Task 5: Capture and Analyze Network Traffic Using Wireshark

Cybersecurity Lab Report

KANDI TEJASREE
ELEVATE LABS

Task 5:

Capture and Analyze Network Traffic Using Wireshark

Index:

1.Install Wireshark.
2.Start capturing on your active network interface.
3.Browse a website or ping a server to generate traffic.
4.Stop capture after a minute.
5.Filter captured packets by protocol (e.g., HTTP, DNS, TCP).
6.Identify at least 3 different protocols in the capture.
7.Export the capture as a .pcap file.
8.Summarize your findings and packet details.

Objective:

Capture live network packets and identify basic protocols and traffic types.

Tools Used:

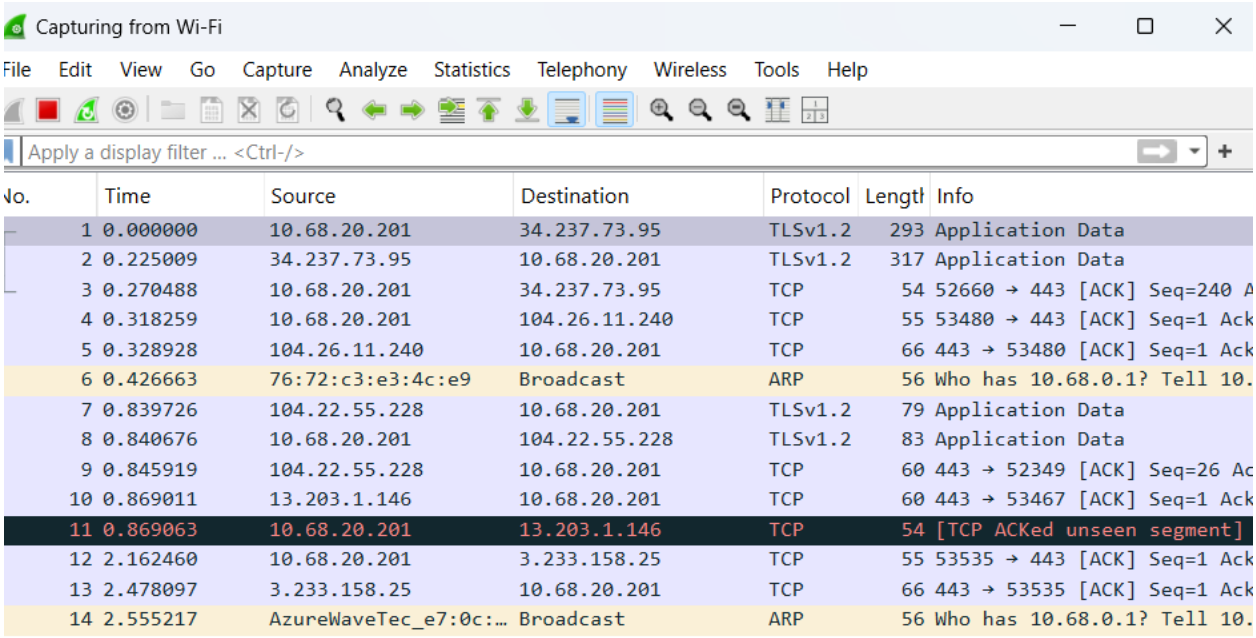
- Wireshark

Steps Performed:

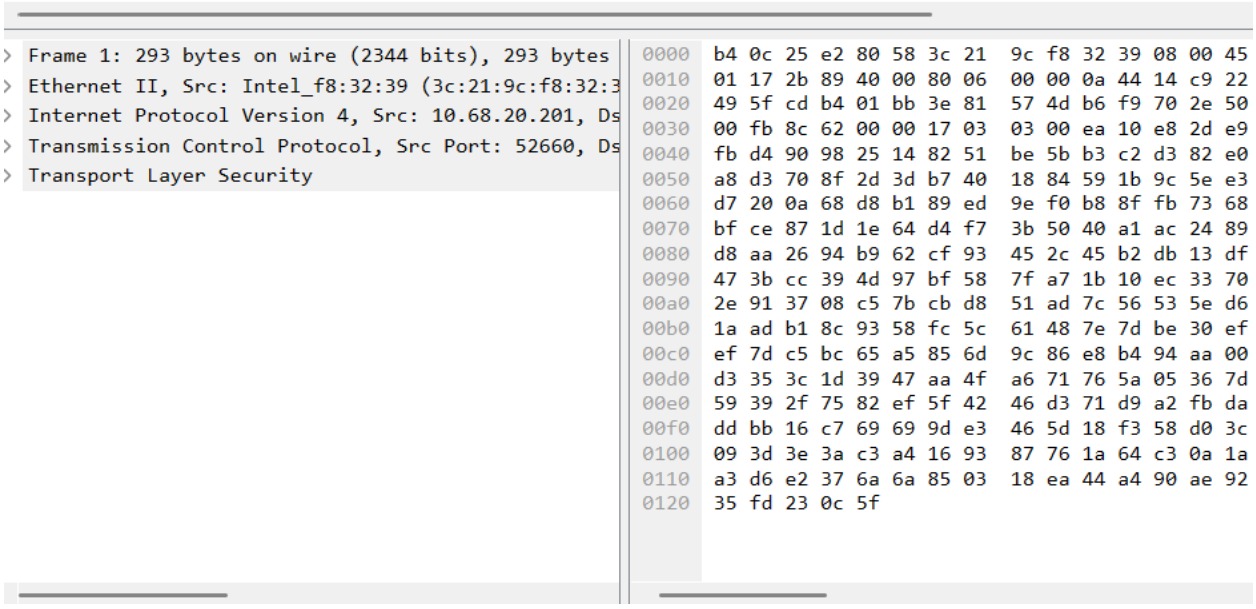
Install **Wireshark**

- Download wireshark and install it.
- During the installation accept the installation of Npcap(necessary for packet capturing).
- Launch wireshark after completion of installation.

Open Wireshark and select your active network interface from the list(usually **Wi-Fi or Ethernet**).
Click on the interface to begin capturing live packets.



No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	10.68.20.201	34.237.73.95	TLSv1.2	293	Application Data
2	0.225009	34.237.73.95	10.68.20.201	TLSv1.2	317	Application Data
3	0.270488	10.68.20.201	34.237.73.95	TCP	54	52660 → 443 [ACK] Seq=240 A
4	0.318259	10.68.20.201	104.26.11.240	TCP	55	53480 → 443 [ACK] Seq=1 Ack
5	0.328928	104.26.11.240	10.68.20.201	TCP	66	443 → 53480 [ACK] Seq=1 Ack
6	0.426663	76:72:c3:e3:4c:e9	Broadcast	ARP	56	Who has 10.68.0.1? Tell 10.
7	0.839726	104.22.55.228	10.68.20.201	TLSv1.2	79	Application Data
8	0.840676	10.68.20.201	104.22.55.228	TLSv1.2	83	Application Data
9	0.845919	104.22.55.228	10.68.20.201	TCP	60	443 → 52349 [ACK] Seq=26 Ac
10	0.869011	13.203.1.146	10.68.20.201	TCP	60	443 → 53467 [ACK] Seq=1 Ack
11	0.869063	10.68.20.201	13.203.1.146	TCP	54	[TCP ACKed unseen segment]
12	2.162460	10.68.20.201	3.233.158.25	TCP	55	53535 → 443 [ACK] Seq=1 Ack
13	2.478097	3.233.158.25	10.68.20.201	TCP	66	443 → 53535 [ACK] Seq=1 Ack
14	2.555217	AzureWaveTec_e7:0c:…	Broadcast	ARP	56	Who has 10.68.0.1? Tell 10.

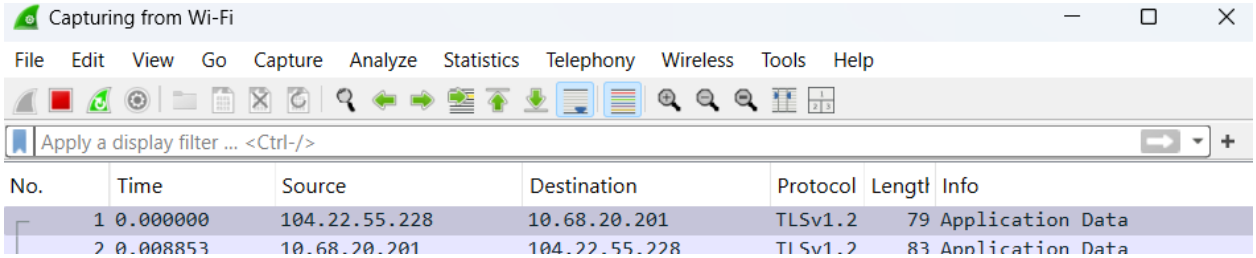


While wireshark is running in background open a web browser and visit any website Ex: https://example.com. Open the command prompt and run command: **ping google.com**

```
C:\Users\tejas>ping google.com

Pinging google.com [142.250.194.46] with 32 bytes of data
:
Reply from 142.250.194.46: bytes=32 time=28ms TTL=118
Reply from 142.250.194.46: bytes=32 time=30ms TTL=118
Reply from 142.250.194.46: bytes=32 time=30ms TTL=118
Reply from 142.250.194.46: bytes=32 time=31ms TTL=118

Ping statistics for 142.250.194.46:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 28ms, Maximum = 31ms, Average = 29ms

C:\Users\tejas>
```

This will create HTTP/HTTPS, DNS, and ICMP traffic.

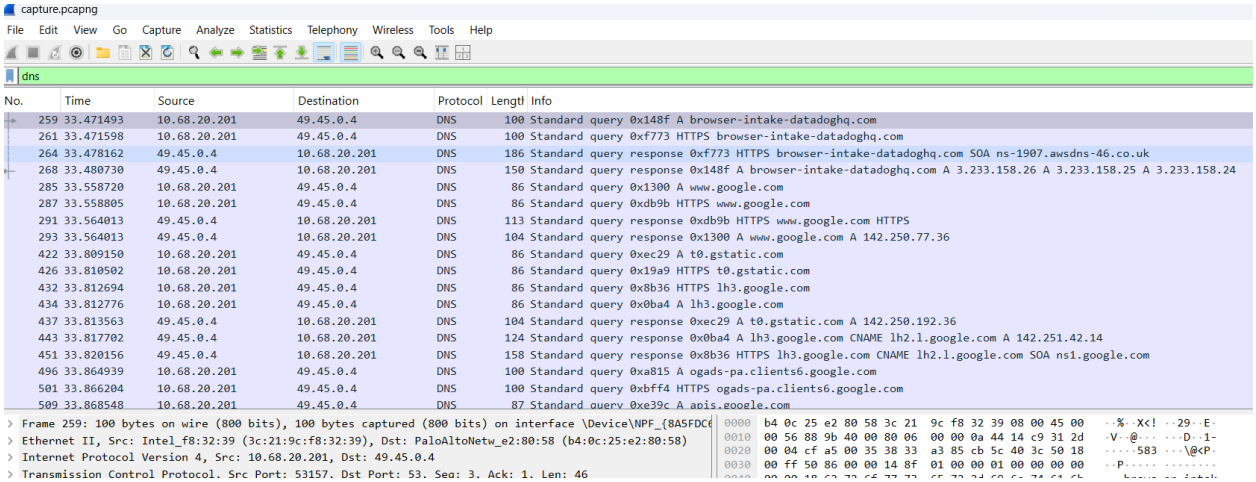
After a minute, click the red square (stop) button on the left of top toolbar.



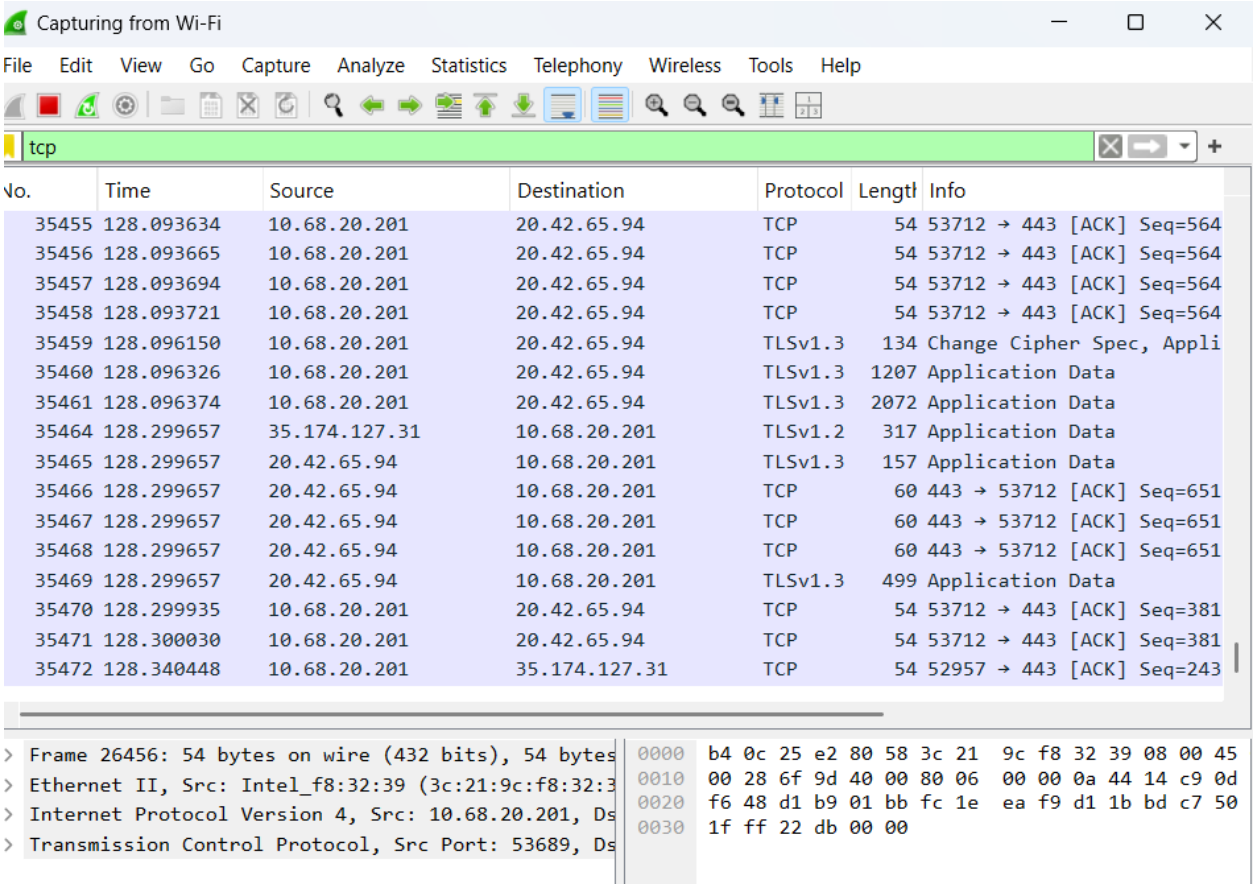
Use the **filter bar** to isolate specific protocols:

Protocol	Filter
DNS	dns
TCP	tcp
ICMP	icmp
ARP	arp

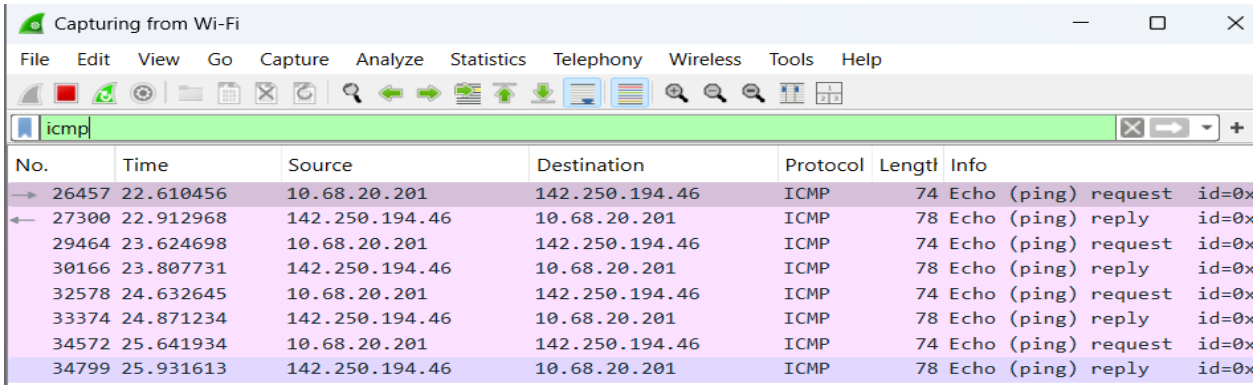
DNS(domain name system): name resolution (e.g., A or AAAA record queries).



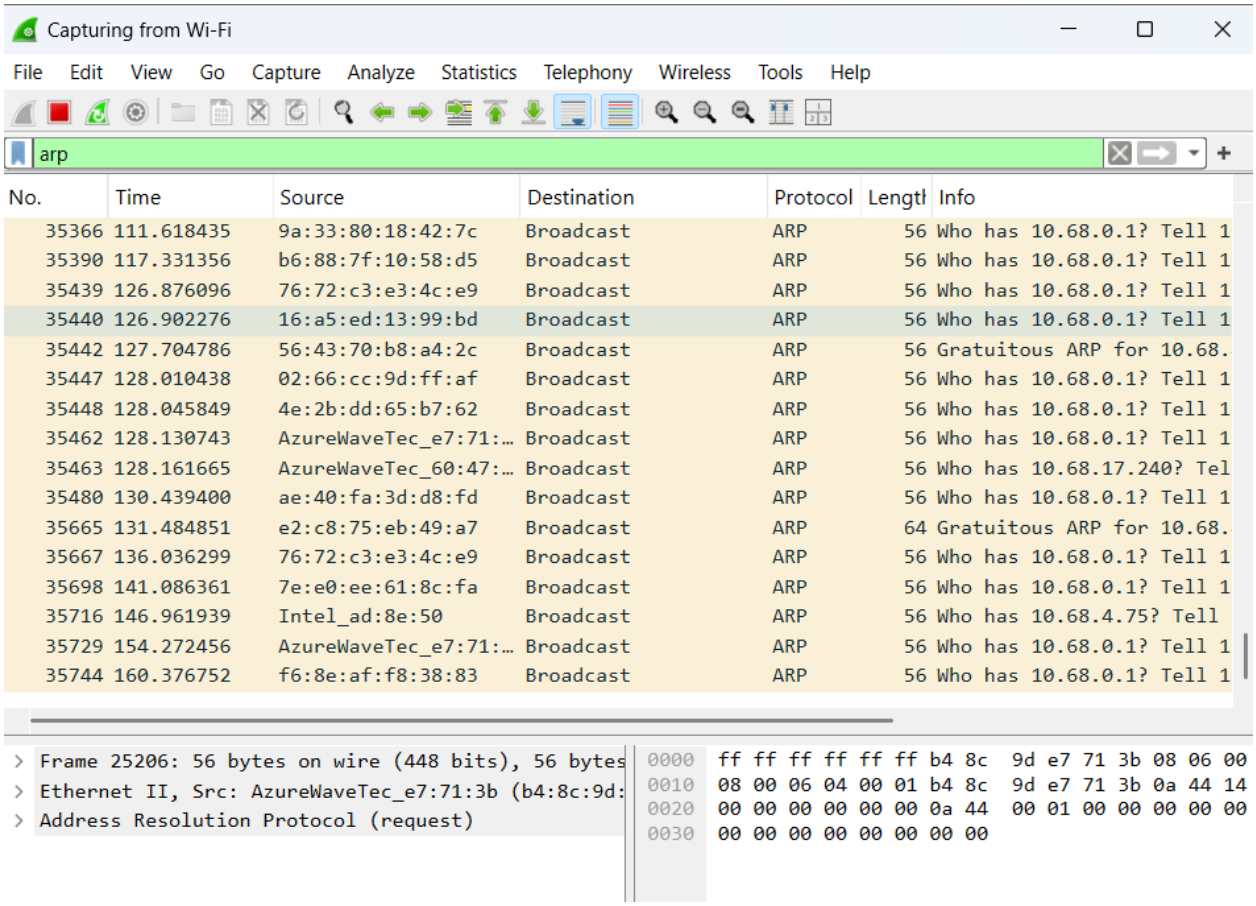
TCP(Transfer Control Protocol): used as transport layer for HTTP.



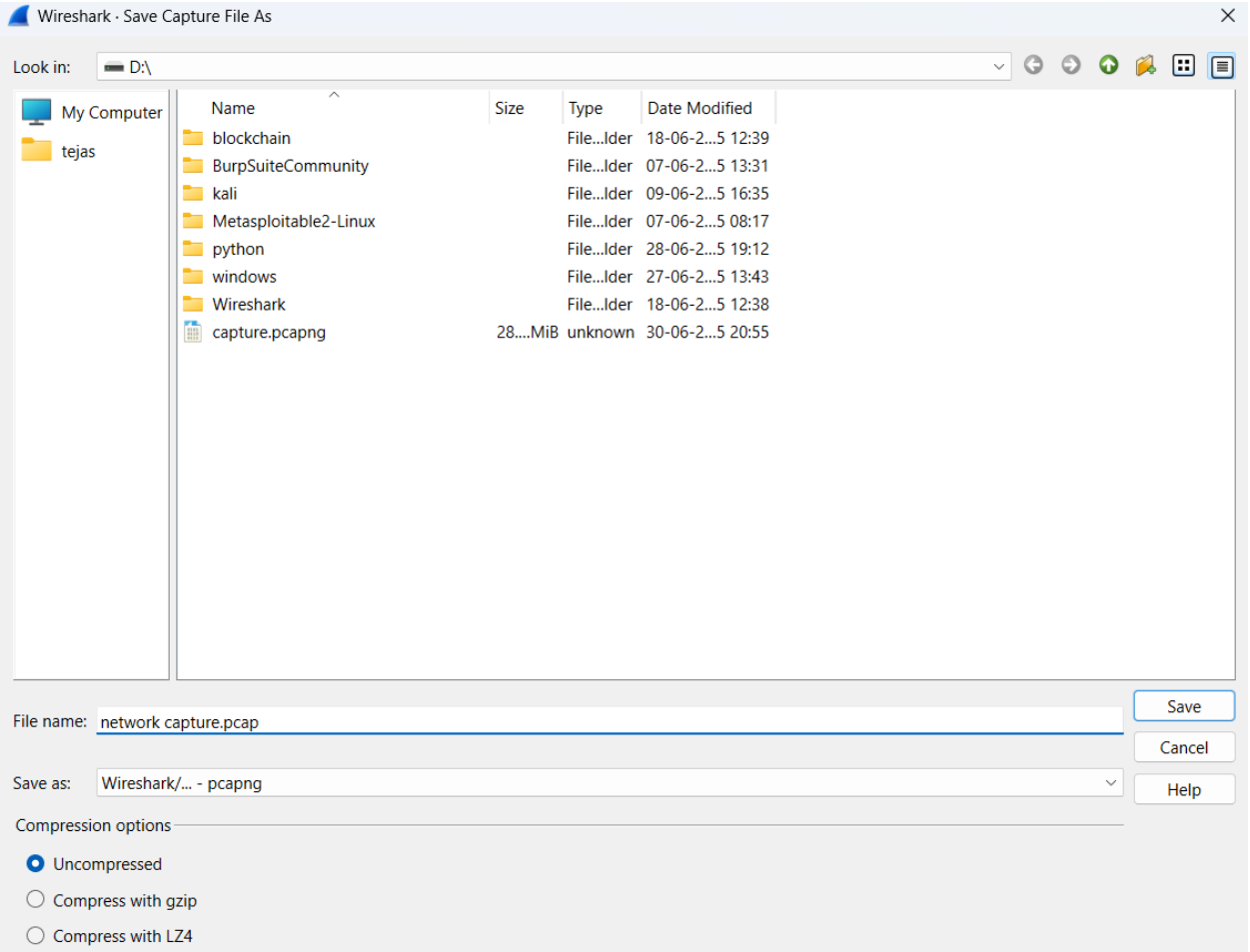
ICMP(Internet Control Message Protocol): ping request and reply.



ARP(Address Resolution Protocol): address resolution between MAC and IP.



Got to **File>Save as** and Name your file: **network capture.pcap** and click **save**(makesure the file type is **.pcap**).



Conclusion:

This task provide valuable hands-on experience in capturing and analyzing live network traffic using Wireshark. By monitoring real-time packet flow, I was able to identify and filter multiple network protocols such as DNS, TCP, ICMP, and ARP. The exercise enhanced my understanding of how data is transmitted over a network, how different protocols function, and how packet-level analysis can reveal important details about system communication.