draft-kumar-ippm-ifa-00

IETF-104

Motivation

Provide simple, protocol agnostic, flexible, inband mechanism to track and collect per flow per packet network telemetry data with low cost and high performance of HW

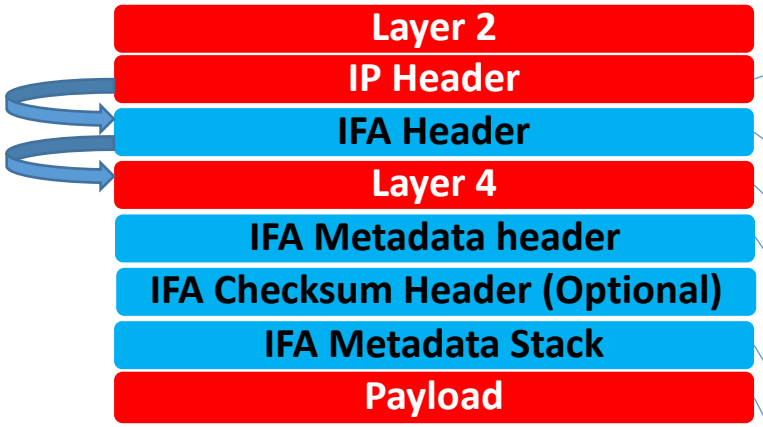
IFA - Recap

- Protocol agnostics proposal – Single draft handles all packet types
  - IPv4/v6 UDP/TCP, GRE, IPinIP, AH/AES, IPv6 EH, SRv6
- Same proposal works for MPLS
  - I-D. draft-Kumar-mpls-mint-00 (Kireeti Kompella, Jai Kumar, John Lemon)
  - NextHdr provides deterministic payload parsing (instead of speculative)
  - NextHdr provides ability to present other metainformation like security tag etc

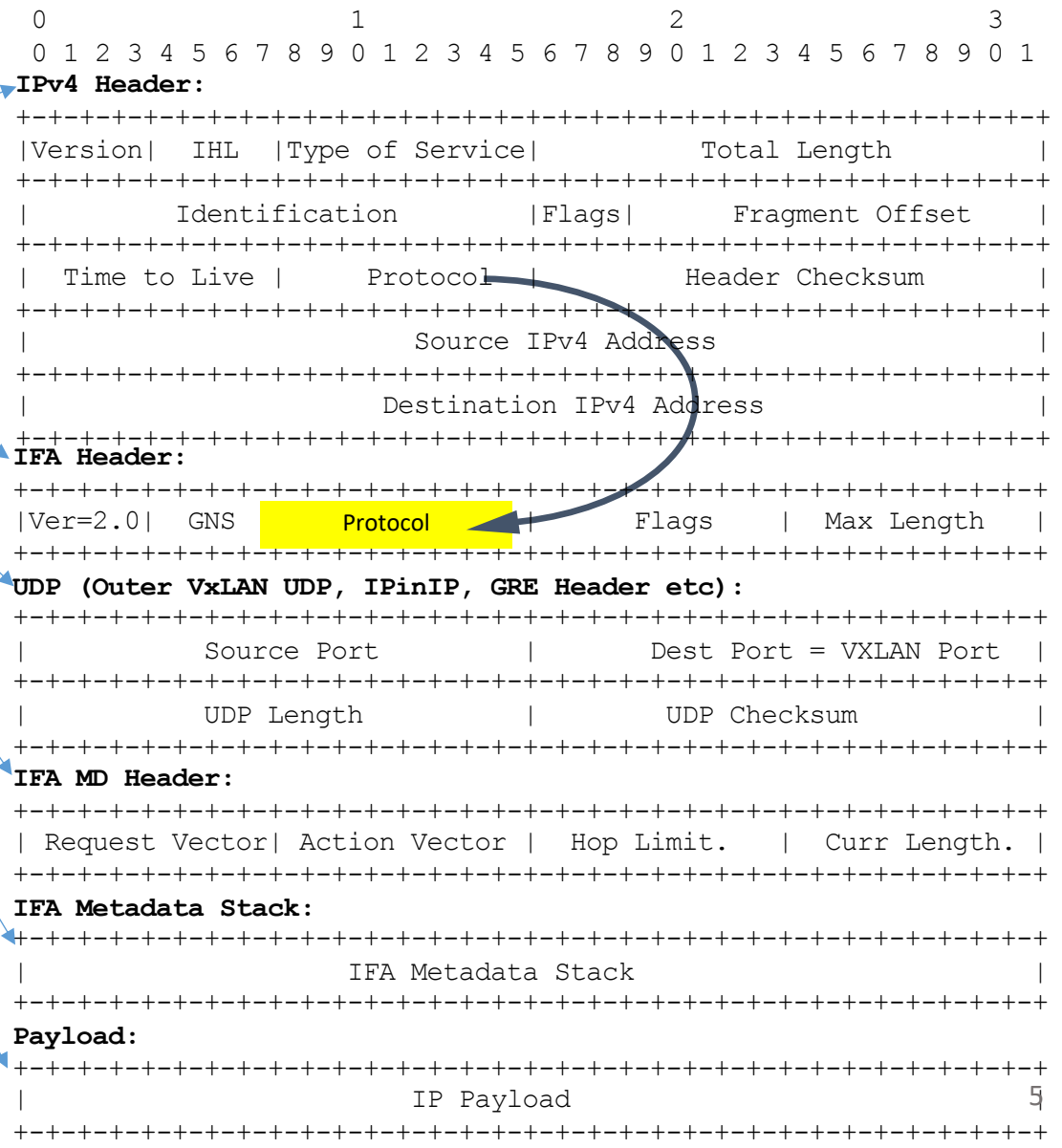
IFA – Updates

- Added support for Fragmentation
  - Needed with very large number of hops in path
  - PMTU is a limitation
- Degenerate case of Fragmentation is Post Card
  - Each node generates each own metadata packet towards collector
  - No insertion of metadata in the packet
- Deployed in MSDC
- Developed and supported by System vendors

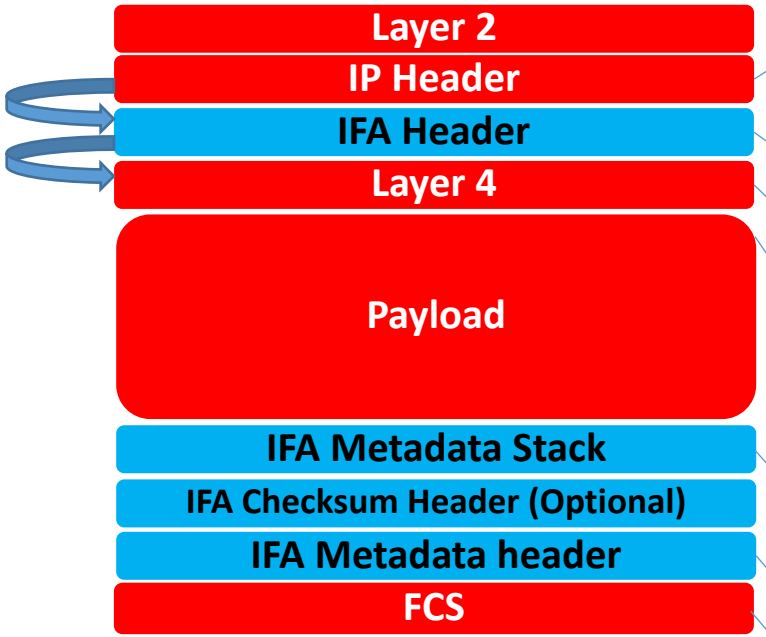
IFA – L3 Extension Header

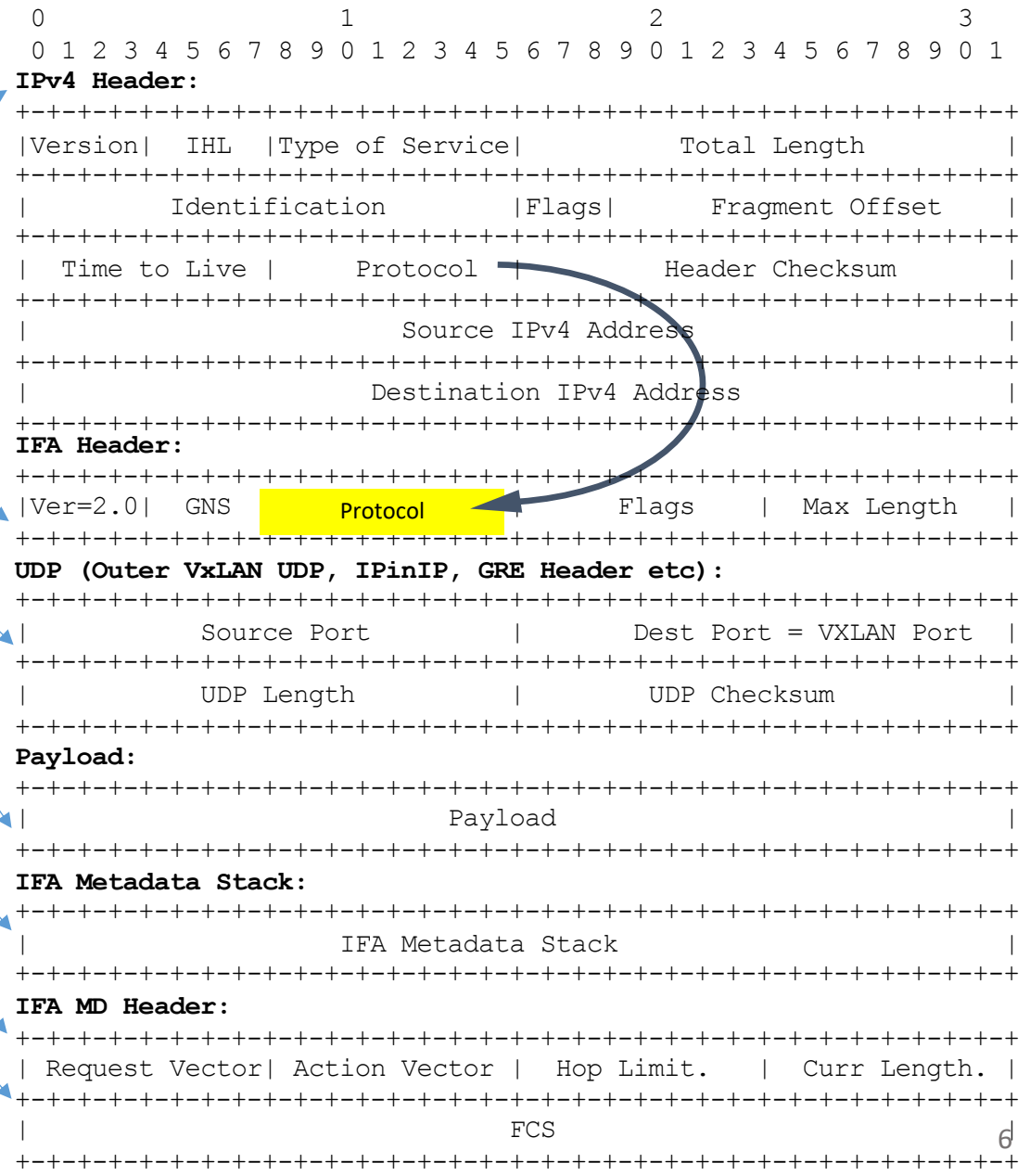


- Use 1 experimental IP protocol type (143 – 252): PROTO_IFA
- Original IP->proto is copied in IFA->proto
- IFA header is treated as L3 upper layer protocol
- Metadata is treated as L4 information
- Metadata is treated as foreign object
  - L4 length and checksum is not updated
  - This is to ease load on hw to strip and forward



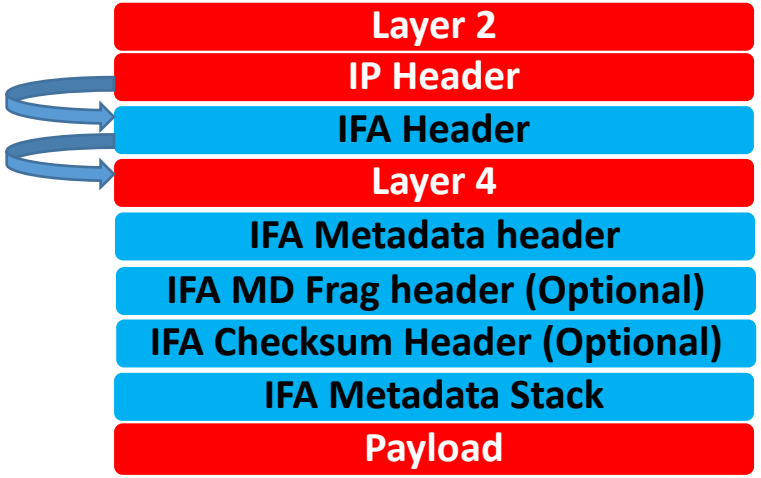
IFA – Tail Insertion





- Metadata Header is at the tail of the packet before FCS
- Metadata is inserted at the bottom of the metadata stack
- MD headers and MD stack looks inverted for packet tail stamp

IFA – Metadata Fragment (MF) Header



- IFA Header Flags “MF” bit is set by Initiator Node
- IFA Frag Header is inserted by Initiator Node
- Frag Header fields Packet ID and Fragment ID set by Initiator Node
- Collector uses timestamp, Packet ID, Frag ID and Flow tuples as a key
- Fragment ID is used to recreate ordered metadata set

```
IFA Header:

       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
       |Ver=2.0|  GNS   |NextHdr=TCP/UDP|      Flags     |   Max Length   |
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+

     (4) Flags (8 bits)
           0: Checksum - Indicates the presence of the checksum header.
           1: TA - Turn Around. Indicates that the IFA packet needs to be
           turned around at the terminating node of the IFA zone.
           2: I - Inband. Indicates this is live traffic. Strip and forward
           MUST be performed by the terminator node if this bit is set.
           3: TS - Tail Stamp.  Indicates the IFA zone is requiring tail
           stamping of metadata.
           4: MF - Fragment.  Indicates the presence of fragment header.
           ignored on receipt.
           5-7: Reserved.  MUST be initialized to 0 on transmission and
           ignored on receipt.

MD Frag Header:
        0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
       |                  Packet ID.                           |  FragID |L|
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

IFA – Metadata Fragmention Function

Any IFA node MAY perform MF function.

Degenerate case each node generates fragment aka IFA report instead of inserting metadata in packet

For Use When

- Very large number of hops in the network
- PMTU limits the packet size

Requirement

- ”MF” bit, set by initiator node
  - 1: *Fragment header present*
  - 0: *Do not Fragment (DF)*
- IFA Frag header inserted by Initiator node
  - [Packet ID, Fragment ID]
- Nodes in path check

If (ifa_hdr.flags.MF == TRUE) && (ifa_hdr.current_length > ifa_hdr.md_hdr.max_length)

*Send a *truncated copy of packet to collector*

Increment Sequence Number in IFA Frag Header

Delete metadata stack (till the fragmentor hop)

Insert metadata of fragmentor hop and forward packet

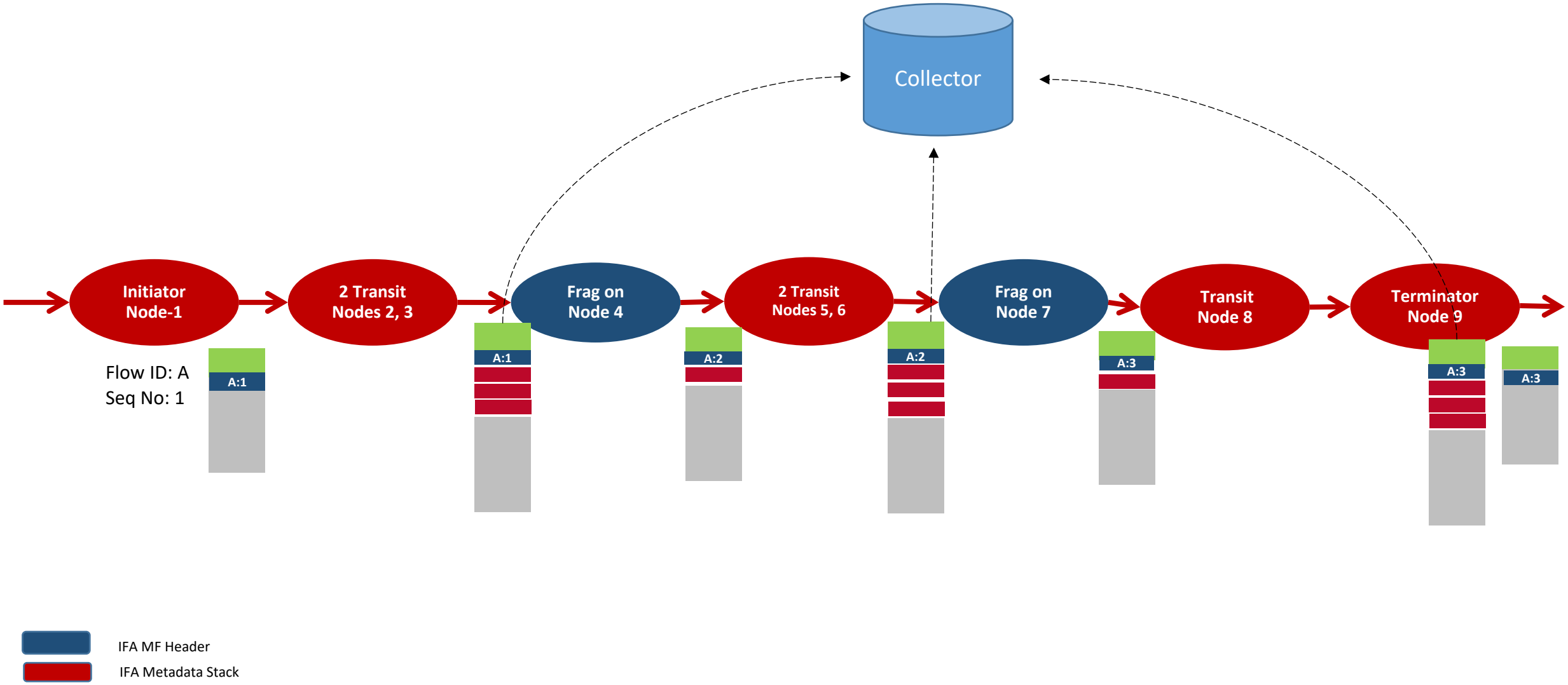
Fragmentation Header and ID Wrap around

- Frag header is a 4 byte overhead
- 5bits of Frag ID will support 32 Fragmentor nodes in the path
- Last Frag “L” bit indicates last fragment
- Metadata Fragmentation is conditional and MAY be performed by any IFA node
  - If (MF bit is set in IFA header) and ((incoming Current length + sizeof(local metadata)) > maximum length)
- Fragmentation involves following
  - Create a copy of packet and send it to collector
  - Delete the metadatastack
  - Update the current length in IFA header
  - Increment the FragID by 1 in IFA MD header
  - Terminator node SET the lastFrag bit in IFA MD header

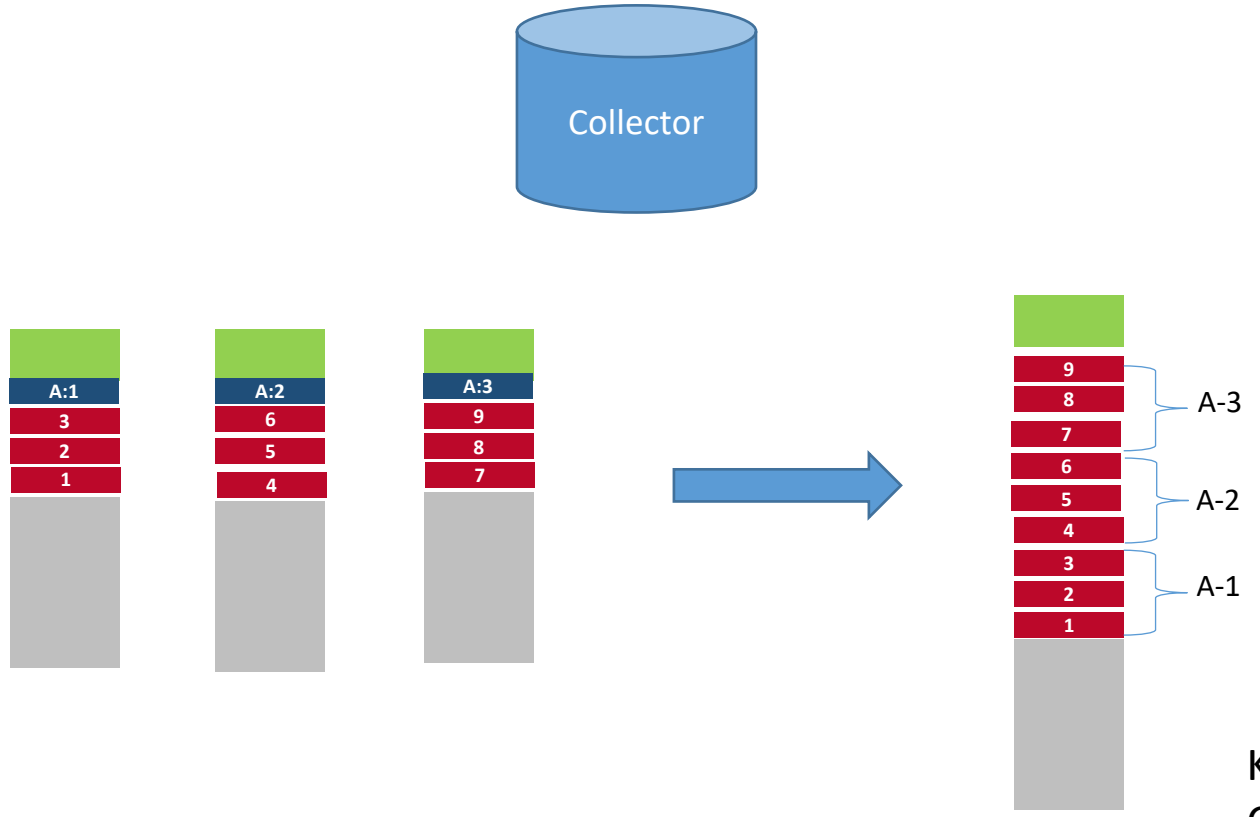
Fragmentation Header and ID Wrap around

- Fragmentation SHOULD be enabled if exceeding PMTU or allowable Metadata stack length
- 26bit ID field MUST be unique within the MDL (maximum datagram life) and 5 tuple
- Average i-mix packet size 512 bytes with 3 sec MDL, bit rate supported for a given node before ID wraparound
  - 183.25 Gbps
  - Sampler per node MUST not exceed traffic rate of 183.25 Gbps
- Initiator node responsible for sampling need to manage the ID space
- First fragment always start with FragID “0”
- Collector should discard fragment if received after MDL

IFA Fragmentor



IFA Fragment Ordered Set



Key: *Timestamp, Flow Tuple, PacketID*
Order Key*: Fragment ID*

*Per Flow Packet ID can also be used to detect packet drops

IFA MF Header
IFA Metadata Stack

Status of I-D

draft-kumar-ippm-ifa-01

https://tools.ietf.org/html/draft-kumar-ippm-ifa-00

- Updated with packet format interworking with tunnels and IPv6 EH
- Updated with support for fragmentation

Request

- Would appreciate review and comments from WG
- WG adoption

Reference

https://tools.ietf.org/html/rfc6864

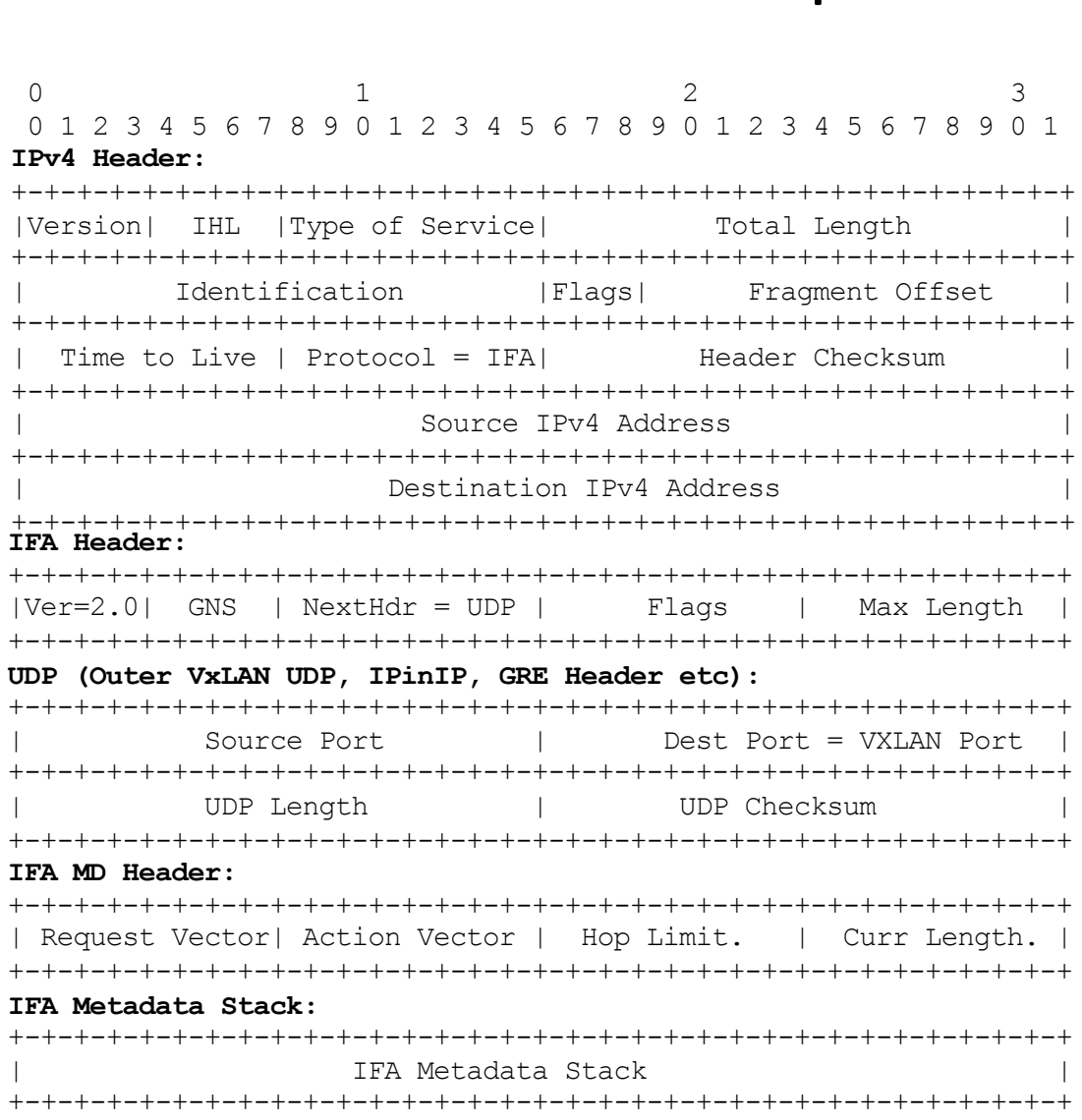
https://tools.ietf.org/html/rfc4963

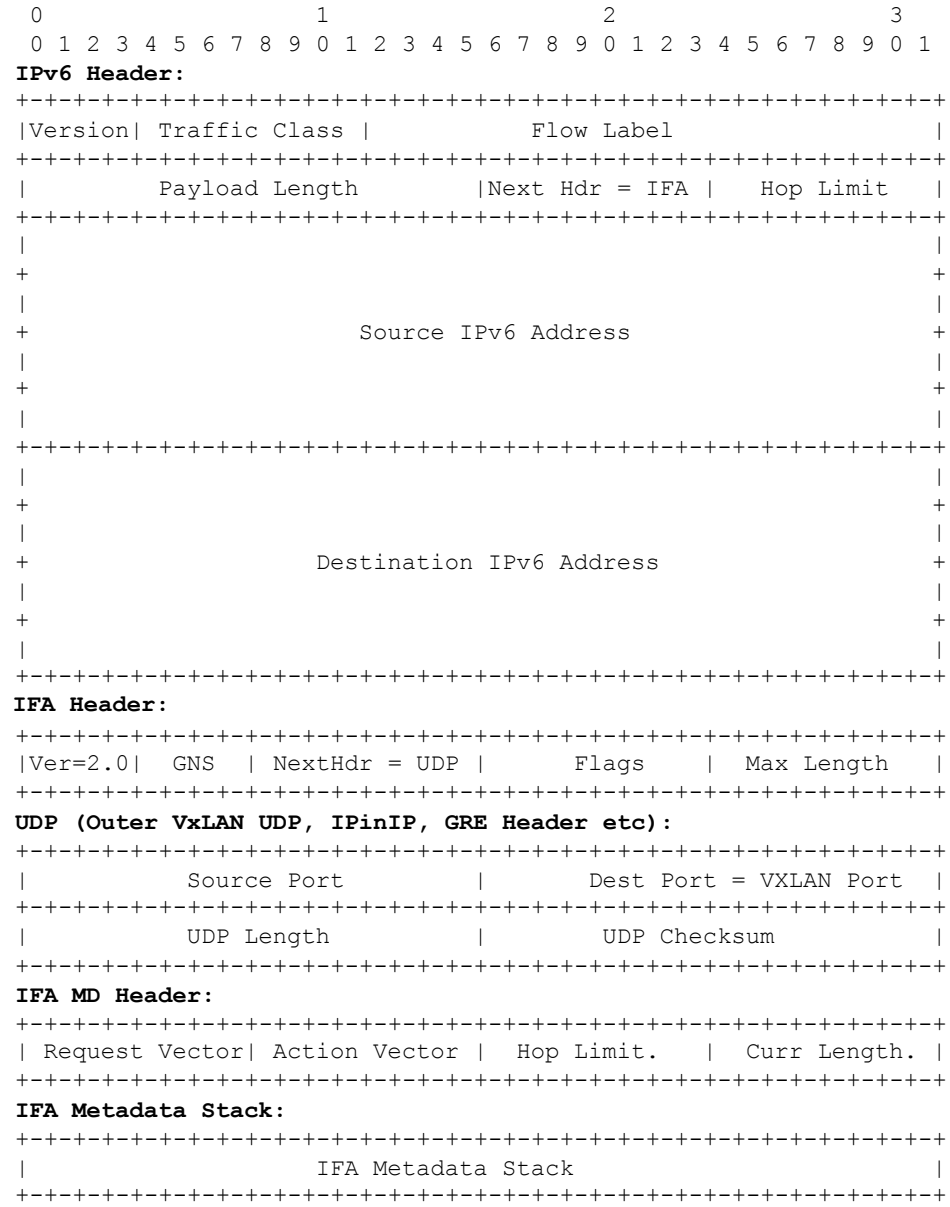
https://tools.ietf.org/html/draft-kumar-mpls-mint-00

Thank You

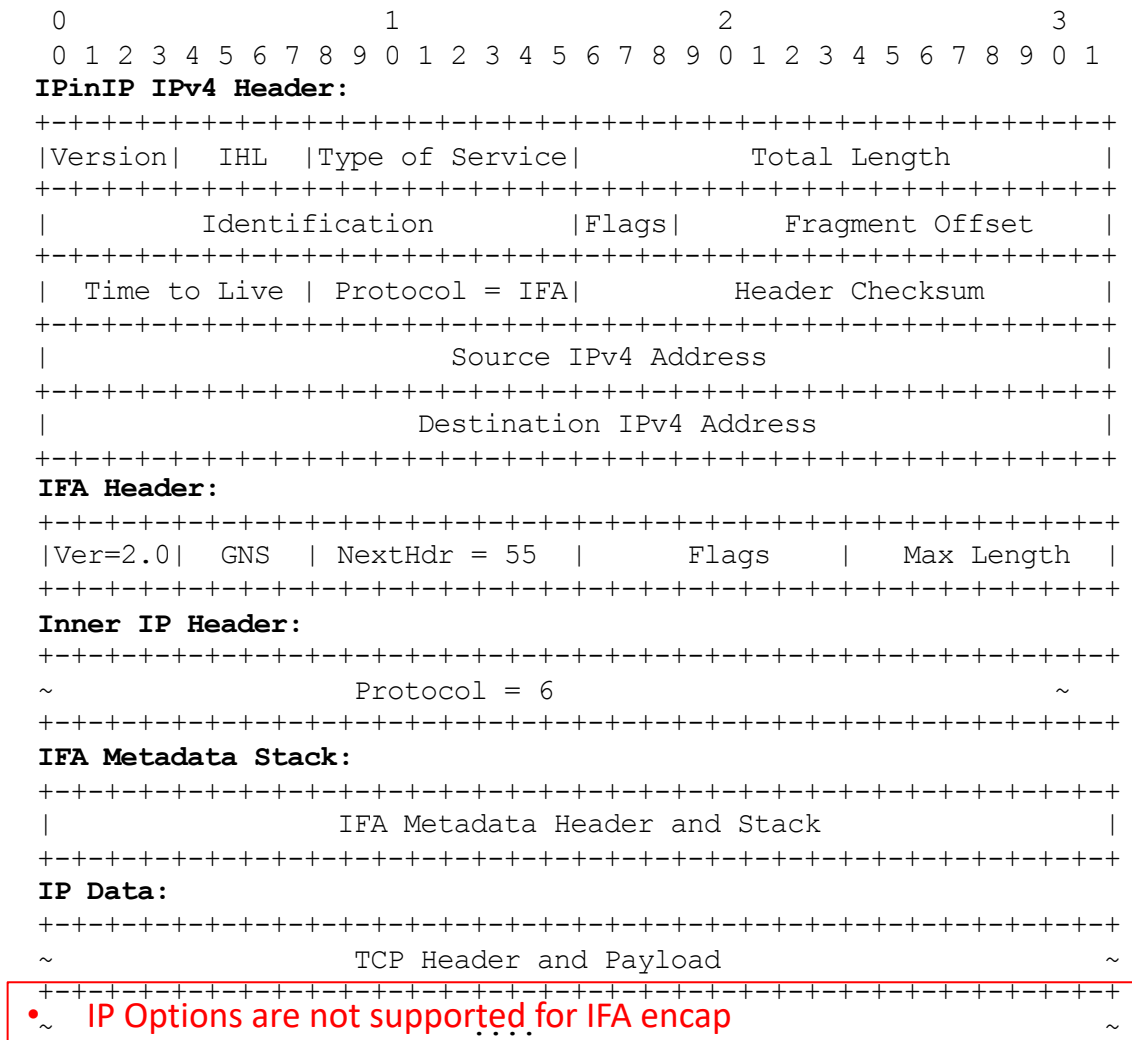
BACKUP

IFA IPv4/IPv6 Encapsulation

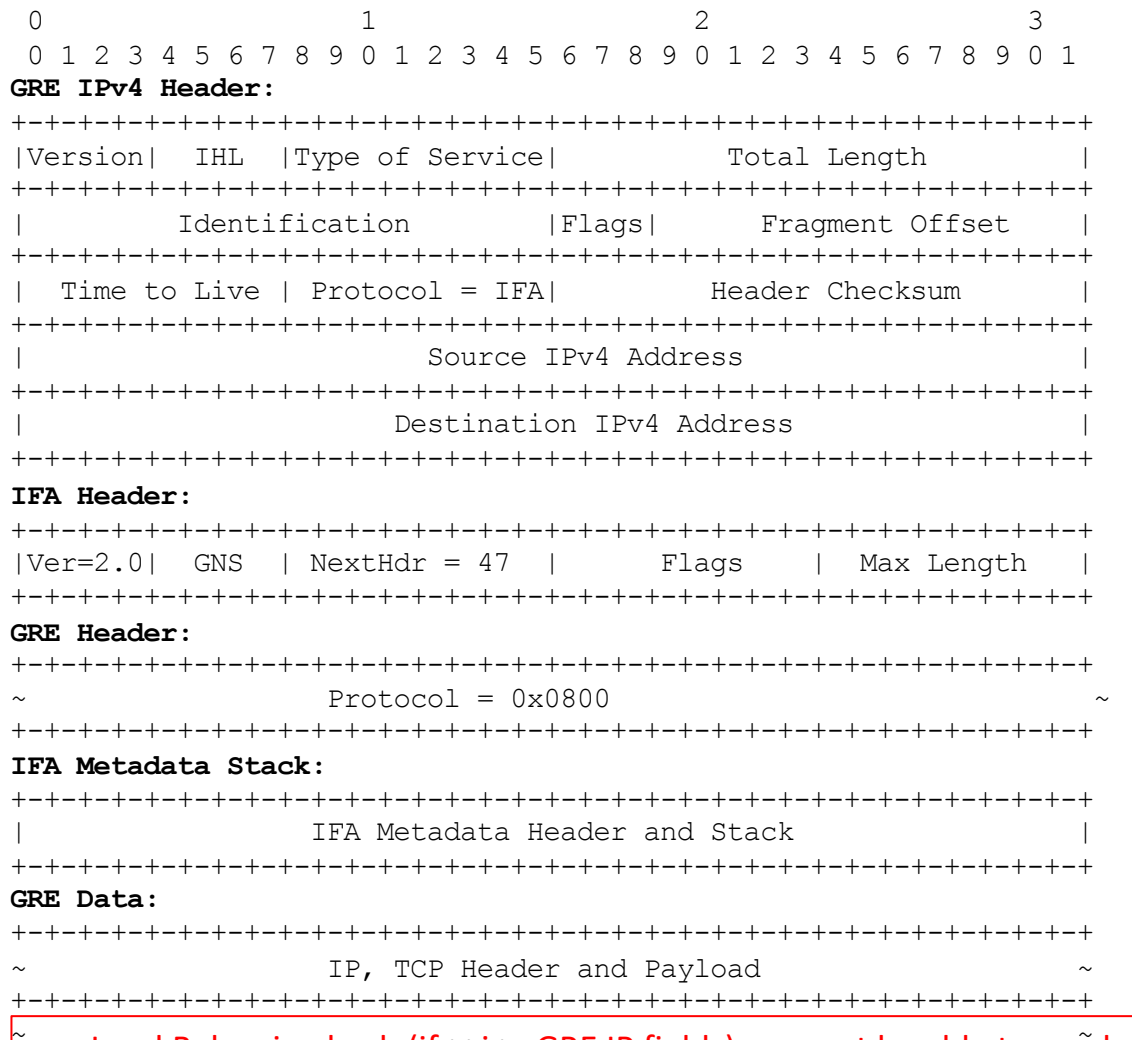




IFA IPv4 IPinIP and GRE Encapsulation

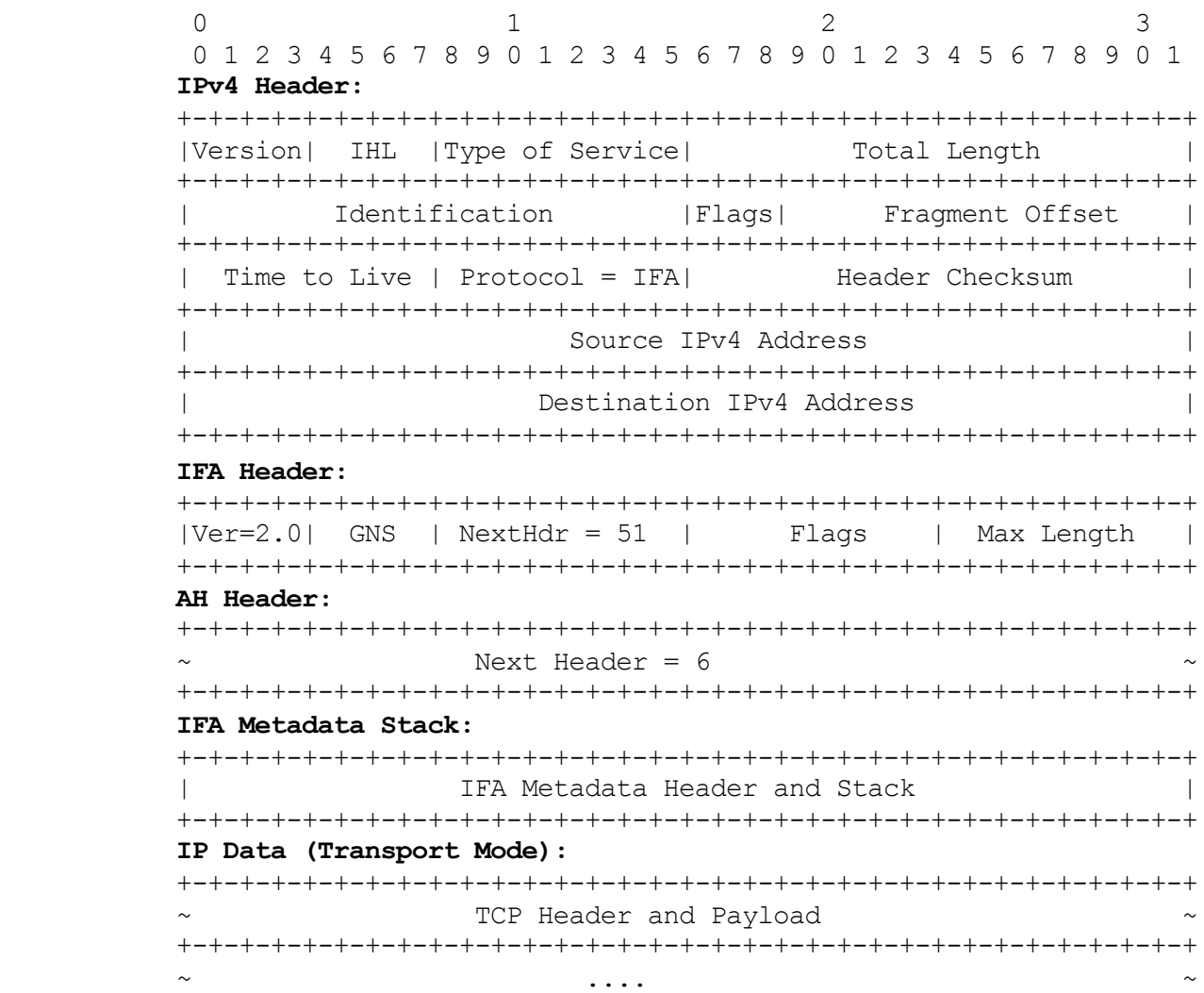


- IP Options are not supported for IFA encap
- Inner IP if has IP options will not be parsed for IFA
- Outer IP if has IP options, will not be parsed for IFA
- Load Balancing hash (if using inner IP L4 fields) may not be able to reach inner L4 fields

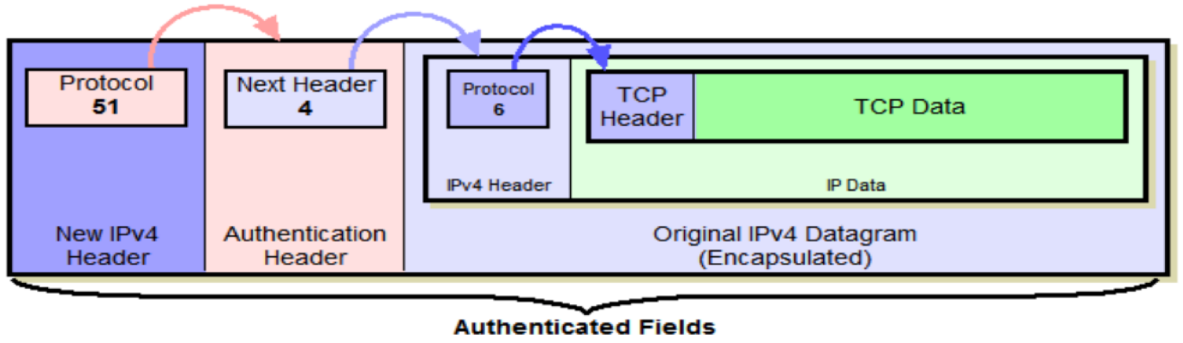


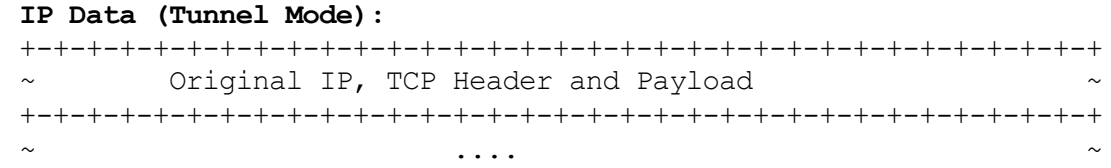
- Load Balancing hash (if using GRE IP fields) may not be able to reach GRE L4 fields
- GRE options C,R,K,S are treated as part of GRE Header as they are bit defined presence fields (not a header chain)

IFA IPv4 AH (TCP payload) Encapsulation



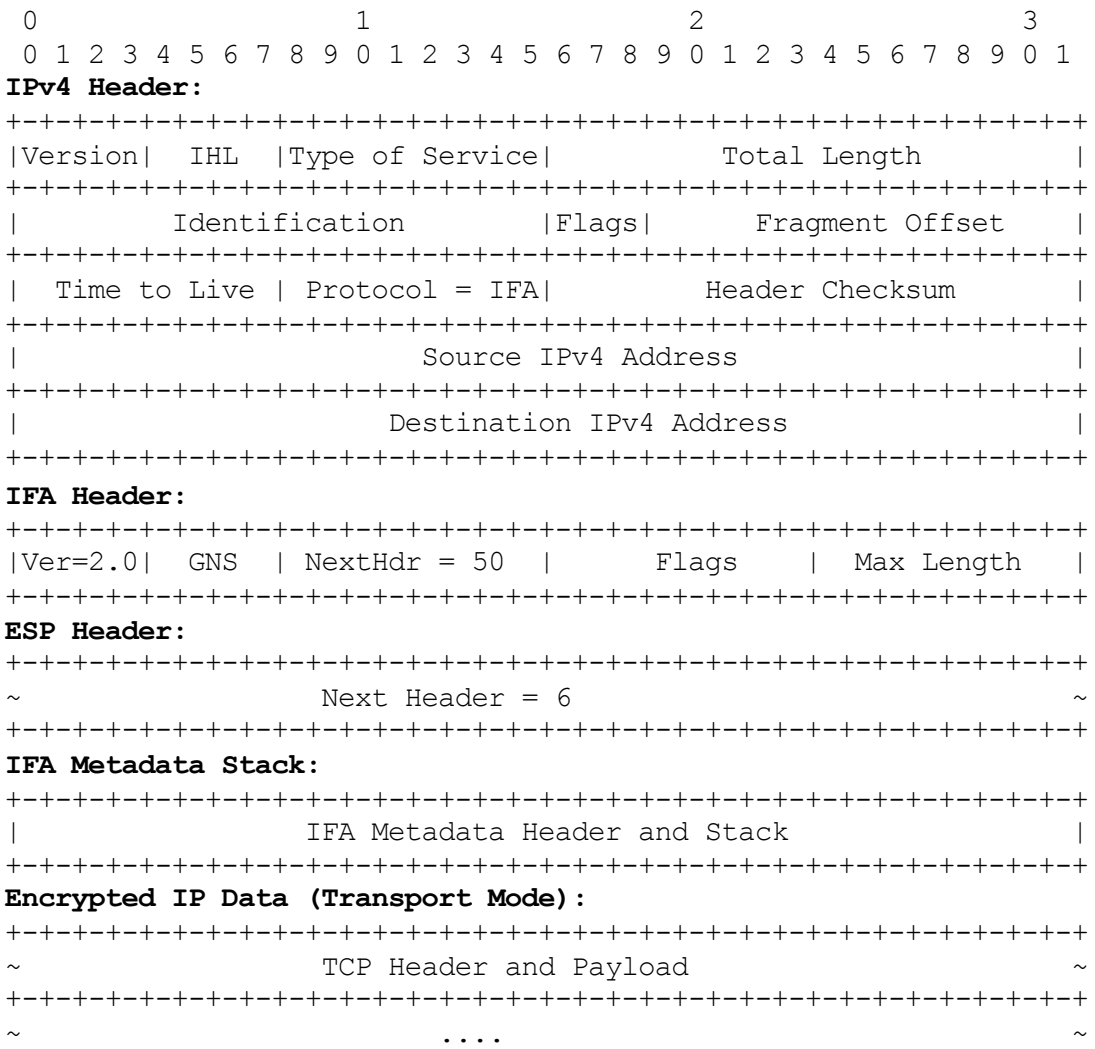
- IPv4 options are separate then protocol extension headers (unlike IPv6)
- IFA base header is inserted after the IP header
- MD stack is inserted after the IP AH header (works for both transport and tunnel mode)
- IPv4 IFA layering remains same if protocol extension headers are present are not
- AH payload is not touched so IPSec tunnel strip and forward is easy



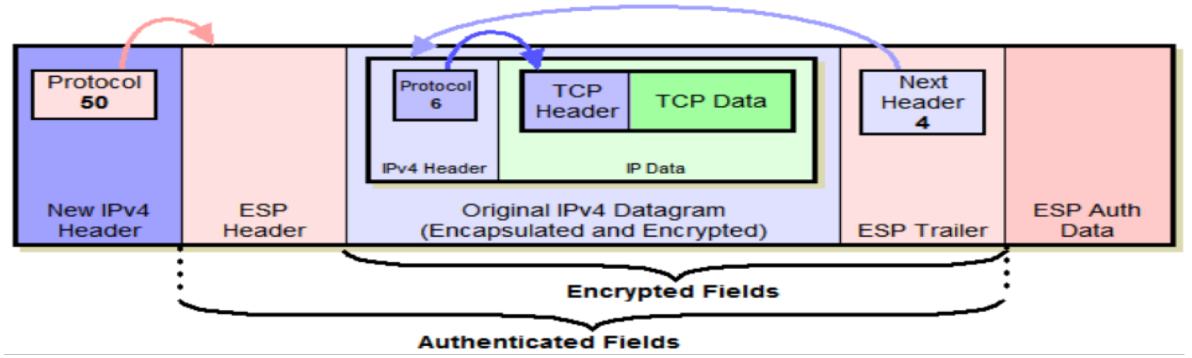


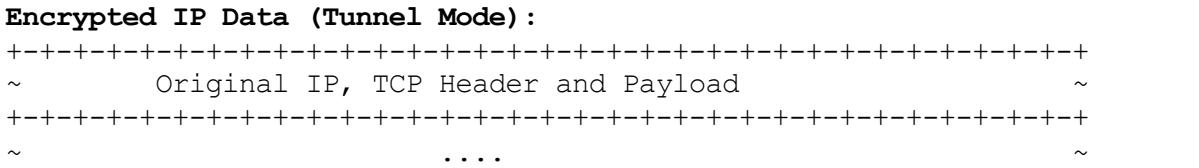
IPv4/AH/IFA/L4/MD Stack/L4Payload

IFA IPv4 ESP (TCP payload) Encapsulation



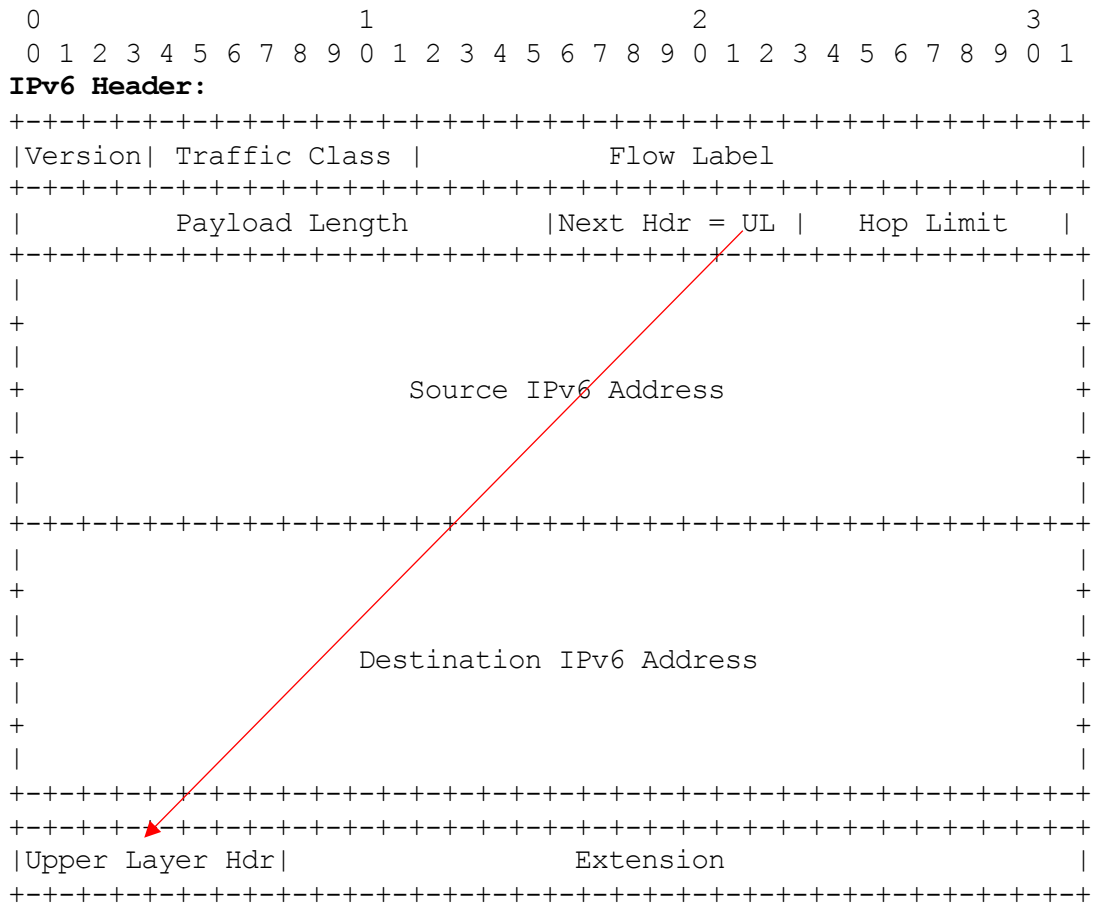
- IPv4 options are separate then protocol extension headers (unlike IPv6)
- IFA base header is inserted after the IP header
- MD stack is inserted after the IP AH header (works for both transport and tunnel mode)
- IPv4 IFA layering remains same if protocol extension headers are present are not
- ESP payload is encrypted and not touched so IPSec tunnel strip and forward is easy





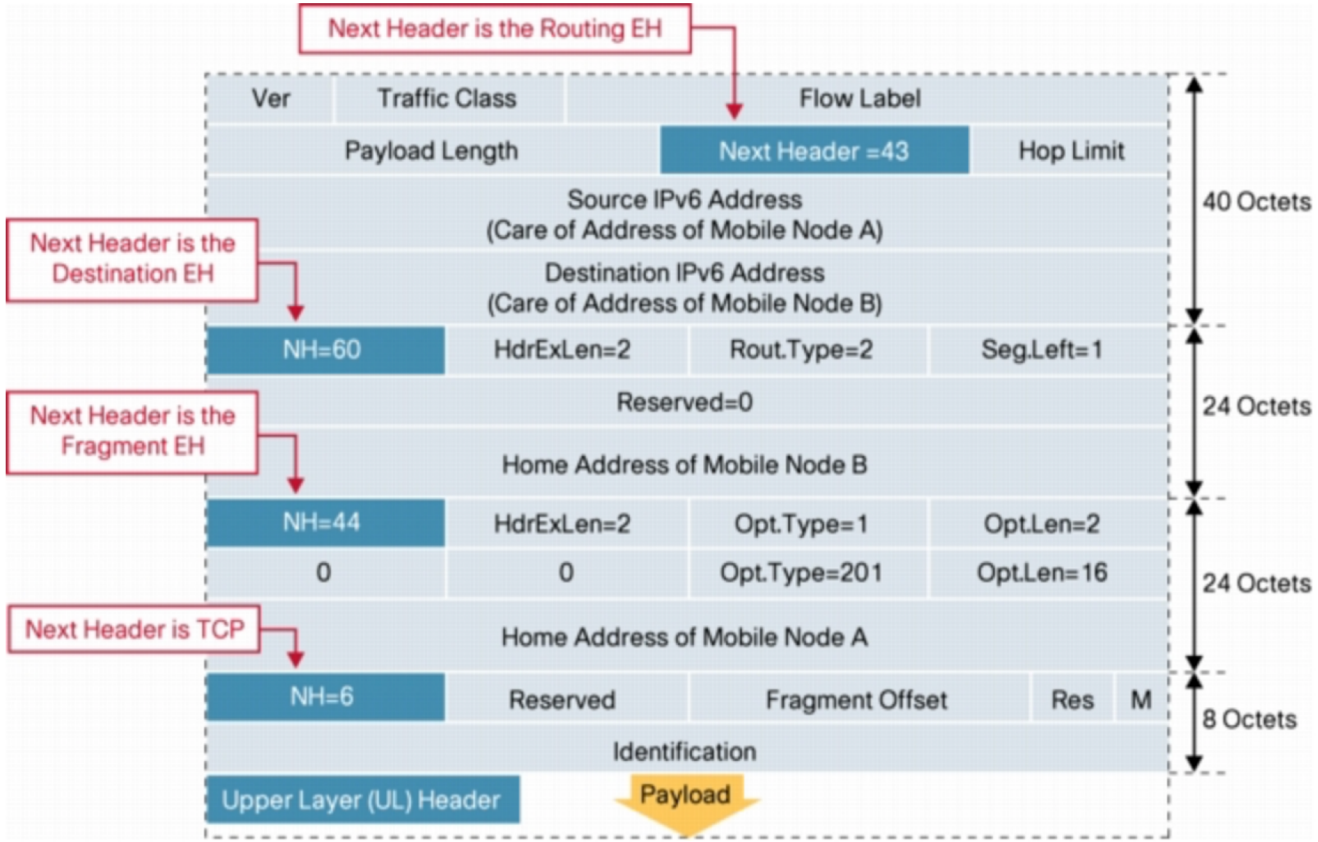
IPv4/ESP/IFA/MD Stack/{L4/L4Payload}

IPv6 Extension Header Processing



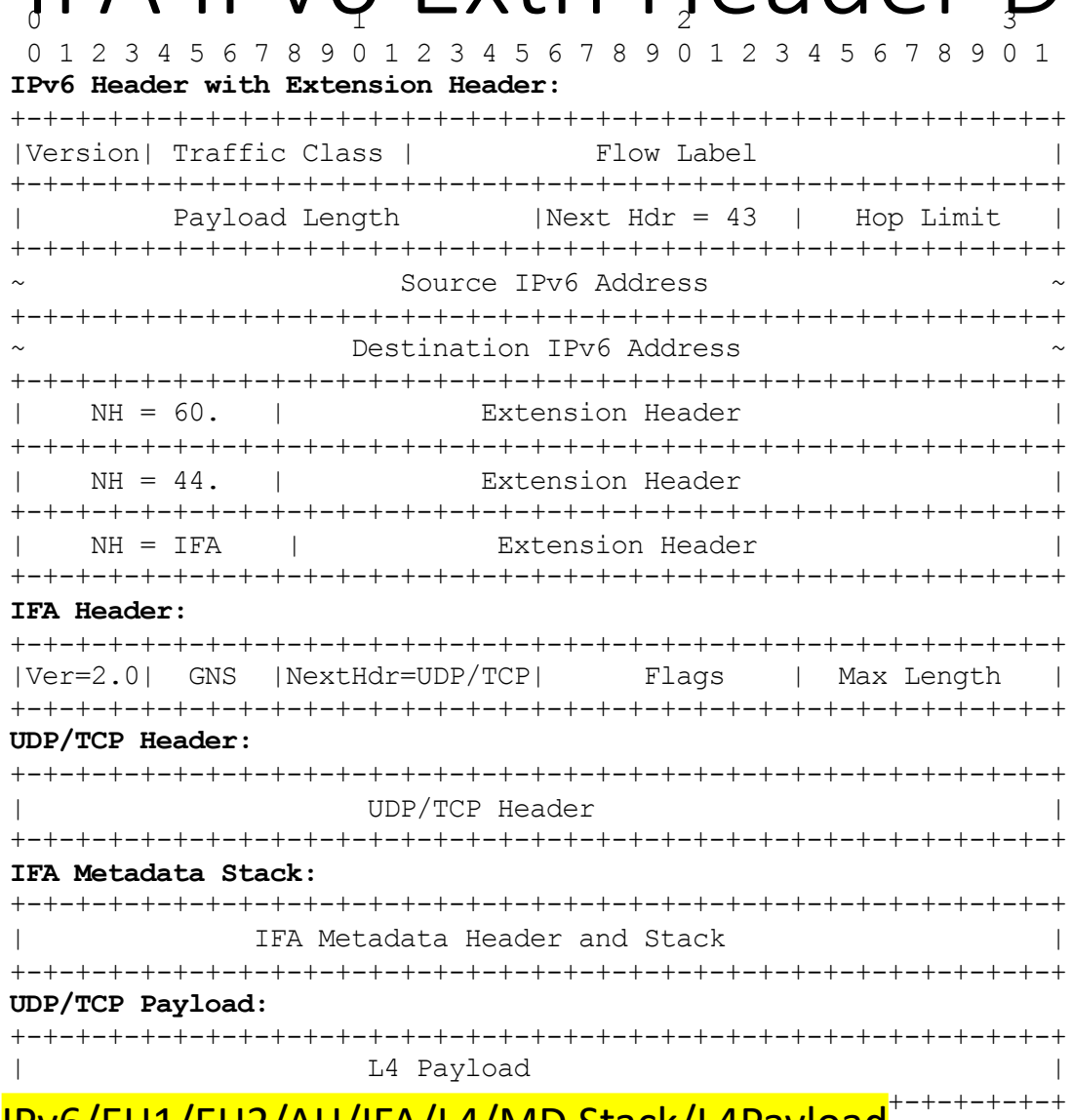
Order	Header Type	Next Header Code
1	**Basic IPv6 Header**	-
2	**Hop-by-Hop Options**	0
3	**Destination Options (with Routing Options)**	60
4	**Routing Header**	43
5	**Fragment Header**	44
6	**Authentication Header**	51
7	**Encapsulation Security Payload Header**	50
8	**Destination Options**	60
9	**Mobility Header**	135
	No next header	59
Upper Layer	TCP	6
Upper Layer	UDP	17
Upper Layer	ICMPv6	58
Upper Layer	IFA	XXX

IPv6 Extn Header Data Traffic Example

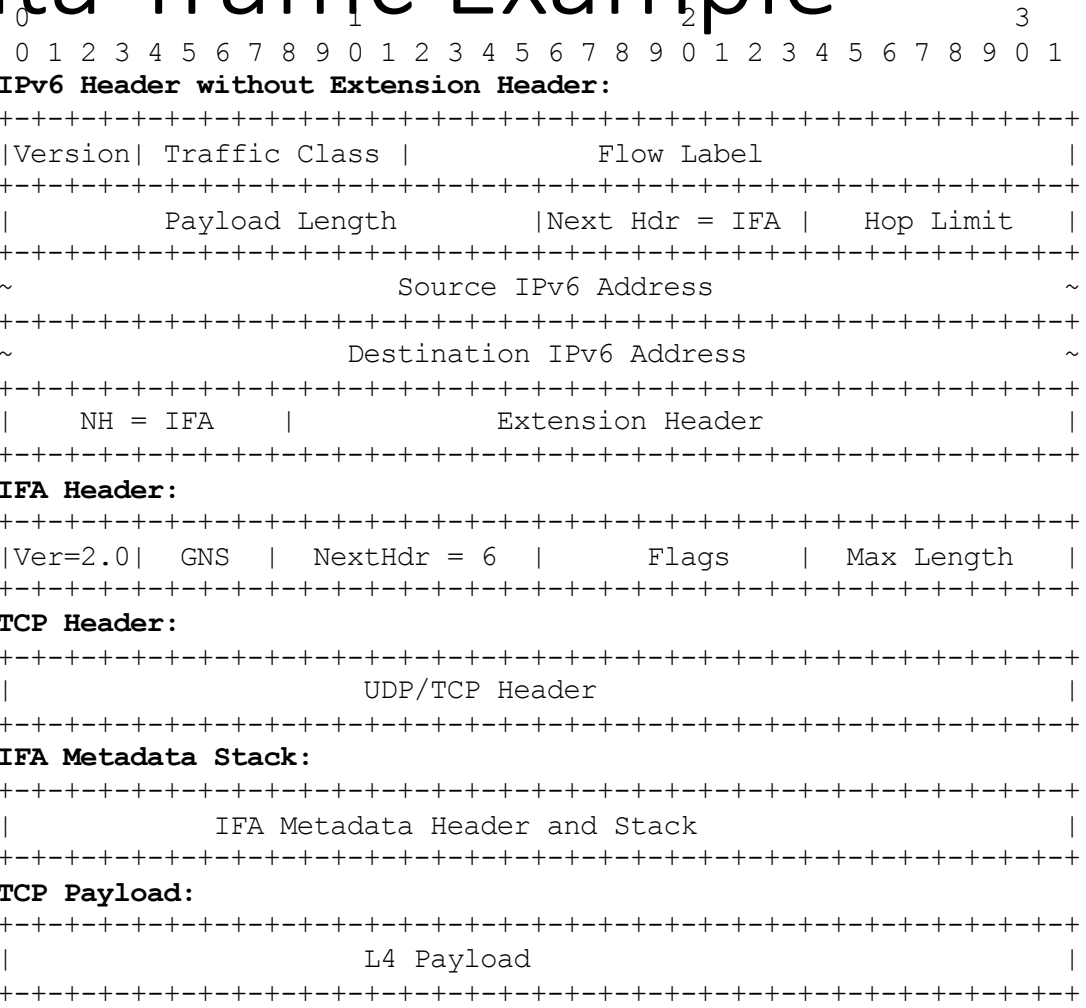


Order	Header Type	Next Header Code
1	**Basic IPv6 Header**	-
2	**Hop-by-Hop Options**	0
3	**Destination Options (with Routing Options)**	60
4	**Routing Header**	43
5	**Fragment Header**	44
6	**Authentication Header**	51
7	**Encapsulation Security Payload Header**	50
8	**Destination Options**	60
9	**Mobility Header**	135
	No next header	59
Upper Layer	TCP	6
Upper Layer	UDP	17
Upper Layer	ICMPv6	58
Upper Layer	IFA	XXX

IFA IPv6 Extn Header Data Traffic Example



IPv6/EH1/EH2/AH/IFA/L4/MD Stack/L4Payload
IPv6/EH1/EH2/ESP/IFA/MD Stack/{L4/L4Payload}



- Parse Extension header as is (May need punt to CPU)
- Insert IFA base header if UL header is detected
- Insert metadata after the UL header
- Ipv6 IFA layering remains same if extension headers are present are not

Thank You