

Wireshark Network Traffic Analysis

DNS, ICMP, TCP and HTTP Investigation

Objective

The objective of this lab was to analyze captured network traffic using Wireshark to identify DNS resolution, ICMP activity, TCP session establishment, and HTTP communication.

The analysis focused on tracking traffic between an internal host (172.21.224.2) and an external IP address (142.250.1.139).

This exercise demonstrates practical packet-level inspection used in network security monitoring and SOC analysis.

Environment

Tool Used: Wireshark

Capture File: sample.pcap

Protocols Observed:

ICMP

DNS

TCP

HTTP

Display Filters Used:

```
udp.port == 53  
tcp  
ip.src == 142.250.1.139  
eth.addr == 42:01:ac:15:e0:02
```

ICMP Traffic Analysis

Observed ICMP Echo (ping) replies from:

Source IP: 142.250.1.139
Destination IP: 172.21.224.2

Packet details included:

ICMP type: Echo reply
Sequence numbers
TTL values
Payload bytes

ICMP traffic confirms basic connectivity between the internal system and the external host.

Security relevance:

ICMP can be used for reconnaissance and network mapping. Monitoring unusual ICMP activity helps detect scanning or lateral movement attempts.

DNS Resolution Analysis

Filter applied:

udp.port == 53

Observed DNS query:

Query: opensource.google.com
Record Type: A
Class: IN

Observed DNS response:

Multiple A records returned including:

142.250.1.139
142.250.1.138
142.250.1.102

Key fields reviewed:

Transaction ID
Flags (Standard query)

Questions count

Answer count

Security relevance:

DNS monitoring is critical because malware often uses DNS for command-and-control communication or data exfiltration. Observing normal resolution behavior builds baseline knowledge.

TCP Three-Way Handshake Analysis

Observed TCP session between:

Internal Host: 172.21.224.2

External Host: 142.250.1.139

Destination Port: 80 (HTTP)

Handshake packets observed:

SYN

SYN-ACK

ACK

Key fields inspected:

Source port: 49652

Destination port: 80

Sequence number

Acknowledgment number

Window size

TCP flags

This confirms successful TCP session establishment.

Security relevance:

Understanding TCP handshake behavior allows detection of:

Port scans

Incomplete connections

SYN flood patterns

Suspicious outbound connections

HTTP Traffic Analysis

Observed HTTP request:

GET / HTTP/1.1

Observed HTTP response:

HTTP/1.1 301 Moved Permanently

This indicates the server redirected the request, which is common for web services.

Security relevance:

HTTP inspection is important for:

- Identifying malicious downloads
- Detecting unusual outbound connections
- Analyzing web-based attacks

Packet Header Breakdown

For selected packets, the following layers were reviewed:

- Ethernet II Header
- Source MAC address
- Destination MAC address

- IPv4 Header
- Source IP
- Destination IP
- TTL
- Header length

- TCP Header
- Flags
- Window size
- Sequence numbers
- Options (MSS, SACK permitted, timestamps, window scale)

Understanding packet structure at each layer is critical for deep network analysis.

Key Observations

Internal host 172.21.224.2 resolved opensource.google.com
DNS returned multiple A records
Internal host initiated TCP connection to 142.250.1.139 on port 80
Successful three-way handshake completed
HTTP GET request observed
Server responded with 301 redirect

This indicates normal web browsing behavior.

Skills Demonstrated

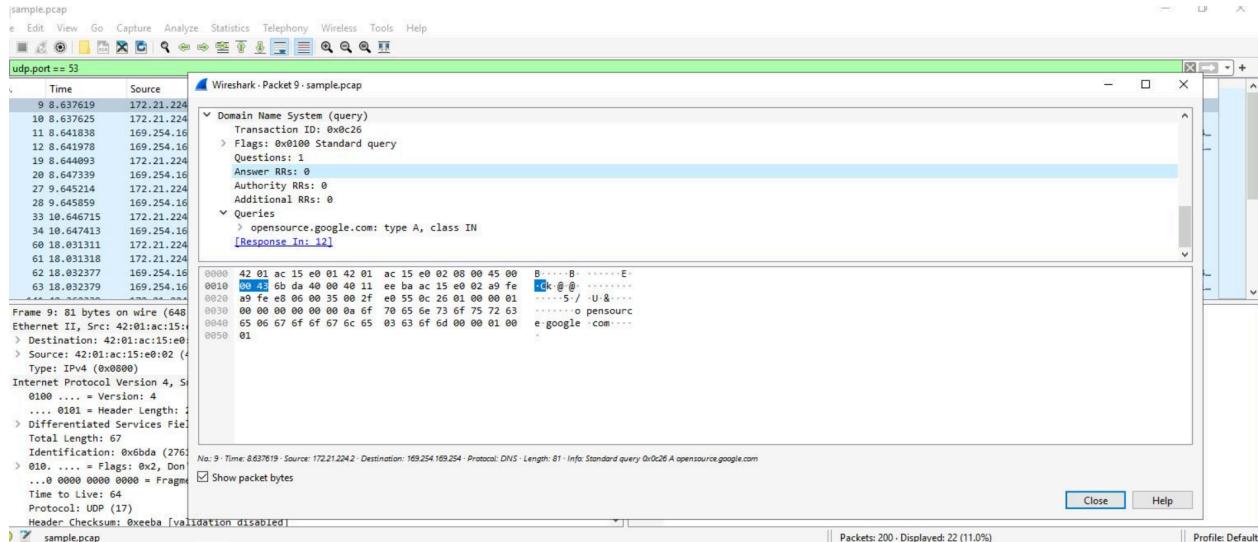
Network traffic filtering
Protocol analysis
DNS resolution tracking
TCP handshake interpretation
ICMP packet inspection
HTTP request analysis
Packet header dissection
Security relevance interpretation

Conclusion

This lab demonstrates hands on network traffic analysis using Wireshark. By filtering and inspecting DNS, ICMP, TCP, and HTTP traffic, the exercise replicates SOC monitoring tasks. The analysis shows how to trace domain resolution to IP address, verify TCP session establishment, and interpret HTTP responses at the packet level.

Appendix – Supporting Evidence

Complete traffic flow showing DNS resolution, TCP three-way handshake, HTTP GET request, and HTTP 301 response between internal host 172.21.224.2 and external server 142.250.1.139.

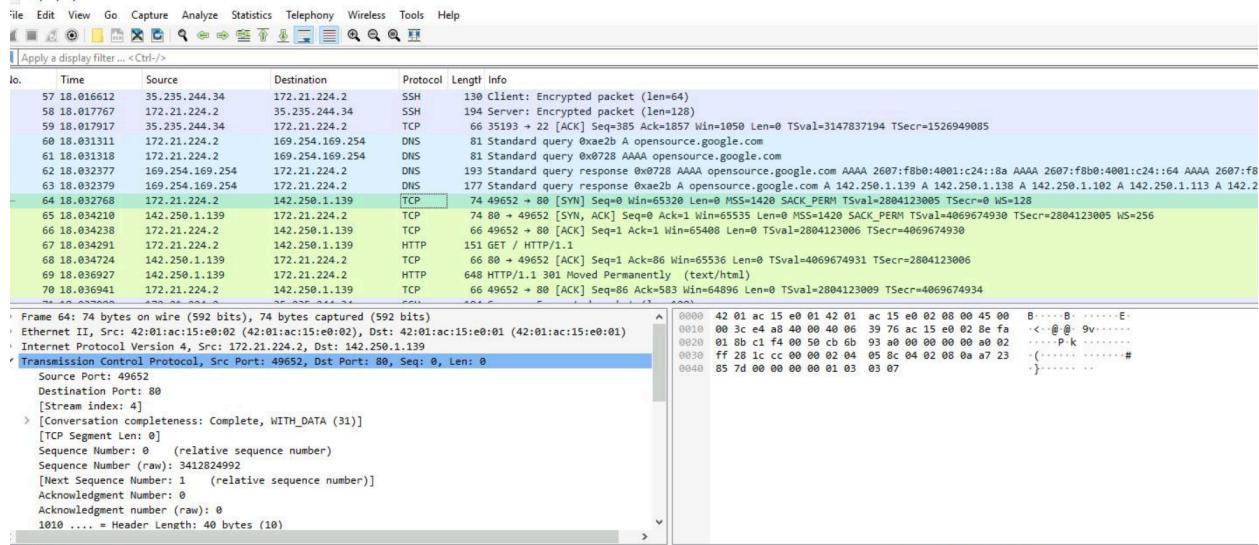


ICMP echo replies observed from external IP 142.250.1.139 confirming network connectivity with internal host 172.21.224.2.

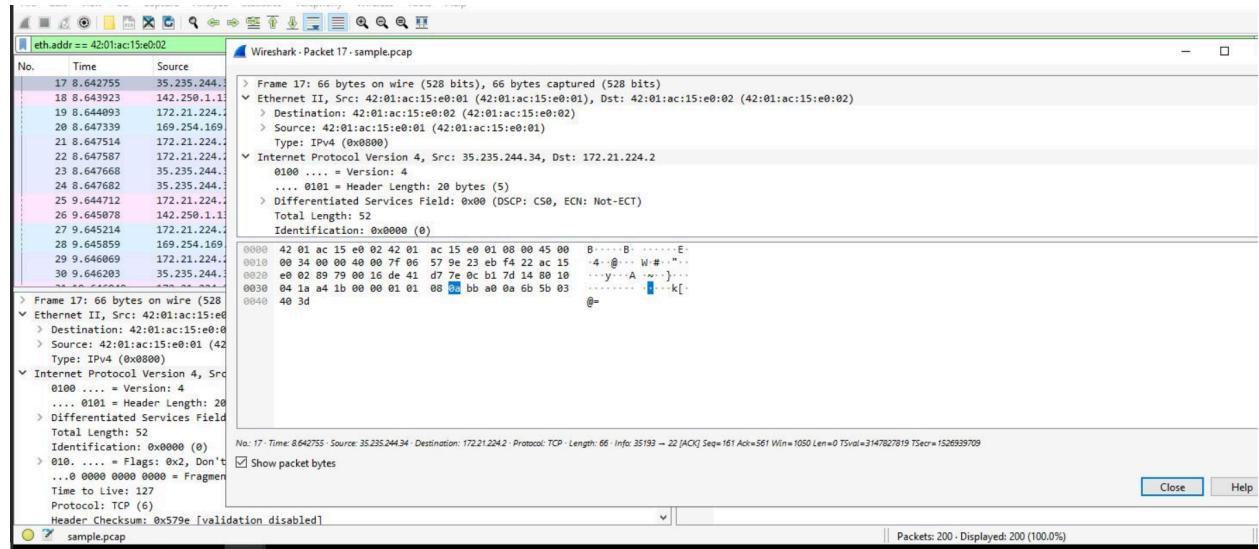
No.	Time	Source	Destination	Protocol	Length	Info
18	8.643923	142.250.1.139	172.21.224.2	ICMP	98	Echo (ping) reply id=0x6831, seq=1/256, ttl=115 (request in 16)
26	9.645078	142.250.1.139	172.21.224.2	ICMP	98	Echo (ping) reply id=0x6831, seq=2/512, ttl=115 (request in 25)
32	10.646563	142.250.1.139	172.21.224.2	ICMP	98	Echo (ping) reply id=0x6831, seq=3/768, ttl=115 (request in 31)
65	18.034210	142.250.1.139	172.21.224.2	TCP	74	80 -> 49652 [SYN, ACK] Seq=0 Ack=1 Win=65536 Len=0 MSS=1420 SACK_PERM TSeq=4069674930 TSecr=2804123005 WS=256
68	18.034724	142.250.1.139	172.21.224.2	TCP	66	80 -> 49652 [ACK] Seq=1 Ack=86 Win=65536 Len=0 TSeq=4069674931 TSecr=2804123006
69	18.036927	142.250.1.139	172.21.224.2	HTTP	648	HTTP/1.1 301 Moved Permanently (text/html)
82	18.037927	142.250.1.139	172.21.224.2	TCP	66	80 -> 49652 [FIN, ACK] Seq=583 Ack=87 Win=65536 Len=0 TSeq=4069674933 TSecr=2804123009

> Frame 32: 98 bytes on wire (784 bits), 98 bytes captured (784 bits) > Ethernet II, Src: 42:01:ac:15:e0:01 (42:01:ac:15:e0:01), Dst: 42:01:ac:15:e0:02 (42:01:ac:15:e0:02) > Internet Protocol Version 4, Src: 142.250.1.139, Dst: 172.21.224.2 > Internet Control Message Protocol	0000 42 01 ac 15 e0 02 42 01 ac 15 e0 01 08 00 45 00 B-----E- 0010 00 54 00 00 00 73 01 2b 0c 8e fa 01 8b ac 15 T-----+----- 0020 e0 02 00 00 3a ef 68 31 00 03 43 14 7e 63 00 08 -----h1 :C<w> 0030 00 00 d9 91 03 00 00 00 00 00 18 11 12 13 14 15 ----- .. !#% 0040 16 17 18 19 1a 1b 1c 1d 1e 1f 20 21 22 23 24 25 ----- 0050 26 27 28 29 2a 2b 2c 2d 2e 2f 30 31 32 33 34 35 8'(*+,- ./012345 0060 36 37 67
---	--

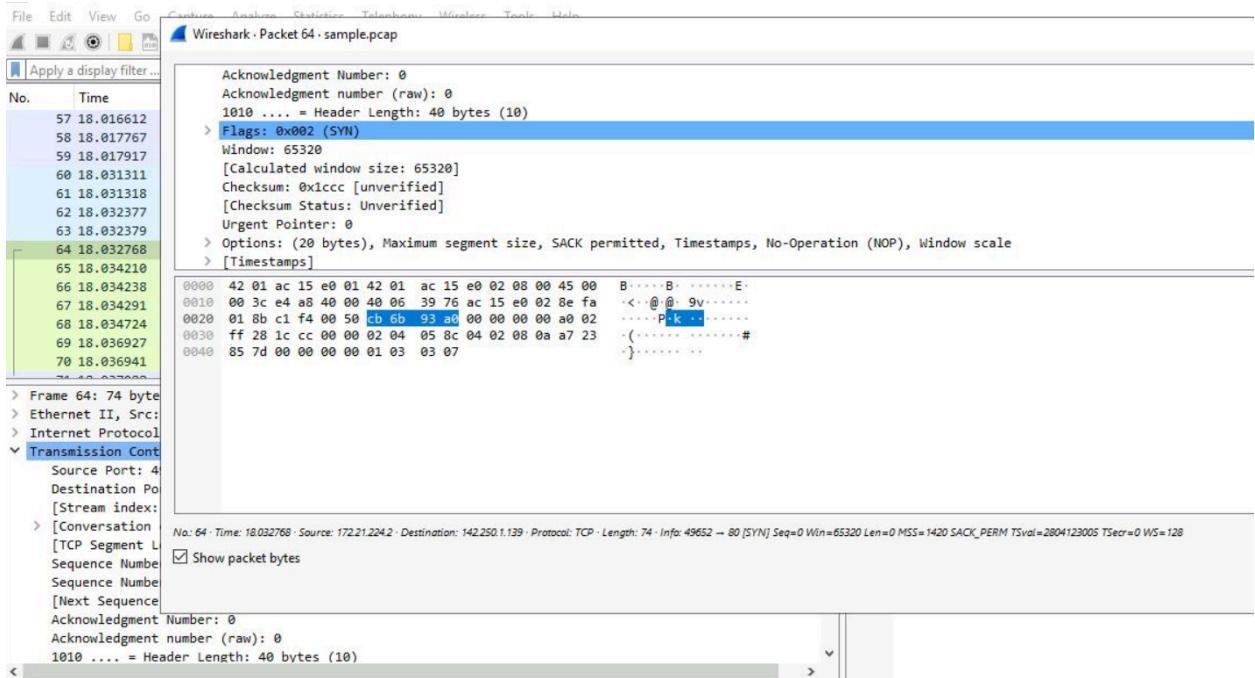
DNS standard query for opensource.google.com (Type A, Class IN) showing domain resolution process prior to TCP connection establishment.



TCP session establishment showing SYN, SYN-ACK, and ACK packets between source port 49652 and destination port 80.



Detailed TCP header analysis including sequence numbers, window size, TCP flags (SYN), and options such as MSS and SACK permitted.



Prepared by: Sambou Kamissoko

LinkedIn: <https://www.linkedin.com/in/sambouk/>