USF Muma
College of Business
UNIVERSITY of SOUTH FLORIDA

OSI ANALYSIS

Professor

Dr. Clinton Daniel, DBA

By Noor Ahamed Vempalle

1. Analysis of 'infiniband.cap'

('infiniband.cap' downloaded from https://wiki.wireshark.org/SampleCaptures)

About InfiniBand:

A strong new architecture called InfiniBand was created to facilitate I/O connectivity for the Internet facilities. All of the main OEM server vendors support InfiniBand as a way to go beyond and develop the next generation of I/O interconnect standard in servers.

Protocol Hierarchy Statistics:

Protocol	Percent Packets	Packets	Percent Bytes	Bytes	Bits/s	End Packets	End Bytes	End Bits/s	PDUs
Frame	100.0	43	100.0	7494	239	0	0	0	43
Extensible Record Format	100.0	43	100.0	7494	239	0	0	0	43
InfiniBand	100.0	43	100.0	7494	239	26	5200	166	43
Internet Protocol Version 4	23.3	10	2.7	200	6	0	0	0	10
Internet Control Message Protocol	14.0	6	5.1	384	12	6	384	12	6
User Datagram Protocol	9.3	4	0.4	32	1	0	0	0	4
Data	9.3	4	3.6	272	8	4	272	8	4
Internet Protocol Version 6	11.6	5	2.7	200	6	0	0	0	5
Internet Control Message Protocol v6	11.6	5	3.8	288	9	5	288	9	5
Address Resolution Protocol	4.7	2	1.5	112	3	2	112	3	2

The above statistics display a network traffic analysis report with various protocols, packets, and bytes information. The summary shows that InfiniBand has the highest percentage of packets, while IPv4 and Extensible Record Format follow closely behind. The "End Packets" shows that InfiniBand has 26 packets, while the other protocols have 0 packets.

The packet capture displays multiple protocols:

- **InfiniBand:** This protocol is utilized in high-performance computing and computer clusters. InfiniBand is a high-performance, low-latency network architecture that supports multiple protocols. It is often used for supercomputing and cloud computing applications.
- **UDP (User Datagram Protocol):** Datagrams can be sent over a network using the connectionless protocol. For applications like DNS or streaming media, where dependability is not a top priority, it is usually utilized.
- **ARP (Address Resolution Protocol):** An IP address can be mapped to a MAC address, or physical machine address, that is recognized by the local network by using this protocol. ARP is used in the capture for address resolution and network discovery.
- **ICMP (Internet Control Message Protocol):** It has features like error reporting, traceroute, and ping (echo request and reply). To check connectivity and troubleshoot network-related problems, ICMP packets are utilized.
- **ICMPv6:** This version of ICMP was created specifically for IPv6. It is customized for the IPv6 protocol and fulfills functions like those of ICMP.

Comprehensive analysis of all packets in the file:

Packets 1-2, 12-13, 41-42: InfiniBand communication—more especially, UD (Unreliable Datagram) Send Only messages is involved in these packets. The message types are '**SubnGet'** and '**SubnGetResp'**, which most likely have to do with retrieving subnet management information.

Packets 3-4, 24-25: These packets, which range from 10.0.0.36 to 10.0.0.255, are UDP messages that give information communication within the same subnet. 49501 is the source port, and 49504 and 49501 are the destination ports.

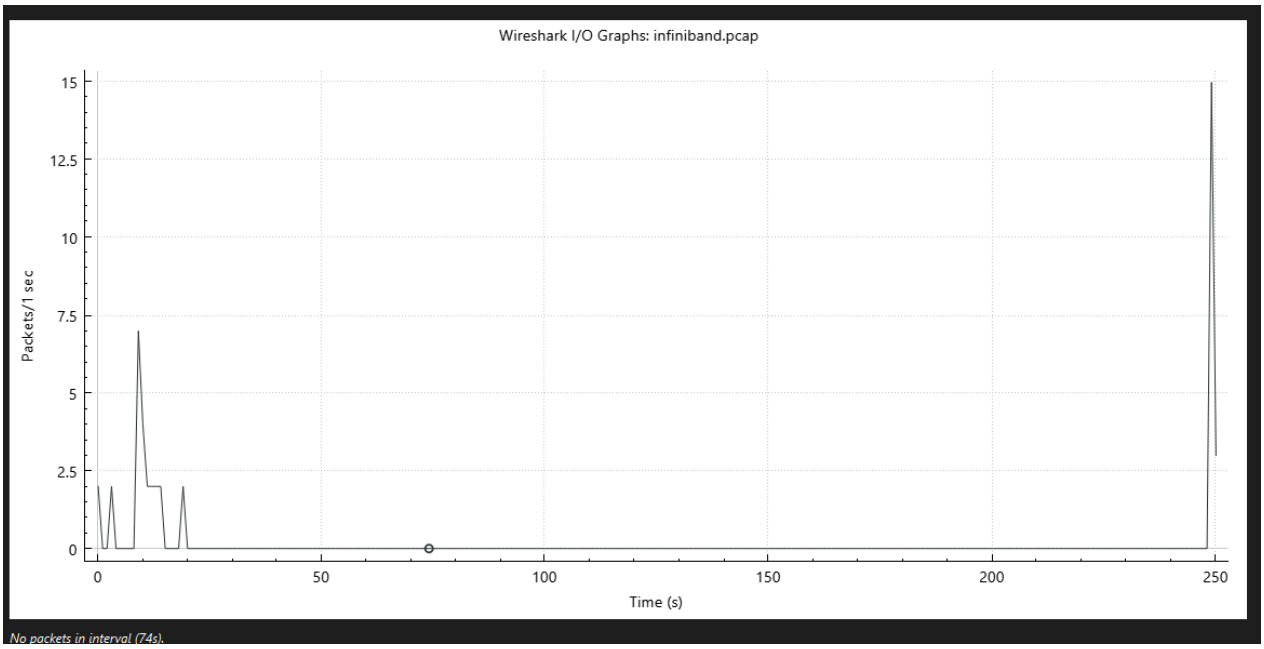
Packets 5-6: These packets involve ARP (Address Resolution Protocol) messages. Packet 5 is an ARP request from IP 10.0.1.34 asking for the MAC address corresponding to 10.0.0.58. Packet 6 is the ARP response providing the MAC address.

Packets 7-9: These packets deal with Connection Management (CM)-related InfiniBand communication. The '**ConnectRequest'** and '**ConnectReply'** messages, which indicate the establishment of a connection, are contained in packets 7 and 8, respectively. Packet 9 is a **ReadyToUse** message indicating that the connection is ready for use.

Packets 10-11, 14-21: Internet Control Message Protocol (ICMP) packets, specifically echo requests (ping) from 10.0.1.34 to 10.0.0.58, are contained in these packets. There is no corresponding echo reply packets, suggesting that the destination is not responding to the ping requests.

Packets 26-31, 36-39, 43: Internet Control Message Protocol version 6 (ICMPv6) communication is involved in these packets. Packets 26-30 are Neighbor Solicitation, Neighbor Advertisement, and echo request-reply messages between IPv6 addresses **fe80::202:c902:24:f636** and **fe80::202:c903:0:1895.**

Wireshark I/O Graph:



- The I/O Graph displaying packet data over time. The graph indicates a spike in packets at the beginning, which then drops sharply. This could mean that there was a burst of data sent or received, followed by a slowdown or a loss of connection.

Packet Lengths:

Topic / Item	Count	Average	Min Val	Max Val	Rate (ms)	Percent	Burst Rate	Burst Start
Packet Lengths	43	174.28	30	290	0.0002	100%	0.1500	249.574
0-19	0	-	-	-	0.0000	0.00%	-	-
20-39	9	30.00	30	30	0.0000	20.93%	0.0300	249.575
40-79	0	-	-	-	0.0000	0.00%	-	-
80-159	12	119.33	94	134	0.0000	27.91%	0.0300	9.180
160-319	22	263.27	166	290	0.0001	51.16%	0.0900	249.574
320-639	0	-	-	-	0.0000	0.00%	-	-
640-1279	0	-	-	-	0.0000	0.00%	-	-
1280-2559	0	-	-	-	0.0000	0.00%	-	-
2560-5119	0	-	-	-	0.0000	0.00%	-	-
5120 and greater	0	-	-	-	0.0000	0.00%	-	-

Conclusion:

Overall, network administrators are better able to comprehend and manage their network infrastructure thanks to the analysis of the packet capture, which offers insightful information about protocol usage, network connectivity, and potential security risks. To resolve any issues found and enhance network security and performance, more research might be necessary.

2. Analysis of 'Ipv4_cipso_option.pcap''

('Ipv4_cipso_option.pcap' **downloaded from https://wiki.wireshark.org/SampleCaptures)**

- CIPSO is used in IPv4 networks—which frequently occur in settings require severe security measures, such as government or military networks—to enforce security standards and provide integrity protection.

Protocol	Percent Packets	Packets	Percent Bytes	Bytes	Bits/s	End Packets	End Bytes	End Bits/s	PDUs
Frame	100.0	6	100.0	764	168	0	0	0	6
Ethernet	100.0	6	11.0	84	18	0	0	0	6
Internet Protocol Version 4	100.0	6	38.7	296	65	0	0	0	6
Internet Control Message Protocol	100.0	6	50.3	384	84	6	384	84	6

The above Hierarchy statistics shows that we have one protocol (Ipv4) observed in the packets.

Ethernet:

It is a **Data Link Layer**. This protocol governs the structure of data packets at the lowest level of the network stack (OSI layer 1). It outlines the arrangement of information within frames, encompassing details such as source and destination MAC addresses, frame type, and the actual data payload.

IPv4:

At the network layer (OSI layer 3), IPv4 manages the transmission of datagrams between networks. It

assigns logical addresses (IP addresses) and dictates the structure of the datagram, encompassing elements like source and destination IP addresses, header details, and the actual data payload.

ICMP (Internet Control Message Protocol) protocol:

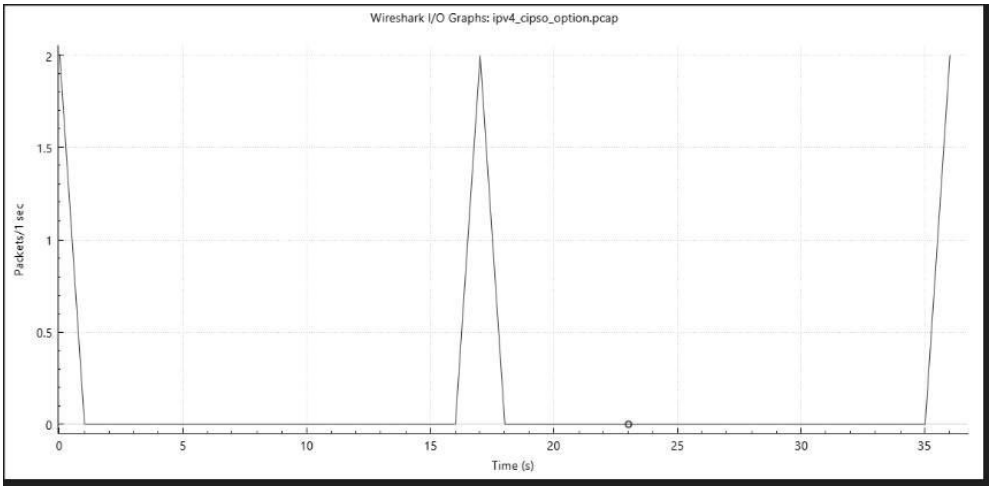
ICMP is commonly used for diagnostic and control purposes in IP networks, including functions like ping (echo request and reply), traceroute, and error reporting. To avoid network routing infinite loops, ICMP packets normally include a TTL (Time-to-Live) field.

Comprehensive Packet Analysis:

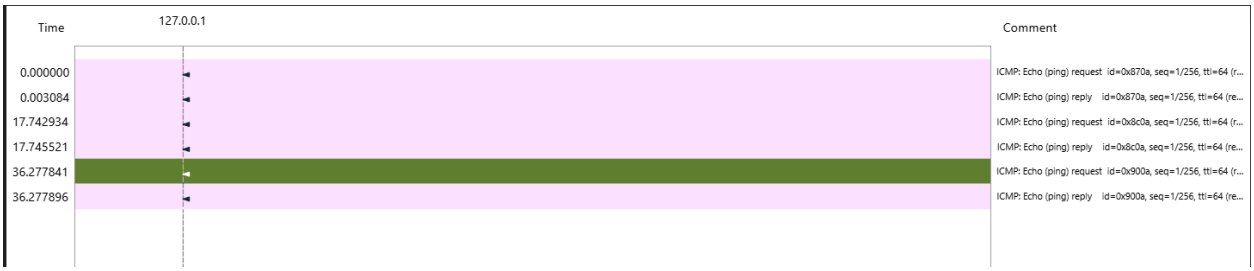
- Here, the loopback communication is likely part of internal network testing or diagnostic procedures within the host's operating system. The identical source and destination IP addresses are used for the packet exchange between the localhost (127.0.0.1). The TTL (Time to Live) value is set to 64, indicating that the packets are meant to stay within the local network.
- Each echo request is followed by a corresponding echo reply with the same id and sequence number. This sequence of requests and replies suggests that the host is sending ICMP ping messages to itself and receiving responses accordingly.
- The consistent TTL (Time-to-Live) value of 64 and the lack of external IP addresses point to local communication within the same system. The rapid round-trip timings and sequential sequence numbers point to effective network stack operation.
- The data shows that the local machine is able to send and receive ping requests and replies correctly, with a consistent TTL value and increasing time delay between each request and reply.

Wireshark I/O Graph:

It can be concluded that the network traffic is relatively stable, with a consistent number of packets per second. The graph shows a steady increase in the number of packets per second, with a slight decrease in the last, this decrease could be due to a temporary drop in network traffic or a change in the traffic patterns.



Flowgraph:



The flowgraph depicts ICMP Echo (ping) requests and replies between a local host (IP address 127.0.0.1). This is represented by the arrow mark.

total bytes sent over time:

Image generated from https://lab.dynamite.ai/

- Both graphs display the trend of data transmission over time, with specific timestamps marked along the x-axis related to network communication or data transfer. These graph shows a peak in bytes sent around January 18, 2007, and the data points indicate the amount of bytes sent from the source to the destination and vice versa.

Conclusion:

Overall, the network administrators and security professionals can gain valuable insights into the usage of CIPSO options within IPv4 packets and ensure the security and integrity of the network infrastructure. We can analyze the overall network traffic to identify patterns, anomalies, or potential security threats and look for any unusual or unexpected behavior within the captured packets that may need further investigation to identify potential threats.

3. Analysis of 'DTMFsipinfo.pcap'

('DTMFsipinfo.pcap' **downloaded from https://wiki.wireshark.org/SampleCaptures)**

Protocol Hierarchy Statistics:

Protocol	Percent Packets	Packets	Percent Bytes	Bytes	Bits/s	End Packets	End Bytes	End Bits/s
Frame	100.0	32	100.0	24893	2537	0	0	0
Ethernet	100.0	32	1.8	448	45	0	0	0
PPP-over-Ethernet Session	100.0	32	0.8	192	19	0	0	0
Point-to-Point Protocol	100.0	32	97.4	24253	2472	0	0	0
Internet Protocol Version 4	100.0	32	2.6	640	65	0	0	0
User Datagram Protocol	100.0	32	1.0	256	26	0	0	0
Session Initiation Protocol	100.0	32	93.6	23293	2374	32	23293	2374

The sample capture protocols are as follows:

- **Ethernet:**
  For local area networks (LANs), Ethernet is a commonly used networking technology. For wired network communication, it specifies the physical and data link layers.
- **PPP-over-Ethernet Session:**
  When two network nodes want to connect directly over Ethernet, they can use the PPP (Point-to-Point Protocol) over Ethernet protocol. PPP traffic can be transmitted over Ethernet networks thanks to its encapsulation of PPP frames within Ethernet frames.
- **Point-to-Point Protocol (PPP):**
  PPP is a data link layer protocol that connects two nodes directly over a variety of physical media, such as Ethernet. It offers a technique for creating a point-to-point connection for data transmission and encapsulating network layer protocols (like IPv4).
- **Internet Protocol Version 4 (IPv4):**
  IPv4 is the fourth version of the Internet Protocol, responsible for addressing and routing packets across networks. It provides logical addressing (IPv4 addresses) to network devices and defines the structure of data packets, including source and destination IP addresses.
- **User Datagram Protocol (UDP):**
  Data packets are sent over IP networks using UDP, a transport layer protocol.
- **Session Initiation Protocol (SIP):**
  SIP is a signaling protocol that is used over IP networks to start, stop, and modify real-time communication sessions, including voice and video calls. It outlines the protocols and messages needed to configure and oversee multimedia sessions amongst endpoints.

Network Graph:

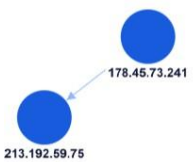


Image generated from https://lab.dynamite.ai/

Comprehensive analysis of all packets in the file:

Overall, this **pcap** file captures the exchange of SIP messages between two endpoints for the initiation, cancellation, and re-negotiation of sessions. Each SIP request is followed by a corresponding response from the receiving endpoint, ensuring proper protocol adherence and session establishment.

1. Session Establishment:

- Packets1-4,5-10,11-15,21-24, and 25-28: These packets show how SIP sessions are started and established using INVITE requests and the replies that follow (status codes 100, 200). To start a conversation, SIP INVITE messages are sent from one endpoint (178.45.73.241) to another (213.192.59.75).

2. Cancellation:

These packets demonstrate the use of CANCEL requests and matching responses (status code 200 OK) to end SIP sessions. To end ongoing SIP sessions that have not yet received a response, CANCEL requests are sent.

3. Session Info:

Packets 17–20, 29–32: In these packets, the endpoints exchange information requests and responses (status code 200 OK). INFO requests are usually used in sessions to transmit non-SIP related information.

4. Session Re-negotiation:

Packets 21–24, 25–28: These packets show the endpoints resuming their SIP sessions (INVITE requests).

5. Status Codes:

- The server has received the INVITE request and is processing it, as indicated by the status code 100 (trying). The request was successful, reflected by the status code 200 (OK).

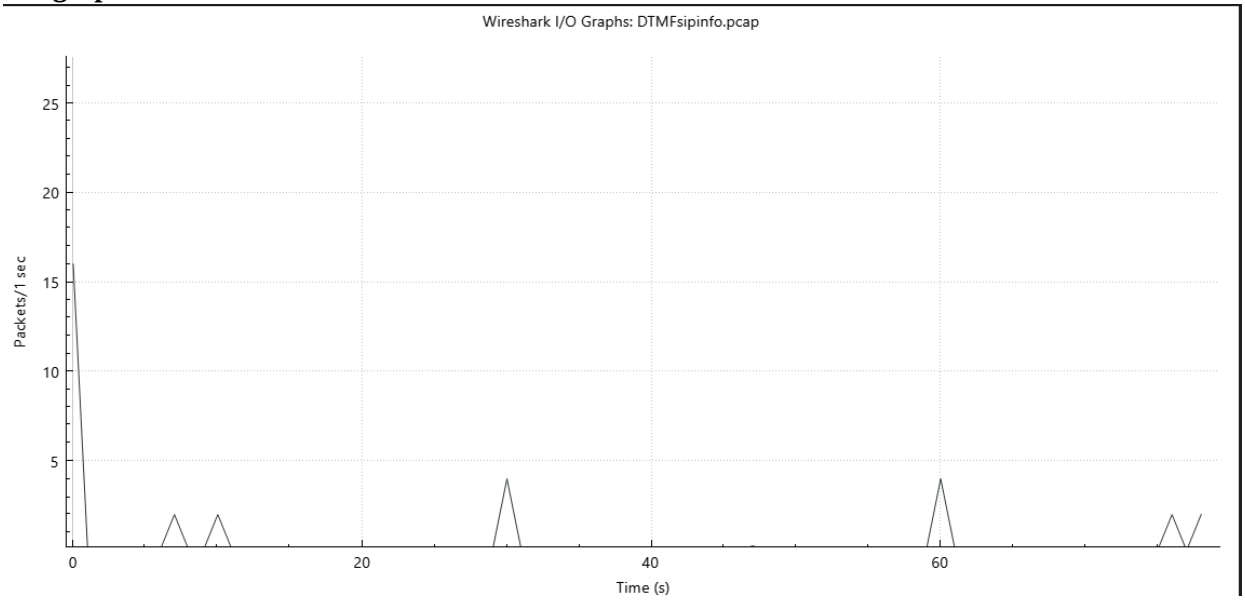
6. Media Negotiation:

- The presence of SDP (Session Description Protocol) in some packets indicates negotiation of media parameters, such as codec types and IP addresses for media transmission.

Flowgraph (Wireshark):

The sample flowgraph shows the request and responses between the two IP addresses (17.64.75.3.241 and 213.192.59.75).

I/O graph:



bytes sent over time:

- The blue line represents bytes sent from the source to the destination and the orange line represents bytes sent from the destination back to the source.
- The two peaks suggest a burst of data transmission from the source to the destination and destination to source during specific timestamp. This indicates bidirectional communication, with data flowing both ways and the simultaneous peaks in both directions suggest an active data exchange.

.

Image generated from https://lab.dynamite.ai/

Conclusion:

These packet captures can be used by developers to verify and test VoIP systems' functionality, especially in relation to DTMF signaling. It aids in making sure the VoIP system interprets DTMF tones correctly and reacts to user input in the right way. Various companies can learn more about their preferences and behavior of their customers during phone conversations by analyzing data from DTMF signaling.