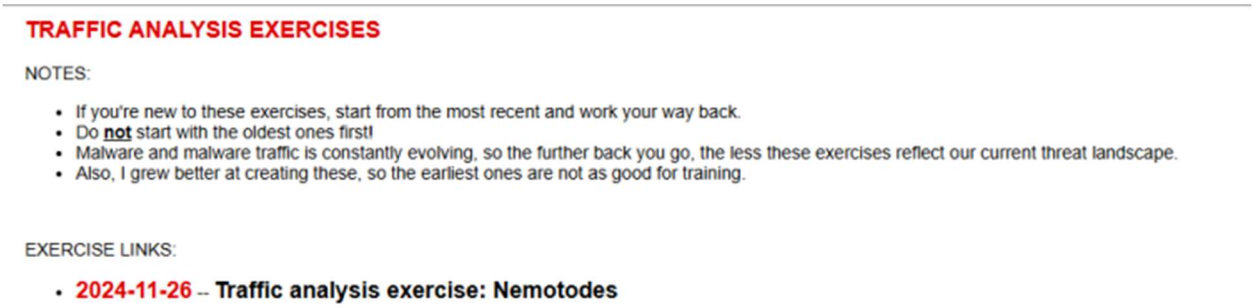
Monitoring and Incident Detection

Welcome back to another project. This time, I'm going to perform monitoring and incident detection using Wireshark. I will use the website "https://www.malware-traffic-analysis.net/%E2%80%9D" to obtain complex captures and perform monitoring.

We click on "Traffic Analysis Exercises", which will take us to another part of the page where it shows a variety of captures with different themes. In my case, I will choose the first one as recommended by the creator, since being the most recent, we can be a bit more up-to-date with the daily challenges in this topic.



We can observe that the creator of the page provides us with two zip files containing alerts and the ".pcap" along with instructions for the password of the zips. Additionally, there is a scenario to perform the corresponding tasks upon receiving a malware alert.

2024-11-26 - TRAFFIC ANALYSIS EXERCISE: NEMOTODES

ASSOCIATED FILES:

- Zip archive of the pcap: **2024-11-26-traffic-analysis-exercise.pcap.zip** 19.7 MB (19,664,067 bytes)
- Zip archive of the alerts: **2024-11-26-traffic-analysis-exercise-alerts.zip** 297.5 kB (297,496 bytes)

NOTES:

- Zip files are password-protected. Of note, this site has a new password scheme. For the password, see the "about" page of this website.

BACKGROUND

You work as a analyst at a Security Operation Center (SOC) for a medical research facility specializing in nemotodes. Alerts on traffic in your network indicate someone has been infected. You don't know which is more disgusting, the nemotodes or the malware.

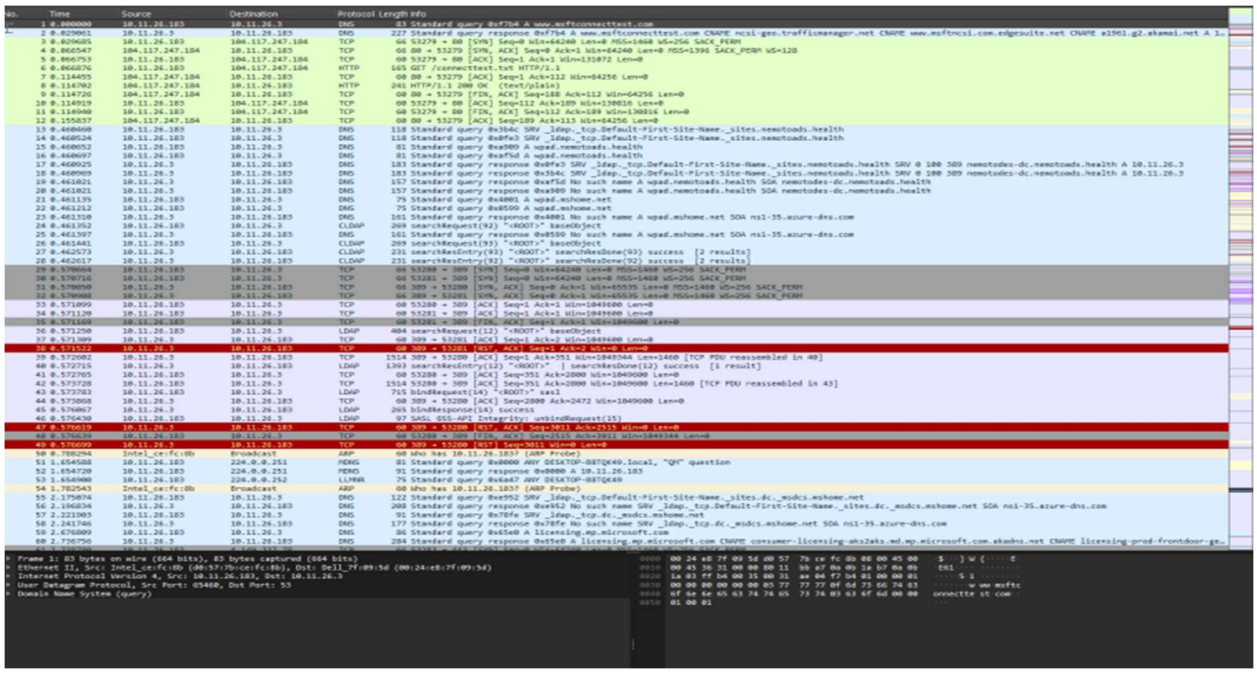
My recommendation is to always keep files organized and named. In my case, I created a folder with the files and captures I will use in this project. Once I have the material I will use, I will proceed to prepare my environment to begin the analysis. We will have Wireshark installed on our computer.

Analysis Plan

1. Initial Analysis:
   - Open the PCAP file in Wireshark to identify initial patterns.
   - Filter by common protocols such as HTTP, HTTPS, DNS and SMB
   - Look for traffic spikes or abnormal patterns.

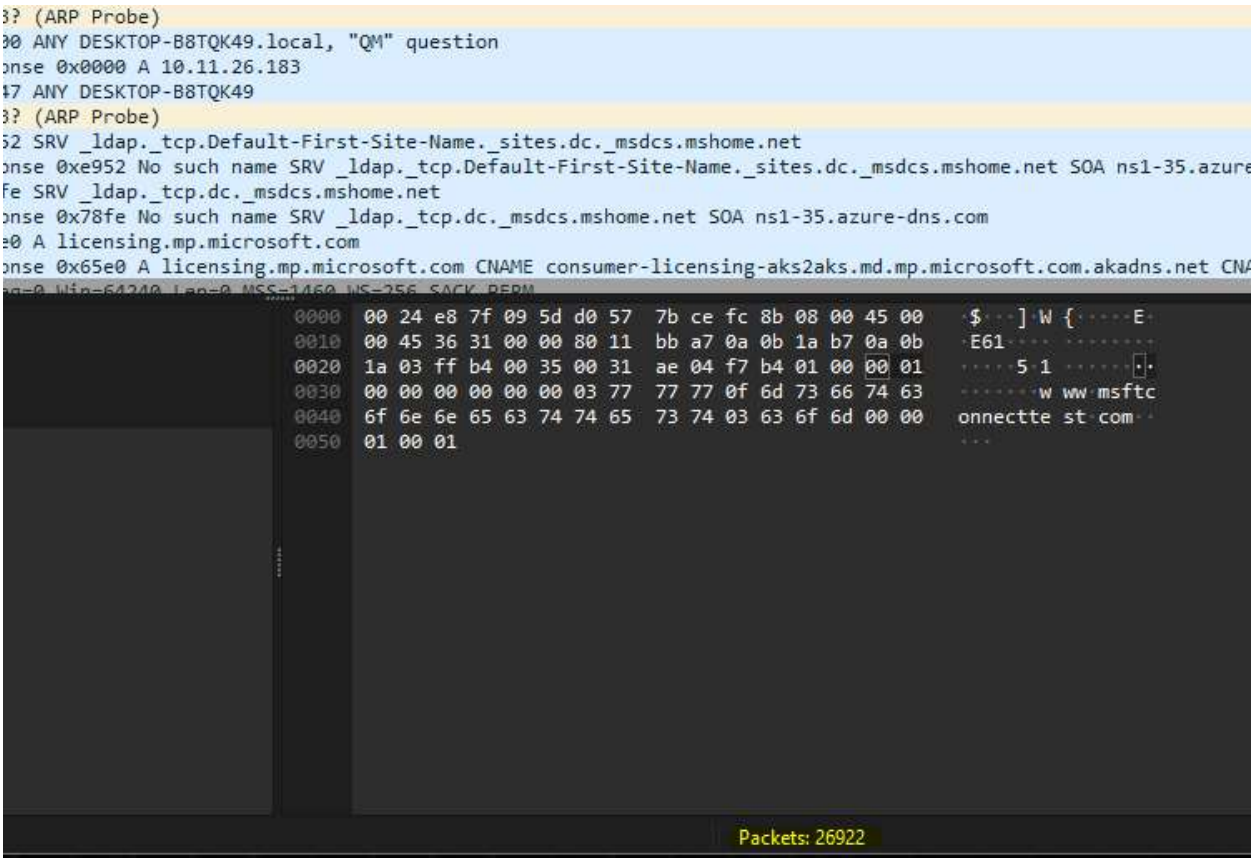
Initial Analysis

We open the ".pcap" file in Wireshark.

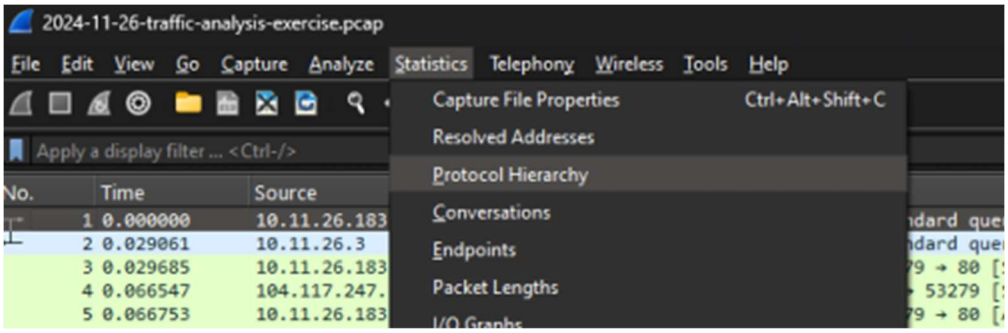


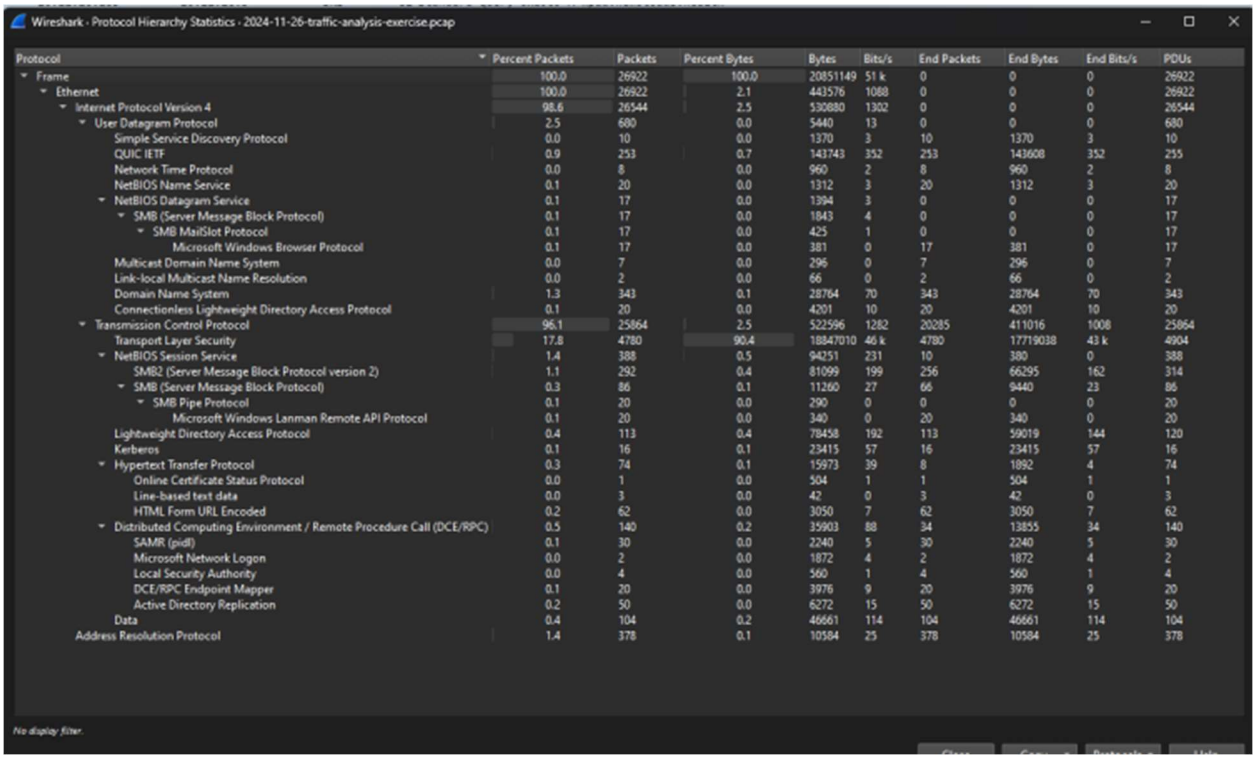
There are the basics we will observe when opening the ".pcap" file in Wireshark. We have a lot of data that we will analyze:

1. Number of Packets: in the bottom bar of Wireshark, we will see the number of packets, which will give us an idea of the traffic size. In total, we have 26,922 packets.



2. Main Protocols: Go to the menu Statistics > Protocol Hierarchy, this shows a breakdown of the present protocols.





We identify the most common protocols and note which ones could be relevant. In this case, I will focus on the HTTP, HTTPS, DNS, and SMB protocols. We can also filter by common protocols.

- HTTP: in the filter bar, we type “http”. We can look for request like GET, POST, or any other HTTP traffic.

.3

- HTTPS: in the filter bar, we type “tls” to see HTTPS traffic. Although it is encrypted, you can search for domains through the SNI (Server Name Indication) field in the TLS Client Hello packets. Right-click on a “tls” packet and select Follow > TLS Stream to analyze the TLS flow.

- DNS: Repeat the same process by applying the “dns” filter. We can examine DNS queries and their responses. Look for long, strange, or failed domains that may indicate malicious traffic.

- SMB: We type “smb” or “smb2” in the filter bar, we can examine whether there are any attempts to access shared resources that could indicate lateral movements or exfiltration attempts.

Identify anomalies (DNS)

We will use “dns” in the filter bar to search for anomalies. What are we looking for?

- Domains with strange or long names.
- DNS queries that have no response (which could be indicators of command and control attempts).

1. Example: WPAD (Web Proxy Auto-Discovery Protocol) Lookups.

No.	Time	Source	Destination	Protocol	Length	Info
187	5.794700	10.11.26.183	10.11.26.3	DNS	75	Standard query 0xe884 A wpad.mshome.net

No.	Time	Source	Destination	Protocol	Length	Info
15	0.460652	10.11.26.183	10.11.26.3	DNS	81	Standard query 0xa909 A wpad.nemotoads.health

Queries to the “wpad” domain indicate that the system is looking for automatic proxy configurations. This can be expected behavior in corporate networks, but it could also be exploited by attackers who set up a malicious WPAD server.

What can we do to detect anomalies?

- Verify if the “wpad” domains are legitimate in your network.
- If there are unexpected response or external domains in the DNS responses related to “wpad”, it could be a malicious redirection attempt.

2. Example: DNS Responses with Multiple CNAME Records.

Time	Source	Destination	Protocol	Src Port	Dest Port	Length	Info
2 0.029061	10.11.26.3	10.11.26.183	DNS	53	65460	227	Standard query response 0xf7b4 A www.msftconnecttest.com CNAME ncsi-geo.trafficmanager.net CNAME www.msftncs

The domain “www.msficonnecttest.com” is a legitimate resource used by Windows to verify internet connectivity. However, a long chain of CNAMEs could be suspicious if it comes from unknown or unrelated domains.

What can we do to detect anomalies?

- Identify if all domains in the CNAME chain are legitimate.
- If there are unrecognized domains, perform additional analysis to check if they are associated with malicious activity.

3. Example: Repeated queries to invalid domains.

Time	Source	Destination	Protocol	Src Port	Dest Port	Length	Info
113 3.248564	10.11.26.3	10.11.26.183	DNS	53	50534	177	Standard query response 0xae3d No such name SRV _ldap._tcp.dc._msdcs.mshome.net

Queries to invalid domains, such as “mshome.net”, can be normal in test environments or misconfigurations. However, if the domains are unrelated to your network, it could indicate unwanted activity.

What can we do to detect anomalies?

- Identify if the domains are related to your network.
- If not, investigate which device is making these queries and if it is legitimate.\

4. Example: Unknown domain with multiple queries.

Time	Source	Destination	Protocol	Src Port	Dest Port	Length	Info
13 0.460460	10.11.26.183	10.11.26.3	DNS	55542	53	118	Standard query 0x3b4c SRV _ldap._tcp.Default-First-Site-Name._sites.nemotoads.h

The domain “nemotoads.health” doesn’t seem to be common or well-known. If it’s not part of the infrastructure, it could be a possible misconfiguration or a malicious attempt to access.

What can we do to detect anomalies?

- Validate if this domain belongs to the organization.
- If it’s not legitimate, track the device that is generating there queries.

5. Example: Common negative response (No such name).

Time	Source	Destination	Protocol	Src Port	Dest Port	Length	Info
56 2.196834	10.11.26.3	10.11.26.183	DNS	53	59980	208	Standard query response 0xe952 No such name SRV _ldap._tcp.Default-First-Site-N

If you see many “No such name” responses, it could be a symptom of misconfigurations or infected devices trying to communicate with malicious dommains.

What can we do to detect anomalies?

- Review the device generating these queries.
- Ensure that the devices are configured correctly.

Additional tools to go deeper

- Domain checking: We can use services like “VirusTotal” to check if the domains or Ips querying have reports of malicious activity.
- Wireshark filters:
  1. Use the filter “dns.flags.rcode != 0” to focus on DNS errors.
  2. Filter by specific domains: “dns.qry.name == wpad.nemotoads.health”.

Example:

```
1374 35.856917   10.11.26.3   10.11.26.183  DNS          53       52957      96 Standard query response 0xa31d A modandcrackedapk.com A 193.42.38.139
```

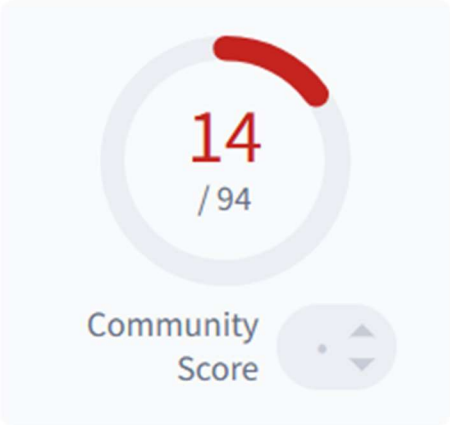
VIRUSTOTAL

SUMMARY | DETECTION | DETAILS | RELATIONS | COMMUNITY

Security vendors' analysis ⓘ — Do you want to automate checks?

alphaMountain.ai	Phishing
BitDefender	Malware
CRDF	Malicious
CyRadar	Malicious
ESET	Phishing
G-Data	Malware
Lionic	Malware
Lumu	Malware
MalwareURL	Malware
Seclookup	Malicious
SOCRadar	Malicious
Sophos	Phishing
VIPRE	Phishing
Webroot	Malicious
Gridinsoft	Suspicious
Abusix	Clean
Acronis	Clean
ADMINUSLabs	Clean

14/94 security vendors flagged this domain as malicious



modandcrackedapk.com

Creation date
1 year ago

Last analysis date
2 days ago

Identify Anomalies (HTTP and HTTPS)

For HTTP traffic, we start by using the filter “http”, or we can be more specific by using the following filters:

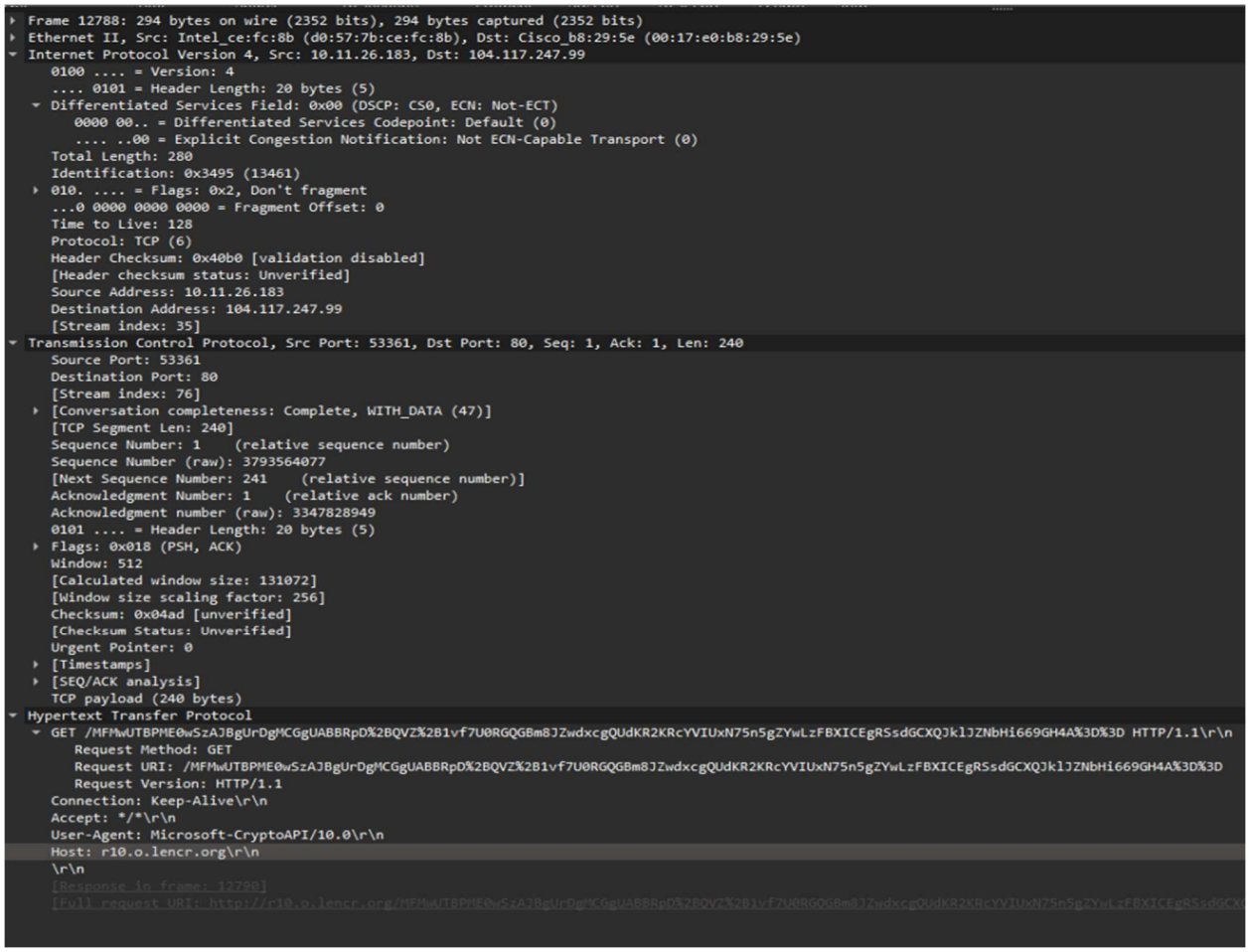
- “http.request.method == “GET” ” (For GET request).
- “http.request.method == “Post” ” (For POST request).

For HTTPS traffic, we use the filter “ssl” or “tls.handshake.type == 1” (To identify TLS packets with SNI).

1. Example:

   We can observe the result of the filter “http.request.method == “GET” ”

This caught my attention, I double-clicked it to inspect the details. Such as the host (domain) and packet information.



1. As highlighted information we have the host (domain):
    - The domain of interest is r10.o.lencr.org. This should be investigate to verify if it is legitimate or suspicious.
2. Packet information:
    - HTTP Method: GET
    - URL:

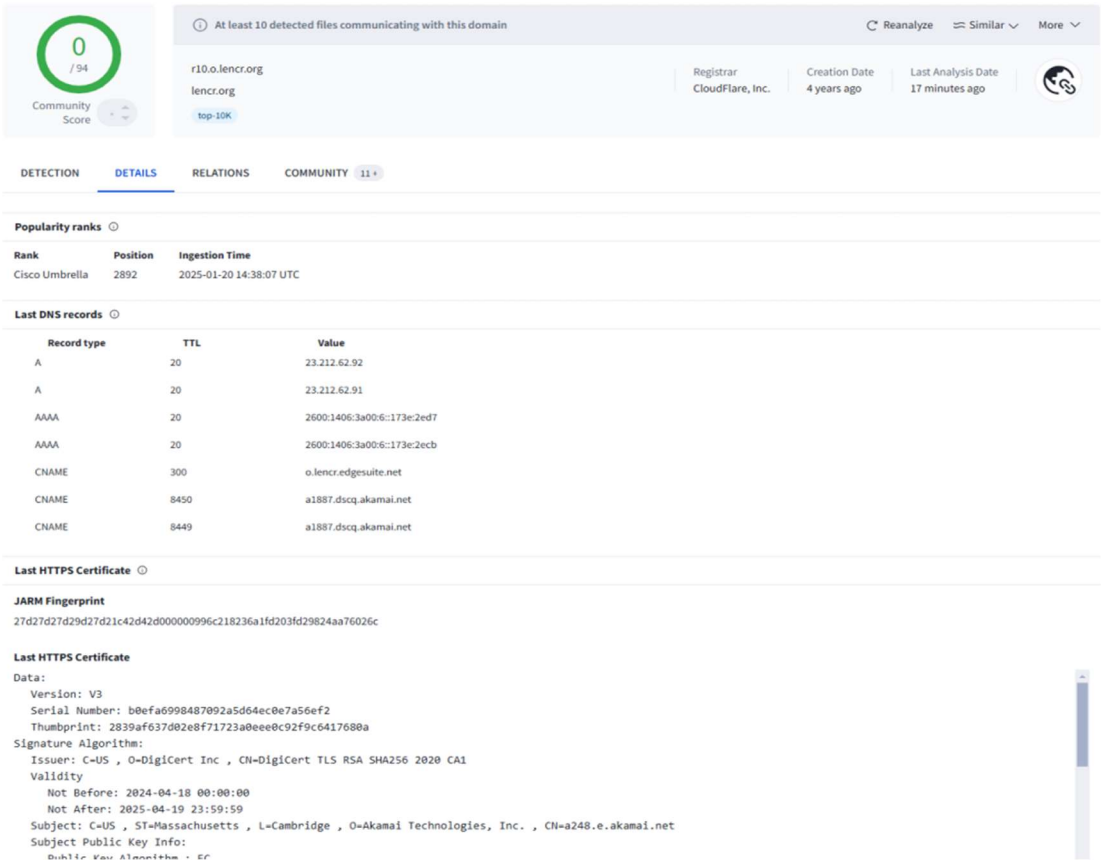
/MFMwUTBPME0wSzAJBgUrDgMCGgUABBRpD%2BQVZ%2B1vf7U0RGQGBm8JZwdxcgQUdKR2KRcYVIUxN75n5gZYwLzFBXICEgRSsdGCXQJklJZNbHi669GH4A%3D%3D

Complex and long, but it is important to analyze if it paints to any malicious or interesting resource.

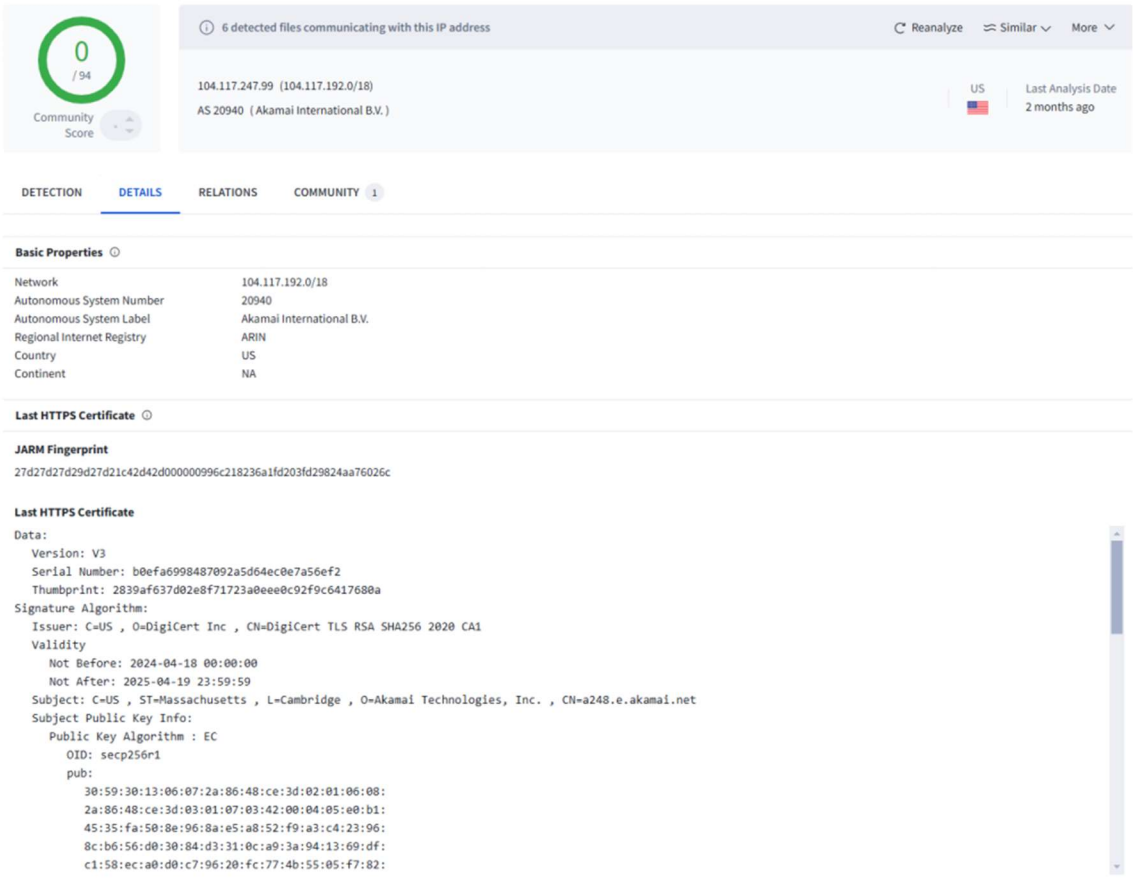
- User-Agent: Microsoft-CryptoAPI/10.0.
- Dest IP: 104.117.247.99, corresponds to the domain but more details about the IP can be verified.
- Connection: Keep-Alive.

The steps to analyze them in depth are not so different from what we did the DNS:

1. We verify the domain in VirusTotal:



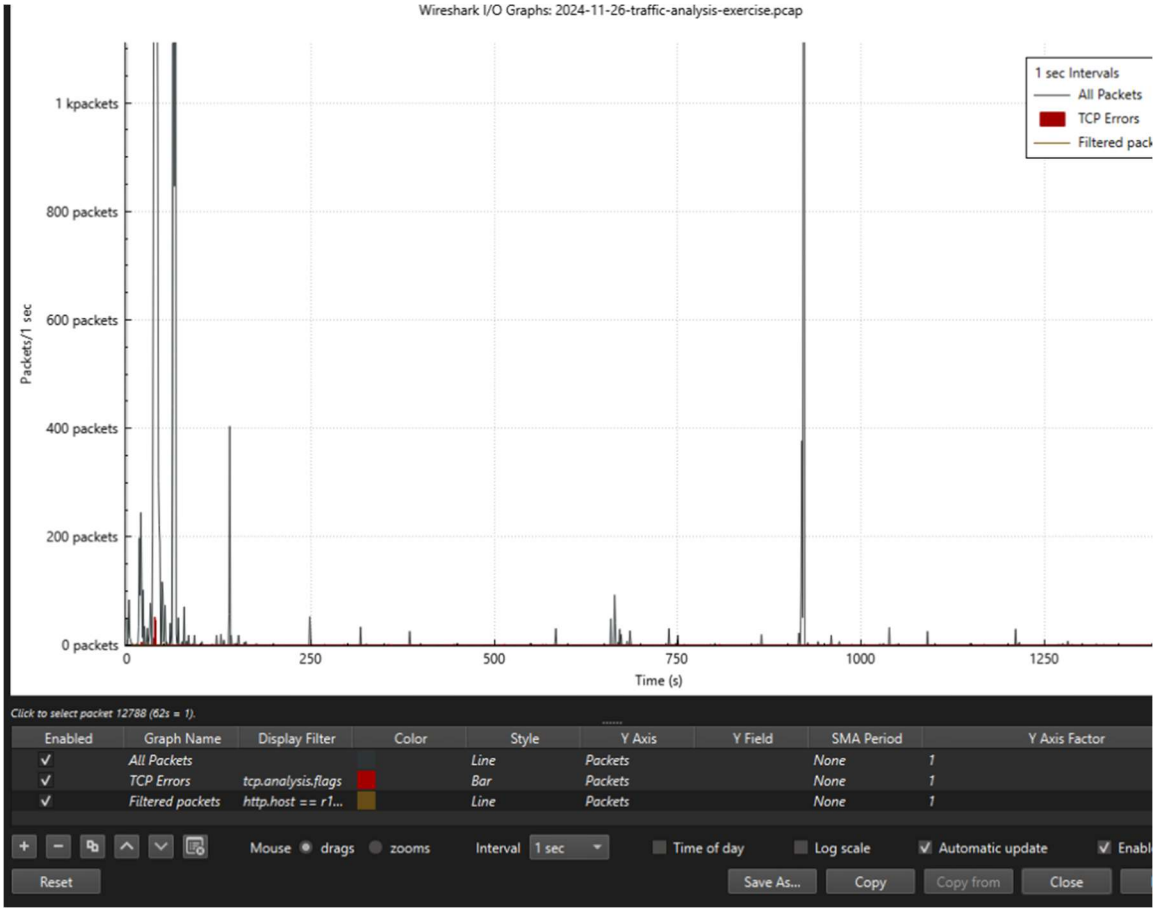
2. We verify the destination IP:



3. Analyze the traffic context:
   - In Wireshark, we filter all requests from the host “http.host == “r10.o.lencr.org” ”, to be able to see all HTTP requests to this domain.

   We can observe that we only have one request.

   - Identify temporal patterns: We can review the “Time” column to see if the requests are mode at intervals or unexpected times, such as outside of business hours.
   - If the requests are frequent and do not correlate with normal activities, it could indicate suspicious behavior.
   - We can use tools like traffic flow graphs to visualize the volume of requests to this domain compared to others,



Identify relevant patterns.

1. **Traffic spikes**: By observing the initial spikes in the graph, such as in the first 500 seconds. This indicates a significant increase in the number of packets per second. These spikes could represent unusual events, such as massive data transfers or attacks.

2. **Sustained or sporadic traffic**: After the spikes, there appears to be more constant traffic with some sporadic activity.

3. **TCP errors**: The red line shows events related to TCP errors. If you see an increase in this line, it could indicate retransmissions or failed connections.

Example 2:

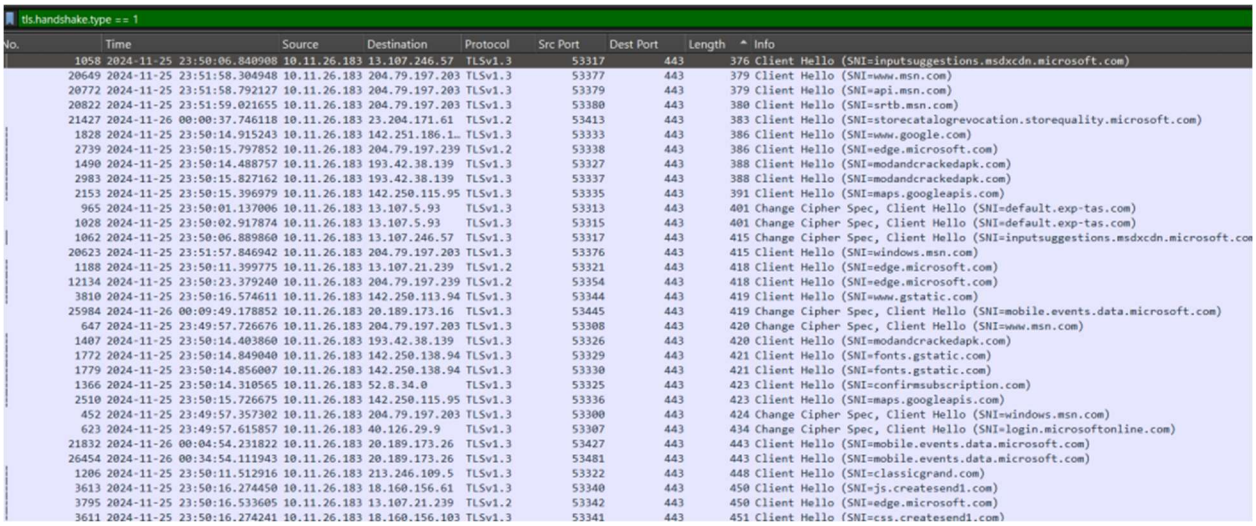
Now we observe using the filter “http.request.method == “POST” ”.

We can repeat the same procedure for this analysis. In this case, there isn't much to do since it's a capture made as part of a practice.

However, I will mention the patterns to observe for HTTP POST:

- **Source IP**: 10.11.26.183 (internal host).
- **Destination IP**: 194.180.191.64 (external host).
- **Destination URL**: http://194.180.191.64/fakeurl.htm.
- **Content-Type**: application/x-www-form-urlencoded.
- **Port**: 443 (HTTPS).

For HTTPS traffic, we search in the filter "ssl" or "tls.handshake.type == 1" to identify TLS packets with SNI.



We can observe at first glance that there are common SNIs in the domains:

- inputsuggestions.msdxcdn.microsoft.com
- www.msn.com
- edge.microsoft.com
- storecatalogrevocation.storequality.microsoft.com
- mobile.events.data.microsoft.com

These domains are legitimate and associated with Microsoft services such as Edge, MSN, and Windows updates.

The traffic appears to be related to automatic updates and basic functions of the operating system.

Google and related services:

- www.google.com
- maps.googleapis.com
- fonts.gstatic.com

These are also legitimate domains used for maps, fonts, and web navigation services. No suspicious activity.

Uncommon or potentially suspicious domains:

- modandcrackedapk.com

This domain is related to modified or pirated software, which represents a potential security risk.

- confirmsubscription.com

Although not necessarily malicious, it could be associated with email or notification systems. It would be best to authenticate this domain.

Example 1:

	Time	Source	Destination	Protocol	Src Port	Dest Port	Length	Info
20779	2024-11-25 23:51:58.861647	10.11.26.183	204.79.197.203	TLSv1.3	53379	443	621	Change Cipher Spec, Client Hello
20842	2024-11-25 23:51:59.092213	10.11.26.183	204.79.197.203	TLSv1.3	53380	443	622	Change Cipher Spec, Client Hello
20627	2024-11-25 23:51:57.911883	10.11.26.183	204.79.197.203	TLSv1.3	53376	443	657	Change Cipher Spec, Client Hello
12376	2024-11-25 23:50:26.773909	10.11.26.183	213.246.109.5	TLSv1.3	53356	443	751	Client Hello (SNI=classicgrand.co
20501	2024-11-25 23:51:02.988273	10.11.26.183	52.113.194.132	TLSv1.3	53370	443	781	Client Hello (SNI=ecs.office.com)
21305	2024-11-25 23:56:03.469275	10.11.26.183	52.113.194.132	TLSv1.3	53400	443	781	Client Hello (SNI=ecs.office.com)

In this example, we have a domain with the SNI “ecs.office.com”. We verify the destination IP on the VirusTotal.com community and get the following report.

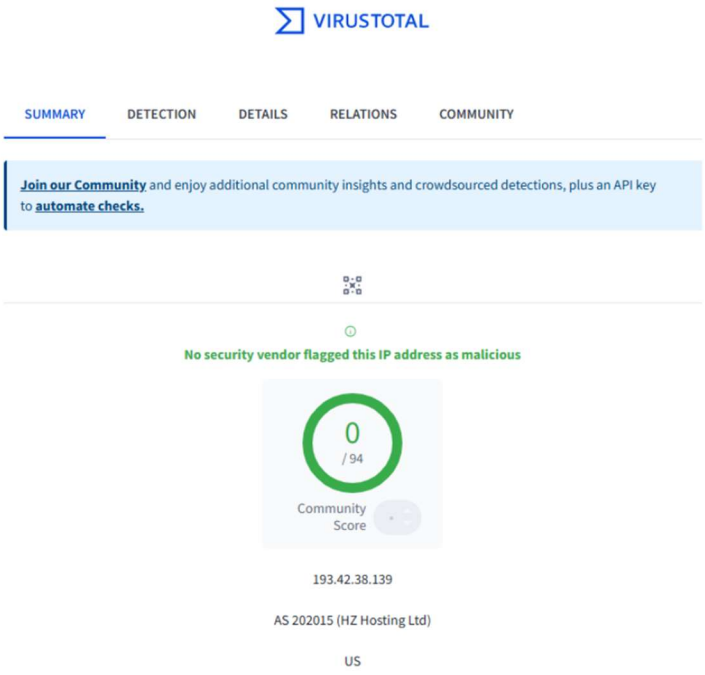
We can observe the presence of malware.

Example 2:

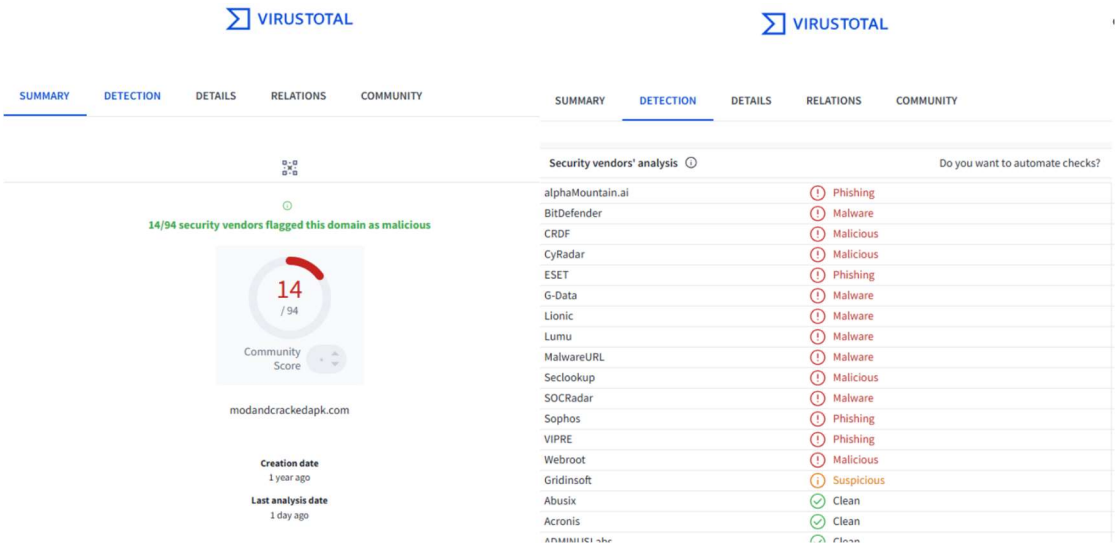
Now let’s proceed with one of the domains I mentioned earlier.

	Time	Source	Destination	Protocol	Src Port	Dest Port	Length	Info
2739	2024-11-25 23:50:15.797852	10.11.26.183	204.79.197.239	TLSv1.2	53338	443	386	Client Hello (SNI=edge.microsoft.com)
1490	2024-11-25 23:50:14.488757	10.11.26.183	193.42.38.139	TLSv1.3	53327	443	388	Client Hello (SNI=modandcrackedapk.com)
2983	2024-11-25 23:50:15.827162	10.11.26.183	193.42.38.139	TLSv1.3	53337	443	388	Client Hello (SNI=modandcrackedapk.com)

When investigating the destination IP, I found nothing relevant or malicious.



However, with the domain, we do have a report indicating malicious activity.



On the other hand, we will mention the protocols and versions, and their differences:

- **TLSv1.2 and TLSv1.3**:
  Most connections use TLSv1.3, which is positive due to its enhanced security features. Some connections still use TLSv1.2 (edge.microsoft.com, among others), which could be a concern if these versions are not updated.

Filter SMB traffic:

SMB (port 445) is a protocol used for sharing files, printers, and other resources on a network. In Wireshark, we can filter SMB traffic using this filter:
“smb || tcp.port == 445”.

To understand the structure of SMB traffic, I'll explain a bit about what we can observe in the capture.

- **Source and Destination**: This indicates that the machine is sending and receiving the packets.
- **Protocol**: We can observe a normal result based on the filter we applied, because SMB packets operate over TCP.
- **Ports**: In the Src Port and Dest Port section, it shows the ports used.
- **Info**: Details about the packets (ACK, RST, Keep-Alive).

1. **Identify normal patterns:**

In typical SMB traffic, you should see:

   - **Established connection**: A TCP handshake (SYN, SYN-ACK, ACK) followed by SMB transfers.
   - **Keep-Alive**: Packets to keep the connection active.
   - **SMB Data**: Packets with specific operations (Tree Connect, Read, Write).

   In the capture, we can observe:

   - **ACK**: Packets confirming reception.
   - **RST, ACK**: Packets that terminate the connection unexpectedly.

2. **Identify anomalies**:

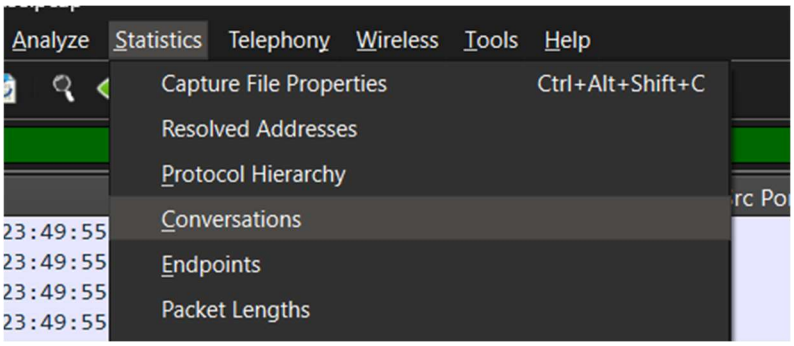
   - Based on the capture, here are potential anomalies:
   - **RST (Reset) packets**: The packets that appear in red (RST, ACK) indicate that one of the machines is closing the connection abruptly.
   - **Investigation**: This could be a legitimate interruption or a failed connection attempt. We can inspect previous packets to identify if there were any errors.
   - **Keep-Alive repetitions**:

Many connections show TCP Keep-Alive packets. This could be normal, but if there is an excess, it might indicate a communication issue.

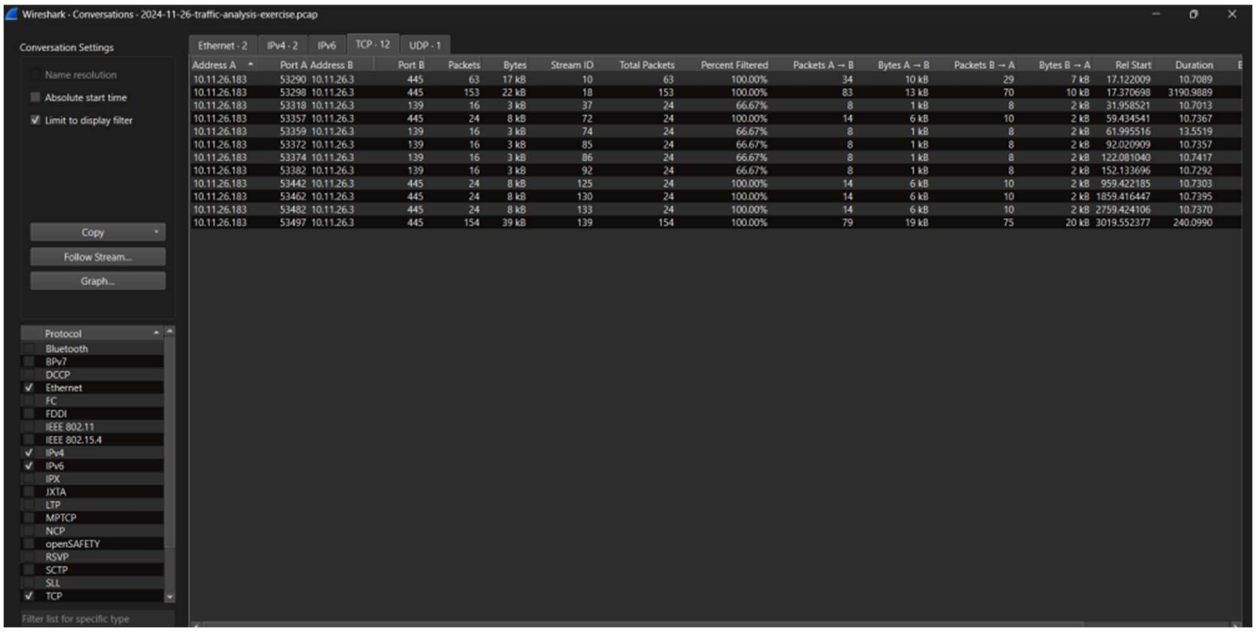
- **Investigation**: We can check if these connections are consuming resources unusually or if they come from a suspicious source.

3. **Unusual source ports**:
   - Although destination port 445 is expected, check the source ports (53290, 53357, etc.) to identify if there are any strange patterns.
   - **Investigation**: We can verify if the source ports change rapidly (this could indicate a port scan).
4. **Tools for deeper analysis**:

Wireshark has several tools we've mentioned earlier, such as viewing traffic statistics. In this case, we'll select one of the packets marked in red and go to the **Statistics > Conversations** tab.



And a tab like this will open.



What you observe in **Statistics > Conversations** in Wireshark shows multiple connections between the same source IP address (10.11.26.183) and the destination IP (10.11.26.3) on port 445 (SMB - Server Message Block). This repetitive traffic could indicate:

1. **Legitimate SMB activity**: If these devices are on a corporate or test network, there could be normal SMB traffic, such as authentications or file transfers.
2. **Brute force attacks or port scans**: If the traffic repeats excessively in short intervals, someone might be trying to brute-force credentials on SMB.
3. **Malware propagation**: Some malware, such as WannaCry or EternalBlue, exploits vulnerabilities in SMB and generates unusual traffic on port 445.

What to do?

- **Filter by IP and review the packets**: Use the filter ip.addr == 10.11.26.183 && tcp.port == 445 to see details of the traffic.
- **Check connection patterns**: If the connections have failed authentication attempts, it could be a brute force attack.
- **Review packet timings and sizes**: Legitimate SMB traffic usually transfers files, while malicious traffic may send small packets with suspicious commands.

I filtered the suspicious IP address to analyze further and came across the following result:

Ethernet · 6 | IPv4 · 60 | IPv6 | TCP · 142 | UDP · 134

Address A	Port A	Address B	Port B	Packets	Bytes	Stream ID	Total Packets	Percent Filtered	Packets A → B	Bytes A → B	Packets B → A	Bytes B → A	Rel Start	Duratio
10.11.26.183	53282	4.149.227.78	443	35	15 kB	3	35	100.00%	17	4 kB	18	11 kB	2.738780	0.4859
10.11.26.183	53283	4.149.227.78	443	34	15 kB	4	34	100.00%	17	4 kB	17	11 kB	3.139470	0.4624
10.11.26.183	53311	4.150.155.223	443	21	10 kB	31	21	100.00%	10	2 kB	11	8 kB	21.815305	0.4295
10.11.26.183	53312	4.150.155.223	443	22	11 kB	32	22	100.00%	11	2 kB	11	9 kB	22.286588	0.4086
10.11.26.183	53280	10.11.26.3	389	15	6 kB	1	15	100.00%	8	3 kB	7	3 kB	0.570664	0.0060
10.11.26.183	53281	10.11.26.3	389	6	372 bytes	2	6	100.00%	3	186 bytes	3	186 bytes	0.570716	0.0008
10.11.26.183	53286	10.11.26.3	88	8	898 bytes	6	8	100.00%	4	467 bytes	4	431 bytes	16.964457	0.0018
10.11.26.183	53287	10.11.26.3	88	10	3 kB	7	10	100.00%	5	607 bytes	5	2 kB	16.970538	0.0026
10.11.26.183	53288	10.11.26.3	88	13	4 kB	8	13	100.00%	6	2 kB	7	2 kB	16.973351	0.0034
10.11.26.183	53289	10.11.26.3	135	14	2 kB	9	14	100.00%	8	964 bytes	6	1 kB	17.121276	13.4971
10.11.26.183	53290	10.11.26.3	445	63	17 kB	10	63	100.00%	34	10 kB	29	7 kB	17.122009	10.7089
10.11.26.183	53291	10.11.26.3	49671	22	6 kB	11	22	100.00%	12	4 kB	10	2 kB	17.122876	33.4894
10.11.26.183	53292	10.11.26.3	88	12	4 kB	12	12	100.00%	6	2 kB	6	2 kB	17.124356	0.0026
10.11.26.183	53293	10.11.26.3	88	12	4 kB	13	12	100.00%	6	2 kB	6	2 kB	17.124569	0.0022
10.11.26.183	53294	10.11.26.3	88	12	4 kB	14	12	100.00%	6	2 kB	6	2 kB	17.127182	0.0014
10.11.26.183	53295	10.11.26.3	49693	12	3 kB	15	12	100.00%	7	2 kB	5	1 kB	17.132732	13.4857
10.11.26.183	53296	10.11.26.3	389	19	8 kB	16	19	100.00%	10	4 kB	9	4 kB	17.135846	0.0212
10.11.26.183	53297	10.11.26.3	389	15	4 kB	17	15	100.00%	8	3 kB	7	929 bytes	17.144210	0.0104
10.11.26.183	53298	10.11.26.3	445	153	22 kB	18	153	100.00%	83	13 kB	70	10 kB	17.370698	3190.9889
10.11.26.183	53299	10.11.26.3	88	12	4 kB	19	12	100.00%	6	2 kB	6	2 kB	17.372243	0.0038
10.11.26.183	53303	10.11.26.3	135	16	3 kB	23	16	100.00%	10	1 kB	6	2 kB	18.965094	14.2702
10.11.26.183	53304	10.11.26.3	49671	36	10 kB	24	36	100.00%	20	6 kB	16	4 kB	18.967733	14.2676
10.11.26.183	53305	10.11.26.3	389	21	8 kB	25	21	100.00%	11	4 kB	10	4 kB	18.979421	0.0315
10.11.26.183	53306	10.11.26.3	389	18	7 kB	26	18	100.00%	9	3 kB	9	4 kB	19.006320	0.0045
10.11.26.183	53318	10.11.26.3	139	24	4 kB	37	24	100.00%	13	2 kB	11	2 kB	31.958521	10.7013
10.11.26.183	53319	10.11.26.3	389	17	4 kB	38	17	100.00%	9	3 kB	8	1 kB	32.552332	0.0131
10.11.26.183	53320	10.11.26.3	389	18	7 kB	39	18	100.00%	9	3 kB	9	4 kB	32.559300	0.0060
10.11.26.183	53357	10.11.26.3	445	24	8 kB	72	24	100.00%	14	6 kB	10	2 kB	59.434541	10.7367
10.11.26.183	53359	10.11.26.3	139	24	4 kB	74	24	100.00%	13	2 kB	11	2 kB	61.995516	13.5519
10.11.26.183	53366	10.11.26.3	135	10	2 kB	80	10	100.00%	6	844 bytes	4	872 bytes	78.068289	0.0332

Key observation:

1. **Connections to multiple HTTPS servers (port 443):**
   - Connections to 4.149.227.78 and 4.150.155.223 through port 443 (HTTPS) with date flow of around 15 kB in packets. This be legitimate web traffic or malicious activity if connecting to multiple servers in a short time.

2. **LDAP/Active Directory connections (port 389):**
    - There are connection attempts to the server 10.11.26.3 in port 389 (LDAP). LDAP is used to authentication and directory management. A high number of connections could indicate user enumeration attempts or stolen credentials.
3. **Traffic on SMB and NetBIOS ports (445, 139, 135):**
    - Port 445 (SMB) with 153 packets and 22kB of data transferred.
    - Port 139 (NetBIOS) and 135 (RPC) with multiple connections.
    - This could indicate attempts to explore or access Windows shared resources.
4. **Multiple connections on port 88 (Kerberos):**
    - Multiple connection attempts to port 88, which is used Active Directory authentication. A high number of connections here could indicate authentication attacks, such as Pass-the-Ticket or Kerberoasting.
5. **Long durations in some sessions:**
    - Some connections last more than 3,000 seconds, suggesting persistent sessions or prolonged activity on the system. This could be legitimate data transfers or attempts at data exfiltration.