

# IPPM Considerations for the IPv6 PDM Extension Header

## IETF 89 - London

Nalini Elkins: Inside Products, Inc.

Mike Ackermann: Blue Cross Blue Shield of Michigan

(Team : Keven Haining: US Bank, Bill Jouris: Inside  
Products)

# Metrics / Fields Needed

- Packet sequence number
  - Speeds diagnostics
  - IPv4 IP ID used as de facto sequence number
- End-to-end response time WITHOUT agents
  - Metrics needed for quick triage:
    - One way delay
    - Server delay
    - Two way delay

# We propose:

## Requirement

- In basic IPv6 transport
- Unmolested by middle systems

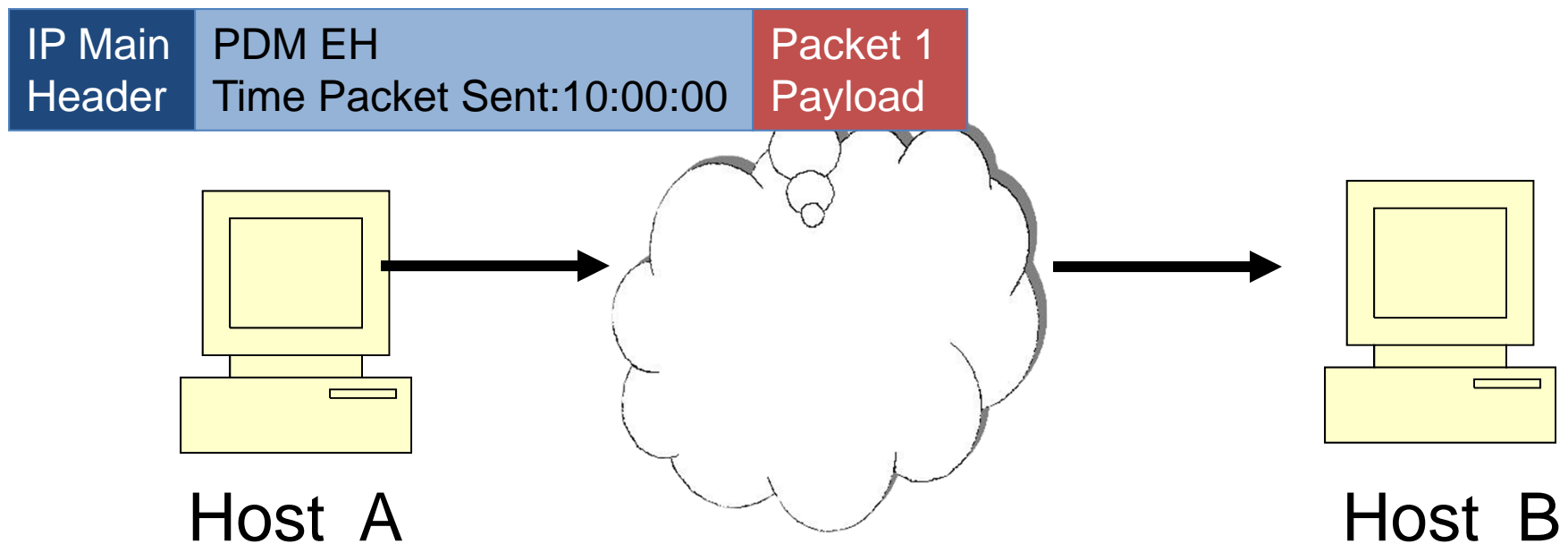
## Solution (IPv6)

- **Implementation** of existing extension header: Destination Options Header (DOH)
- Performance and Diagnostic Metrics (PDM) DOH
- PDM 1: Requires time synchronization
- PDM 2: No time synchronization

# Response Time Measurement

## Step 1

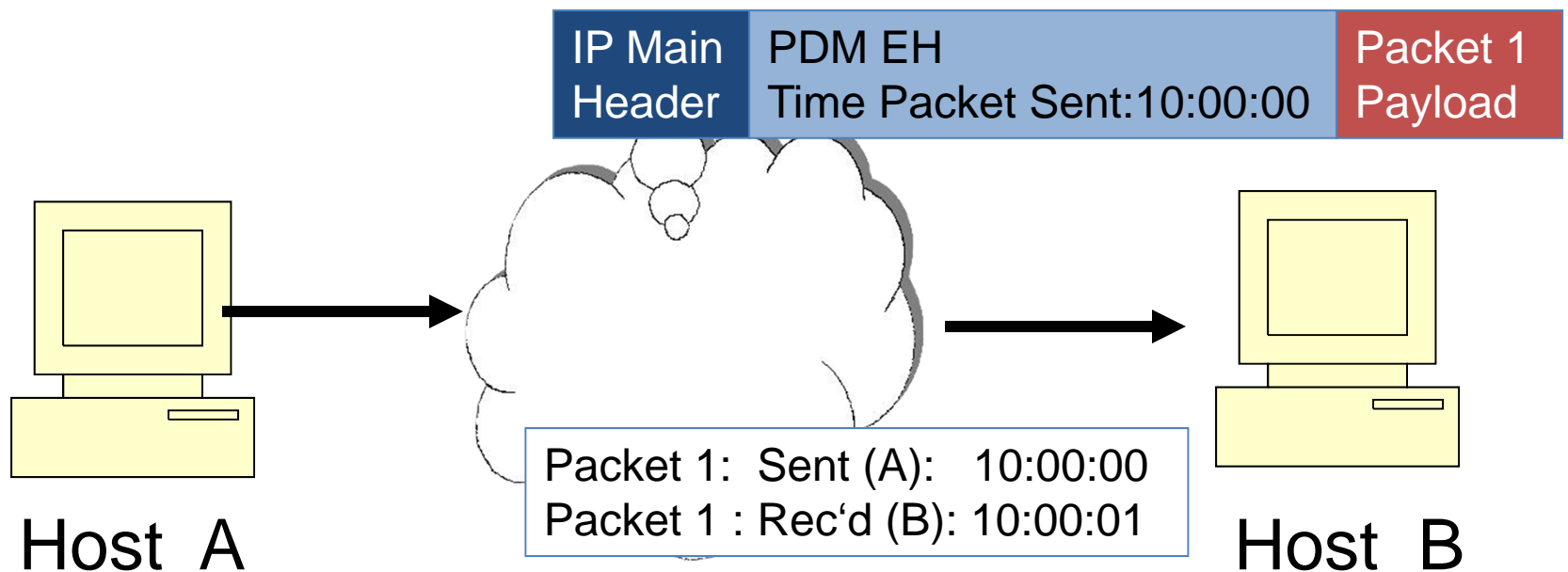
- Packet 1 sent from source host A
- Time-stamped leaving Host A
- Timestamp is in PDM extension header



# Response Time Measurement

## Step 2

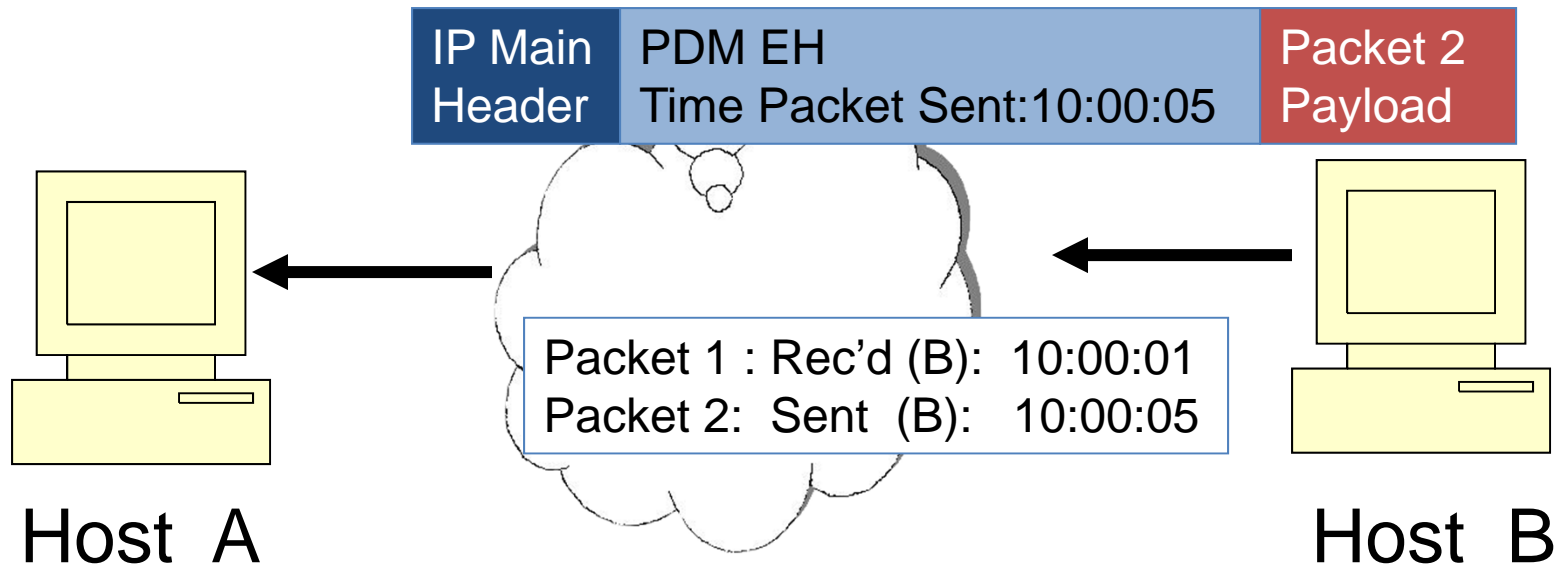
- Packet 1 received at Host B
- Time-stamped leaving Host A
- Inbound network time = Packet 1 rec'd (B) – Packet 1 sent (A)



# Response Time Measurement

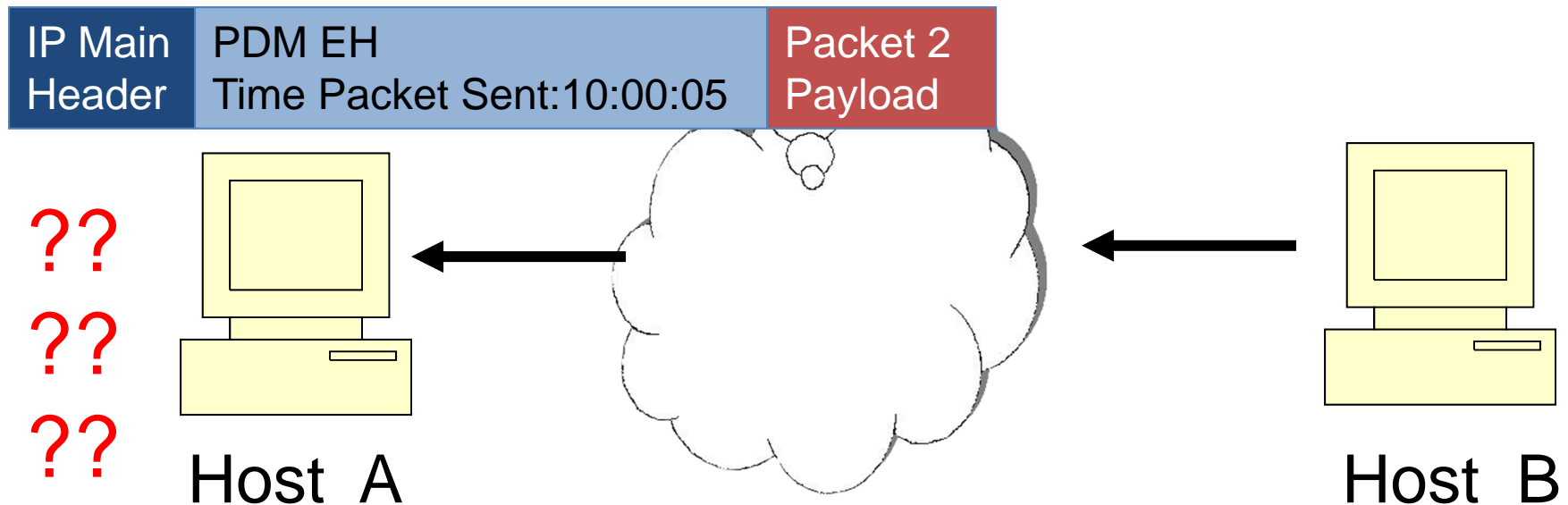
## Step 3

- Packet 2 sent from Host B (response to Packet 1)
- Time-stamped leaving Host B
- Processing Time = Packet 2 sent (B) - Packet 1 rec'd (B)



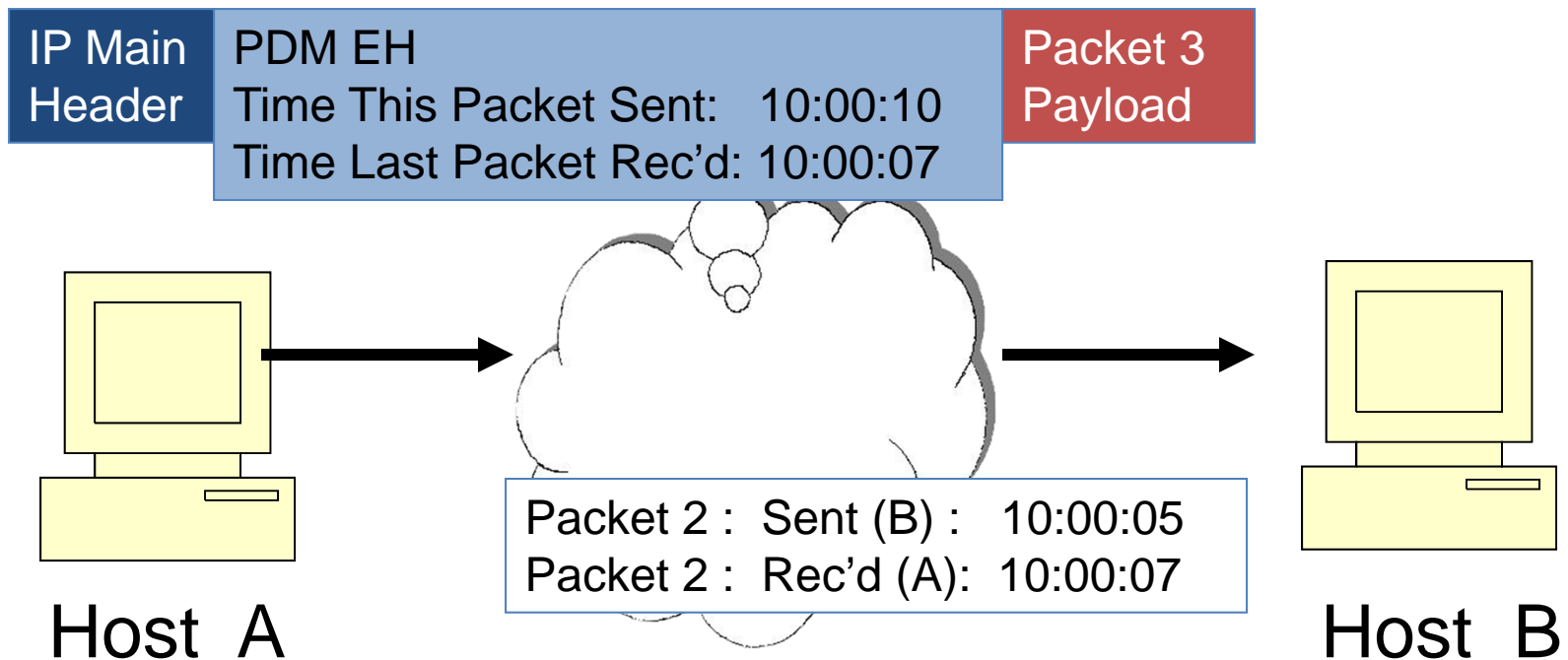
# When Did it Get to Host A?

- When did Packet 2 to arrive at Host A?
- Return route may not be the same, may be congestion, packet might never arrive.



# What is Needed?

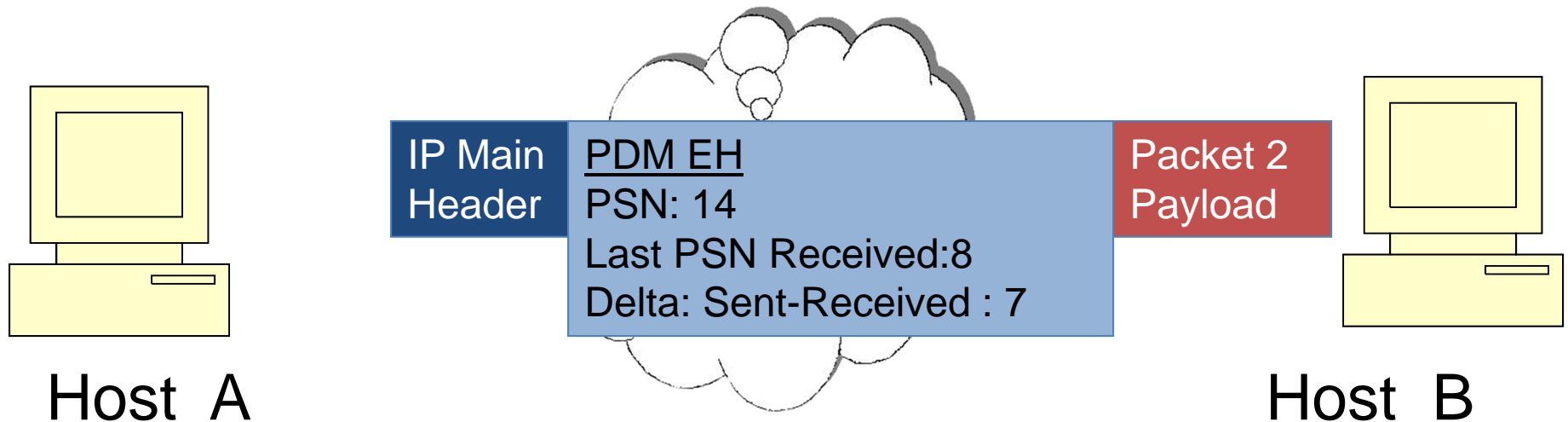
- With each packet, add “Time Last Packet Received” in PDM EH
- When Packet 3 sent, has when Packet 2 got to Host A
- Outbound Network time = Last rec'd (A) – Time sent (B)
- Processing Time (A) = Packet 3 sent (A) - Last rec'd (A)





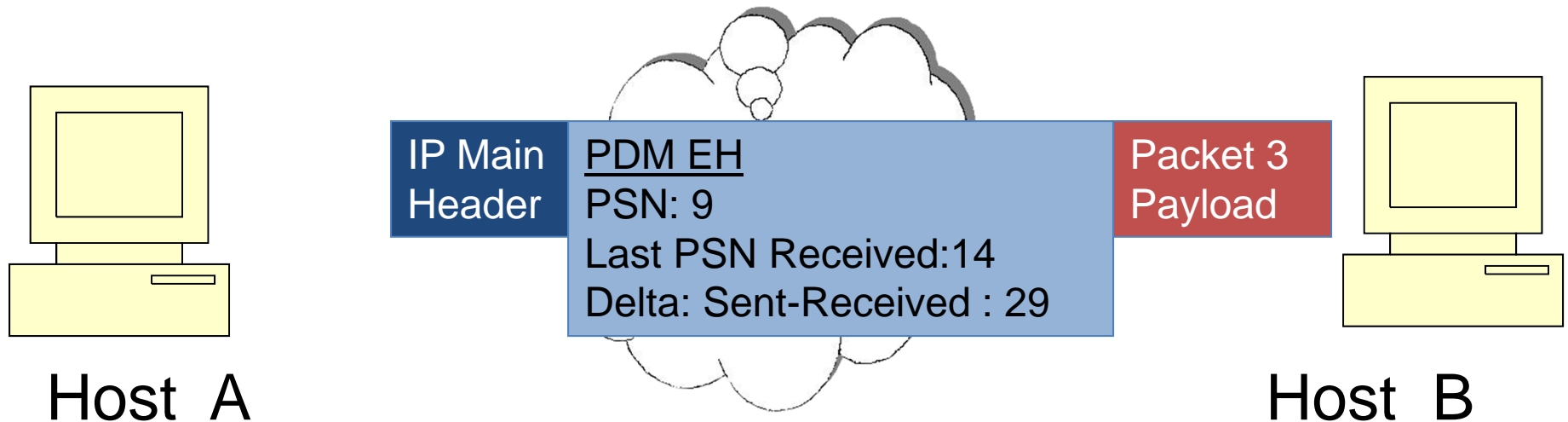
# At Host B: PDM Type 2

- Packet 2 sent (10:00:14) – Packet 1 received (10:00:07)
- Server delay time ( 7 seconds)
- PSN-A starts at 8 (random)
- PSN-B starts at 14 (random)



# At Host A: PDM Type 2















- Packet 1 sent (10:00:01) – Packet 2 received (10:00:30)
- Round trip time: 29 seconds
- Server delay time : 7 seconds
- Two way delay: 22 seconds
- PDM is sent in NEXT packet



	PDM 2 (Delta)	PDM 1 (Timestamp)	TCP Timestamp	TCP Seq Number	AH / ESP Header	Agents or OWAMP TWAMP
1. Must apply to all upper layer protocols	Yes	Yes	No	No	Yes	Yes
2. Has diagnostic <u>and</u> performance value	Yes	Yes	No	No	No	Varies
3. Calculate server <u>and</u> network delays	Yes	Yes	Yes	No	No	Yes
4. Low overhead (expense, infrastructure to implement, etc)	Yes	Yes	Yes	Yes	No	No
5. Not synthetic traffic	Yes	Yes	Yes	Yes	Yes	Varies
6. Easy user implementation / operation	Yes	Yes	Yes	Yes	No	No
7. Can be only at one end	Yes	Yes	No	No	No	No
8. Effective without time synchronization	Yes	Varies	Yes	Yes	Yes	Varies

Show IPv6 PDM Type 16 Header  
 Using: Trace File:pdm16  
 Sort Order : Packet Number

This is PDM Type 2

-	-	Packet Number	Packet Date	Extension Header	Source Address	Destination Address	This Packet ID	Packet Last Received	Packet Last Sent	Delta Last Received (Microseconds)	Delta Last Sent (Microseconds)
1		4	2014-01-10 13:30:22.857512	60	2001::2	2001::1	0	0	0	0	0
2		5	2014-01-10 13:30:22.860452	60	2001::1	2001::2	0	0	0	0	0
3		6	2014-01-10 13:30:23.865714	60	2001::2	2001::1	1	0	0	1006	983
4		7	2014-01-10 13:30:23.877588	60	2001::1	2001::2	1	0	0	1017	1017
5		8	2014-01-10 13:30:24.870476	60	2001::2	2001::1	2	1	1	1008	974
6		9	2014-01-10 13:30:24.871949	60	2001::1	2001::2	2	1	1	994	994
7		13	2014-01-10 13:30:25.879201	60	2001::2	2001::1	3	2	2	1005	995
8		14	2014-01-10 13:30:25.88565	60	2001::1	2001::2	3	2	2	1013	1013
9		17	2014-01-10 13:30:26.886962	60	2001::2	2001::1	4	3	3	1008	985
10		18	2014-01-10 13:30:26.897091	60	2001::1	2001::2	4	3	3	1011	1011
11		19	2014-01-10 13:30:27.891001	60	2001::2	2001::1	5	4	4	1007	974
12		20	2014-01-10 13:30:27.901722	60	2001::1	2001::2	5	4	4	1004	1004
13		27	2014-01-10 13:30:28.894605	60	2001::2	2001::1	6	5	5	1004	982
14		28	2014-01-10 13:30:28.905136	60	2001::1	2001::2	6	5	5	1003	1003

```
.... 0 .... = IG bit: Individual address (unicast)
Type: IPv6 (0x86dd)
Internet Protocol Version 6, Src: 2001::2 (2001::2), Dst: 2001::1 (2001::1)
0110 .... = Version: 6
  [0110 .... = This field makes the filter "ip.version == 6" possible: 6]
.... 0000 0000 .... = Traffic class: 0x00000000
  .... 0000 00.. .... = Differentiated Services Field: Default (0x00000000)
  .... ..0. .... = ECN-Capable Transport (ECT): Not set
  .... ..0 .... = ECN-CE: Not set
.... 0000 0000 0000 0000 0000 = Flowlabel: 0x00000000
Payload length: 71
Next header: IPv6 destination option (60)
Hop limit: 64
Source: 2001::2 (2001::2)
[Source Teredo Server IPv4: 0.0.0.0 (0.0.0.0)]
[Source Teredo Port: 65535]
[Source Teredo Client IPv4: 255.255.255.253 (255.255.255.253)]
Destination: 2001::1 (2001::1)
[Destination Teredo Server IPv4: 0.0.0.0 (0.0.0.0)]
[Destination Teredo Port: 65535]
[Destination Teredo Client IPv4: 255.255.255.254 (255.255.255.254)]
[Source GeoIP: Unknown]
[Destination GeoIP: Unknown]
Destination Option
  Next header: TCP (6)
  Length: 2 (24 bytes)
  IPv6 Option (PadN)
    Type: PadN (1)
    Length: 6
    PadN: 000000000000
  IPv6 Option (Unknown 16)
    Type: Unknown (16)
    Length: 14
    Unknown Option Payload: 000103000003ee000003d700b326
Transmission Control Protocol, Src Port: 45862 (45862), Dst Port: 80 (80), Seq: 452248946, Len: 7
Source port: 45862 (45862)
Destination port: 80 (80)
[Stream index: 2]
```

Breakout in  
WireShark



# Appendix

- Further information

# Session Classification

- The PDM may be used to classify sessions as follows:
  - One way traffic flow
  - Two way traffic flow
  - One way traffic flow with keep-alive
  - Two way traffic flow with keep-alive
  - Multiple send traffic flow
  - Multiple receive traffic flow
  - Full duplex traffic flow
  - Half duplex traffic flow
  - Immediate ACK data flow
  - Delayed ACK data flow
  - Proxied ACK data flow