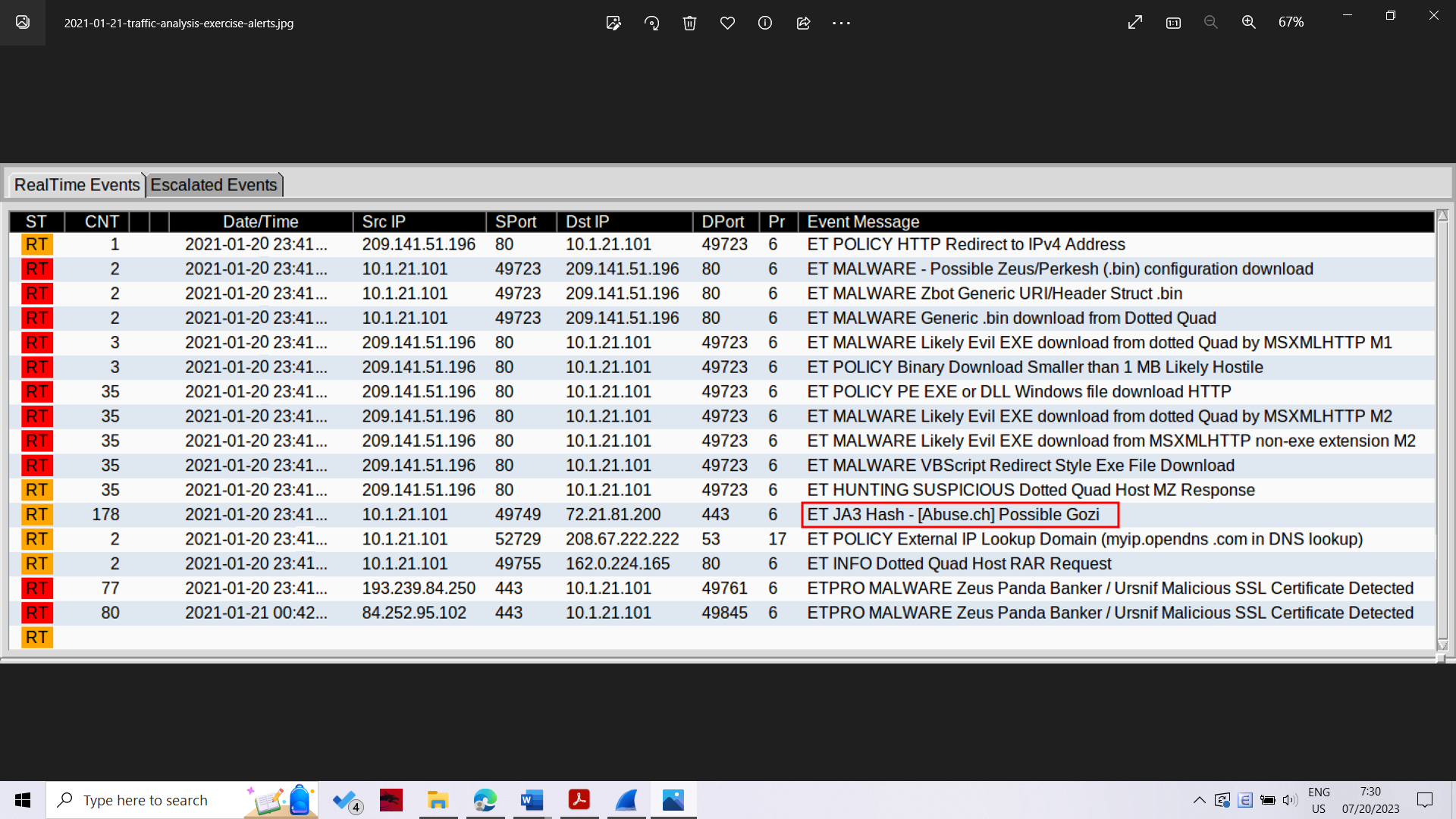
**TRAFFIC ANALYSIS EXERCISE - WOKEMOUNTAIN**

# SCENARIO:

* + LAN segment range: 10.1.21.0/24 (10.1.21.0 through 10.1.21.255)
  + Domain: wokemountain.com
  + Domain Controller: WokeMountain-DC
  + LAN segment gateway: 10.1.21.1
  + LAN segment broadcast address: 10.1.21.255

# TASK:

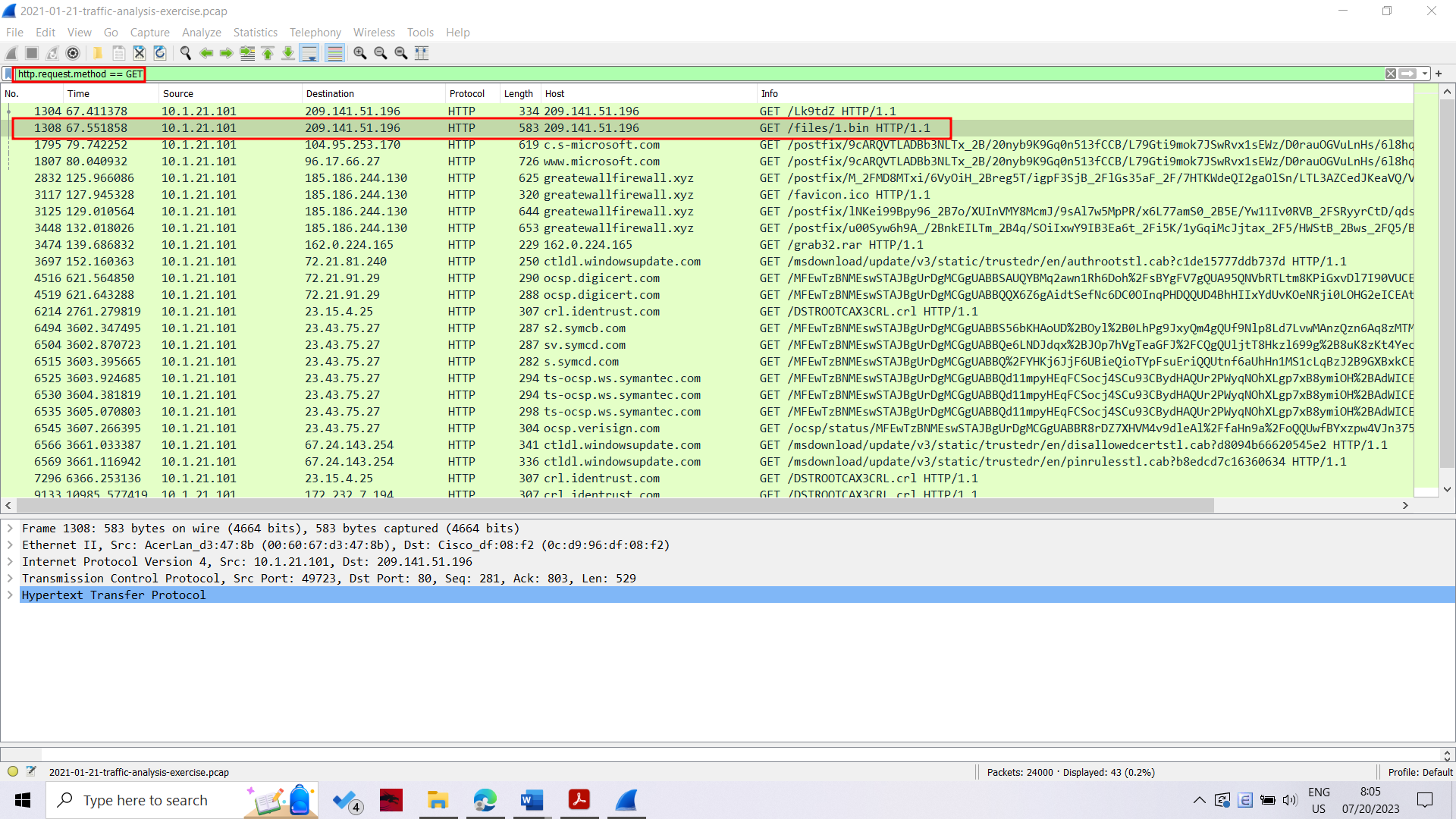
* + Write an incident report based on the pcap and alerts.
  + The incident report should contain the following:
    - Executive Summary
    - Details (of the infected Windows host)
    - Indicators of Compromise (IOCs)

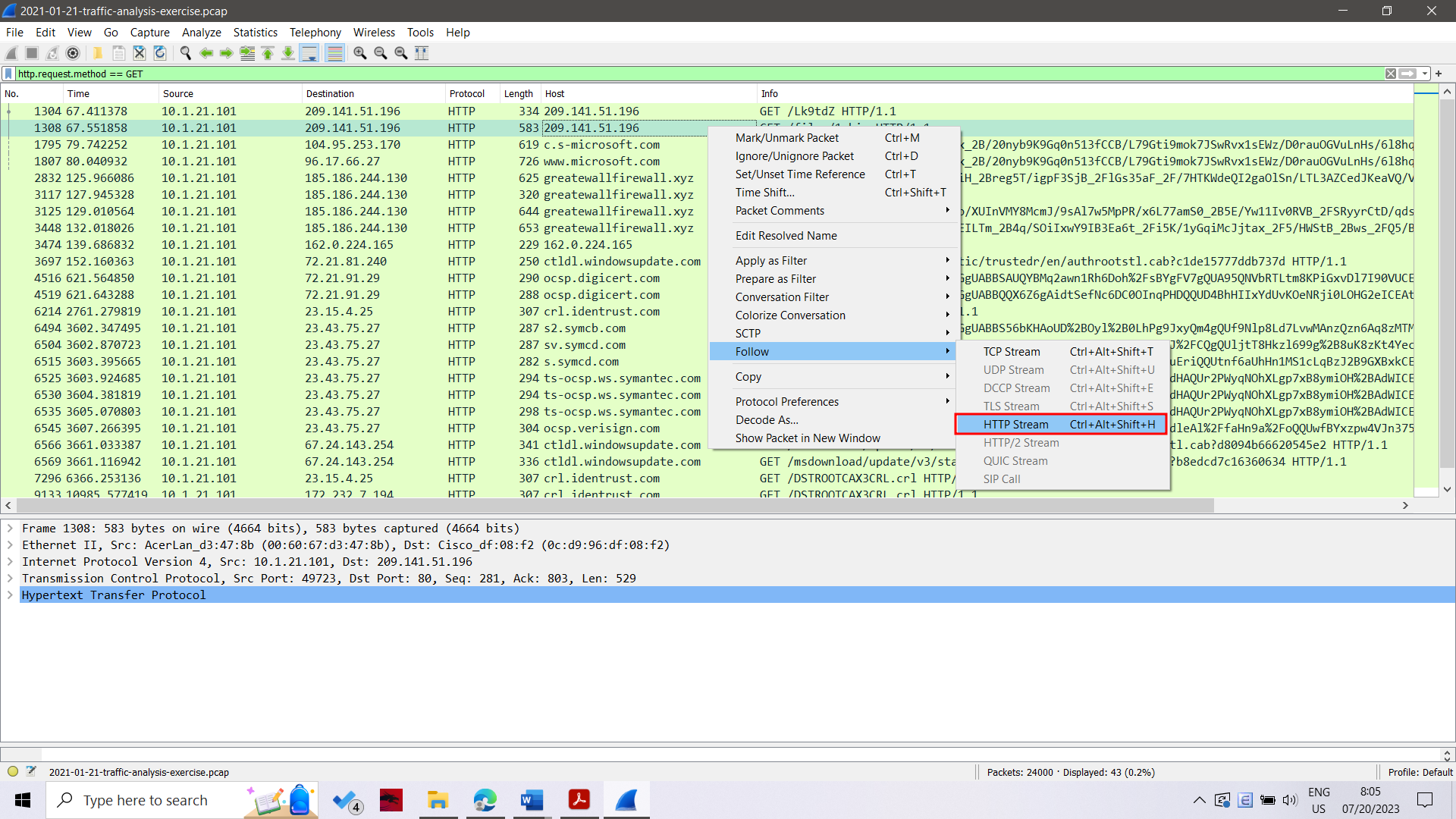


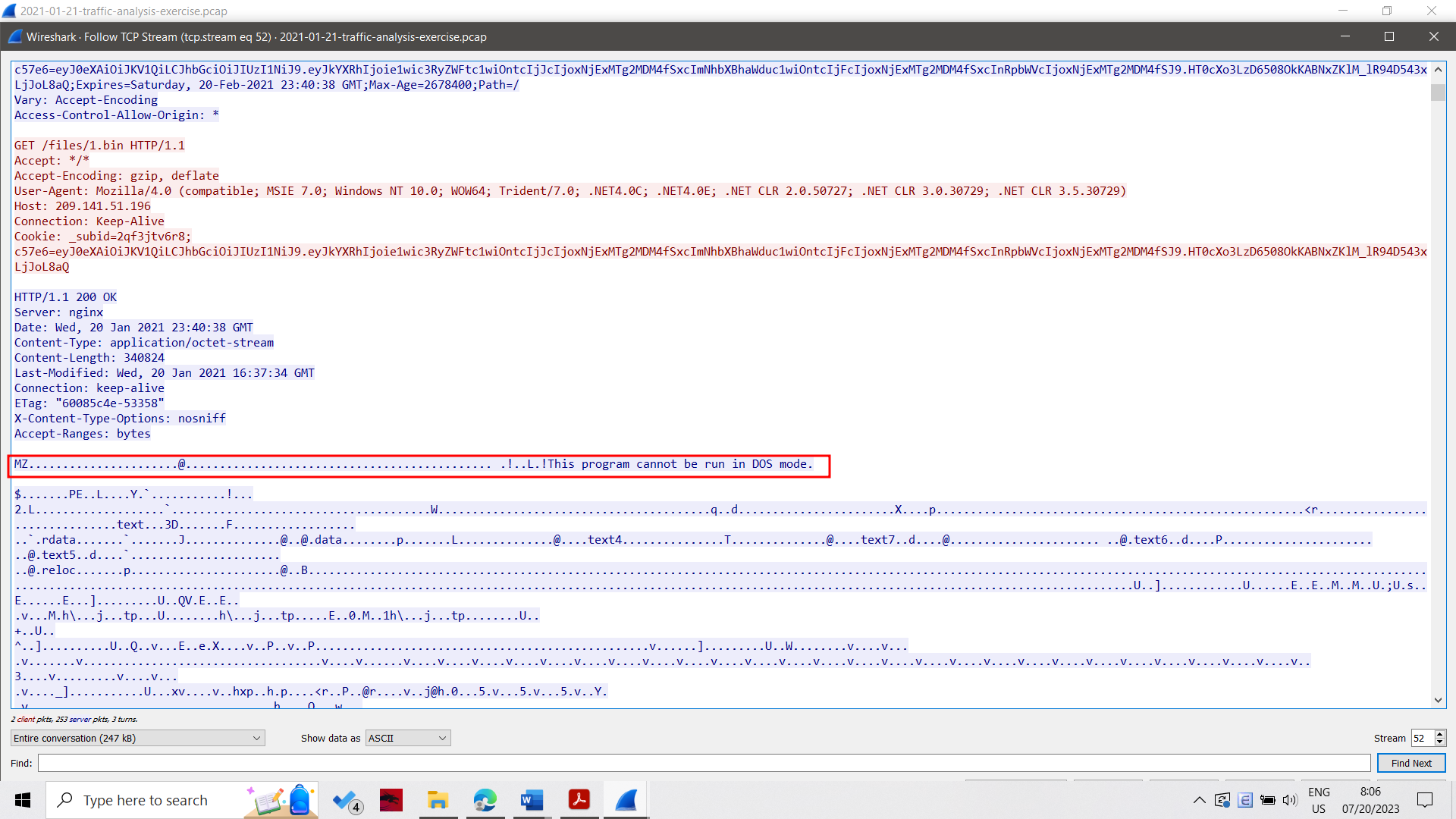
False positive alert

There was a malware DLL returned from http://209.141.51.196/files/1.bin. Unfortunately, you cannot export the above file from the pcap using the File --> Export Objects --> HTTP

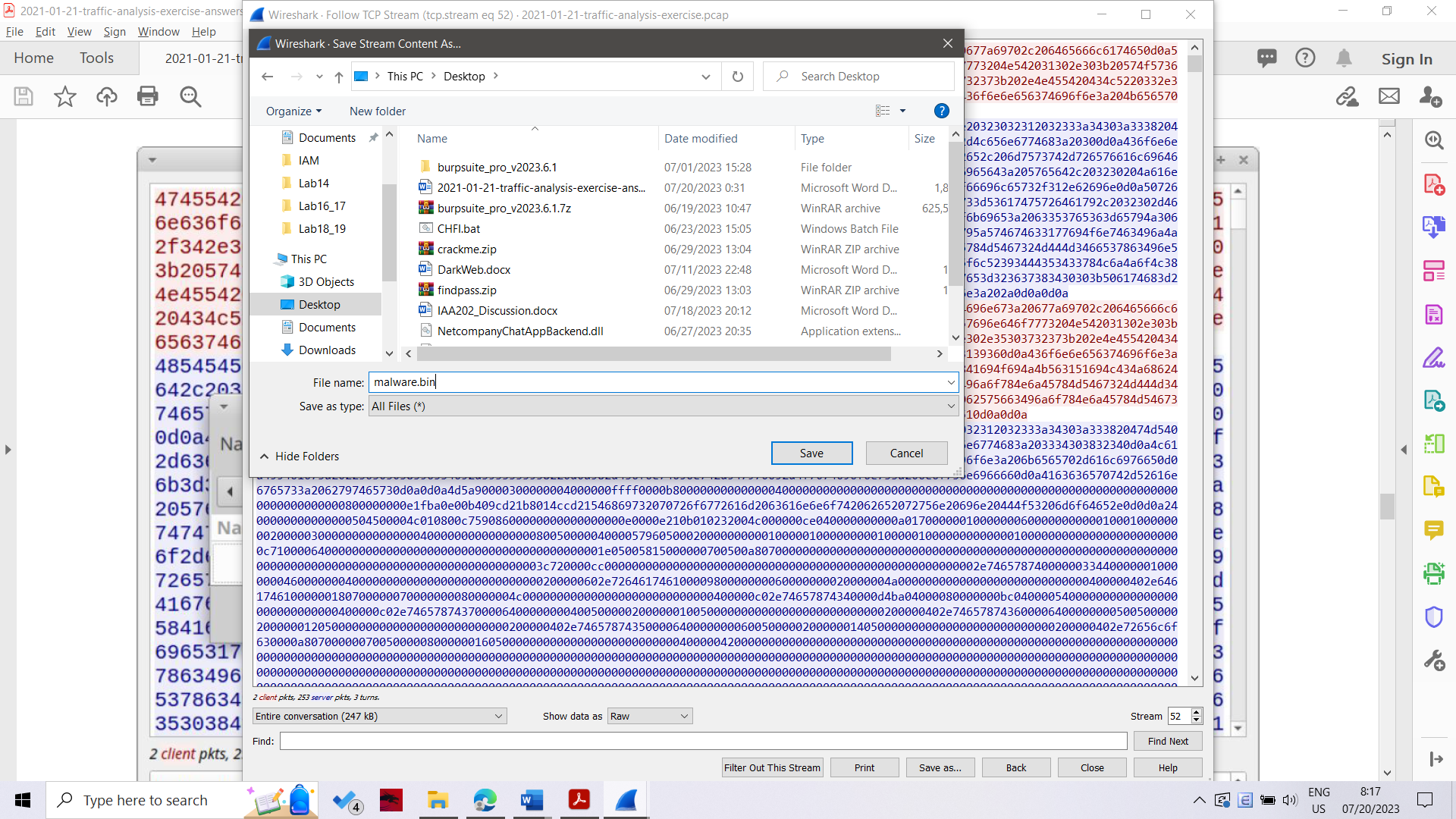
However, you can extract that DLL from the pcap using a work-around. First, you follow the TCP stream for the HTTP GET request to 209.141.51.196 for /files/1.bin. Scroll down a little bit to confirm an EXE or DLL file was returned (there's two HTTP requests in this TCP stream).



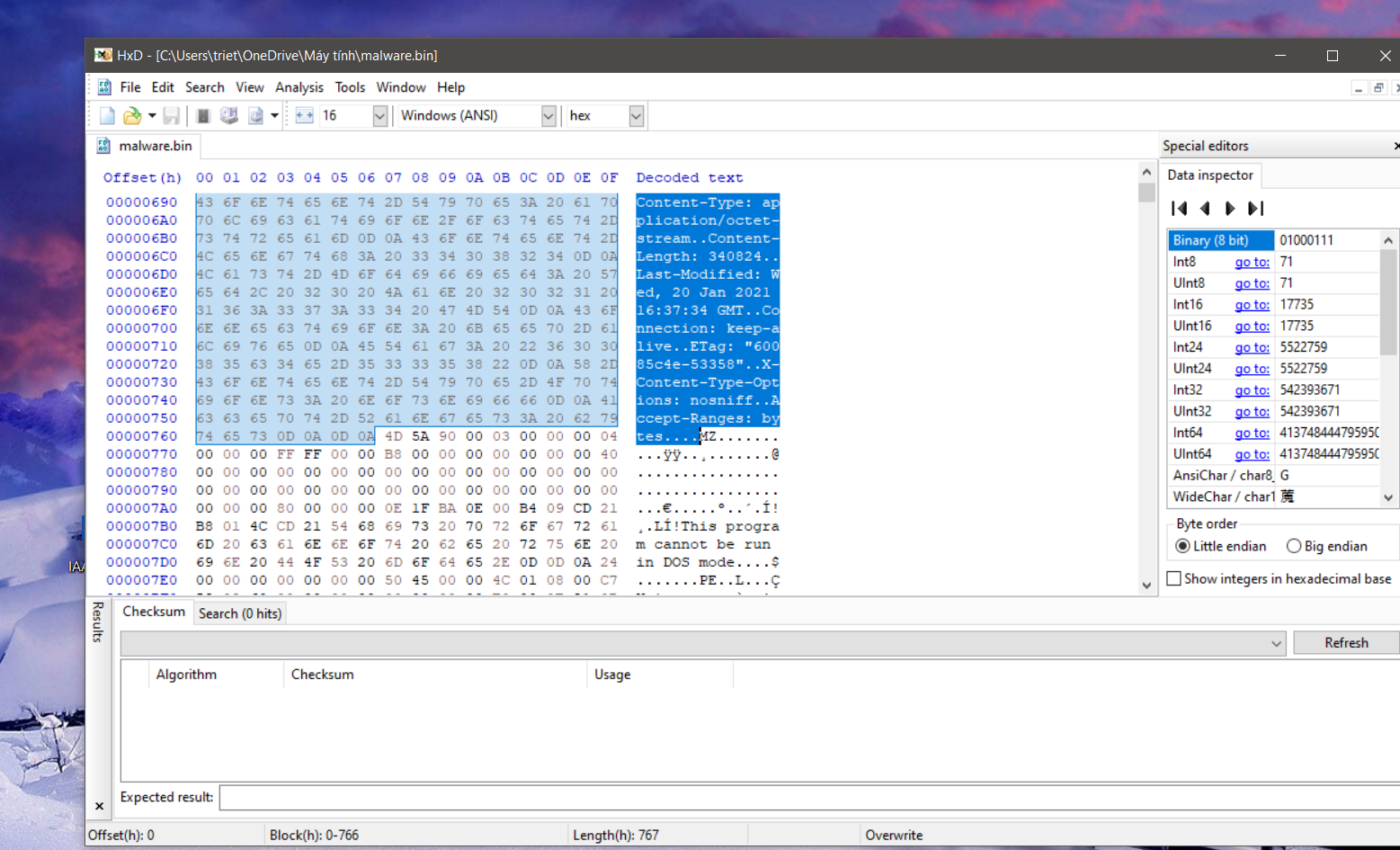




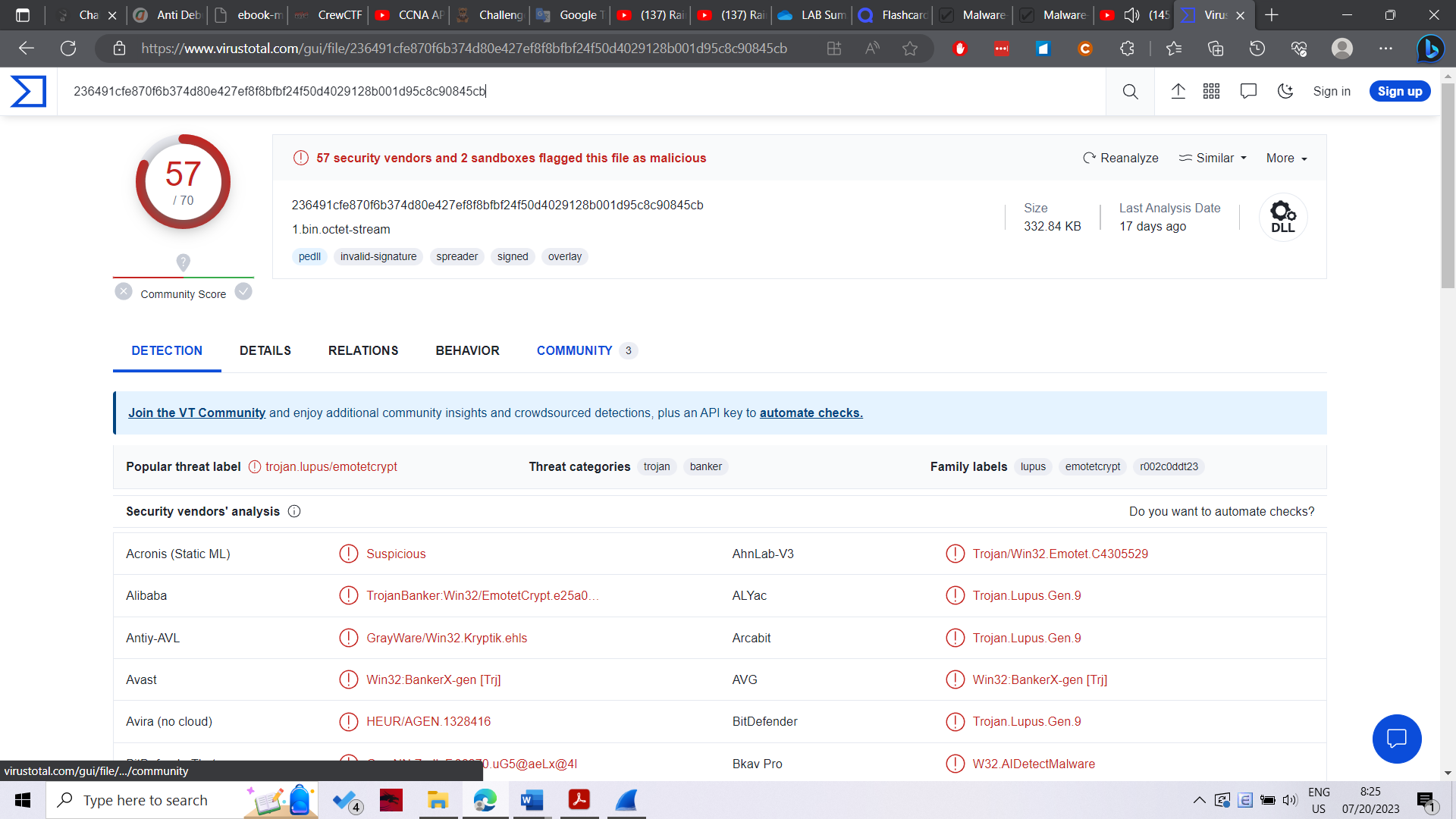




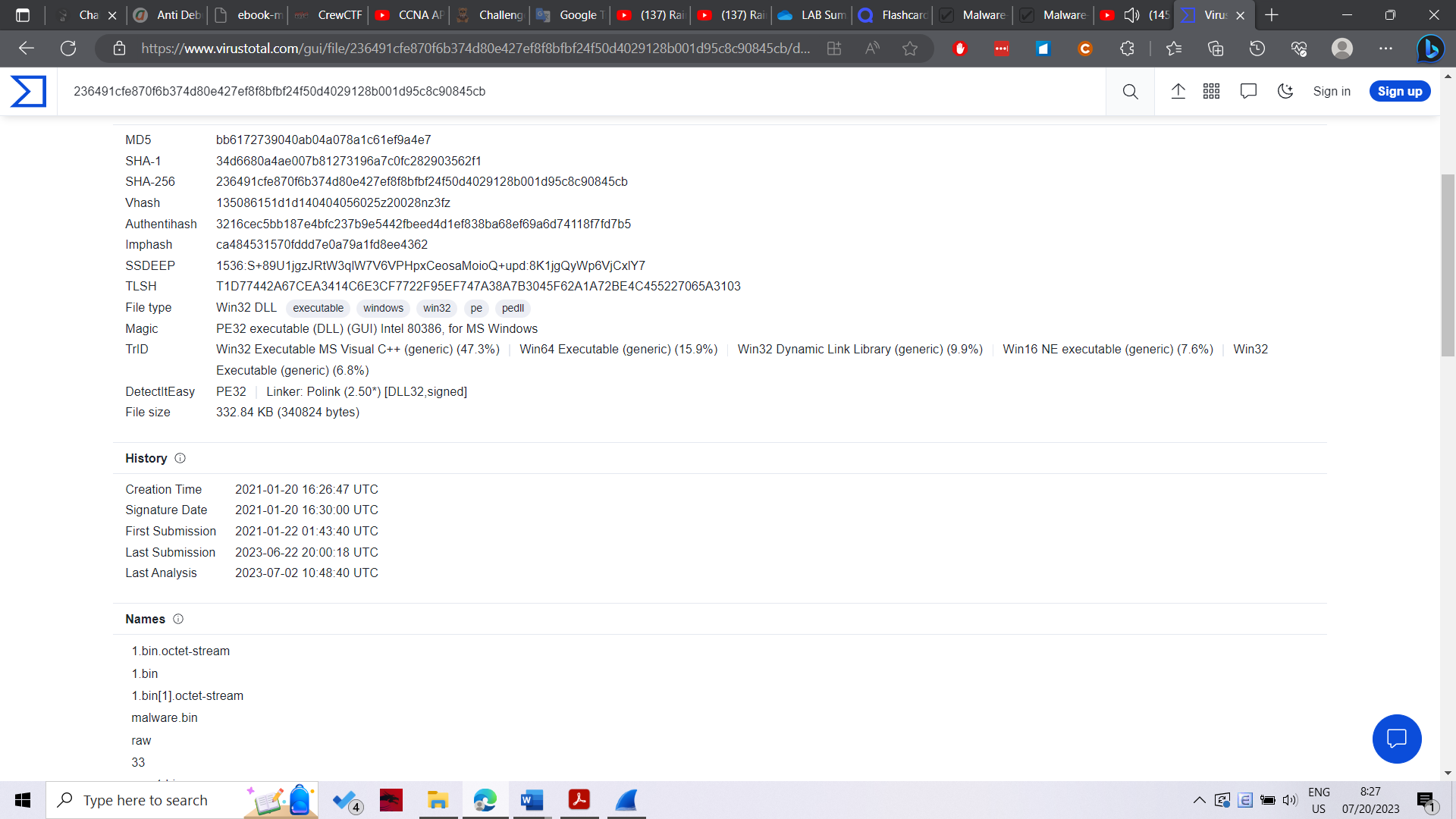
Once you have the binary opened in a hex editor, just select everything before the first two bytes of the malware EXE or DLL file, represented by the characters MZ after the HTTP response header. Delete that section of your exported binary, then save the updated file.

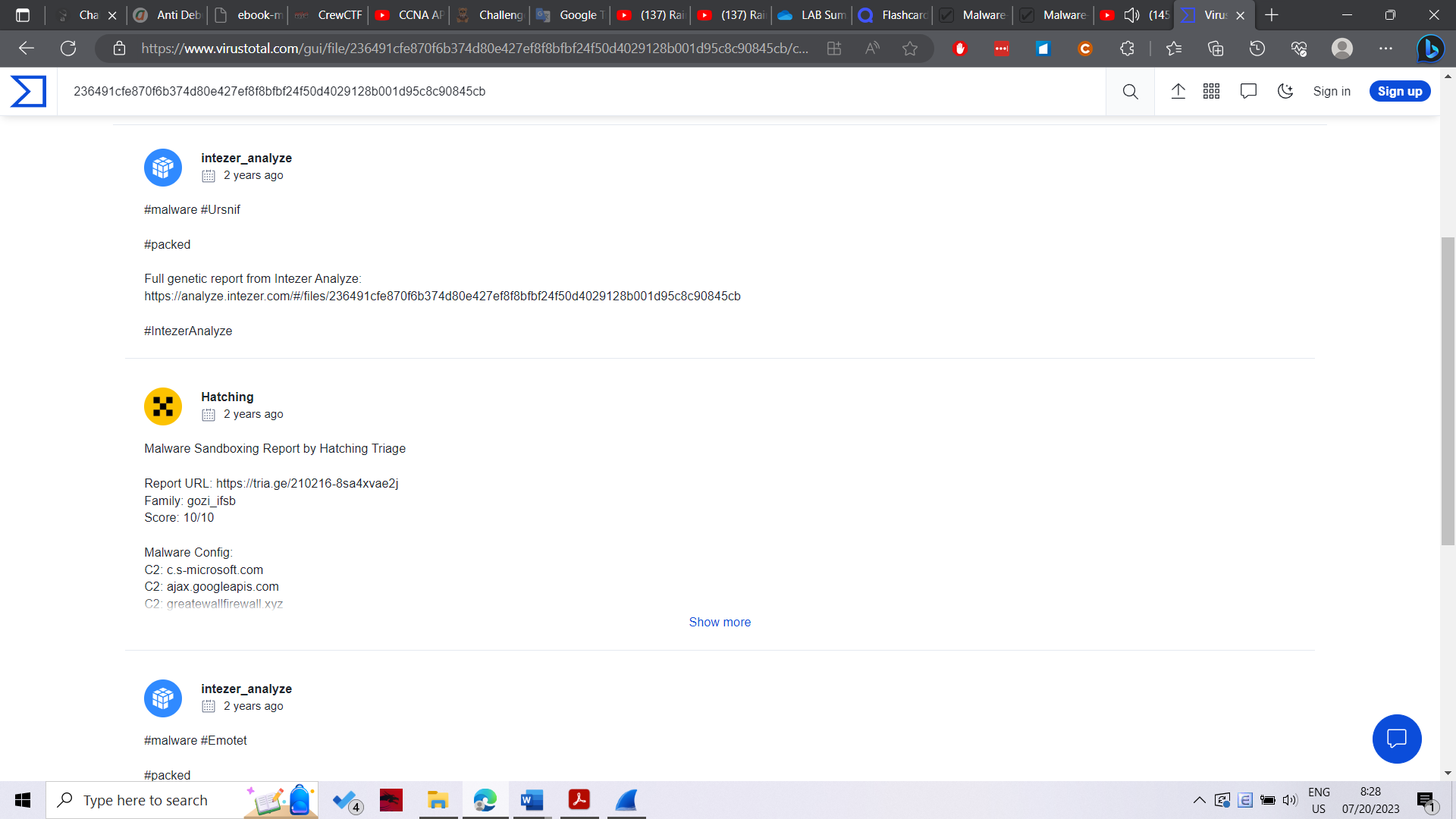


From there, you can submit the file to VirusTotal to confirm it is malware and a DLL. Here's what I found when I did it:



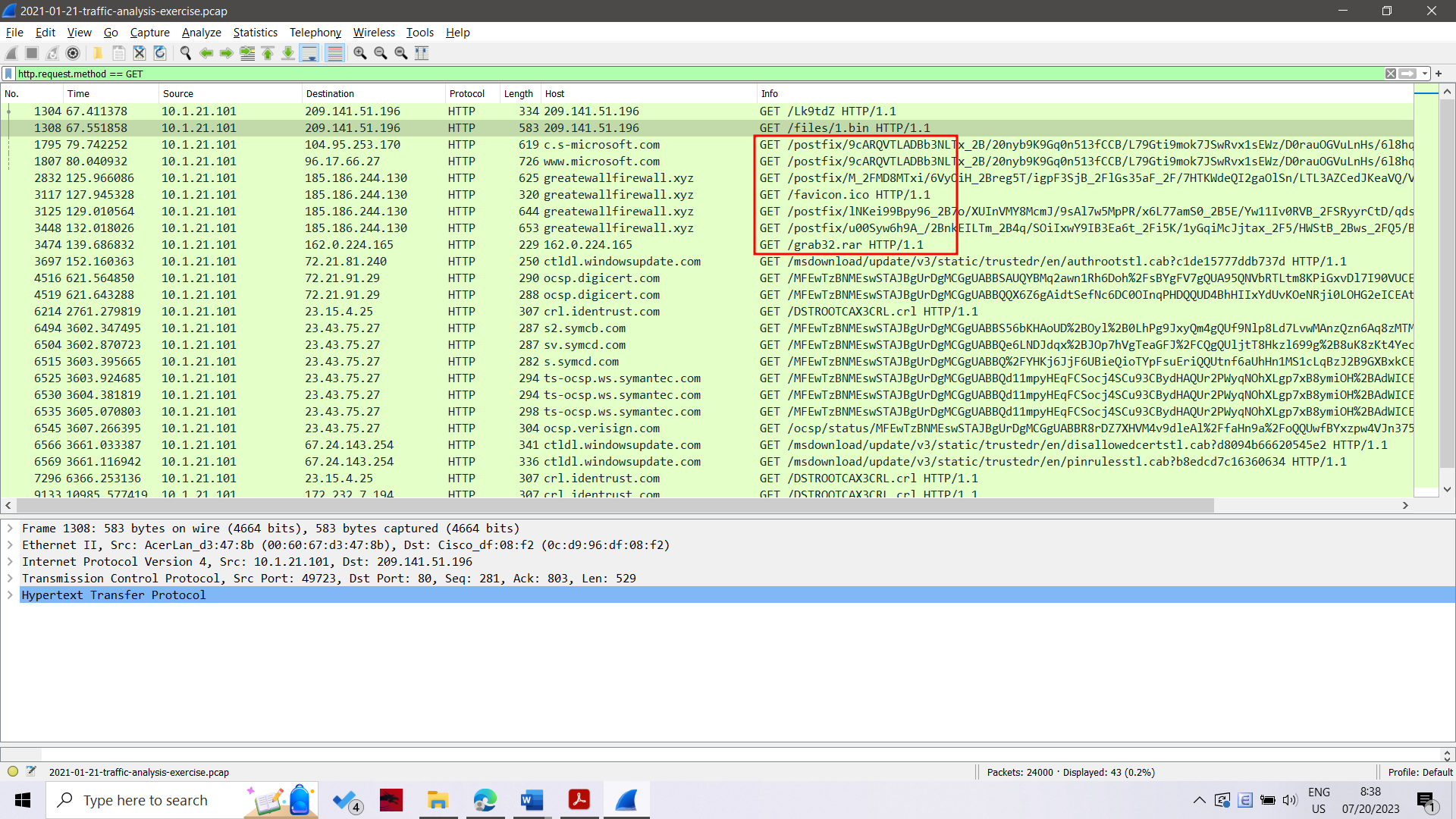
[VirusTotal - File - 236491cfe870f6b374d80e427ef8f8bfbf24f50d4029128b001d95c8c90845cb](https://www.virustotal.com/gui/file/236491cfe870f6b374d80e427ef8f8bfbf24f50d4029128b001d95c8c90845cb)



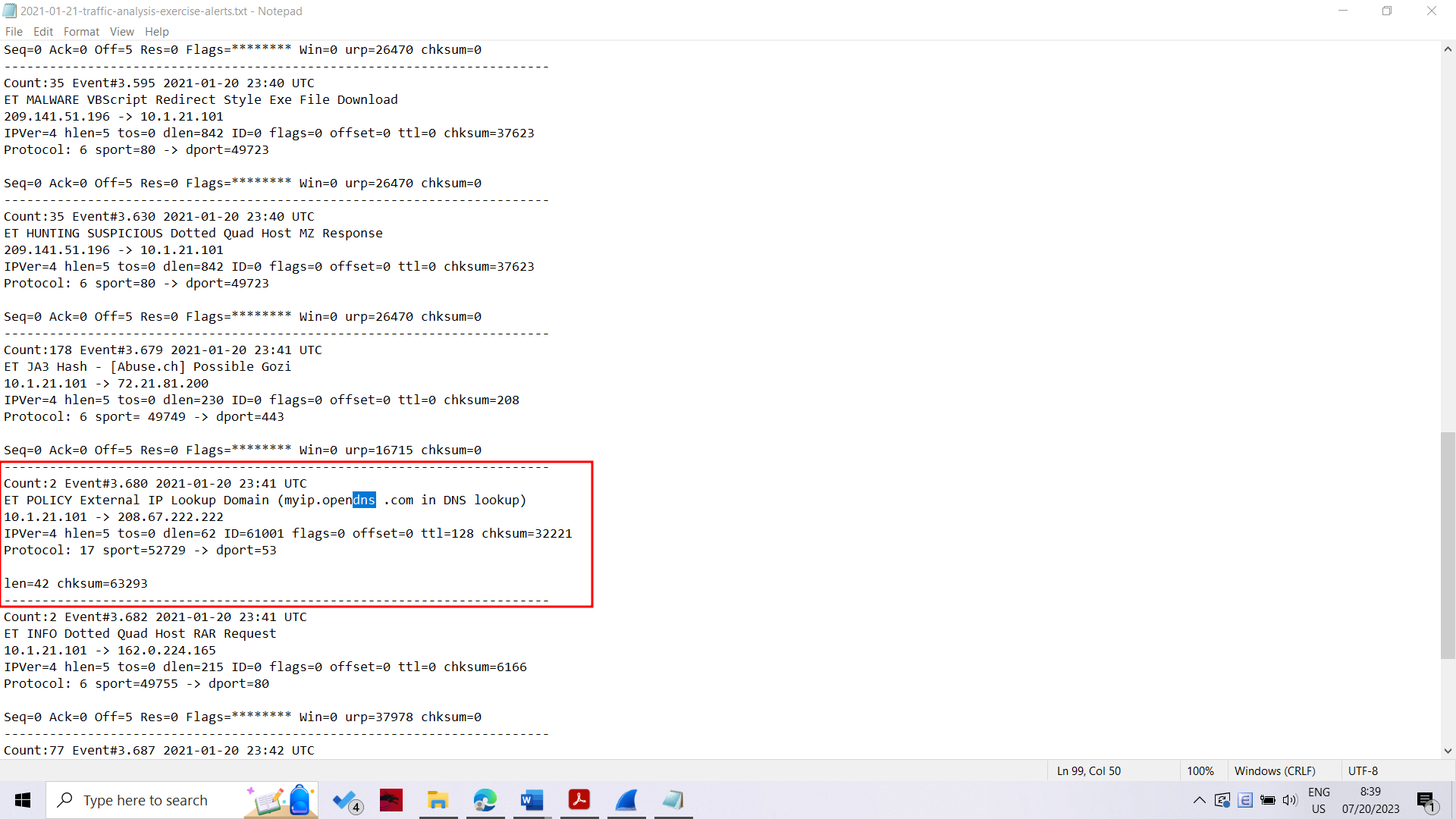


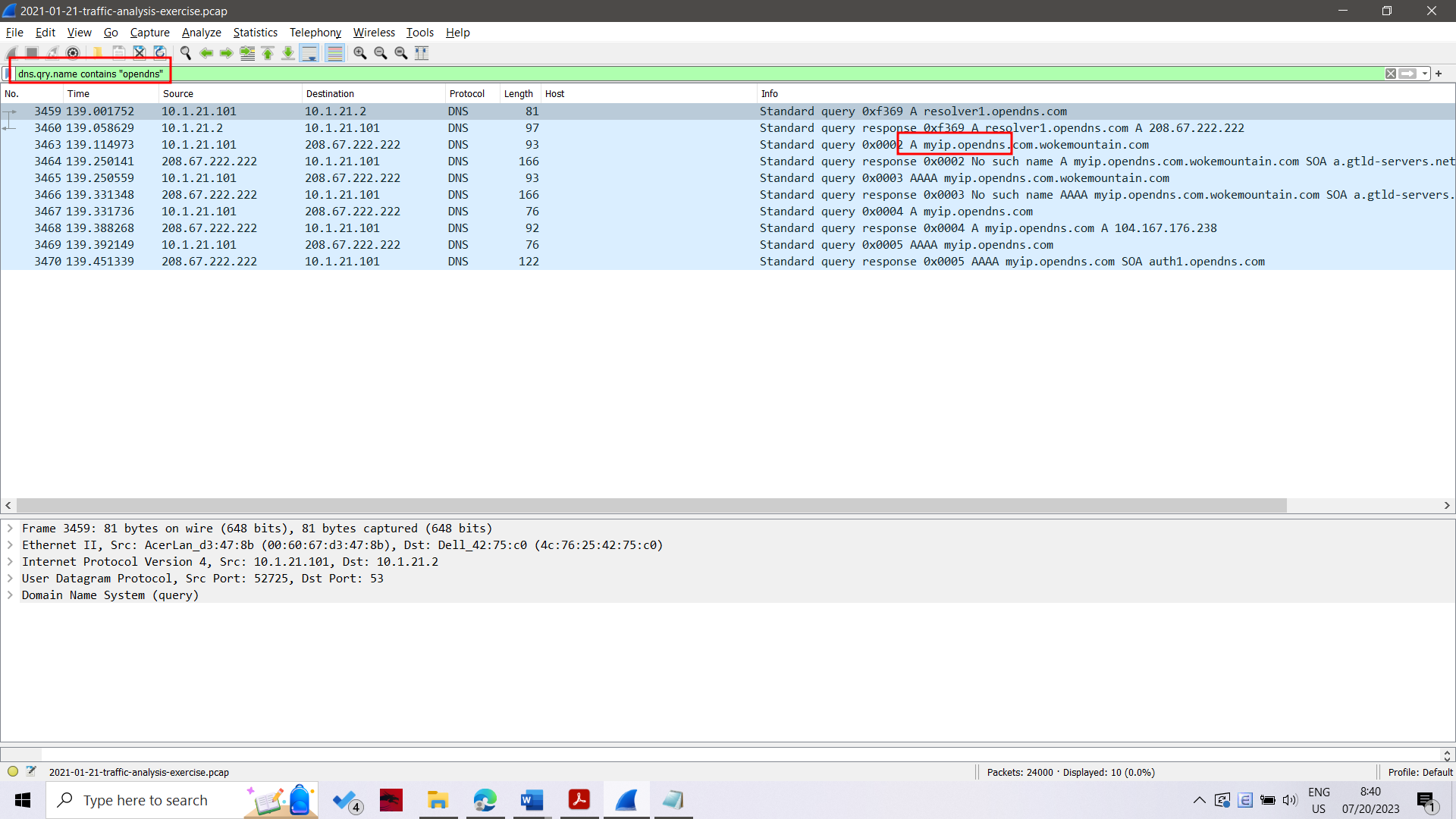
The alerts don't mention HTTP traffic for Ursnif on 185.186.244.130 over TCP port 80 to greatewallfirewall.xyz. This is visually distinctive, and once you've seen it a few times, you can recognize these very long URLs as the type used by Ursnif.

Use your basic web filter and scroll down a little bit until you see these HTTP GET requests.



The myip.opendns.com traffic is also interesting, because it's DNS traffic to a public IP address, and I rarely see it used legitimately, which is why there's an alert on it, I suppose.





**Indicators of Compromise (IOCs)**

• 209.141.51.196 port 80 - 209.141.51.196 - GET /Lk9tdZ

• 209.141.51.196 port 80 - 209.141.51.196 - GET /files/1.bin

• 185.186.244.130 port 80 - greatewallfirewall.xyz - GET /postfix/[long base64 string with backslashes and underscores].yml

• port 53 - IP address check to myip.opendns.com

• 162.0.224.165 port 80 - 162.0.224.165 - GET /grab32.rar

• 193.239.84.250 port 443 - [no domain] - HTTPS/SSL/TLS traffic

• 193.239.84.250 port 443 - booloolo3.com - HTTPS/SSL/TLS traffic

• 84.252.95.102 port 443 - [no domain] - HTTPS/SSL/TLS traffic

• 84.252.95.102 port 443 - booloolo4.com - HTTPS/SSL/TLS traffic

SHA256 hash: 236491cfe870f6b374d80e427ef8f8bfbf24f50d4029128b001d95c8c90845cb

• File size: 340,824 bytes

• File location: http://209.141.51.196/files/1.bin

• File description: malware DLL file associated with Ursnif (Gozi/ISFB)