INTERNET-DRAFT
Intended Status: Experimental

J. Kumar
S. Anubolu
J. Lemon
R. Manur
Broadcom Inc.
H. Holbrook
Arista Networks
A. Ghanwani
Dell EMC
D. Cai
H. OU
AliBaba Inc.
Y. Li
Huawei

Expires: August 25, 2019
February 21, 2019

Inband Flow Analyzer
draft-kumar-ippm-ifa-01

Abstract

Inband Flow Analyzer (IFA) records flow specific information from an end station and/or switches across a network. This document discusses the method to collect data on a per hop basis across a network and perform localized or end to end analytics operations on the data. This document also describes a transport-agnostic header definition that may be used for tunneled and non-tunneled flows alike.

Status of this Memo

This Internet-Draft is submitted to IETF in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

The list of current Internet-Drafts can be accessed at

http://www.ietf.org/1id-abstracts.html

The list of Internet-Draft Shadow Directories can be accessed at http://www.ietf.org/shadow.html

Table of Contents

1. Introduction

This document describes Inband Flow Analyzer (IFA) which is a mechanism to mark packets in a flow to enable the collection of metadata regarding the analyzed flow. IFA defines an IFA header to mark the flow and direct the collection of analyzed metadata per marked packet per hop across a network. The ability to mark a packet using an IFA OAM header can also be leveraged to create synthetic flows meant for network data collection. This document describes a mechanism that may be used to monitor live traffic and/or create synthetic flows. This document also describes IFA zones, IFA reports, and IFA metadata. IFA does not require changes to protocol headers in order to collect metadata or analyze flows. IFA puts minimal requirements on switching silicon.

1.1 Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119].

IFA: Inband Flow Analyzer

MTU: Maximum Transmit Unit

1.2 Scope

This document describes IFA deployment, the type of traffic that is supported, header definitions, analytics, and data path functions.

IFA deployment involves defining an IFA zone and understanding the requirements in terms of traffic overhead and points of data collection. Given that IFA provides the ability to perform local analytics on the collected data, this document describes the scope of the analytics function as well. The scope of IFA is from an end station and/or ToR, through any/all nodes in the network, and terminating in a network switch and/or an end station.

IFA can create a synthetic stream of traffic and use it to collect metadata along the path. This sampled stream is later discarded. IFA can also insert metadata on a per packet basis in live traffic. Inband insertion of metadata can be done within the payload or via tail stamping.

This draft defines an identification mechanism using a dedicated protocol type in the IP header for identifying IFA.

1.3 Applicability

IFA is capable of providing traffic analysis in an encapsulation-agnostic manner. Simple TCP and UDP flows, as well as tunneled flows, can be monitored. IFA can be enabled on an end station, or it can be enabled just on network switches. Enabling IFA on an end station provides better scalability and visibility by monitoring intra end station or inter end station traffic. IFA performs best when there is hardware assistance for deriving the flow metadata in the data path. This document describes data path functions for IFA.

1.4 Motivation

The main motivation for IFA is to collect analyzed metadata from packets within a flow for a given application. The definition of the IFA header ensures that it works for any IP packet, and with minimal impact on hardware performance.

2. Requirements

IFA requirements are defined with operational efficiency, performance of the network, and cost of hardware in mind.

2.1 Encapsulation Requirements

IFA packets MUST be clearly marked and identifiable so that a networking element in the flow path can insert metadata or perform other IFA operations.

IFA packets need to be easily identified for performance reasons. IFA packet identification MUST be the same for all the IP packet types. This means that expensive hardware modifications are not needed for supporting new protocol types.

Since IFA packet processing is a data path function, the IFA header MUST keep the processing overhead minimal. Simple parsing in the switch hardware with localized read/write fields in IFA header will optimize the switch performance and cost.

A single IFA encapsulation MUST support IPv4 and IPv6 protocol types for tunneled and non-tunneled packets, preserving the fields used for load balancing hash computation.

IFA MAY support a checksum for the entire IFA metadata stack instead of a checksum per metadata element.

2.2 Operational Requirements

IFA MUST preserve the flow path across the network.

IFA MUST incur minimal traffic overhead.

IFA MUST provide an option to clone and truncate a packet to avoid disrupting the PMTU discovery of a network.

Cloning SHOULD be supported. Sampling of cloned traffic MUST be at a sampled ratio to keep the network overhead to a minimum.

IFA MUST provide the ability to insert metadata on cloned traffic.

IFA MUST provide the ability to insert metadata on live traffic.

IFA MAY provide the ability to specify checksum validation on the IFA header and metadata.

IFA MUST provide the ability to define a zone using hop count.

IFA MUST provide the ability for a networking element to perform metadata insertion in the payload.

IFA MAY provide the ability for networking element to insert metadata as tail stamping.

IFA MUST be able to support an IFA zone name space, also referred to as a global name space.

IFA MUST be able to support a per hop name space, also referred to as a local name space.

IFA MAY be able to support fragmentation of metadata. Fragmentation is needed to support a large number of hops in the network path.

2.3 Cost and Performance Requirements

The IFA header and metadata MUST be treated as foreign data present in the application data. IFA SHOULD be able to insert or strip the IFA header and metadata without modifying the layer 4 headers. This will help keep the cost of hardware down with no degradation in performance.

IFA MUST support the ability to clone and/or truncate, live traffic for IFA metadata insertion. This is needed for PMTU protocols to work within the IFA zone.

The IFA header MUST provide the ability to differentiate between a cloned packet and an original packet. This is needed for hardware to be able to identify and filter the cloned traffic at the edge of an IFA zone.

IFA encapsulation MUST provide mechanism to avoid impacting the parse depth of hardware for packet processing.

IFA MUST NOT require pre-allocation for reserving the space in a packet. The overhead of managing reserved space in a packet can result in performance degradation.

3. IFA Operations

IFA performs flow analysis, and possible actions on the flow data, inband. Once a flow is enabled for analysis, a node with the role of "Initiator" makes a copy of the flow or samples the live traffic flow, or tags a live traffic flow for analysis and data collection. Copying of a flow is done by sampling or cloning the flow. These new packets are representative packets of the original flow and possess the exact same characteristics as the original flow. This means that IFA packets traverse the same path in the network and same queues in the networking element as the original packet would. Figure 1 shows the IFA based Telemetry Framework. The terminating node is responsible for terminating the IFA flow by summarizing the metadata of the entire path and sending it to a Collector.

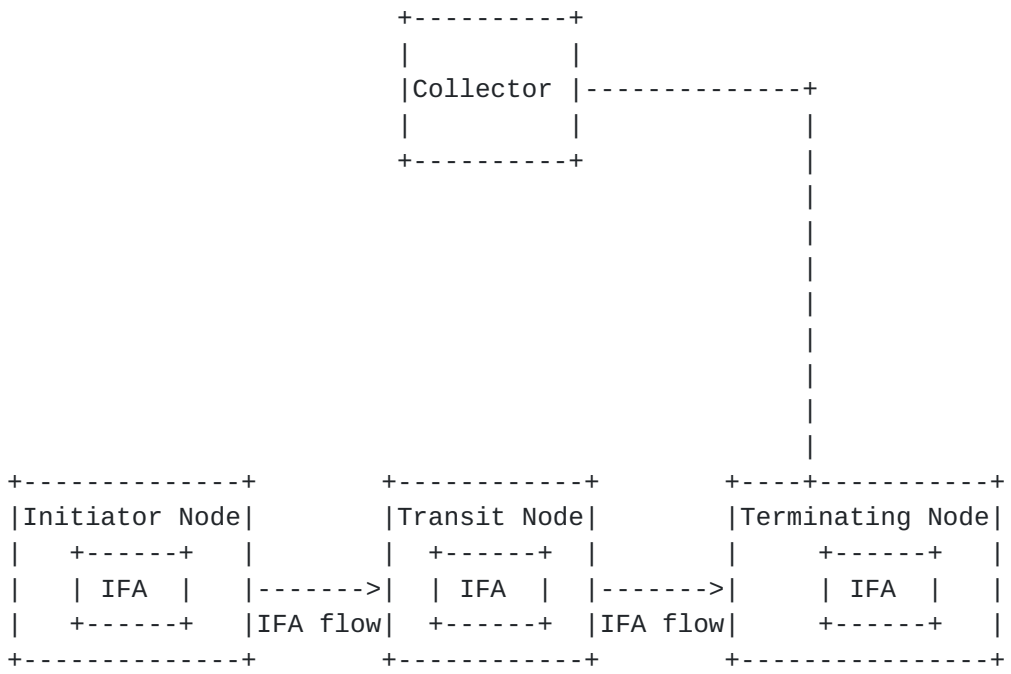


Figure 1 IFA Zone Framework without fragmentation

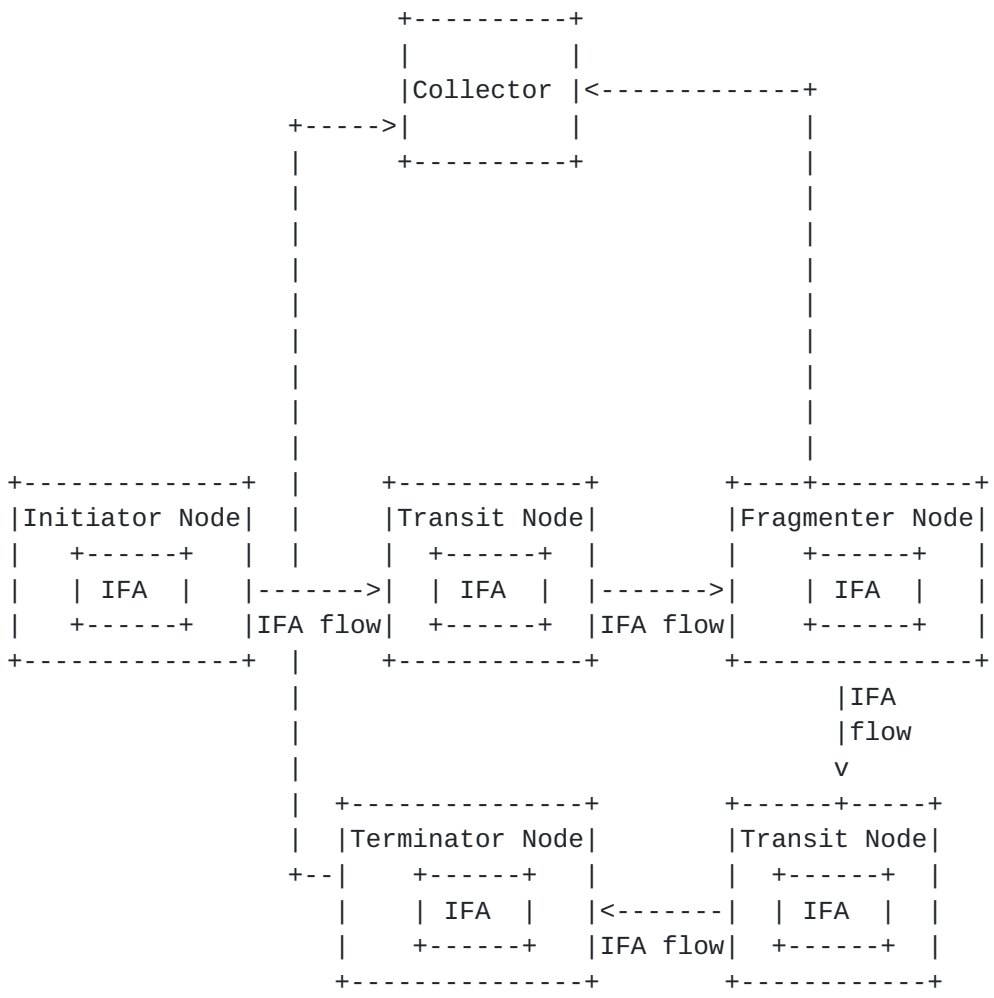


Figure 2 IFA Zone Framework with metadata fragmentation

3.1 IFA Zones

An IFA zone is the domain of interest where IFA monitoring is enabled. An IFA zone MUST have designated IFA function nodes. An IFA zone can be controlled by setting an appropriate TTL value in the L3 header. Initiating and Terminating function nodes are always at the edge of the IFA zone. Internal nodes in the IFA zone are always Transit function nodes.

3.2 IFA Function Nodes

There are three types of IFA functional nodes with respect to a specific or set of flows. Each node MAY perform metadata fragmentation function as well.

3.2.1 Initiating Function Node

An end station, a switch, or any other middleware can perform the IFA initiating function. It is advantageous to keep this role closest to

the application to maximize flow visibility. An IFA initiating function node performs the following functions for a flow:

- Samples the flow traffic of interest based on a configuration.
- Converts the traffic into an IFA flow by adding an IFA header to each sample.
- Updates the packet with initiating function node metadata.
- MAY mandate a specific template ID metadata by all networking elements.
- MAY mandate tail stamping of metadata by all networking elements.

3.2.2 Transit Function Node

An IFA transit node is responsible for inserting transit node metadata in the IFA packets in the specified flow.

3.2.3 Terminating Function Node

An IFA terminating node is responsible for the following for a flow:
- Inserts terminating node metadata in an IFA packet.
- Performs a local analytics function on one or more segments of metadata, e.g., threshold breach for residence time, congestion notifications, and so on.
- Filters an IFA flow in case of cloned traffic.
- Sends a copy or report of the packet to collector.
- Removes the IFA headers and forwards the packet in case of live traffic.

3.2.4 Metadata Fragmentation Function

There are cases where the size of metadata may grow too big for link MTU or path MTU, or where it imposes excessive overhead for the terminating function node to remove it. This is specially true in networks with a large number of hops between initiator function node and terminating function node. This is also true where the size of per hop metadata itself is large. For such cases, IFA defines a metadata fragmentation function. Metadata fragmentation function allows, removal of metadata from the packet and send a copy/report of the packet to collector. Correlation of metadata fragments and recreation of metadata stack for the entire flow path is done by the collector.

There is no dedicated node performing the metadata fragmentation function. As an IFA packet traverses the hops in an IFA zone, any node MAY detect the need to fragment the packet's metadata stack and perform metadata fragmentation.

Metadata fragmentation is done if the IFA header in the packet has "MDF" bit set and the current length of the metadata would exceed the

maximum length after the addition of metadata by the current node. A node MAY create a copy of the packet or create an IFA report, remove the existing metadata stack from the packet, insert its own metadata, and finally forward the packet. A node MAY also update the IFA MDF (Meta Data Fragment) header fragment identifier, current length, IP length, and IP header checksum.

The maximum length in an IFA header, if set to "0", MAY trigger the metadata fragmentation special function. This mechanism can be used to generate IFA reports at each hop and never insert metadata in the packet. If maximum length is set to "0", a node MAY ONLY create an IFA report or copy of the packet including its own metadata. A node MUST NOT update the IFA MD header current length, IP length, or insert metadata in the IFA packet. The node MUST increment the IFA MDF header fragment identifier field.

3.3 IFA Cloning, Truncation, and Drop

IFA allows cloning of live traffic. It is expected that cloned traffic will have the same network path characterization as the original traffic i.e. follow the same network path, use the same queues etc.

Cloned traffic can be truncated to accommodate the PMTU of the IFA zone.

Cloned traffic MUST be dropped by the terminating function node of the IFA zone.

3.4 IFA Header

The IFA header is described below. An experimental IP protocol number is used in the IP header to identify an IFA packet. The IP header protocol type field is copied into the IFA header NextHdr field for hardware to correctly interpret the layer 4 header.

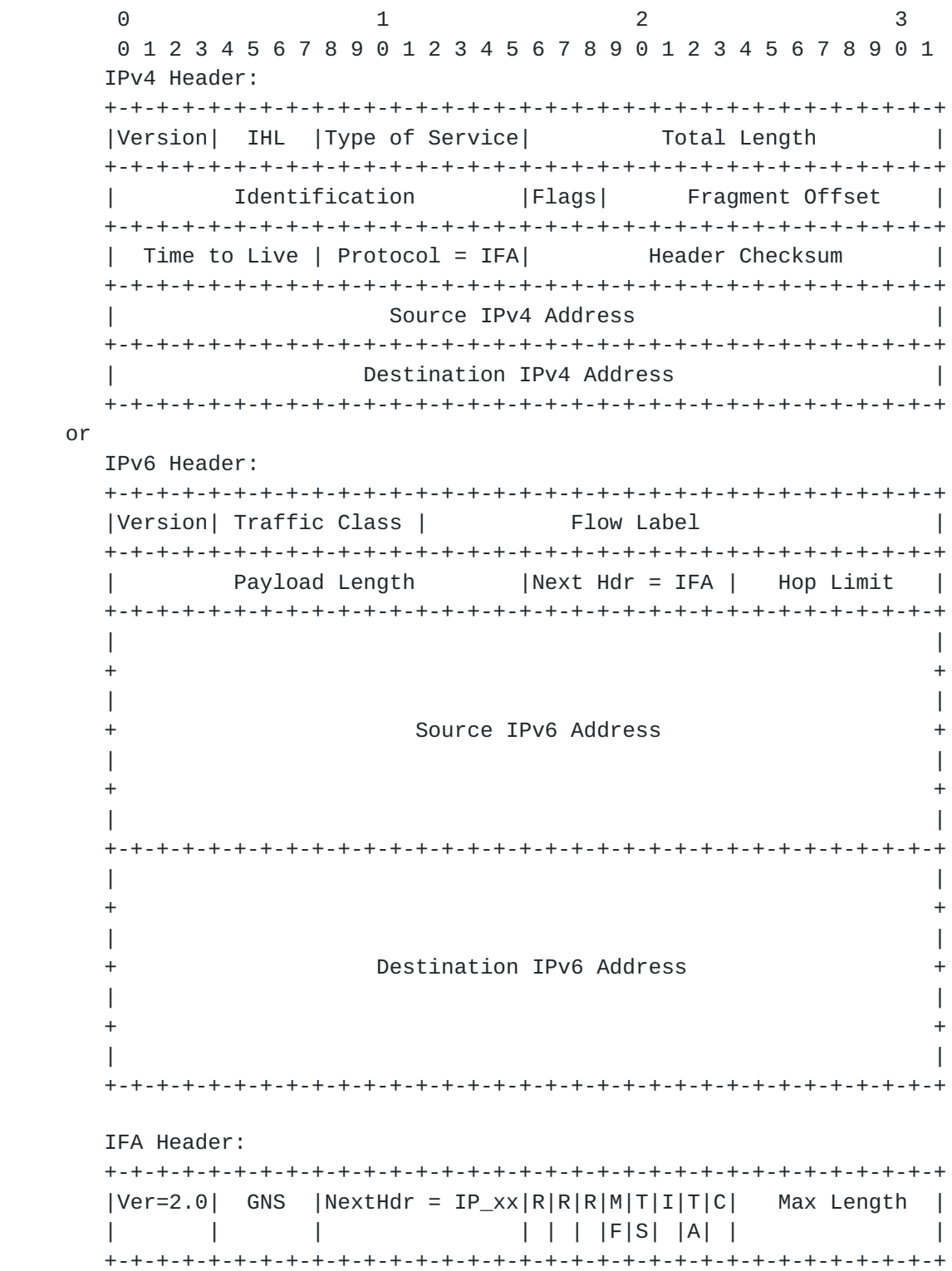


Figure 3 IFA 2.0 Header Format

(1) Version (4 bits) - Specifies the version of IFA header.

(2) GNS (4 bits) - Global Name Space. Specifies the IFA zone scoped name space for IFA metadata.

(3) Protocol Type (8 bits) - IP Header protocol type. This is copied from the IP header.

(4) Flags (8 bits)

0: R - Reserved. MUST be initialized to 0 on transmission and ignored on receipt.

1: R - Reserved. MUST be initialized to 0 on transmission and ignored on receipt.

2: R - Reserved. MUST be initialized to 0 on transmission and ignored on receipt.

3: MF - Metadata Fragment. Indicates the presence of the optional metadata fragment header. This header is inserted and initialized by the initiator node. If the MF bit is set, nodes in the path MAY perform fragmentation of metadata stack if the current length exceeds the maximum length.

4: TS - Tail Stamp. Indicates the IFA zone is requiring tail stamping of metadata.

5: I - Inband. Indicates this is live traffic. Strip and forward MUST be performed by the terminator node if this bit is set.

6: TA - Turn Around. Indicates that the IFA packet needs to be turned around at the terminating node of the IFA zone and sent back to source IP address. This bit MAY be used for probe packets where probes are collection bidirectional information in the network. This is same as echo request and echo reply. A packet MAY be generated with TA bit set and collects metadata in one direction and after it is turned around by the terminating function node, collects metadata in the reverse direction.

7: C - Checksum - Indicates the presence of the optional checksum header. The checksum MUST be computed and updated for the IFA header and metadata at each node that modified the header and/or metadata. A node MAY perform checksum validation before updating the checksum.

(5) Max Length (8 bits) - Specifies the maximum allowed length of the metadata stack in multiples of 4 octets. This field is initialized by the initiator node. Each node in the path MUST compare the current length with the max length, and if the current length equals or exceeds the max length, the transit nodes MUST stop inserting

metadata.

3.4.1 IFA Metadata Header

The IFA metadata header is always present.

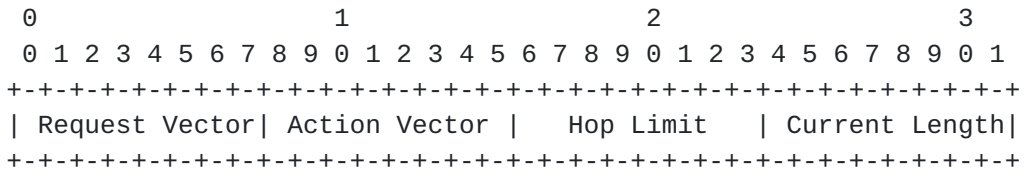


Figure 4 IFA Metadata Header Format

Request Vector (8 bits) - This vector specifies the presence of fields as specified by GNS. Fields are always 4-octet aligned. This field can be made extensible by defining a new GNS for an IFA zone.

Action Vector (8 bits) - This vector specifies node-local or end-to-end action on the IFA packets.

Hop Limit (8 bits) - Specifies the maximum allowed hops in an IFA zone. This field is initialized by the initiator node. The hop limit MUST be decremented at each hop. If the incoming hop limit is 0, current nodes MUST NOT insert metadata. A value of 0xFF means that the Hop limit check MUST be ignored.

Current Length (8 bits) - Specifies the current length of the metadata in multiples of 4 octets.

3.4.2 IFA Checksum Header

The IFA checksum header is optional. Presence of the checksum header is indicated by the C bit in the flags field of the IFA header.

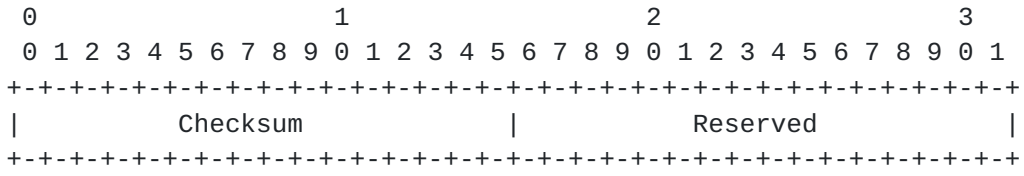


Figure 5 IFA Checksum Header Format

Checksum (16 bits) - The checksum covers the IFA header and metadata

stack. Initiator function node MAY compute the full checksum including IFA header and metadata. Other nodes MAY compute delta checksum for the inserted/deleted metadata.

Reserved (16 bits) - Reserved. MUST be initialized to 0 on transmission and ignored on receipt.

3.4.3 IFA Metadata Fragmentation (MF) Header

The IFA metadata fragmentation (MF) header is optional. Presence of the fragmentation header is indicated by the MF bit in the flags field of the IFA header.

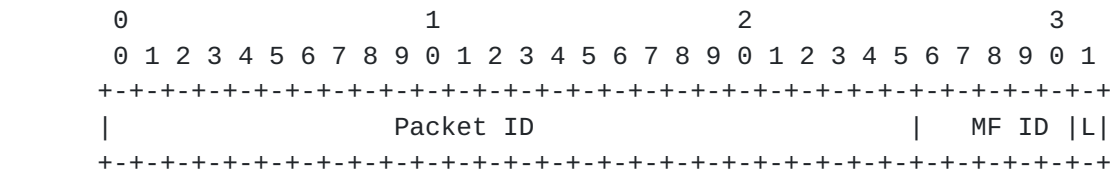


Figure 6 IFA MF Header Format

Packet ID (26 bits) - Packet identification value generated by the initiator node. This value is node scoped.

Metadata Fragment ID (5 bits) - The initiator MUST initialize this value to 0. A node performing metadata fragmentation function MUST increment the value by 1.

L (1 bit) - This bit is set by the node creating the last metadata fragment. This will ALWAYS be the terminating function node. If incoming hop limit is "0", terminating function node will still generate copy/report of the packet and MUST set L bit. Collector MUST implement mechanism to recover from lost packets/reports with L bit set.

The MF header is a fixed overhead of 4 octets per packet. A network operator MUST identify the need for using IFA metadata fragmentation. The following network conditions can be considered:

- If an IFA packet may exceed the link or path MTU of the flow path

- If there are large number of hops in a flow path and MAY trigger link or path MTU breach

- If the length of metadata creates excessive overhead for terminating function node to delete the metadata.

- If each hop needs to generate its own IFA report (postcard mechanism)

With 26 bits of packet id, a maximum datagram lifetime (MDL) of 3 seconds, and an average Internet mix (IMIX) packet size of 512 bytes, we get 183.25 Gbps of IFA traffic bit rate per node before the packet identifier wraps around. The collector can use [device id, packet id, MF id, L] to rebuild the fragmented packet.

5 bits of MF id will support 32 metadata fragments.

3.5 IFA Metadata

The IFA metadata is the information inserted by each hop after the IFA header. The IFA metadata can be inserted at the following offsets:

- Payload Stamping: Immediately after the layer 4 header. This is the default setting.
- Tail Stamping: After the end of the packet. This is controlled by the TS bit in the flags field of the IFA header.

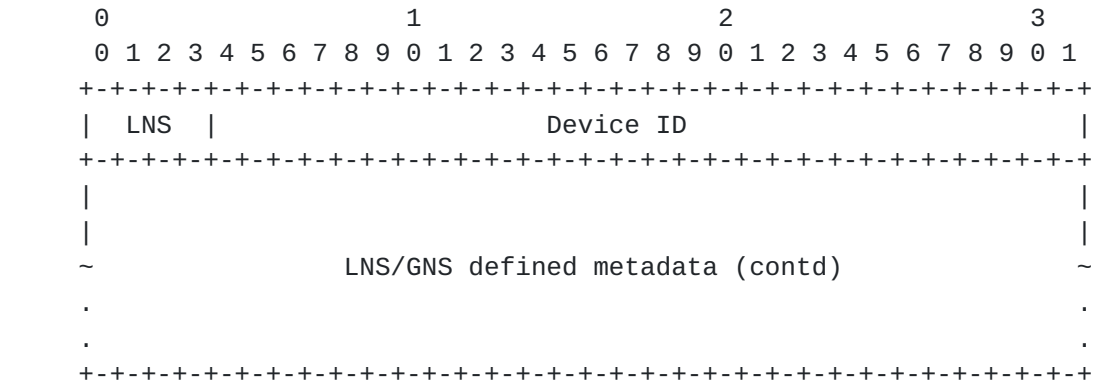


Figure 7 IFA Metadata Format

The IFA metadata header contains a set fields as defined by the name space identifier. Two types of name space identifiers are proposed.

3.5.1 Global Name Space (GNS) Identifier

A Global Name Space (GNS) is specified in the IFA header by the initiator node. The scope of a GNS is an IFA zone. All networking elements in an IFA zone MUST insert metadata as per the GNS ID specified in the IFA header. This is defined as the "Uniform Mode" of deployment.

A GNS value of 0xF indicates that metadata in an IFA zone is defined by the LNS of each hop.

The advantage of using the uniform mode is having a simple and uniform metadata stack. This means less load on a collector for parsing.

The disadvantage is that metadata fields are supported based on the least capable networking element in the IFA zone.

3.5.2 Local Name Space (LNS) Identifier

A Local Name Space (LNS) is specified in the metadata header. A GNS value of 0xF in the IFA header indicates the presence of an LNS. This is defined as the "Non-uniform Mode" of deployment.

A switch pipeline MUST parse the GNS field in the IFA header. The parsing result will dictate the name space ID that the hop needs to comply with.

The advantage of using the non-uniform mode is having a flexible metadata stack. This allows each hop to include the most relevant data for that hop.

The disadvantage is more complex parsing by a collector.

3.5.3 Device ID

A 28-bit unique identifier for the device inserting the metadata. If a GNS other than 0xF is present, then the device ID can be expanded to a 32 bit value. This is to support including an IPv4 loopback address as a Device ID.

3.6 IFA Network Overhead

A common problem associated with inserting metadata on a per packet per flow basis is the amount of traffic overhead on the network. IFA 2.0 is defined to minimize the overhead on the network.

```
IFA Base Header          : 4 octets
IFA Metadata Header      : 4 octets
IFA Checksum Header      : 4 octets
IFA Fragmentation Header : 4 octets


Minimum Overhead:
   IFA header          : 4 octets
```

IFA Metadata Header : 4 octets

Total Min Overhead : 8 octets per packet

3.7 IFA Analytics

There are two kinds of actions considered in this proposal.

(1) Action Bit MAP in IFA Header - This is encoded in the IFA header. Each node in the path MAY use the action bitmap to insert or not insert the metadata based on exceeding a locally-specified threshold. Not inserting the metadata is indicated by setting the field value to -1 (all 1s).

(2) Terminating Node Actions - A terminating node may decide to perform threshold or other actions on the set of metadata in the packet. This information is not encoded in the IFA header.

3.8 IFA Packet Format

The IFA header is treated as a layer 3 extension header. IFA header and metadata stack length is reflected in IP total length field. IPv6 extension headers are ordered. The IFA header MUST be the last extension header in the IPv6 extension header chain. Similarly in case of IPv4 AH/ESP/WESP extension headers, IFA header MUST be the last extension header.

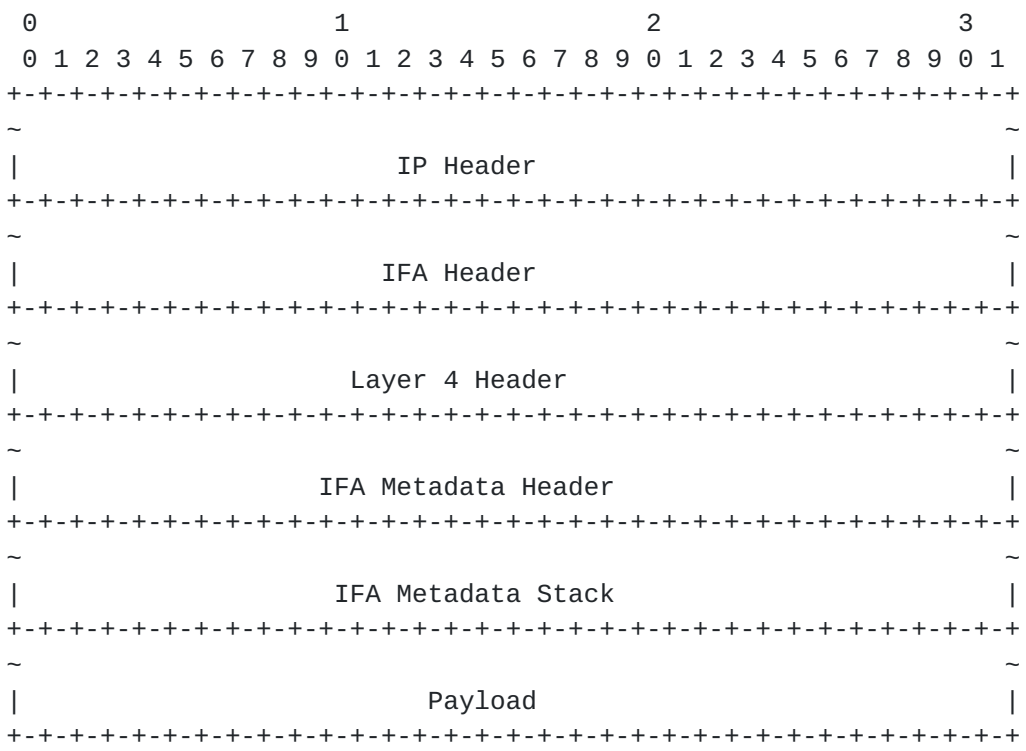


Figure 8 IFA Packet Format

3.8.1 IFA Packet Format with TS Flag Set

In case the Tail Stamp flag is set in the IFA header, the IFA metadata header and metadata stack are inserted at the end of the packet just before the FCS. Each node inserts metadata at the bottom of IFA metadata stack.

One of the key advantages of using TS is to support legacy devices and/or appliances that need to look at the layer 5 data. The IP length and IP header checksum are updated at each hop inserting metadata. This is the same as without the TS flag.

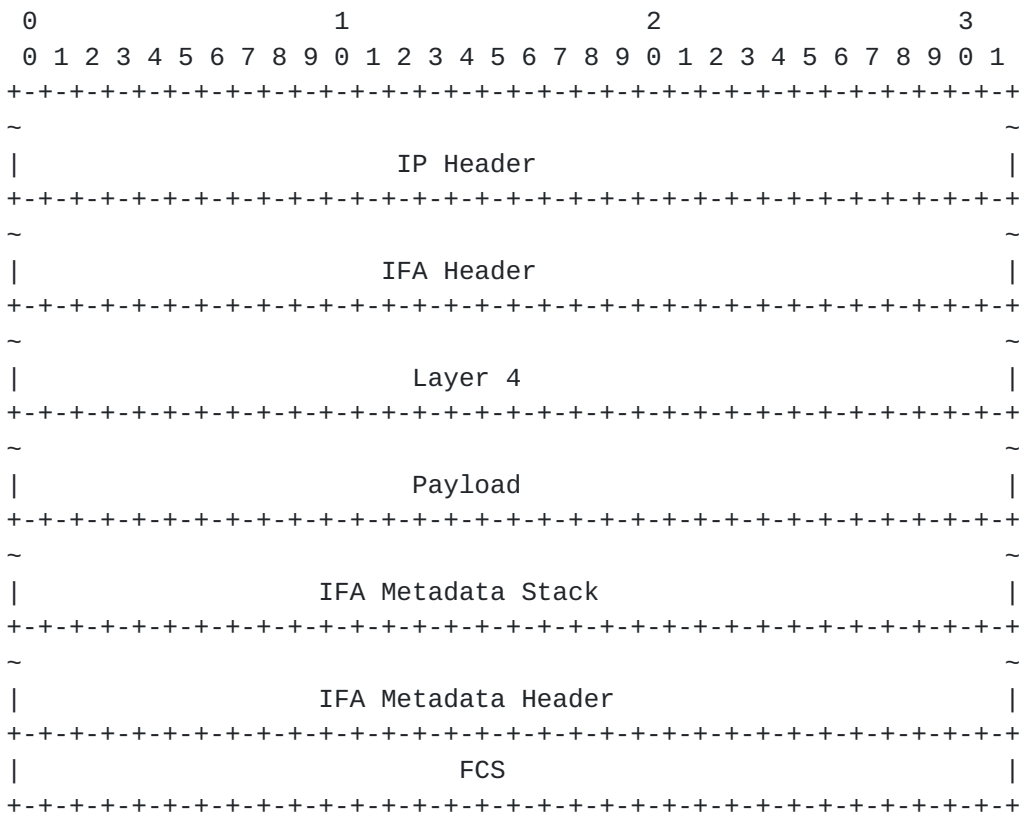
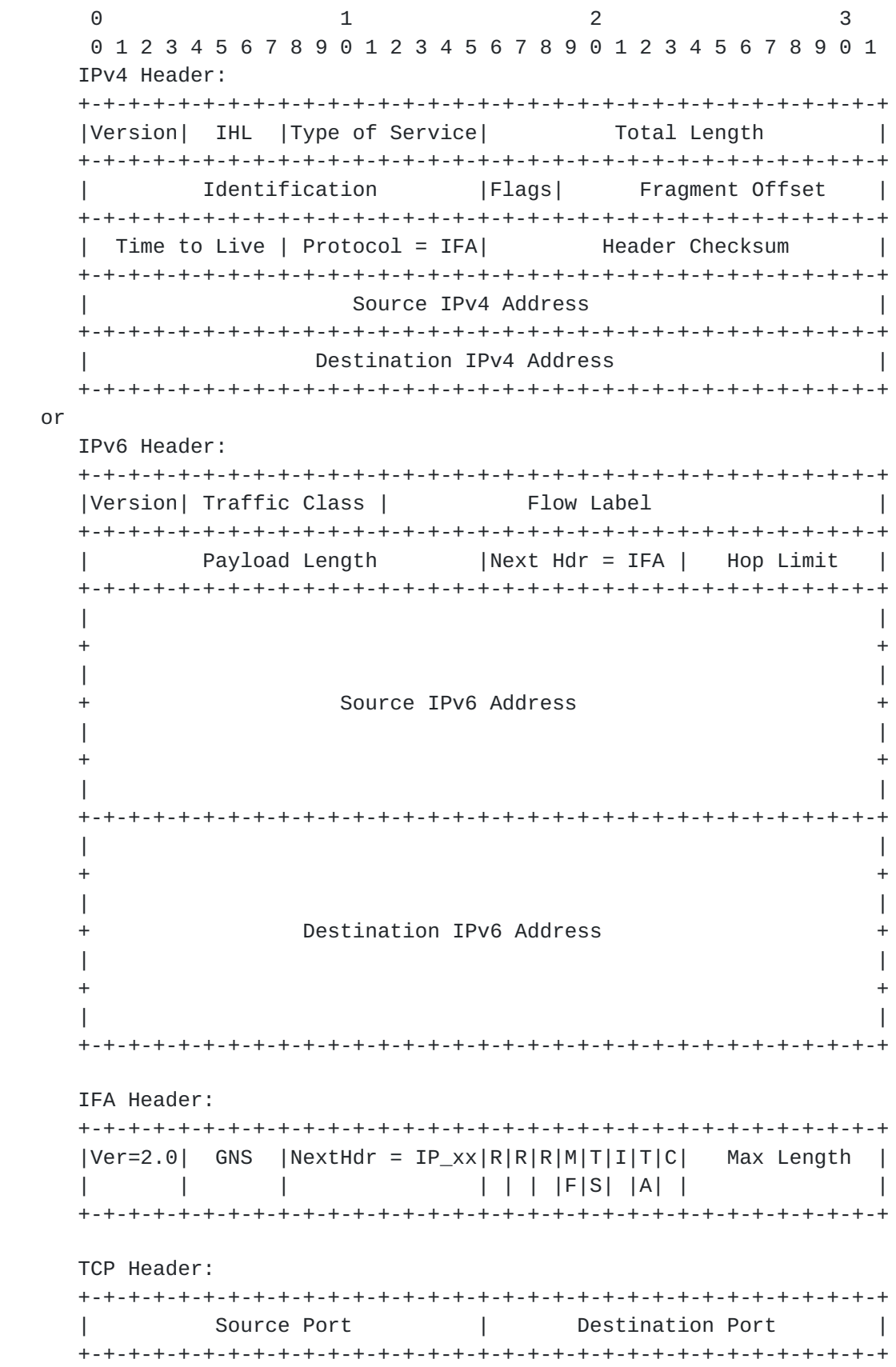


Figure 9 IFA Packet Format with TS

3.8.1 TCP/UDP Packet


0 1 2 3
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
IPv4 Header:
Version
IHL
Type of Service
Total Length
Identification
Flags
Fragment Offset
Time to Live
Protocol = IFA
Header Checksum
Source IPv4 Address
Destination IPv4 Address
or
IPv6 Header:
Version
Traffic Class
Flow Label
Payload Length
Next Hdr = IFA
Hop Limit
Source IPv6 Address
Destination IPv6 Address
IFA Header:
Ver=2.0
GNS
NextHdr = IP_xx
R R R M T I T C
Max Length
TCP Header:
Source Port
Destination Port

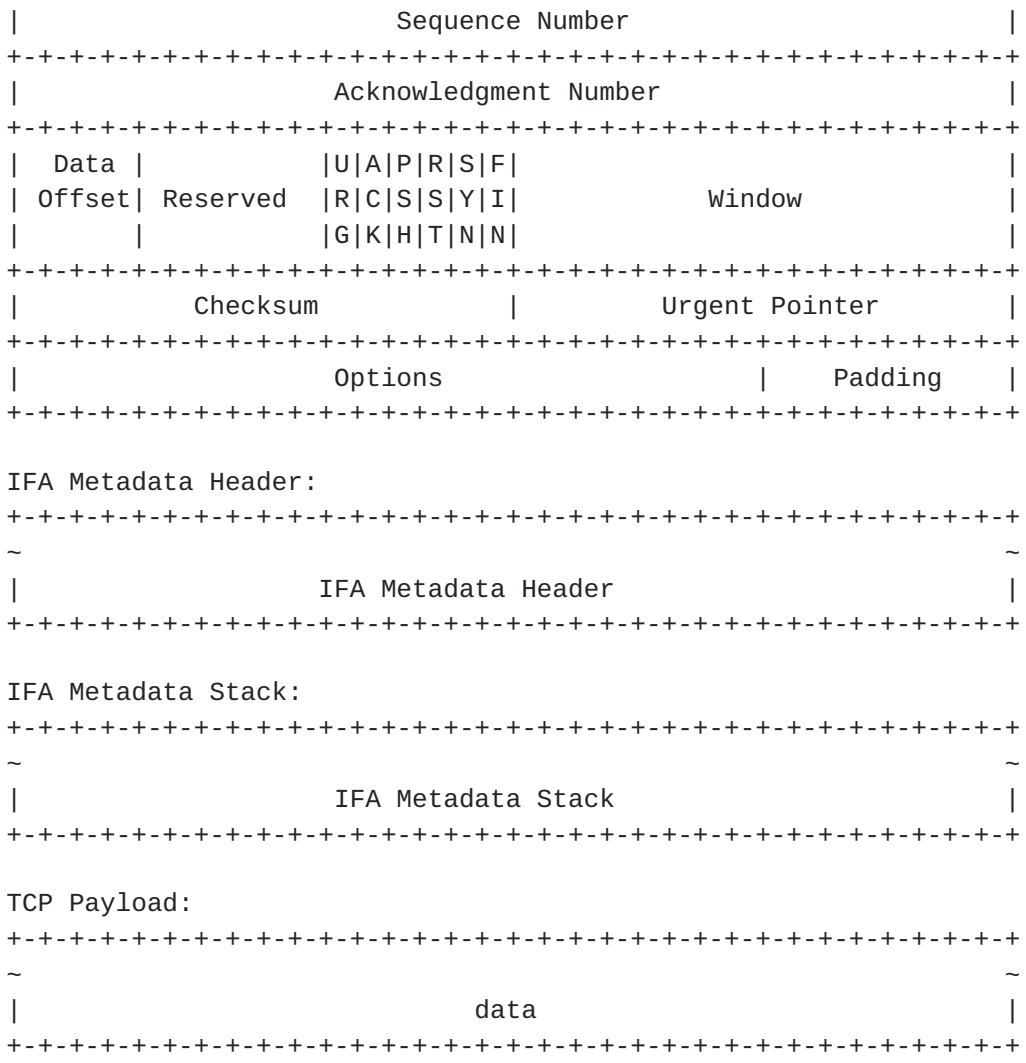
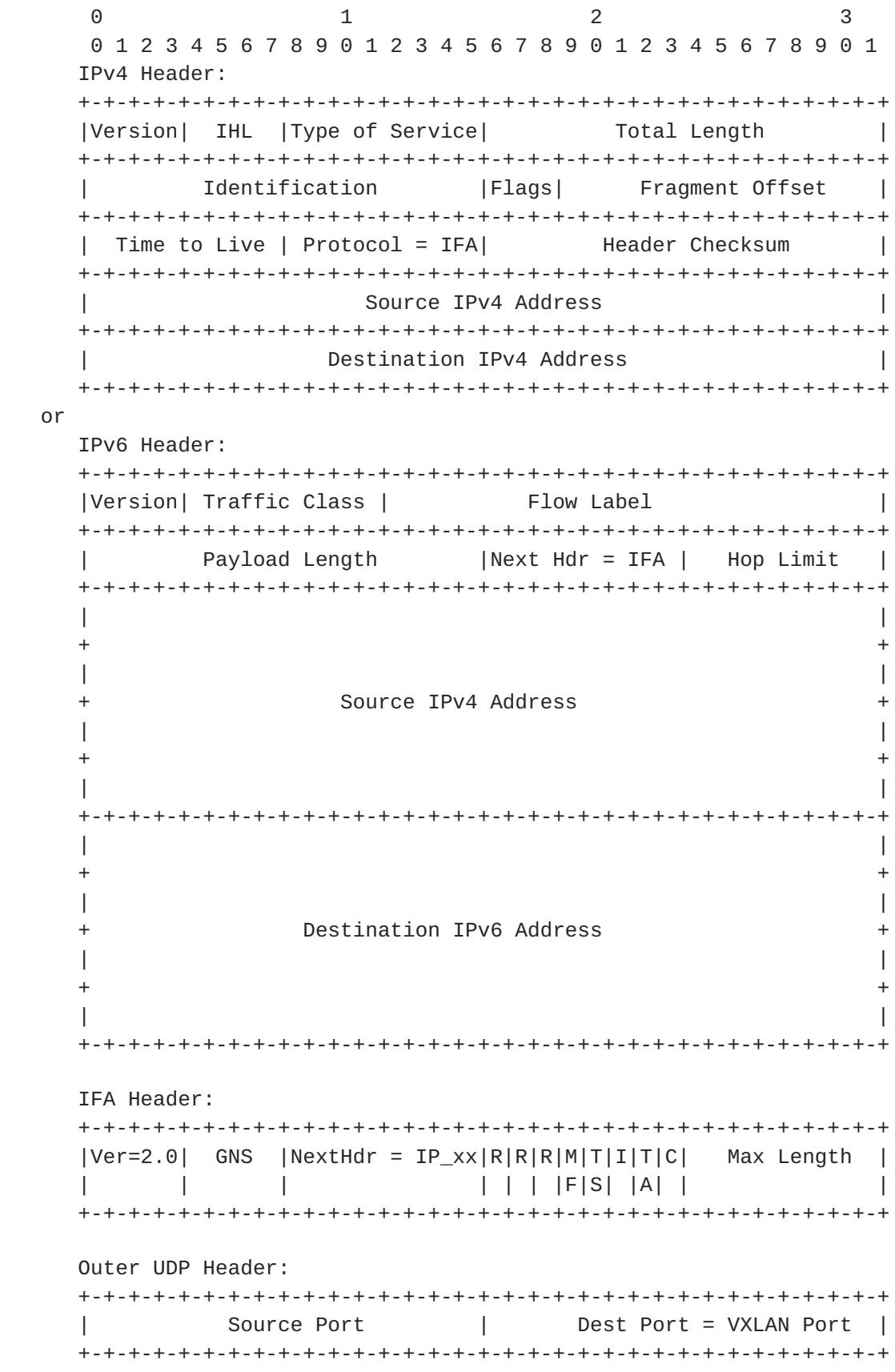


Figure 10 TCP/UDP IFA Packet Format

3.8.2 VxLAN Packet


0 1 2 3
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
IPv4 Header:
Version
IHL
Type of Service
Total Length
Identification
Flags
Fragment Offset
Time to Live
Protocol = IFA
Header Checksum
Source IPv4 Address
Destination IPv4 Address
or
IPv6 Header:
Version
Traffic Class
Flow Label
Payload Length
Next Hdr = IFA
Hop Limit
Source IPv4 Address
Destination IPv6 Address
IFA Header:
Ver=2.0
GNS
NextHdr = IP_xx
R
R
R
M
T
I
T
C
Max Length
F
S
A
Outer UDP Header:
Source Port
Dest Port = VXLAN Port

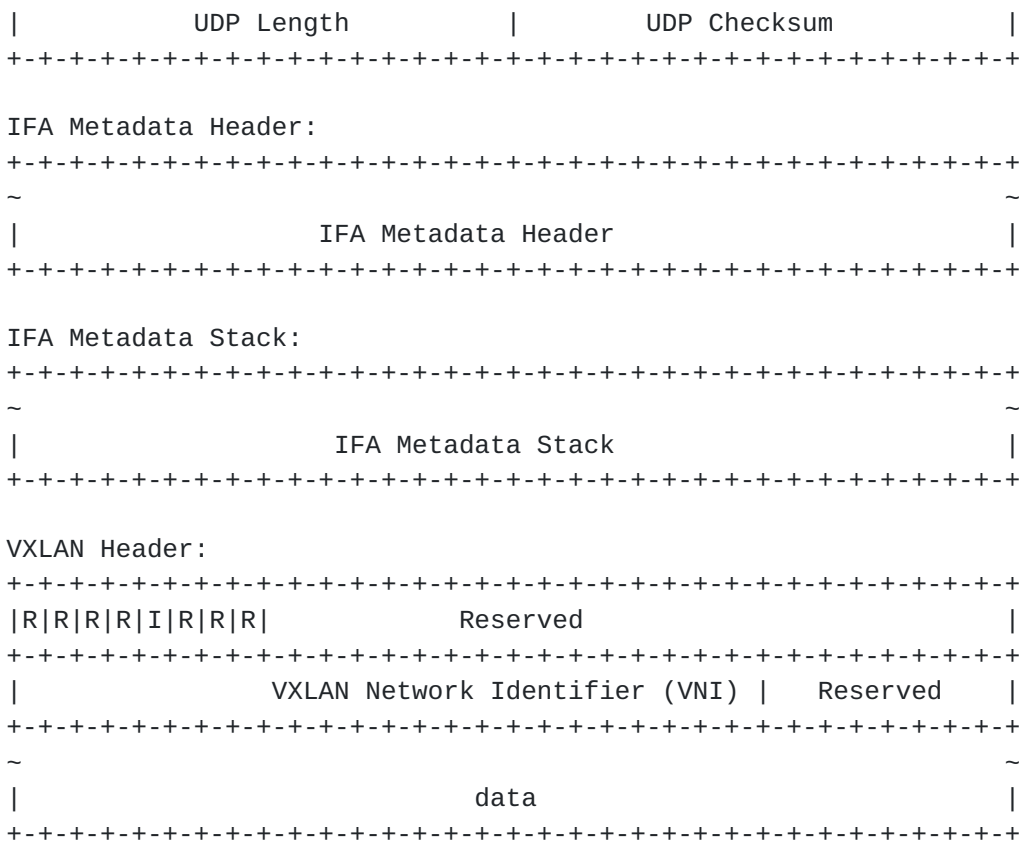
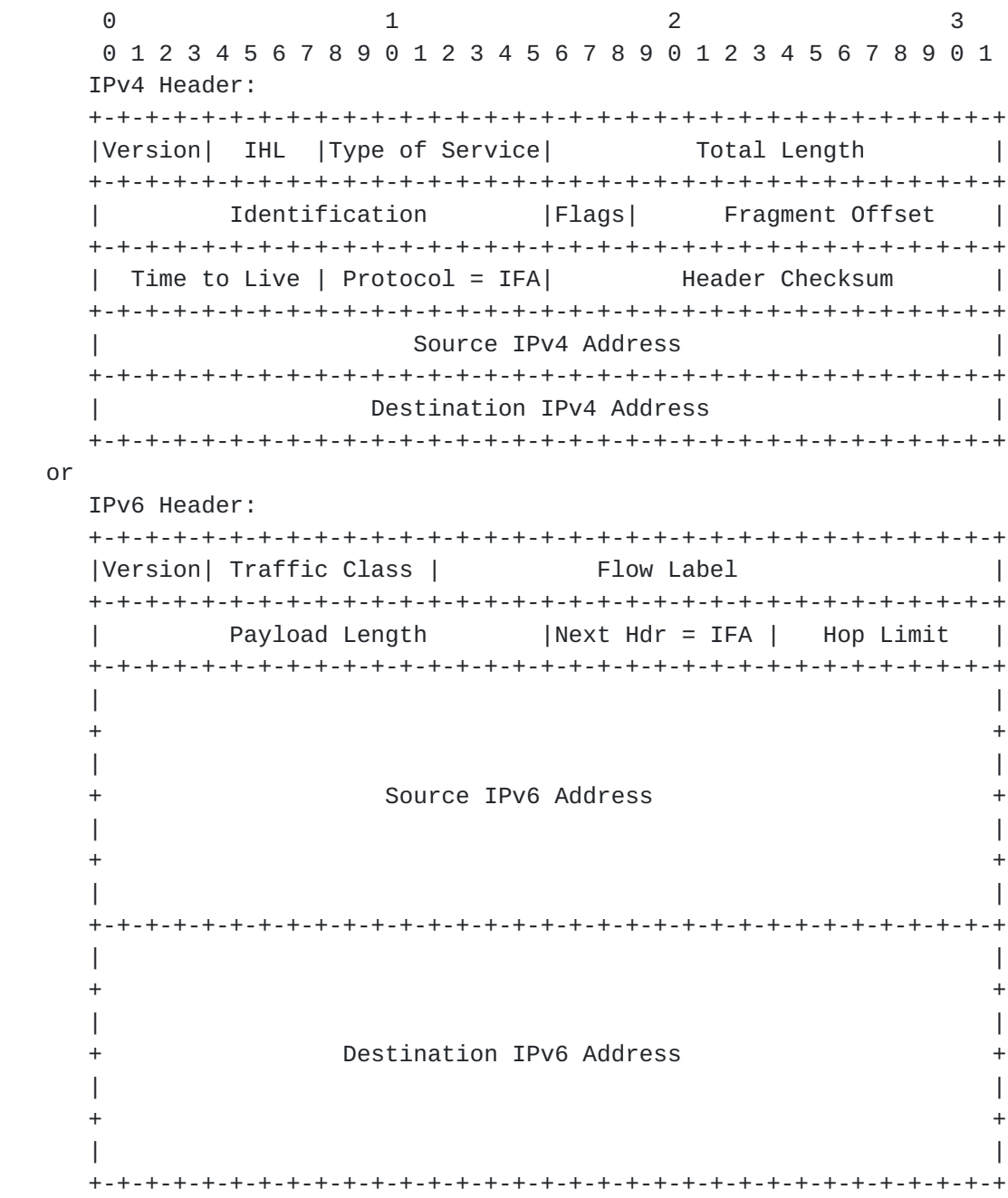


Figure 11 VxLAN IFA Packet Format

3.8.3 GRE Packet


0 1 2 3
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
IPv4 Header:
Version
IHL
Type of Service
Total Length
Identification
Flags
Fragment Offset
Time to Live
Protocol = IFA
Header Checksum
Source IPv4 Address
Destination IPv4 Address
or
IPv6 Header:
Version
Traffic Class
Flow Label
Payload Length
Next Hdr = IFA
Hop Limit
Source IPv6 Address
Destination IPv6 Address
IFA Header:
Ver=2.0
GNS
NextHdr = IP_xx
R
R
R
M
T
I
T
C
Max Length
GRE Header:
C
Reserved0
Ver
Protocol Type

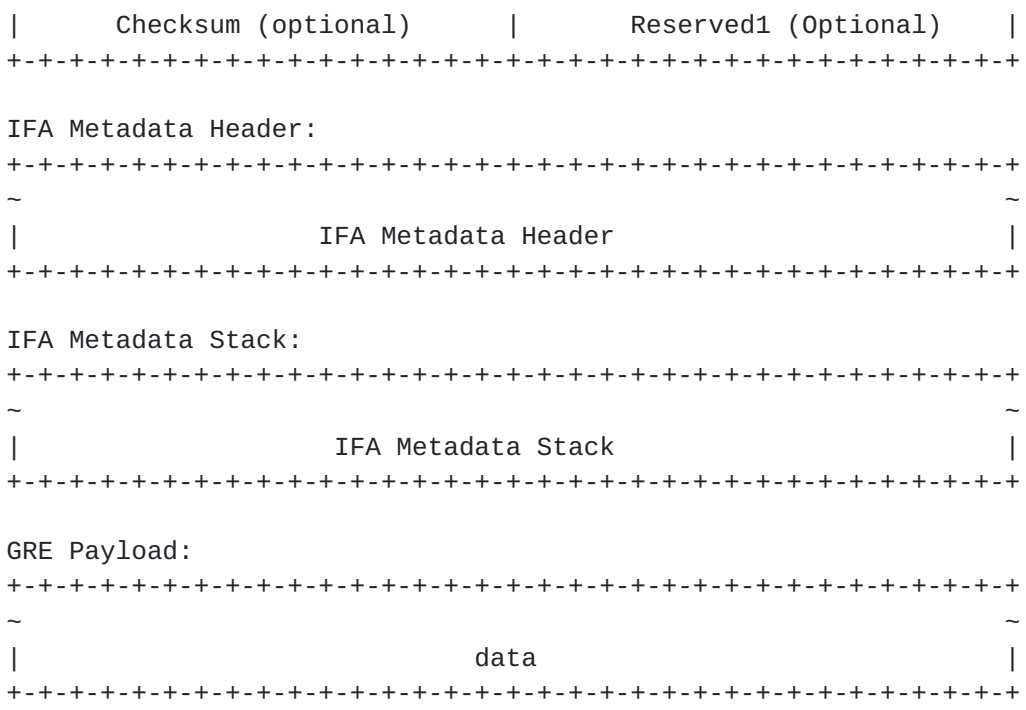
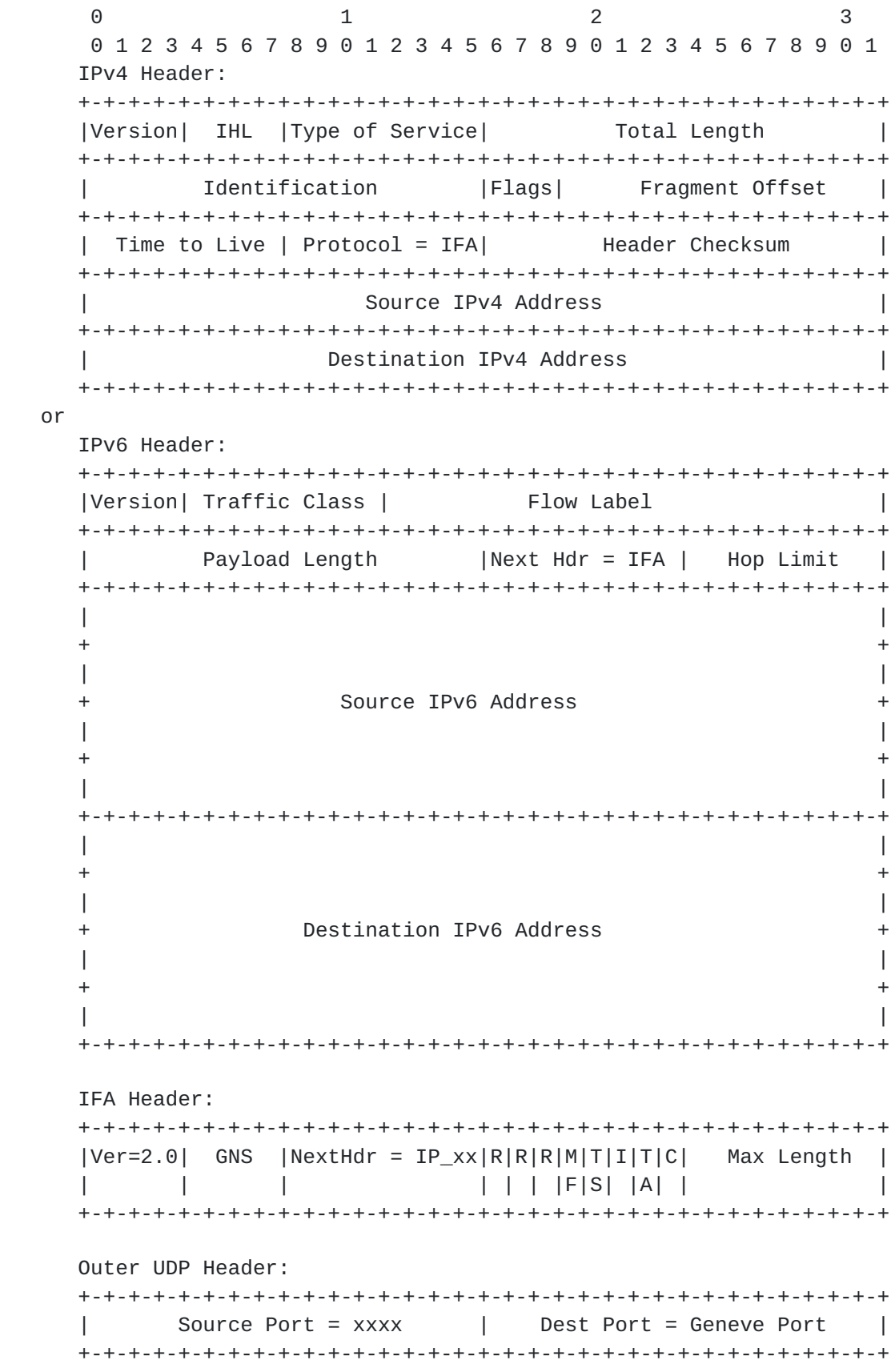


Figure 12 GRE IFA Packet Format

3.8.4 Geneve Packet


0 1 2 3
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
IPv4 Header:
Version
IHL
Type of Service
Total Length
Identification
Flags
Fragment Offset
Time to Live
Protocol = IFA
Header Checksum
Source IPv4 Address
Destination IPv4 Address
or
IPv6 Header:
Version
Traffic Class
Flow Label
Payload Length
Next Hdr = IFA
Hop Limit
Source IPv6 Address
Destination IPv6 Address
IFA Header:
Ver=2.0
GNS
NextHdr = IP_xx
R R R M T I T C
Max Length
Outer UDP Header:
Source Port = xxxx
Dest Port = Geneve Port

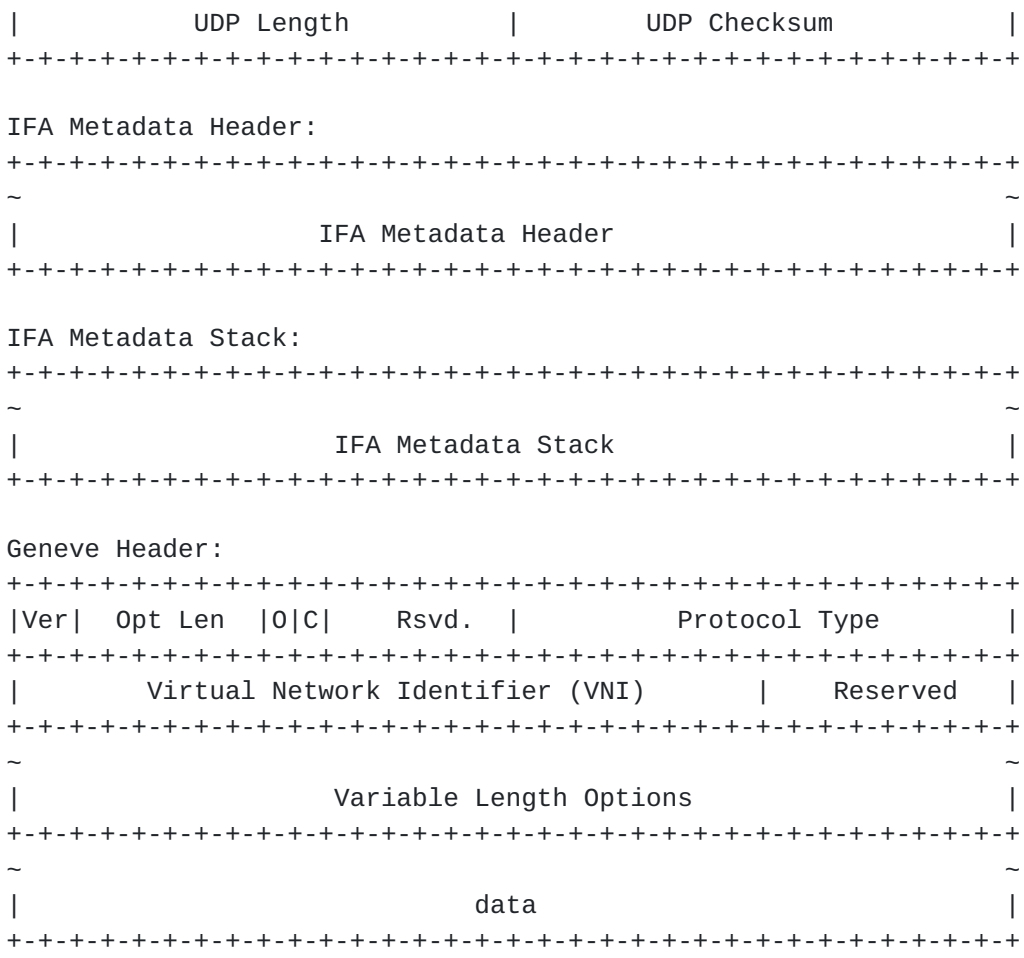
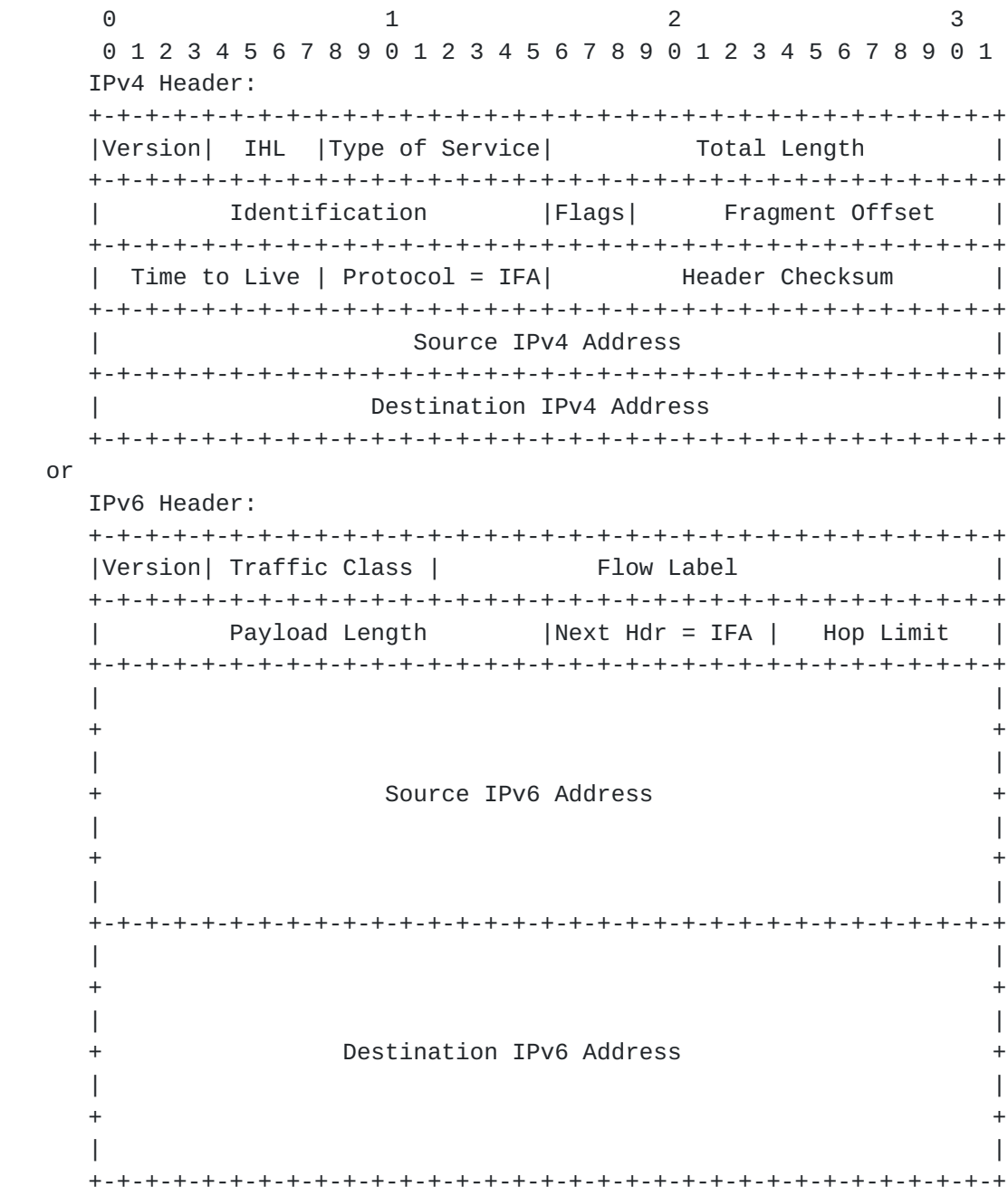


Figure 13 Geneve IFA Packet Format

3.8.5 IPinIP Packet


0 1 2 3
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
IPv4 Header:
Version
IHL
Type of Service
Total Length
Identification
Flags
Fragment Offset
Time to Live
Protocol = IFA
Header Checksum
Source IPv4 Address
Destination IPv4 Address
or
IPv6 Header:
Version
Traffic Class
Flow Label
Payload Length
Next Hdr = IFA
Hop Limit
Source IPv6 Address
Destination IPv6 Address
IFA Header:
Ver=2.0
GNS
NextHdr = IP_xx
R
R
R
M
T
I
T
C
Max Length
F
S
A
Inner IP Header:
IPv4 or IPv6 Header

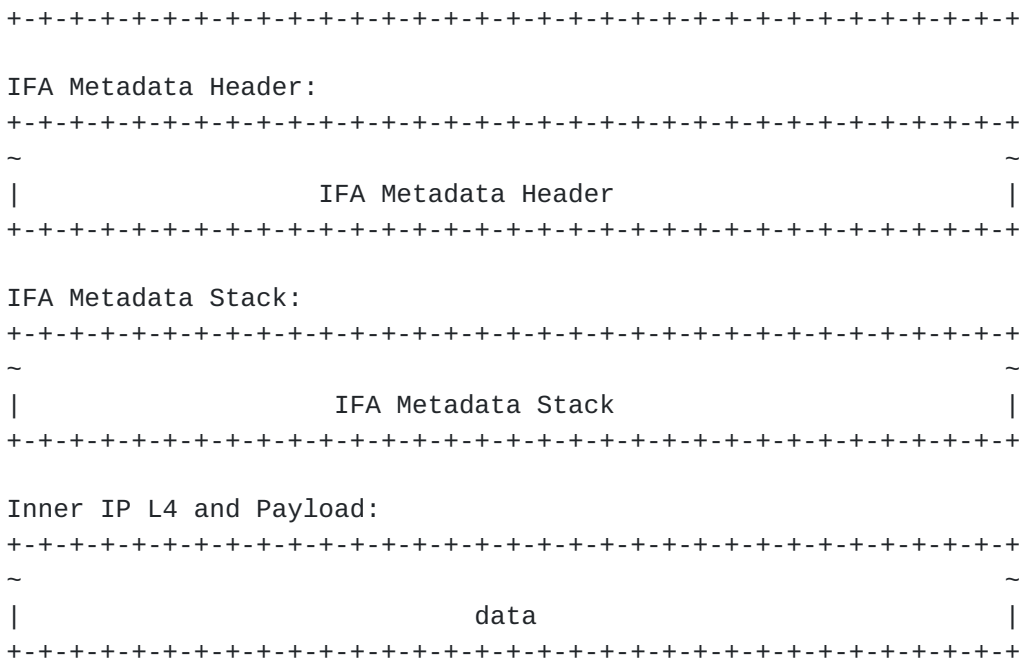


Figure 14 IPinIP IFA Packet Format

3.8.6 IPv6 Extension Headers with IFA

The IFA header is always the last extension header in the IPv6 extension header chain. The last extension header's next header field is stored in the IFA next header field and is replaced by the IFA protocol value.

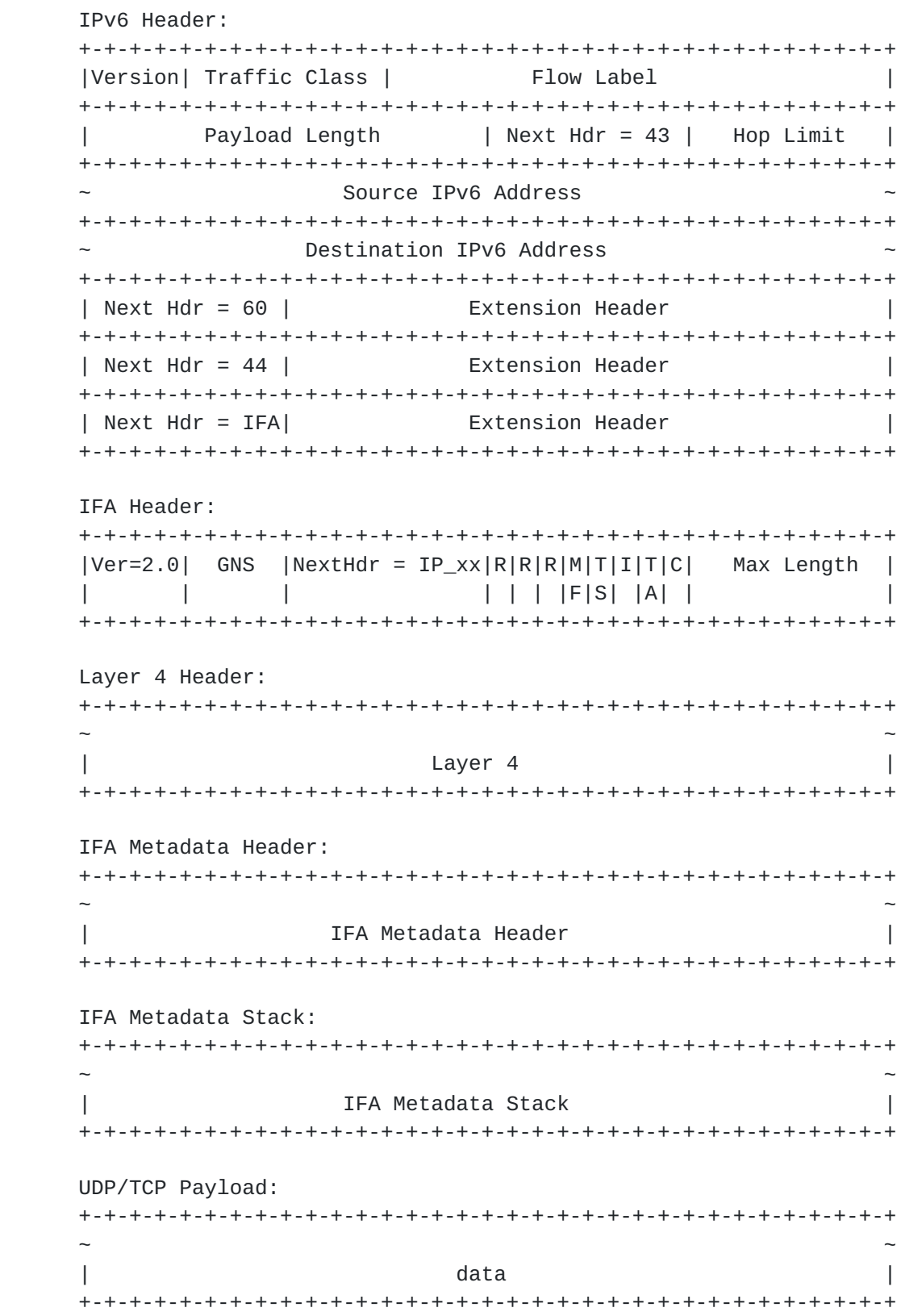
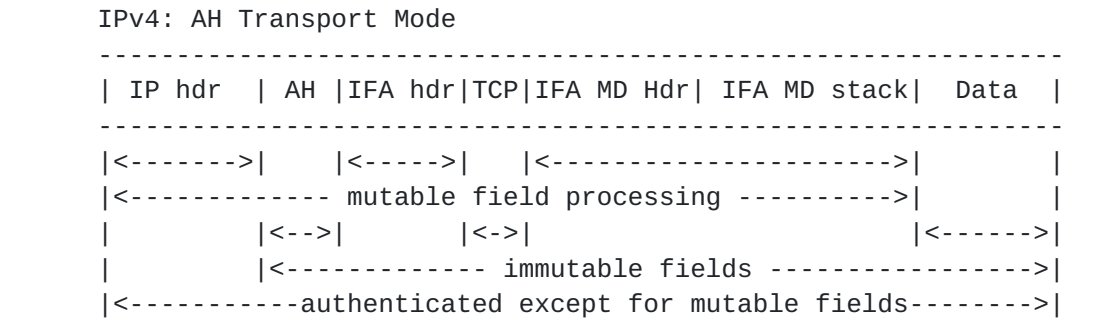


Figure 15 IPv6 Extension Header with IFA Packet Format

3.8.6 IP AH/ESP/WESP Packet

An AH, ESP, or WESP header is treated as a chained header in IPv4. The IPv4 protocol field is replaced by the AH/ESP/WESP protocol value and the IPv4 protocol field value is stored in the AH/ESP/WESP next header field.

The IFA header is ALWAYS placed as the last header in a header chain. In case of ESP/WESP where layer 4 and payload is encrypted, IFA metadata stack is placed immediately after IFA header.



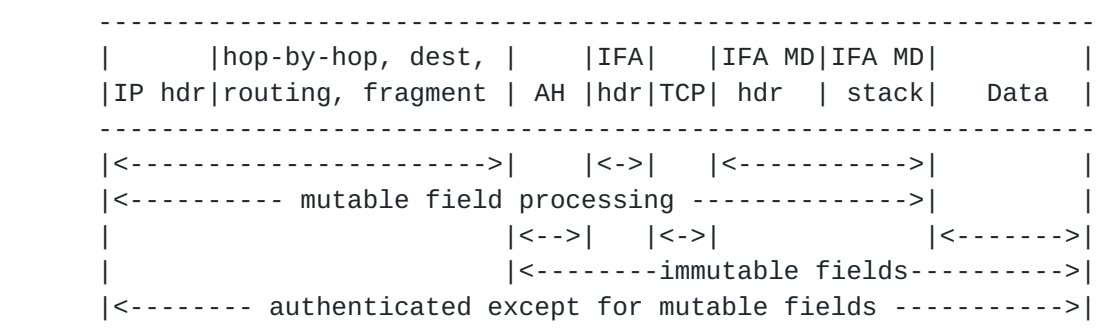
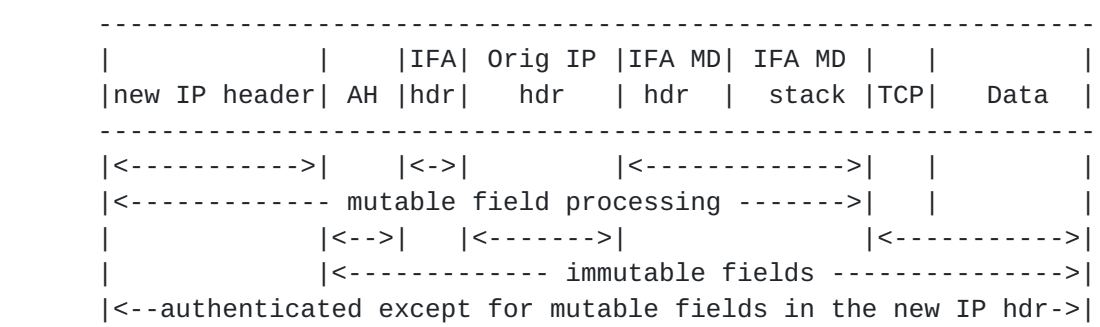


Figure 16 IP AH Transport Mode IFA Packet Format



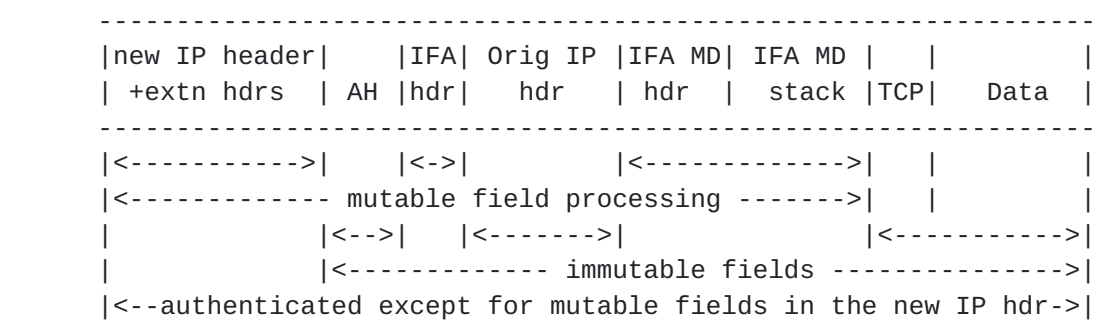


Figure 17 IP AH Tunnel Mode IFA Packet Format

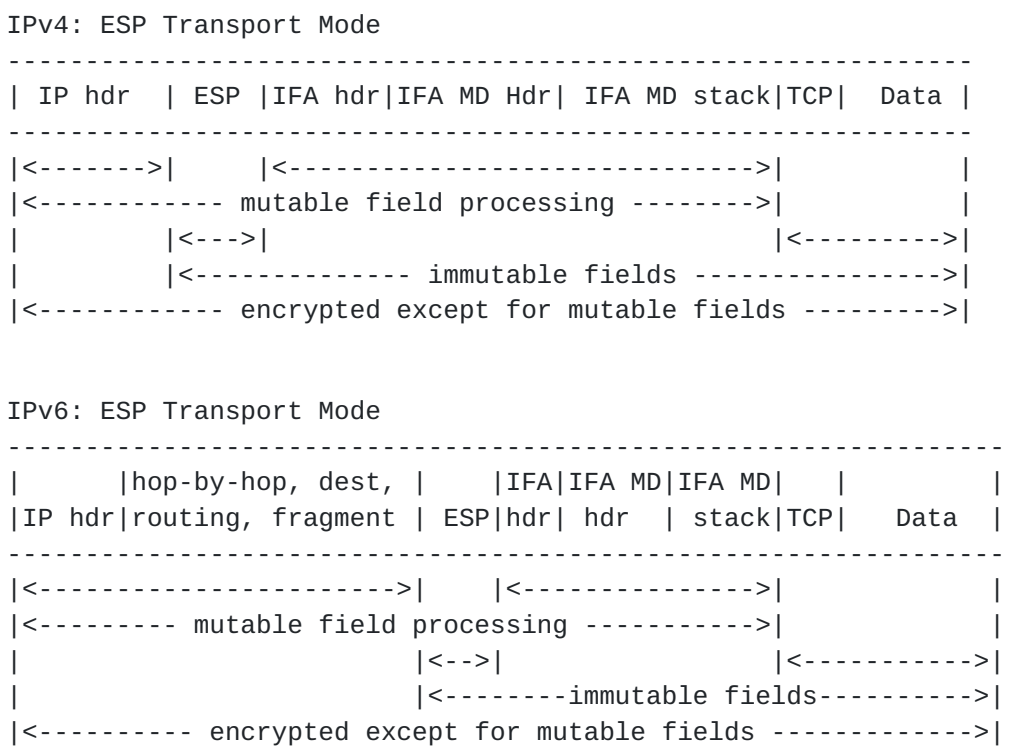


Figure 18 IP ESP Transport Mode IFA Packet Format

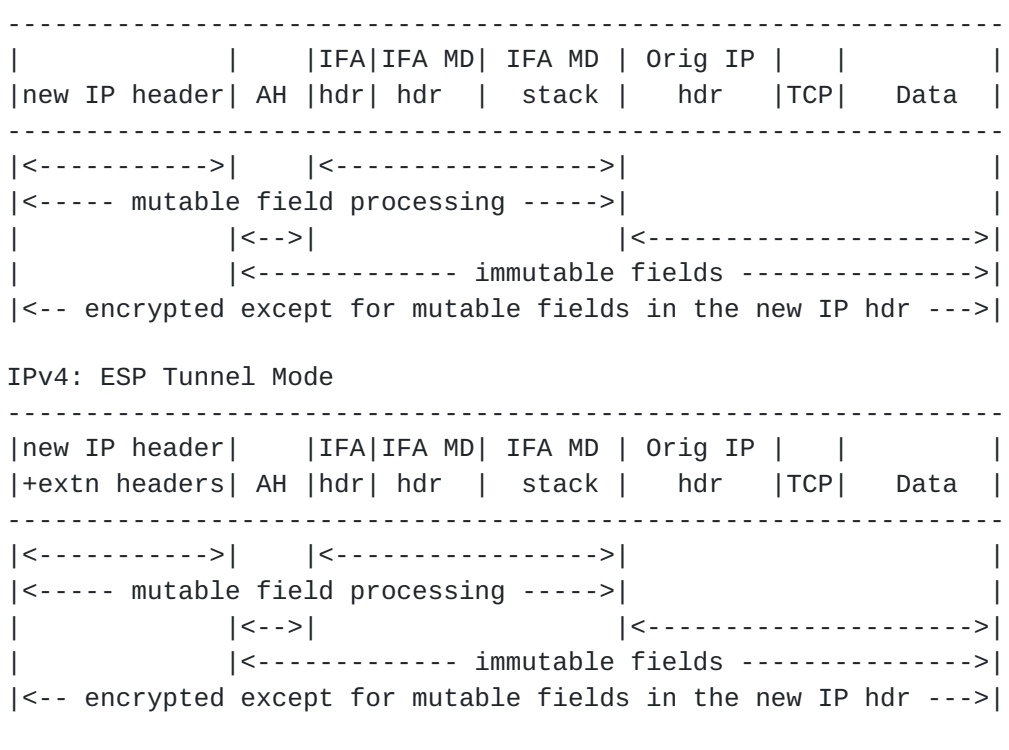


Figure 19 IP AH Tunnel Mode IFA Packet Format

3.9 IFA Load Balancing

IFA changes the IP protocol field value to the IFA protocol number. The IP protocol field value is included in the hash computation. This will impact load balancing of flows.

The forwarding plane MUST support reading the IP protocol field value stored in the IFA NextHDR field for hash computation.

The layer 4 header is available at a fixed offset from the IFA header and is available for hash computation.

Hash computation based on the layer 4 payload will depend on the length of the IFA metadata stack present.

4. Interoperability Considerations

Version 2.0 of this protocol specification is not backward compatible with version 1.0.

5. Security Considerations

A successful attack on an OAM protocol can prevent the detection of failures or anomalies, or create a false illusion of nonexistent ones.

The metadata elements of IFA can be used by attackers to collect information about the network hops.

Adding IFA headers or adding to IFA metadata can be used to consume resources within the path being monitored or by a collector.

Adding IFA headers or adding to IFA metadata can be used to force exceeding the MTU for the path being monitored resulting in fragmentation and/or packet drops.

IFA is expected to be deployed within controlled network domains, containing attacks to that controlled domain. Limiting or preventing monitoring or attacks using IFA requires limiting or preventing unauthorized access to the domain in which IFA is to be used, and preventing leaking IFA metadata beyond the controlled domain.

6. References

6.1 Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

6.2 Informative References

[RFC791] https://tools.ietf.org/html/rfc791

[RFC6864] https://tools.ietf.org/html/rfc6864

[RFC3514] https://tools.ietf.org/html/rfc3514

[IFA 1.0] https://tools.ietf.org/html/draft-kumar-ifa-00

Authors' Addresses

Jai Kumar
Broadcom Inc.
Email: jai.kumar@broadcom.com

Surendra Anubolu
Broadcom Inc.
Email: surendra.anubolu@broadcom.com

John Lemon
Broadcom Inc.
Email: john.lemon@broadcom.com

Rajeev Manur
Broadcom Inc.
Email: rajeev.manur@broadcom.com

Hugh Holbrook
Arista Networks
Email: holbrook@arista.com

Anoop Ghanwani
Dell EMC
Email: anoop.ghanwani@dell.com

Dezhong Cai
AliBaba Inc.
Email: d.cai@alibaba-inc.com

Heidi OU
AliBaba Inc.

Email: heidi.ou@alibaba-inc.com

Yizhou Li
Huawei Technologies
EMail: liyizhou@huawei.com

Appendix A

Appendix A is for informational purposes only. The following options were considered for the IFA protocol.

A.1 Probe Marker

One of the challenges of using probe signatures in an IFA header is a false positive.

The IFA version 2.0 header takes care of large header sizes and identification based on probe markers. Probe markers can cause false positives if there is a match on the first 64 bits of the layer 4 payload.

This approach is not a preferred approach, but is supported by this draft as a version 1.0 header.

A.2 DSCP

[RFC791] EXP/LU Pool 3 can be used for identifying IFA packets. CU bits can be used for identifying IFA packets.

The problem with using TOS bits is that they are pervasively used in the network deployment and are responsible for affecting the forwarding decision.

This approach is not supported or recommended by this draft.

A.3 IP Options

[RFC791] The IP options provide for control functions that are needed or useful in some situations but unnecessary for the most common communications. The IP options include provisions for timestamps, security, and special routing.

There are various problems with this approach.
(1) The IPv4 header size can become arbitrarily large with the presence of options.

(2) A switch pipeline typically handles IP option packets as exception path processing and punts them to a host CPU.
(3) IP options make the construction of firewalls cumbersome, and are typically disallowed or stripped at the perimeter of enterprise networks by firewalls.

This approach is not supported or recommended by this draft.

A.4 IPv4 Identification or Reserved Flag

[RFC6864] [RFC3514] Another suggestion is to use the IPv4 identification field or reserved flag. This suggestion is also discarded and not supported for the following reasons:

[RFC6864] prohibits usage of id field for any other purposes.

[RFC3514] prohibits using flags bit 0 for security reasons.