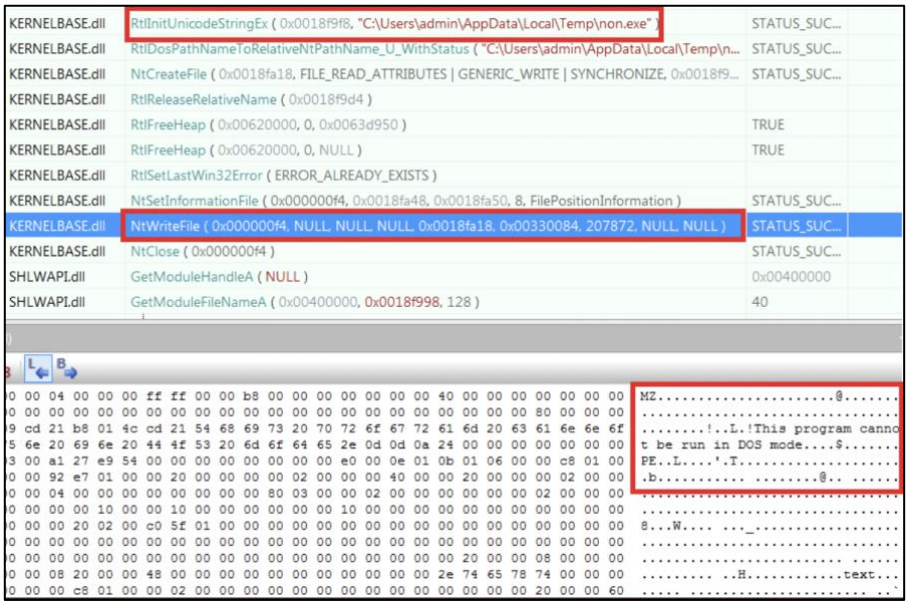
Figure 5. Malware checks if certain processes are active.

- *If none of the checked processes are active, the malware can continue the actual infection: write the actual payload of Nanocore RAT into the "%TEMP%" folder.*

```
405167   RegOpenKeyExA (HKCU\Software\Borland\Locales)
405185   RegOpenKeyExA (HKLM\Software\Borland\Locales)
76ff3391   ExitThread()
4051a3   RegOpenKeyExA (HKCU\Software\Borland\Delphi\Locales)
7725f12d   VirtualAllocEx(h=ffffffff, addr=0, sz=100000,type=2000, prot=1) = 1c0000
7725f12d   VirtualAllocEx(h=ffffffff, addr=1c0000, sz=4000,type=1000, prot=4) = 1c0000
73f24f05   IsDebuggerPresent() = 0
7725f12d   VirtualAllocEx(h=ffffffff, addr=0, sz=1000,type=1000, prot=40) = 2c0000
7725f12d   VirtualAllocEx(h=ffffffff, addr=0, sz=20a72ced,type=3000, prot=4) = 100b0000
463147   VirtualFree(addr=100b0000, sz=0, type=8000) (region_sz=20a73000)
7725f12d   VirtualAllocEx(h=ffffffff, addr=0, sz=1f4b0000,type=2000, prot=1) = 100b0000
7725f12d   VirtualAllocEx(h=ffffffff, addr=100b0000, sz=1f4a4000,type=1000, prot=4) = 100b0000
4017c3   VirtualFree(addr=100b0000, sz=1f4a4000, type=4000) (region_sz=1f4a4000)
401681   VirtualFree(addr=100b0000, sz=0, type=8000) (region_sz=1f4b0000)
7725f12d   VirtualAllocEx(h=ffffffff, addr=1c4000, sz=8000,type=1000, prot=4) = 1c4000
7725f12d   VirtualAllocEx(h=ffffffff, addr=0, sz=100,type=3000, prot=4) = 2f0000
7725f12d   VirtualAllocEx(h=ffffffff, addr=0, sz=1000,type=3000, prot=40) = 380000
7725f12d   VirtualAllocEx(h=ffffffff, addr=0, sz=8000005,type=3000, prot=4) = 2e10000
1c2e3a   VirtualFree(addr=2e10000, sz=0, type=8000) (region_sz=8001000)
7725f12d   VirtualAllocEx(h=ffffffff, addr=0, sz=101d0,type=3000, prot=4) = 390000
1c391f   CreateToolhelp32Snapshot(flags:2, pid:0)
7725f12d   VirtualAllocEx(h=ffffffff, addr=0, sz=dd,type=3000, prot=4) = 3c0000
7725f12d   VirtualAllocEx(h=ffffffff, addr=0, sz=32c84,type=3000, prot=4) = 510000
10005617   CreateToolhelp32Snapshot(flags:2, pid:0)
10005681   Process32Next() HIDING api_logger.exe
76d4ac0e   OpenProcess(pid:854) trasferimento.exe - BLOCKED
10005617   CreateToolhelp32Snapshot(flags:2, pid:0)
10005681   Process32Next() HIDING api_logger.exe
76d4ac0e   OpenProcess(pid:854) trasferimento.exe - BLOCKED
757d9acc   CreateMutexA(Local\ZonesCacheCounterMutex) = 0x2a0
757d9acc   CreateMutexA(Local\ZonesLockedCacheCounterMutex) = 0x2a4
1c313f   VirtualFree(addr=510000, sz=0, type=8000) (region_sz=33000)
7725f12d   VirtualAllocEx(h=ffffffff, addr=0, sz=74bd2,type=3000, prot=4) = 3610000
```

Figure 6. API calls used to check the open processes.

- *The NanoCore payload written by the loader and related API calls will be further loaded into the memory. Interestingly, the payload has not been encrypted or obfuscated.*



Yara rule:

```
rule Nanocore_RAT_Gen_2 {
        meta:
                description = "Detetcs the Nanocore RAT"
                author = "Abhishek Verma"
                reference = "https://www.joesandbox.com/analysis/462129/0/html"
        strings:
                $x1 = "NanoCore.ClientPluginHost" fullword ascii
                $x2 = "IClientNetworkHost" fullword ascii
                $x3 = "#=qjgz7ljmpp0J7FvL9dmi8ctJILdgtcbw8JYUc6GC8MeJ9B11Crfg2Djxcf0p8PZGe" fullword ascii
                $c = "ProjectData"
                $d = "DESCrypto"
                $e = "KeepAlive"
                $g = "LogClientMessage"
        condition:
                 all of them
}
```

- ***NanoCore.ClientPluginHost:*** *String "NanoCore.ClientPluginHost" that belong to NanoCore RAT is found in the memory. [5], [6]*
- ***IClientNetworkHost*** *: Nanocore Client plugin directory. [7]*
- ***ProjectData****: ProjectData.EndApp() is called to terminate the malware exe process. [8]*
- ***DESCryptoServiceProvider*** *[9]*
- ***KeepAlive:*** *A keepalive is a signal sent from one device to another to maintain a connection between the two devices. This may be between a client and a server, but it could apply to any number of devices or technologies. Keepalives are used in network environments to maintain an open communication pathway, or to regularly check the status of a connection to a remote device. [10]*

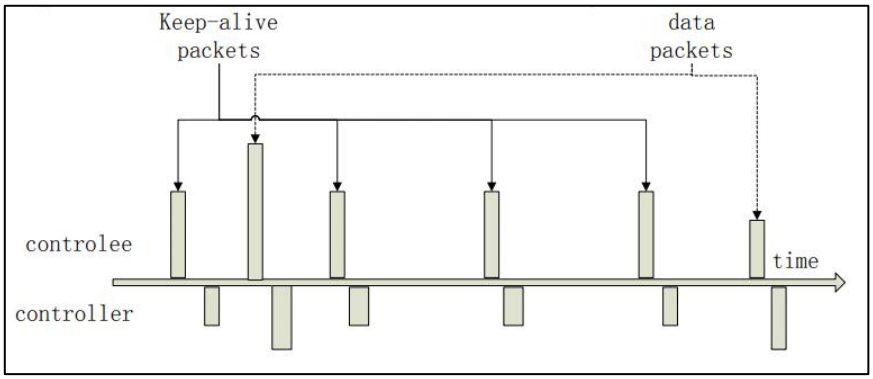


Figure 7. How keepalive works. [11]

References

1. https://yara.readthedocs.io/en/v3.4.0/writingrules.html
2. https://yoroi.company/research/dissecting-nanocore-crimeware-attack-chain/
3. https://www.sentinelone.com/blog/teaching-an-old-rat-new-tricks/
4. https://www.lastline.com/wp-content/uploads/2020/01/ThreatAlert_MalwareSnapshot_K-12.pdf
5. https://securitynews.sonicwall.com/xmlpost/nanocore-rat-delivered-through-phishing-campaigns/
6. https://blogs.cisco.com/security/talos/sysadmin-phish
7. https://github.com/LRNAB/NanoCore-Plugins/blob/master/NanoCore%20Libraries/ClientPlugin.xml
8. https://www.fortinet.com/blog/threat-research/-net-rat-malware-being-spread-by-ms-word-documents
9. https://resolverblog.blogspot.com/2019/01/a-crypto-defeat-story-of-malware.html
10. https://www.microsoft.com/en-us/wdsi/threats/malware-encyclopedia-description?Name=Trojan%3AWin32%2FKeepAlive.A
11. https://www.sciencedirect.com/science/article/pii/S1877050913002391