**Analyzing a reverse Shell Correlating IOCs**

# Executive Summary

This document outlines a systematic approach to analyzing potentially malicious software using static and dynamic analysis techniques. The process begins with static analysis, where tools like CMDR and Floss are utilized to inspect the sample for any discernible information. If no relevant data is found, the analysis proceeds to dynamic analysis.

Dynamic analysis involves running the sample in a controlled environment, such as REMnux with INETSIM and Wireshark running to capture network traffic. By observing DNS queries and analyzing traffic, suspicious behavior can be identified. In the case of the examined malware, it was found attempting to reach a DNS record on port 443, indicating potential malicious intent.

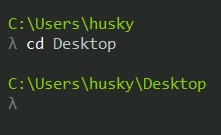
Further investigation includes using tools like Procmon to monitor process activity. By executing the malware and observing its behavior, it was discovered that the malware had reverse shell capabilities, allowing it to establish a connection with a remote listener.

Overall, this analysis demonstrates a comprehensive methodology for identifying and understanding the behavior of potentially harmful software, enabling effective threat mitigation and response strategies.

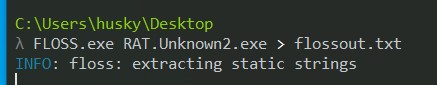
# Static Analysis

Open up CMDR

Cd to Desktop



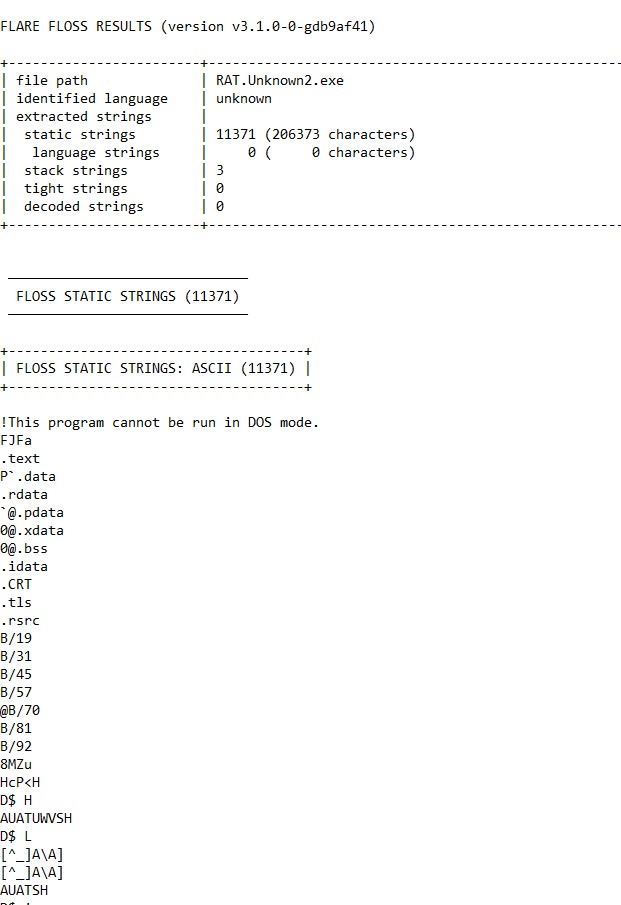
Run Floss against our sample and convert that into a text file



Open up flossout.txt

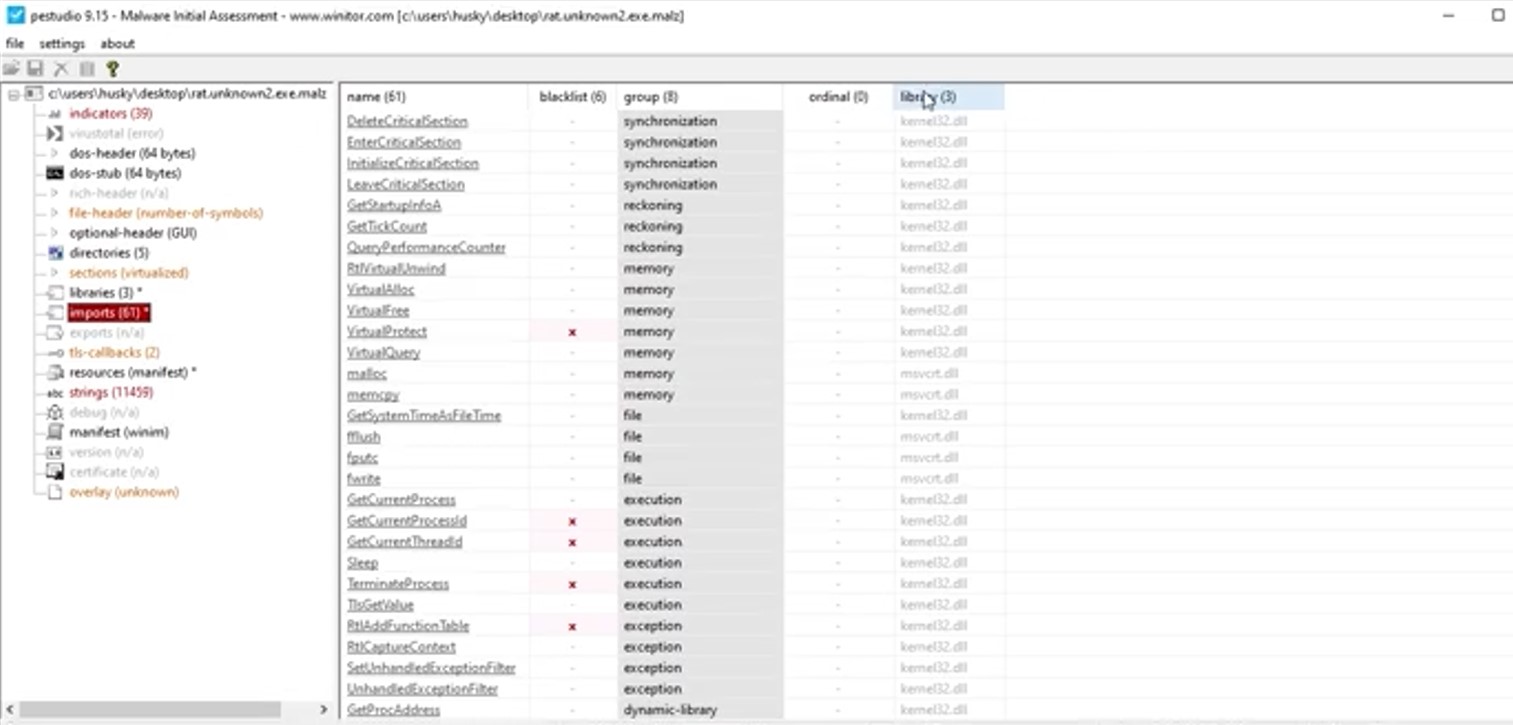


Upon inspection, we did not find any relevant information we could use within the text file.

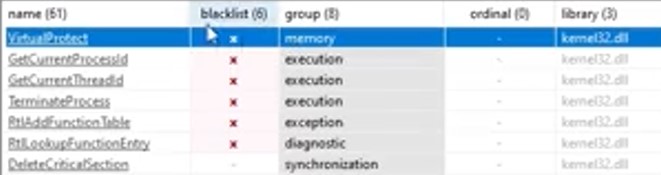
That's ok, but we really want to make sure we are diligent and always check first.

Let's go ahead and close out the flossout.txt file, as well as CMDER.

Navigate to PEStudio



Navigate to the blacklist and see if anything stands out.

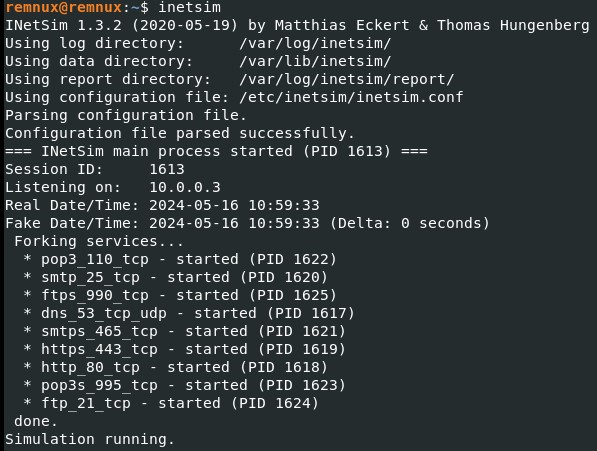


You may not know exactly what an API call is or what it is doing, but you can always go to the Microsoft website, and read up if you aren't familiar with them. Familiarize yourself with them so you can determine which ones are worth your efforts.

Not much is going on here either. Moving on to our dynamic analysis.

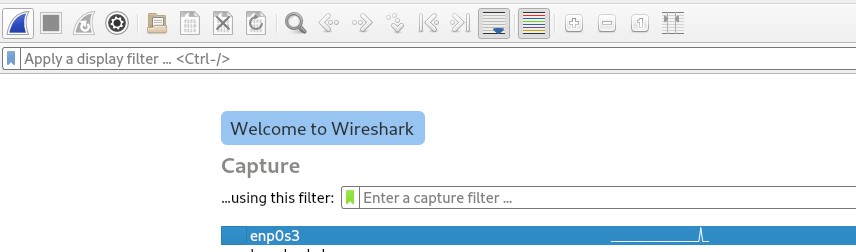
# Dynamic Analysis

Make sure we have REMnux up with INETSIM running.



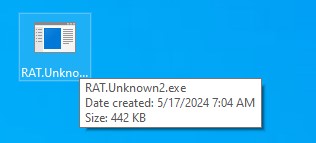
Open another terminal and run Wireshark as well to capture all of the network traffic.

Click on the capture in the top left and let's observe.



We already suspect that there is socket capability present, so we need to be extremely alert.

Let's arm and run the sample as admin.

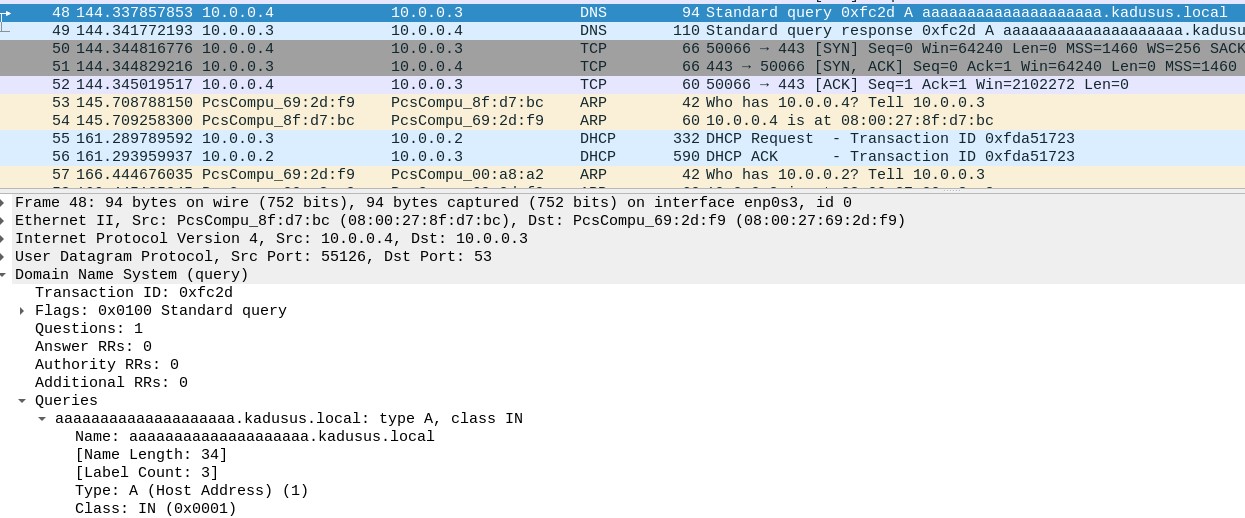


Navigate to the REMnux terminal running Wireshark to see what happened.

We can see some DNS queries coming up.



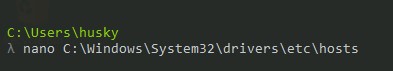
Let's take a look.



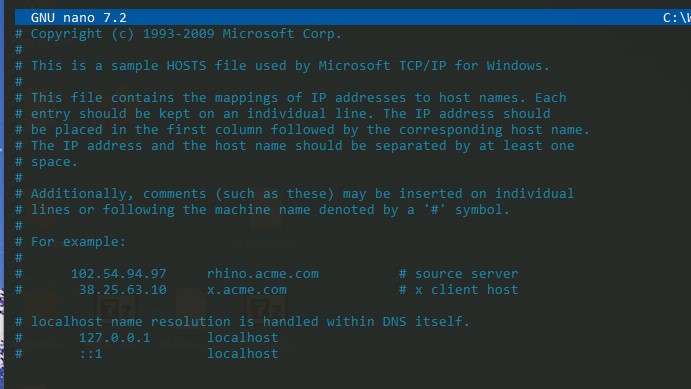
We found a Record DNS: aaaa....kadusus.local



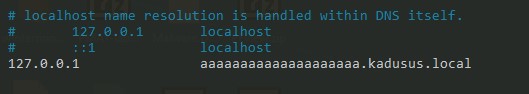
Let's move over to our FlareVM machine. Navigate to CMDER and run as admin. Type the following command.



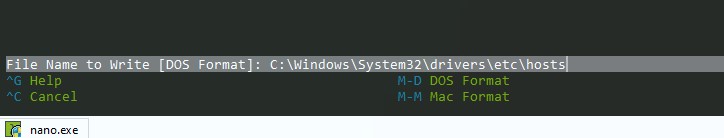
Once the command has been run, this will populate.



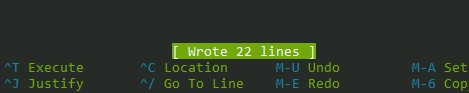
Move your down arrow key to the bottom of the text, and input this.



Save this in nano by holding down control + o/. Keep the file name standard.



Once you hit Enter, it will say it wrote 22 lines.

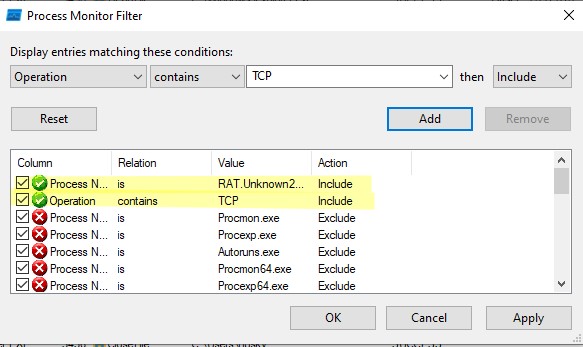


Exit out of nano by holding control + x.

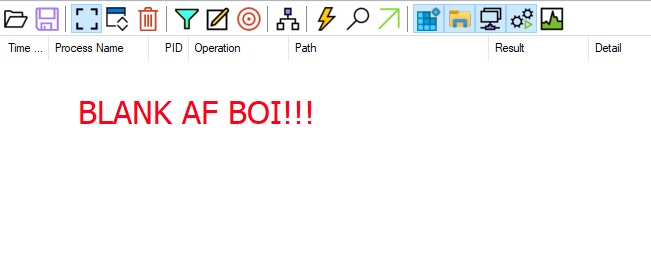
At this particular point, we can surmise this malware is attempting to reach a DNS record, but our Wireshark information is not sufficient. It states that that it is pointed a DNS record, but does not give us a port or protocol. We don't know how this is trying to connect to the DNS record.

\*at this point I start to use Flameshot for screenshots

Let's try to see what information we can get from host. Navigate to Procmon and fire it up. Let's ensure we have the below filters added and applied.



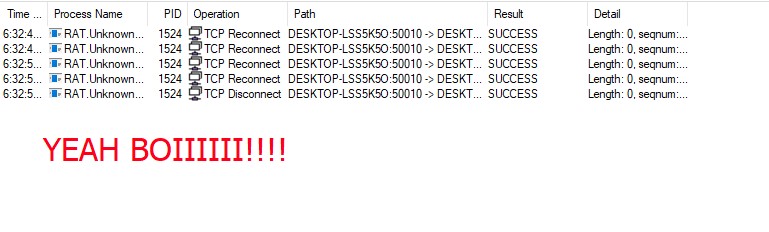
Absolutely nothing populates just yet.



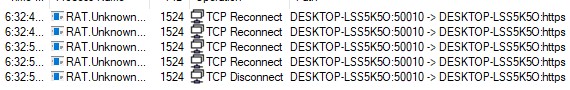
Time to execute the malware.



Oh snap. We have some information.



Let's expand the path section so we can see what is going on.



HTTPS.... Oh snap, that's port 443. Now we know that this malware is calling out to a specified DNS record and what port it is using.

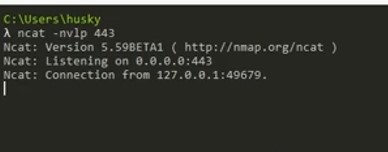
Time to open up CMDER again. Make sure we run as admin. Run this command.



Run the malware again.



Look what popped up in CMDER.

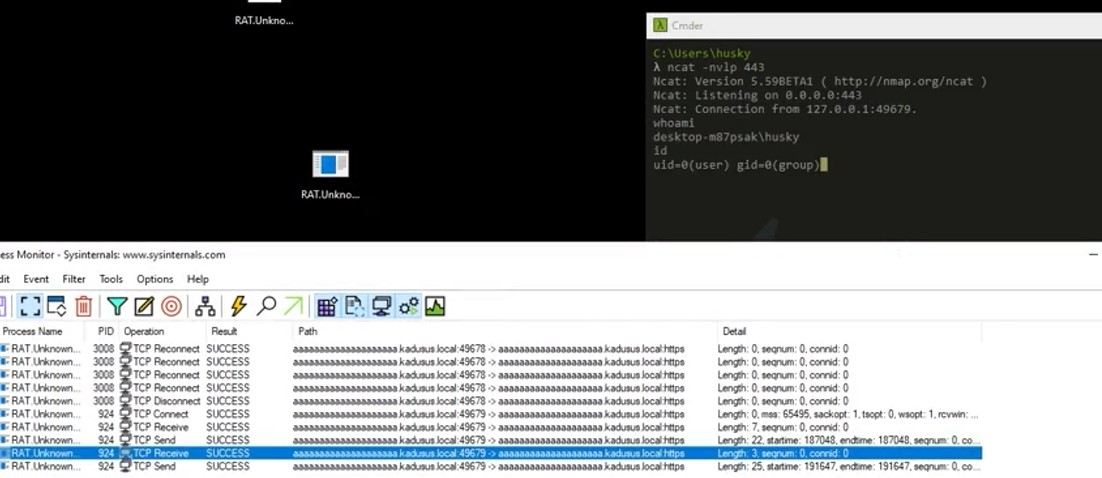


Now we have an open socket that we tricked the malware into thinking was us.

If we had no idea this was a RAT to begin with, we could ask CMDER "whoami".



Essentially what we have done is setup a listener, when the malware was executed, it reached out to the listener and connected. This opened a command shell. We can input commands within the shell, and see them populate in Procmon. Turns out this malware has reverse shell capabilities.



In the next document, we will cover a parent-child analysis. Stay tuned.