**DIGITAL FORENSICS LAB**

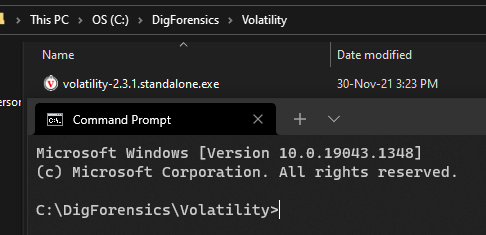
|  |  |
| --- | --- |
| Exercise 14 | |
| Name | S Shyam Sundaram |
| Registration Number | 19BCE1560 |
| Slot | L39+L40 |
| Faculty | Dr. Seshu Babu Pulagara |
| Date | 30th November, 2021 |

**AIM**

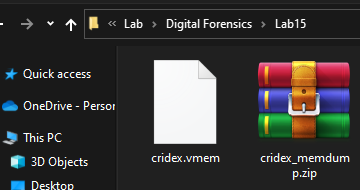
Employing memory forensic tool to analyse a memory dump.

**PROCEDRE**

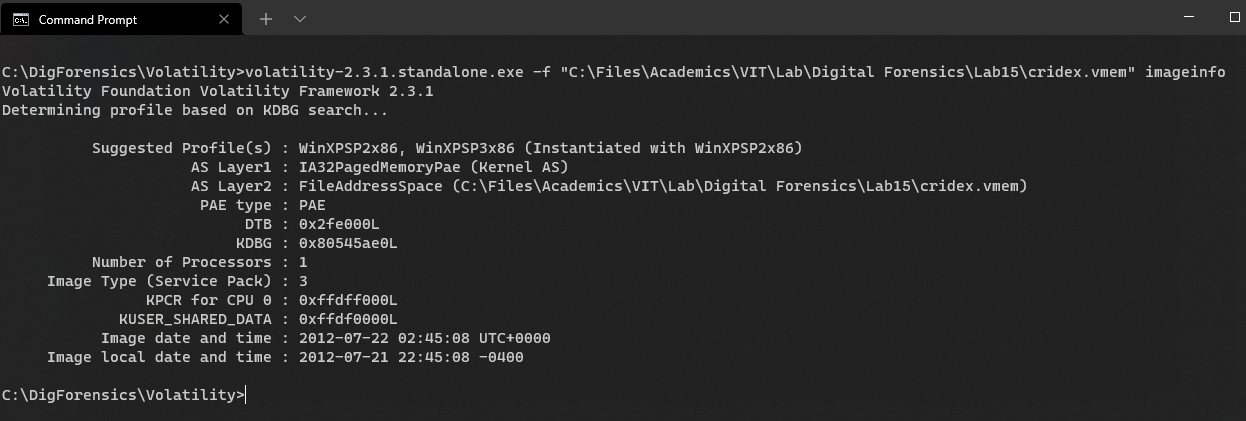
1. Go to the folder where Volatility is downloaded and open the terminal there.



1. Download a memory image file or create your own using ‘Magnet RAM Capture’. Here I have downloaded a memory file with the Cridex malware.

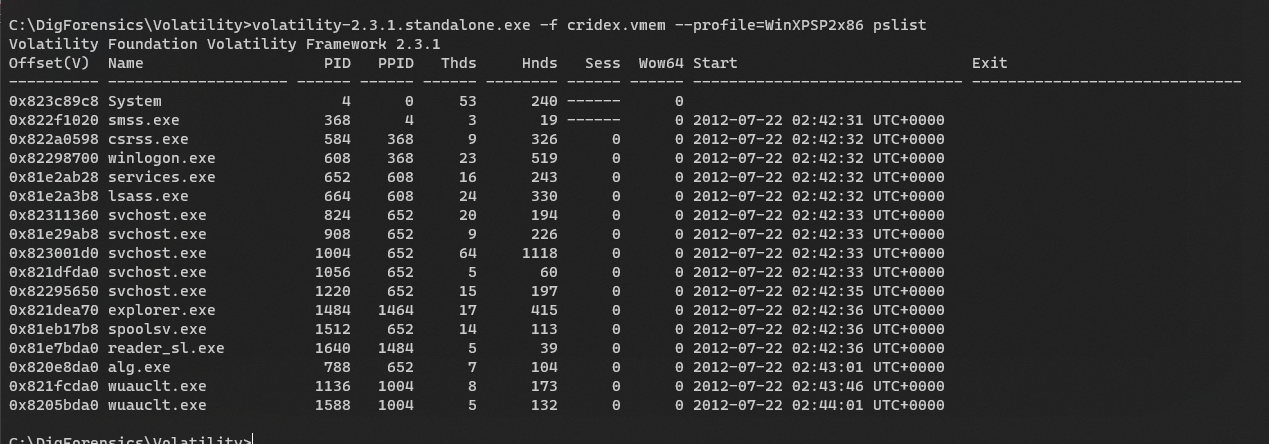


1. Now run the following command to get more info about the memory file.

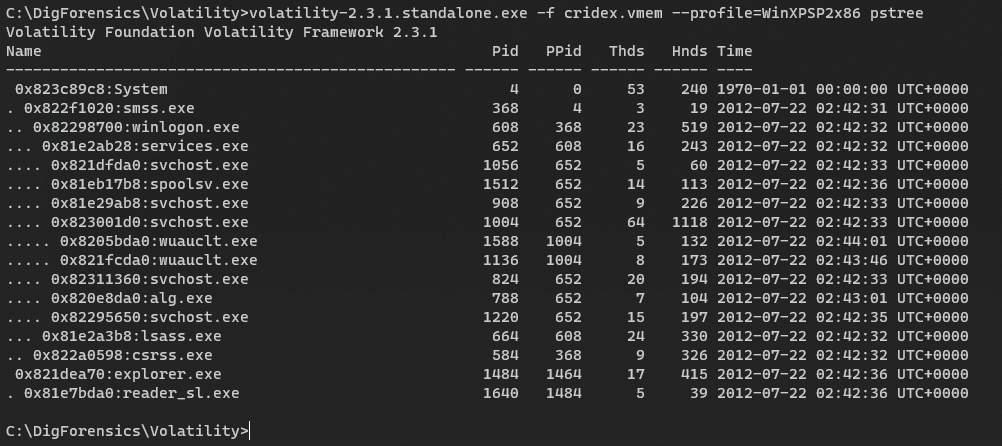


This gives us a list of suggested profiles to use for the cridex.vmem image.

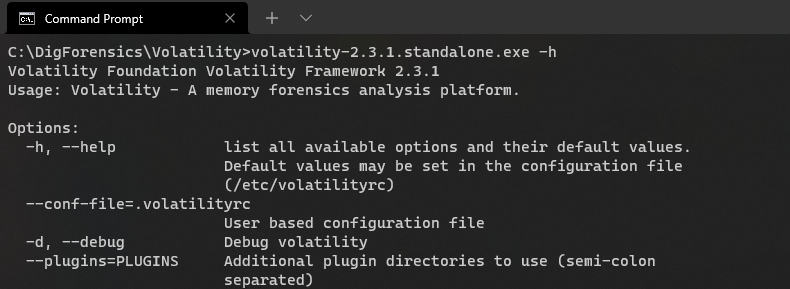
1. Now we see what were the processes running in the memory with the following command.



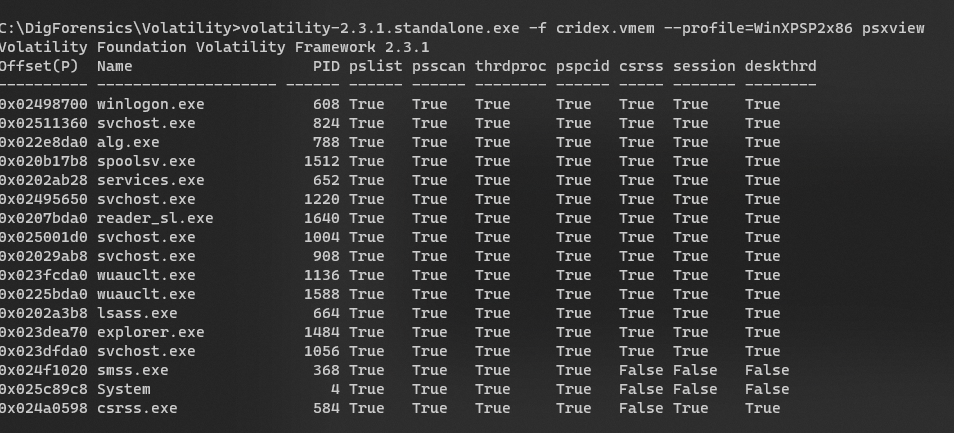
1. To see it in another format, we replace pslist with pstree.



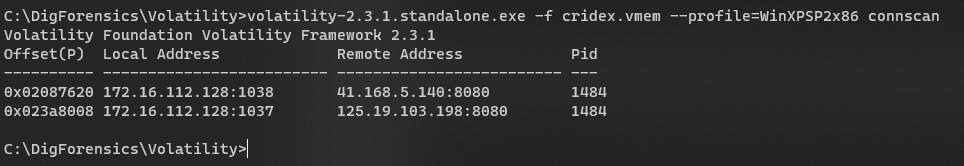
1. To see more about the tool’s options and get some help, we execute this:



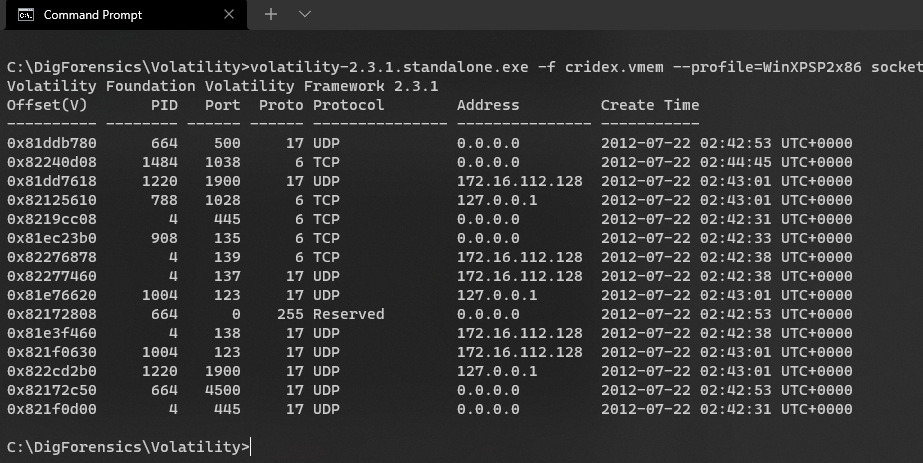
1. To uncover any hidden process, use psxview as below.



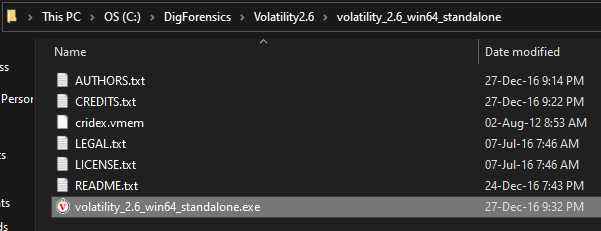
1. To check the running open TCP conections, we can use connscan.



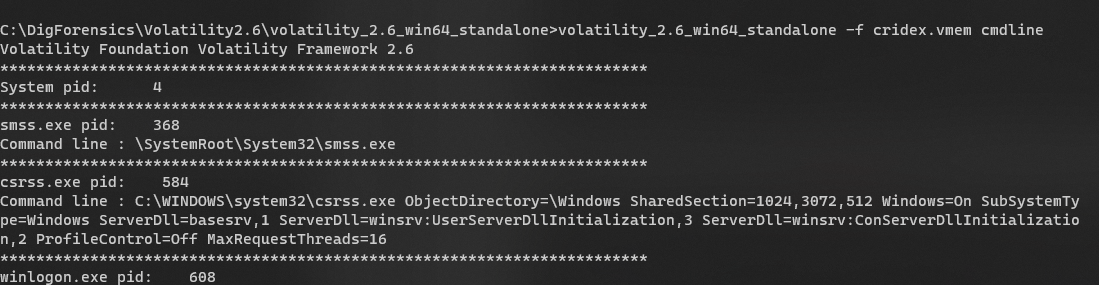
1. To view both TCP and UDP connections, use ‘sockets’



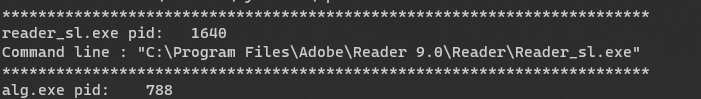
1. **FOR THE FOLLOWING, USE VOLATILITY 2.6. This was saved in another folder called “Volatility2.6” in “DigForensics” folder.**



To display the commandline arguments of each process, use cmdline:



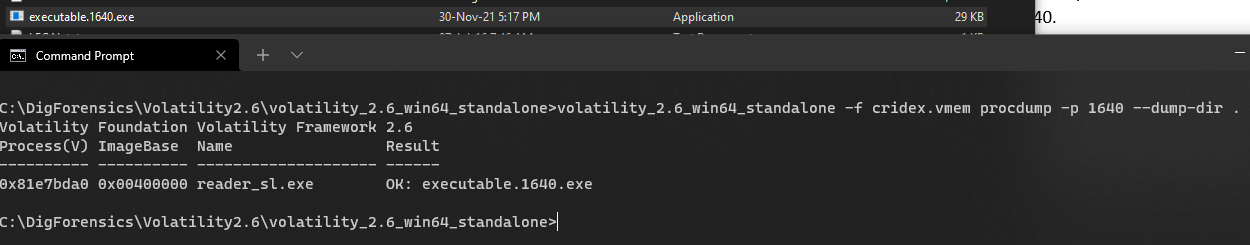
Scolling down we see the command line arguments and the path where reader\_sl was stored. It is found using its process id 1640.



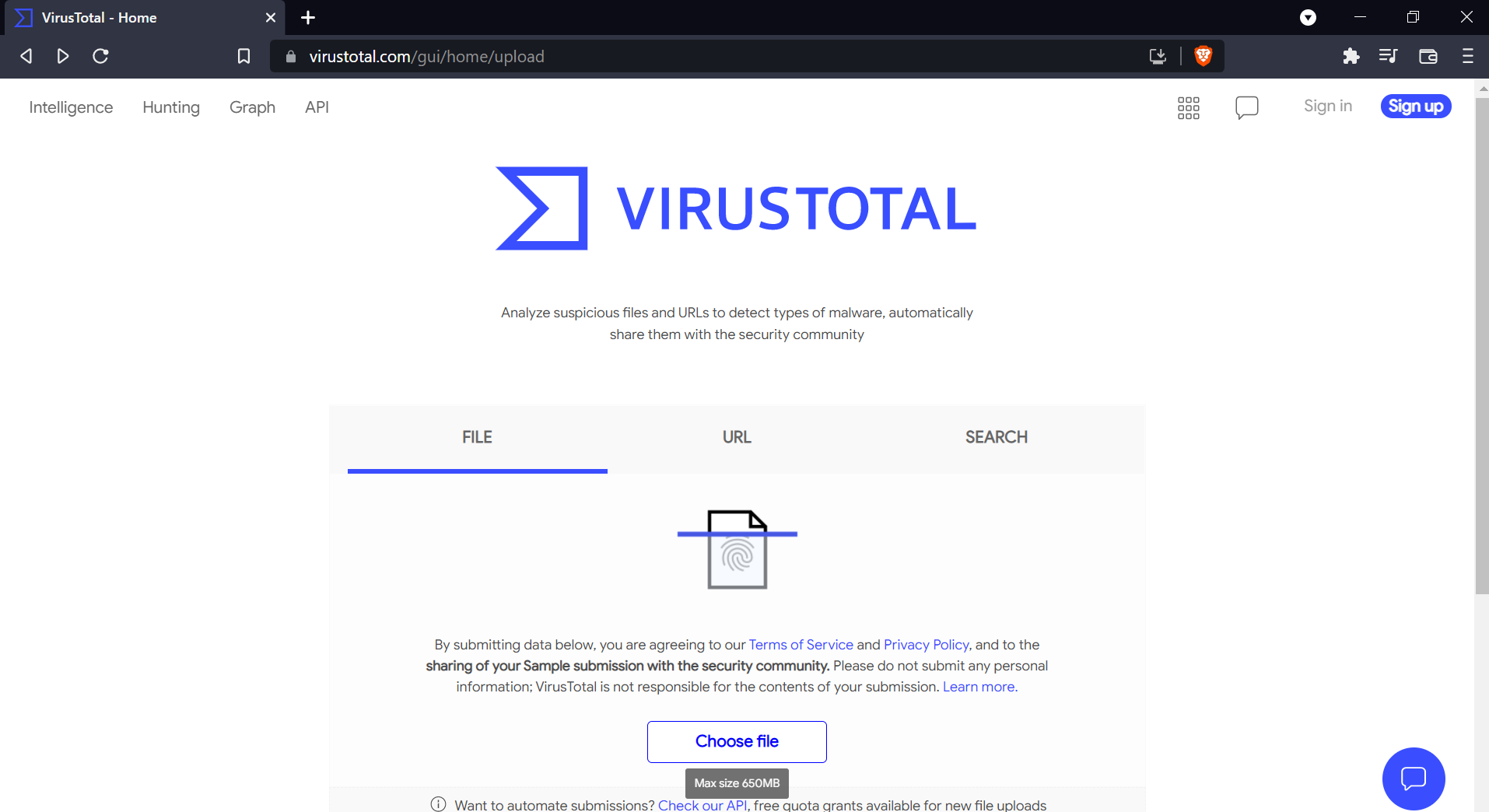
1. We now create a dump of this process and check it out. The command for this is given below using procdump and specifying the PID of the process.

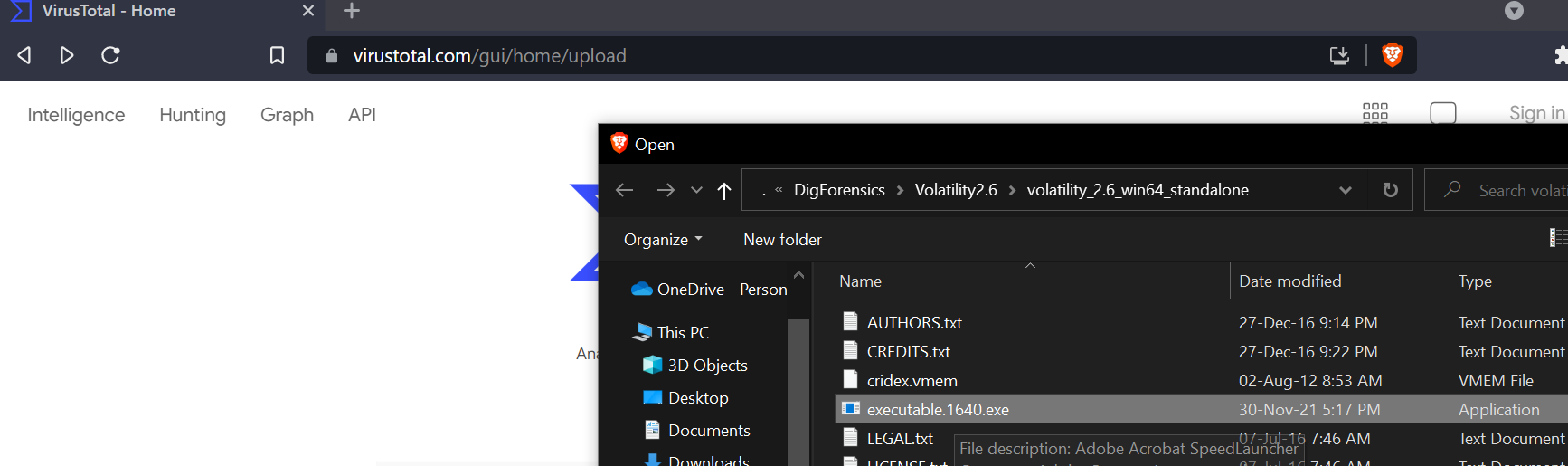
**Command run:** volatility\_2.6\_win64\_standalone -f cridex.vmem procdump -p 1640 --dump-dir .

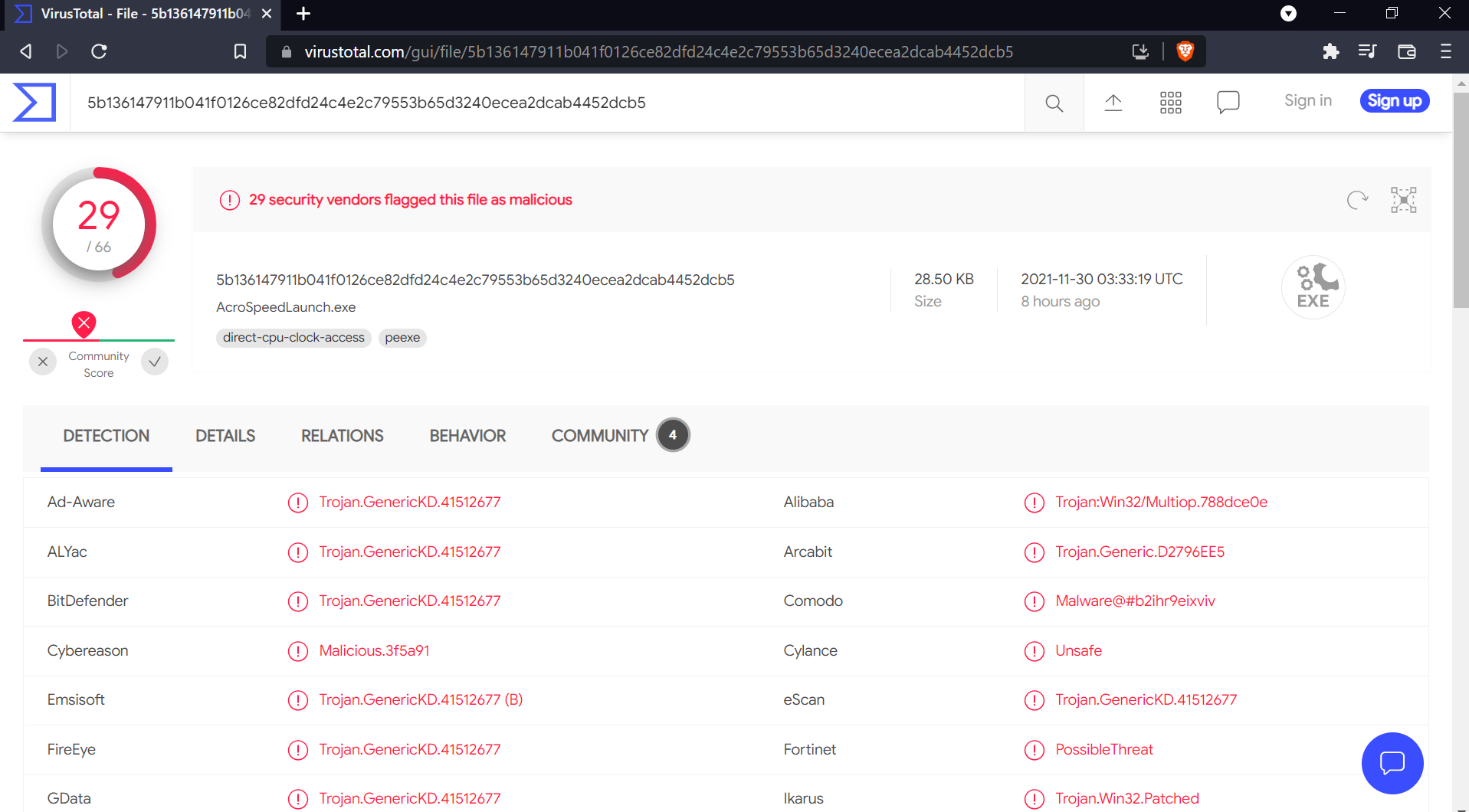
**NOTE: DO NOT DOUBLE CLICK OR RUN THIS NEWLY CREATED EXE/DUMP FILE!**



1. Now, go to virustotal.com and upload this newly created exe file.







From the image above, we see that VirusTotal recognized this file as a Trojan malware. Thus, reader\_sl.exe is a malware.

**OBSERVATIONS**

In the fifth image, we see that there is a process named “reader\_sl.exe” with “explorer.exe” as its parent process. Upon checking the connections and sockets, we see in image 8 that its parent process 1484 makes a connection to some location with address 41.168.140:8080. This is a bit suspicious as the name says it is a process of Adobe Reader but there is no reason why Adobe reader would have to make a connection to some remote location.

Thus, this process’s dump was made and uploaded to VirusTotal which recognized this file as a malware.

**CONCLUSION**

We have used a memory image to figure out which process was the malware.