Sicko Mode

Seems to be some type of meterpreter backdoor; according to virustotal

**Findings: Use cutter.exe to our advantage, take your time to explore the binary in cutter, use the graph feature to help see the logical flow of the binary!!**

**- What language is the binary written in?**

It is written in NIM

**- What is the architecture of this binary?**

64-bit executable size=559499 bytes

**- Under what conditions can you get the binary to delete itself?**

The binary deletes itself upon broken connection, no connection, or once it has exfiltrated all data

**- Does the binary persist? (continuance of effect after removal) If so, how?**

No it does not, but it does create files on the machine that stay after the malware is created but they don’t provide any further functionality

**- What is the first callback domain?**

update.ec12-4-109-278-3-ubuntu20-04.local

(oddly no URLs were seen during basic static)

**- Under what conditions can you get the binary to exfiltrate data?**

Whenever is has something to connect to eg inetsim

**- What is the exfiltration domain?**

**Graphical user interface, application

Description automatically generated**

**- How does exfiltration take place?**

The data from cosmo.jpeg is encrypted, the malware performs a GET request to the above domain and places the encrypted contents as a parameter in the URL, which can later be retrieved in the access logs of the exfiltration domain.

**- What URI is used to exfiltrate data?**

**Graphical user interface, application

Description automatically generated**Although this is a get request, as you can see by the post parameter this data could be some data from the target machine encoded or encrypted and then later retrieved in the access logs of the attackers domain.

**- What type of data is exfiltrated (the file is cosmo.jpeg, but how exactly is the file's data transmitted?)**

It is encrypted using RC4 with the contents of the password.txt file as the encryption key.

**Make sure to use the graph feature in cutter.exe to your advantage, and really analyze main() and the surrounding functions in the cutter.exe graph!!!!**

**Also if we have hints, use the search feature of cutter.exe to find stuff!!**

**- What kind of encryption algorithm is in use?**

RC4

**- What key is used to encrypt the data?**

Its sickomode, as contained in the password.txt file created by the program in C:/users/public

**- What is the significance of `houdini`?**

This is the call used by the malware to delete itself, upon the conditions listed above.

****

**All of these files are referenced seems like the password.txt is created by the binary and the cosmo.jpg file is likely exfiltrated out**

Graphical user interface, text, application

Description automatically generated

Dns Request for update.ec12-4-109-278-3-ubuntu20-04.local

Graphical user interface, text, application, email

Description automatically generated

HTTP request to the same address

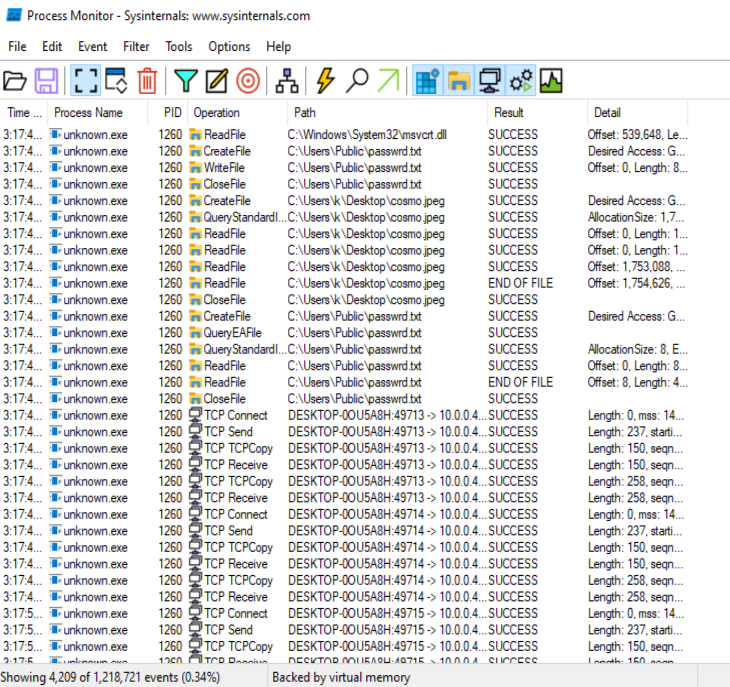
unknown.exe,1460,TCP,Syn Sent,10.0.0.3,50000,10.0.0.4,80,1/19/2023 6:05:40 AM,unknown.exe,,,,

Text

Description automatically generated

We were able to make a connection but it didn’t last long and the malware still deleted itself -> we see strings like to socket, TCP Reconnect, @ HTTP/1.1\r\n, http and other buzz words hinting at external connection

Creates a file at runtime called C:\Windows\Prefetch\UNKNOWN.EXE-10D34576.pf could this be how the program persists upon deletion?



Here we see the reading of the cosmo.jpeg file and the creation of the password.txt file (seems to only be created upon successful connection) (cosmo.jpeg never gets deleted )