IDR Working Group
Internet-Draft
Intended status: Standards Track
Expires: 5 December 2024

S. Hares
Hickory Hill Consulting

3 June 2024

BGP Flow Specification Version 2 - More IP Actions
draft-hares-idr-fsv2-more-ip-actions-01

Abstract

The BGP flow specification version 2 (FSv2) for Basic IP defines user ordering of filters along with FSv1 IP Filters and FSv1 actions. This draft suggests additional IP actions for Flow Specification FSv2 in Extended Communities and Community path attribute.

Status of This Memo

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

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at https://datatracker.ietf.org/drafts/current/.

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."

This Internet-Draft will expire on 5 December 2024.

Table of Contents

1. Introduction

Version 2 of BGP flow specification was originally defined in [I-D.ietf-idr-flowspec-v2] (denoted FSv2). However, the full FSv2 specification contains more than initial implementers desired. Therefore, this original FSv2 draft remains an WG draft, but the content will be split out into functions that implementers can manage. Section 1.3 contains the list of documents resulting from the split of the original FSv2 documents.

FSv2 defines new user-ordered filters that will be used with the IPv4 (AFI=1) and IPv6 (AFI=2) 2 new SAFIs (TBD1, TBD2) for FSv2 to be used with 5 AFIs (1, 2, 6, 25, and 31) to allow user-ordered lists of traffic match filters for user-ordered traffic match actions encoded in Communities (Wide or Extended).

This draft specifies defines extensions to the FSv2 Basic IP package [I-D.hares-idr-fsv2-ip-basic] to support additional IP filters for IP packet and payload. The filters are passed in the Extended IP Filters (type 2) of the subTLVs. This filter form contains a filter version number so filters can be added easily.

BGP Flow Specification version 1 (FSv1) as defined in [RFC8955], [RFC8956], and [RFC9117] specified 2 SAFIs (133, 134) to be used with IPv4 AFI (AFI = 1) and IPv6 AFI (AFI=2). FSv2 specifies 2 new SAFIs (TBD1, TBD2) for FSv2 to be used with 5 AFIs (1, 2, 6, 25, and 31) to allow user-ordered lists of traffic match filters for user-ordered traffic match actions encoded in Communities (Wide or Extended). The first SAFI (TBD1) will be used for IP forwarding, and the second SAFI (TBD2) will be used with VPNs. The supported AFI/SAFI combinations in FSv2 are:

* IPv4 (AFI=1, SAFI=TBD1),
* IPv6 (AFI=2, SAFI=TBD1),
* L2 (AFI=6, SAFI=TBD1),
* SFC (AFI=31, SAFI=TBD1),
* BGP/MPLS IPv4 VPN (AFI=1, SAFI=TBD2),
* BGP/MPLS IPv6 VPN (AFI=2, SAFI=TBD2),
* BGP/MPLS L2VPN (AFI=25, SAFI=TBD2), and
* SFC VPN (AFI=31, SAFI=TBD2)

FSv1 and FSv2 use different AFI/SAFIs to send flow specification filters. Since BGP route selection is performed per AFI/SAFI, this approach can be termed â\200\234ships in the nightâ\200\235 based on AFI/SAFI.

FSv2 specifies allows new IP filters to be used with the IPv4 (AFI=1) and IPv6 (AFI=2). FSv2 for Basic IP suggests these new filters are added in a component called "Extended IP Filters [I-D.hares-idr-fsv2-ip-basic]. [I-D.hares-idr-fsv2-more-ip-filters] provides a summary of the additional IP filters which have been adopted by the IDR WG or proposed to the IDR WG. It is anticipated that the filters will be added in groups as needed by applications.

The original FSV2 work [I-D.ietf-idr-flowspec-v2] proposed an user-ordered of traffic match actions encoded in the Community Path Attribute. This document proposes:

* a specification order for IP actions in Extended Communities, and

* a user ordering for IP Actions in the Community path Attribute.

Section 2 contains a description of the format of the FSv2 actions in Extended Communities and Wide Communities.

Since Extended Communities can be attached to a FSv2 NLRI with filters, it is possible that the Extended Communities conflict with each other. Section 3 lists the FSv1 actions that conflict with one another if attached to the same filter.

This document proposes that Extended Community actions attached to FSv2 NLRI use a pre-defined order that occurs after the User defined orders. The predefined order for Extended Communities in FSv2 NLRI would be set by the action type value (see section 6.3). For example, if the user-actions range from 1-1000, the Extended Community order would start after those actions.

Sections 4 and 5 provide a short description of new actions proposed for FSv2. Section 4 provides a description of the Action Chain Ordering proposed for the FSv2 for Basic IP ([I-D.hares-idr-fsv2-ip-basic]. Section 5 lists the potential new actions from FSv2 actions described in IDR WG drafts or proposed as individual drafts.

Sections 6, 7, and 8 provide information on validation, ordering, and error handling of FSv2 actions. Section 6 describes validation of actions and ordering of actions (user ordered and default). Section 7 describes error handling for FSv2 actions. Section 8 provides an example for ordering of actions.

Section 9 provides details on IANA considerations

1.1. Definitions and Acronyms

AFI - Address Family Identifier

AS - Autonomous System

BGPSEC - secure BGP [RFC8205] updated by [RFC8206]

BGP Session ephemeral state - state which does not survive the loss of BGP peer session.

Configuration state - state which persists across a reboot of software module within a routing system or a reboot of a hardware routing device.

DDOs - Distributed Denial of Service

Ephemeral state - state which does not survive the reboot of a software module, or a hardware reboot. Ephemeral state can be ephemeral configuration state or operational state.

FSv1 - Flow Specification version 1 [RFC8955] [RFC8956]

FSv2 - Flow Specification version 2 (this document)

NETCONF - The Network Configuration Protocol [RFC6241].

RESTCONF - The RESTCONF configuration Protocol [RFC8040]

RIB - Routing Information Base

ROA - Route Origin Authentication [RFC6482]

RR - Route Reflector.

SAFI â\200\223 Subsequent Address Family Identifier

1.2. RFC 2119 language

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 BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals as shown here.

1.3. FSv2 Series of Specifications

The full FSV2 information is contained in [I-D.ietf-idr-flowspec-v2].

Feedback from the implementers indicate that the Flow Specification v2 needs to broken into drafts based on the use cases the technology supports. These include IPv4/IPv6 IP Basic Filters for DDOS, IPv4/ IPv6 filters beyond DDOS, BGP/MPLS IPv4 VPN, BGP/MPLS IPv6 VPN, BGP/ MPLS L2VPN, Segment routing (SRMPLS, SRv6), SFC, SFC VPN, L2, L2 VPNs, and tunneled traffic (e.g., nv03 WG tunnels).

The following is the list of planned drafts:

* FSv2 IP Basic ([I-D.hares-idr-fsv2-ip-basic])

* FSv2 More IP Filters ([I-D.hares-idr-fsv2-more-ip-filters])

* FSv2 More IP Actions ([I-D.hares-idr-fsv2-more-ip-actions])

* FSv2 Non-IP Filters (draft-hares-idr-fsv2-non-ip-Filters)

* FSv2 Non-IP Actions (draft-hares-idr-fsv2-non-ip-actions)

[I-D.hares-idr-fsv2-ip-basic] has a description of each draft. The original FSV2 information is contained in [I-D.ietf-idr-flowspec-v2].

2. Format of FSv2 Actions

The FSv2 actions may be sent in an Extended Community or a Community Path Attribute. User ordering of FSv2 actions requires using the Community Path Attribute. This section reviews the describes the format of FSv2 actions in Extended Communities or Community Path Attributes.

The Extended Community encodes the Flow Specification actions in the Extended IPv4 Community format [RFC4360] or in the Extended IPv6 Community format [RFC5701]. The Extended Community actions cannot be ordered by the user, but will be ordered by default. The implementer and the operator must be aware of interactions between any FSv2 actions must be specified in an Extended Community.

FSv2 IP Basic proposes that Extended Community actions attached to FSv2 NLRI use a pre-defined order that occurs after the User defined orders. The predefined order for Extended Communities in FSv2 NLRI would be set by the action type value (see section 6.3). For example, if the user-actions range from 1-1000, the Extended Community order would start after those actions.

The Community attribute [I-D.ietf-idr-wide-bgp-communities] describes an attribute with flexible format for specifying community information.

This section first describes the following Information related to FSv2 Actions in Extended Communities:

* Generic Transitive Extended Communities for FSv2 Actions (FS-TG-EC) [RFC8955]

* Transitive Extended Communities for redirect. This includes:

  - (Generalized redirection ID with Sequencing and copy) [I-D.ietf-idr-flowspec-path-redirect]

  - Redirect plus Copy bit [I-D.ietf-idr-flowspec-redirect-ip]

  - Transitive IPv6-Address Extended Community formats for FSv2 actions [RFC8956]

2.1. Encoding FSv2 Actions in Generic Transitive Communities

The FSv2 actions encoded in Generic Transitive Communities inherit the FSv1 actions in Generic Transitive Communities.

The Extended Community encodes the Flow Specification actions in the Extended Community format as Generic Transitive Extended Communities per [RFC4360] per [RFC8955], [RFC9117], and [RFC9184].

The format of the these actions can be:

Generic Transitive Extended Community (0x80): where the Sub-Types are defined in the Generic Transitive Extended Community Sub-Types registry.

Generic Transitive Extended Community Part 2 (0x81): where the Sub-Types are defined in the Generic Transitive Extended Community Part 2 Sub-Types registry.

Transitive Four-Octet AS-Specific Extended Community (0x82): where the Sub-Types defined in the Generic Transitive Extended Community Part 3 Sub-Types registry.

Generic Transitive Extended Community Part 3 (0x83): where the Sub-Types defined in the Transitive Opaque Extended Community Sub-Types" registry.

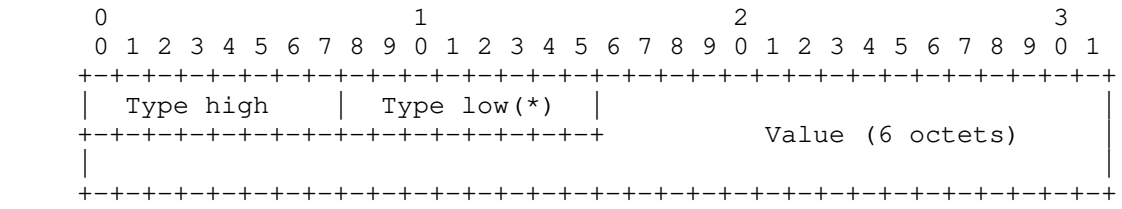


Figure 2-1

Table 2-1 Generic Transitive Extended Community
Part 1 - (0x80)

IPv4 Extended Communities (Type 0x80)

Value	Description	Name	Reference
0x01	FSv2 Action Chain Ordering	ACO	[This document]
0x06	FSv2 traffic-rate-byte	TRB	[RFC8955]
0x07	Flow spec traffic-action	TAIS	[RFC8955]
0x08	Flow spec rt-redirect AS-2 octet format	RDIP	[RFC8955]
0x09	Flow spec Remark DSCP	TMDS	[RFC8955]
0x0C	Flow Spec Traffic-rate-packets	TRP	[RFC8955]
0xOD	Flow Spec for SFC classifiers	SFCC	[RFC9015]

Table 2-2 Generic Transitive Extended Community
Part 2 (0x81)

IPv4 Extended Communities FSv2 action (Type 0x81)

Value	Description	Name	Reference
0x08	Flow spec rt-redirect	RDIP	[RFC8955]

Table 2-3 Generic Transitive Extended Community
Part 3 (Type 0x82)

Value	Description	Name	Reference
0x08	Flow spec rt-redirect AS-4 octet format	RDIP	[RFC8955]

Table 2-4: Traffic Action bits

Bit	Name	Name	Reference
47	Terminal Action	TAct	[RFC8955]
46	Sample	Samp	[RFC8955]
45	Copy	Copy	[this document]
44	Drop	drop	[this document]

2.2. Encoding Path Forwarding in IPv4 Transitive Extended Communities

FSv2 needs to refine the following Transitive Extended Communities that are not "Transitive Generic Communities" to a specific set of functions. These features provide overlapping functions. While some of these features are implemented, these actions should be be reviewed.

There are three types of functions:

* Active filters on interfaces in group for inbound or outbound data traffic

* Redirect to an IP address. Optionally perform a traffic action (copy)

* Redirect to an Indirection ID of a specific type. Optionally perform a traffic action (copy).

Table 2-5 Transitive Extended Community types (T-EC-types)

sub-type	FSv1 Description	Name	Reference
0x07	FS Interface set	Ifset	[IDR-WG-ifset]
0x08	FS Redirect/ Mirror	RIPv4	[IDR-WG-redirect-ip]
0x09	FS Redirect to Indirection-id	RGID	[IDR-WG-redirect-iid]

[IDR-WG-ifset] [I-D.ietf-idr-flowspec-interfaceset]

[IDR-WG-redirect-ip] [I-D.ietf-idr-flowspec-redirect-ip]

[IDR-WG-redirect-iid] [I-D.ietf-idr-flowspec-path-redirect]

2.3. Encoding FSv2 Actions in IPv6 Extended Community

The IPv6 Extended Community encodes the Flow Specification actions in the Extended Community format ([RFC5701]) in the transitive opaque format (See [RFC8956], [RFC9117], and [RFC9184])

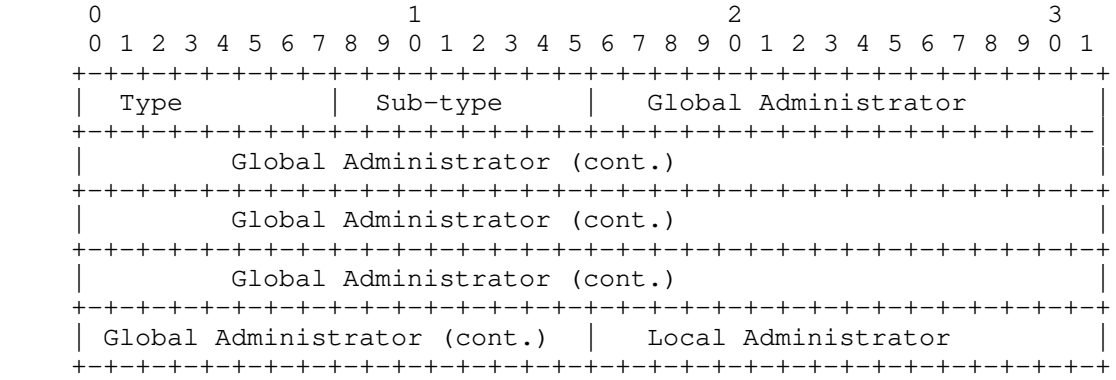


Figure 2-2

The 20 octets of value are given in the following format:

Global Administrator: IPv6 address assigned by Internet Registry
Local Administrator: 2 bytes of Local Administrator

Table 2-6 transitive IPv6-Address-Specific Actions

Value	Description	Name	Reference
0x01	Flow Spec Action Chain	ACO	[This document]
0x0C	Flow Spec redirect-v6-flag	RD6F	[IDR-WG-RD6f]
0x0D	Flow Spec rt-redirect IPv6 format	RDv6	]RFC8956]

Figure 3-16

[RFC8956] - [RFC8956]

IDR-WG-RD6F - [I-D.ietf-idr-flowspec-redirect-ip]

2.4. FSv2 Actions encoded in Community Attribute

The user-ordered FSv2 Actions use the Communities Path Attribute defined in [I-D.ietf-idr-wide-bgp-communities]. The Community BGP Path attribute is a flexible Community container which allows different formats for different community applications. The FSv2 Community Type ([FSv2-CA], type=TBD4) is only used for FSv2 user-ordered actions.

If both the Extended Community FSv2 Actions (FSv2-EC) and the FSv2 Community Attribute actions are specified (FSv2-CA) then the default preference will be the FSv2-CA prior to the FSv2-EC. This preference may be changed based on a configuration knob. However, this configuration MUST be consistent within an Autonomous System or a group of Autonomous Systems where both the FSv2-CA and FSv2-EC are deployed.

2.4.1. FSv2 Actions in Community Path Attribute

The Community attribute (Type = BGP Community Container, value 34)

Community Path attribute common header

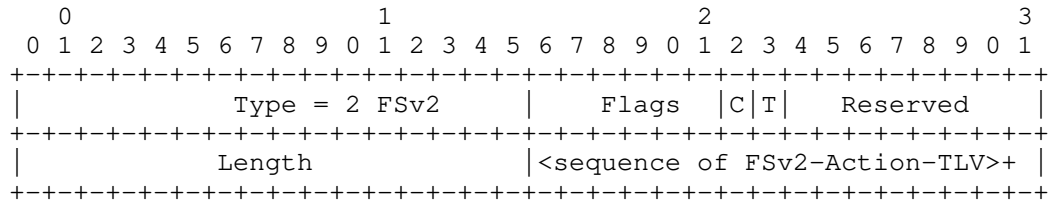


figure 2-3

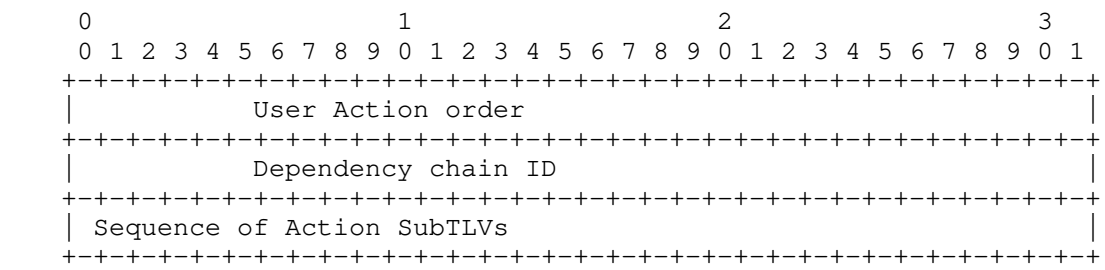
where:

Type = the type of community (Type 1 or Type 2)
Flag = an octet of bits with only two
bit that can be set
T = 1 - Transitive across AS boundaries
T = 0 - Non-Transitive across AS boundaries
C = 1 - Transitive across Confederation boundaries
C = 0 - Non-Transitive across Confederation boundaries

figure 2-4

The FSv2 Action TLVs have the following format:

Common Header for Action TLVs



Each Action SubTLV has the format:

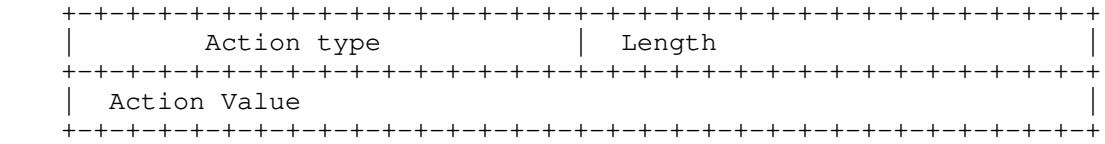


figure 2-5

where:

User Action Order - 4 octet field indication user defined action order.

Dependency chain ID - 4 octet field indicating dependency chain identification. (Editor's note: Dependency chain needs further discussion from WG.}

Sequence of Action SubTLVs - The type-length-value fields specified per Action type.

3. Conflicts in FSv2 Extended Communities Actions

3.1. Conflicts between FSv1 Actions

Table 3-1: Conflicts between FSv2 Transitive Generic IPv4 actions

IPv4 Extended Communities (Type 0x80)		
Value	Name	Conflicts with
0x01	ACO	none
0x06	TRB	TRP
0x07	TAIS	duplication also done in RDIP, RIPv4, RGID
0x08	RDIP	redirection done in RIPv4, RGID copy done in TAIS
0x09	TMDS	none
0x0C	TRP	TRB
0xOD	SFCC	none

Table 3-2 Transitive IPv6-Address-Specific Actions

Value	Name	Conflicts with
0x01	ACO	none
0x0C	RD6F	RDv6
0x0D	RDv6	RD6F

4. Actions Proposed for FSv2 for Basic IP

The long-term goal of the FSv2 actions is to allow user ordering of the flow specification actions. Only the Community Path Attribute provides enough structured space for user ordering of actions. The IDR WG draft [I-D.ietf-idr-flowspec-v2] contains the long-term plan for FSv2 filters with actions. Any new Actions for FSv2 should be specified in both the Extended Community format and the Community Path Attribute formatt.

The FSv2 for Basic IP will support existing IPv4 from [RFC8955], and existing IPv6 actions from [RFC8956] and one additional feature for action chain ordering (ACO).

An action chain for FSv2 Extended Community actions is defined as a series of Extended Communities which are attached to a set of filters.

The action chain ordering (ACO)action provides a set of flags that define a clear action if failure occurs. One of the issues with FSv1 is the lack of a clear definition on what happens if multiple actions interact. The existence of the Action chain ordering action enforces that the actions will have a deterministic outcome during failures.

The AC-Failure types are:

* 0x00 â\200\223 default â\200\223 stop on failure

* 0x01 â\200\223 continue on failure (best effort on actions)

* 0x02 â\200\223 conditional stop on failure (depends on AC-Failure-value/ policy)

* 0x03 â\200\223 rollback do all or nothing (depends on AC-Failure-value/ policy)

Editors note: The following options for encoding ACO exist.

Option 1: redefine bits in Traffic Action subtype

Option 2: create a new Extended Community

4.1. Action Chain Ordering

4.1.1. Option 1: Action Chain operation IPv4 Extended (ACO)(1, 0x01)

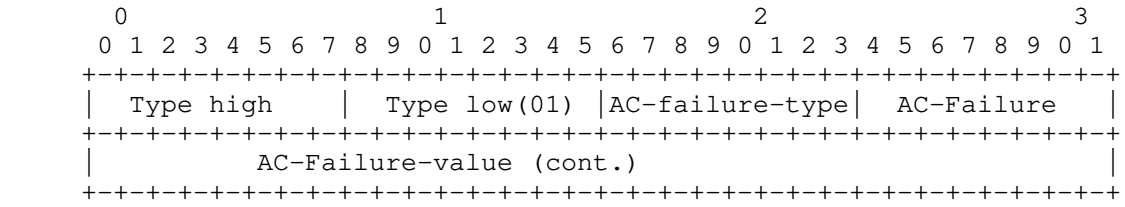


Figure 4-1

SubTLV: 0x01

Length: 6 octets

Value:

AC-dependence - 1 octet byte of flag regarding dependency

AC-failure-type â\200\223 1 octet byte that determines the action on failure

AC-failure-value â\200\223 variable depending on AC-failure-type.

Actions may succeed or fail and an Action chain must deal with it. The default value stored for an action chain that does not have this action chain is â\200\234stop on failureâ\200\235.

where:

AC-Failure types are:

- 0x00 â\200\223 default â\200\223 stop on failure

- 0x01 â\200\223 continue on failure (best effort on actions)

- 0x02 â\200\223 conditional stop on failure â\200\223 depending on AC-Failure-value

- 0x03 â\200\223 rollback â\200\223 do all or nothing - depending in AC-Failure-value

AC-Failure values: TBD

Interactions with other actions: Interactions with all other Actions

Ordering within Action type: By AC-Failure type

4.1.2. Option 2: Action Chain operation encoded in IPv4/IPv6 Community Traffic Action (0x07)

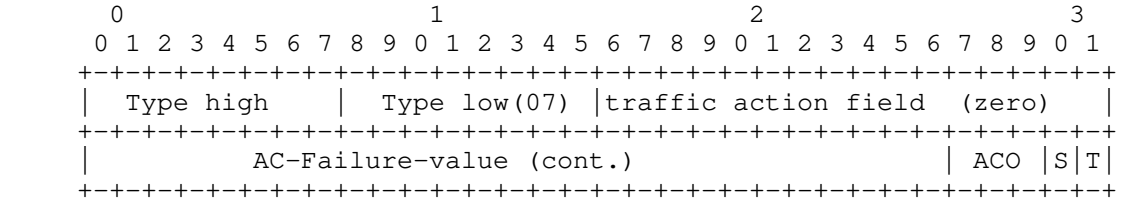


figure 4-2

Where
ACO - is the Action Chain failure types (0x00 to 0x03)
00 - stop on failure
01 - continue on failure
02 - conditional stop on failure (by policy)
03 - rollback on failure (with policy)
S - Sample flag
T - Terminal action

5. Actions Proposed for FSv2 Actions

5.1. Summary of FSv2 Extended Community [FSv2-EC] in WG (actual or proposed)

Table 5-1 shows the actions for Extended-Communities FSv2 actions from IDR RFCs, IDR WG drafts and drafts proposed to IDR. The link between the short names and the IETF draft names is shown in Table 5-2. Table 5-3 has FSv2 actions proposed for IPv6.

The FSv2 Extended Communities actions (FSv2-EC actions) would take a default ordering based on the numerical order of the action id (act-id). For example, ACO (act-id = 1) would be processed before Traffic Actions per interface group (act-id = 3).

A match on an IP Filters can request a non-IP action. Table 5-3 gives a list of Non-IP functions defined for FSv2-EC action. The order of processing the non-IP action is done by the action id (act-id). FSv2 rules can link non-IP actions can be to IP filters. For example, the SFC filters for IP link an SFC classifier action (name: TISFC, Action-id = 31).

Table 5-4 contains FSv2 actions for IPv6. The size of the IPv6 fields require a unique space for some redirect actions in Extended Communities. (Editor's note: IPv6 Extended Community actions do not have to be unique in the TLV formats in the Community Attribute.)

Table 5-1 All IP Actions in Extended Communities

act-id	Name	Description	Document
00	RSV	Reserved	[this document]
01	ACO	action chain order	[this document]
02	TBA	Unassigned	[this document]
03	TAIS	traffic actions per interface group	[IDR-FSv2-ifset]
04	TBA	to be assigned	[this document]
05	TBA	to be assigned	[this document]
06	TRB	traffic rate limited by bytes	[RFC8955]
07	TA	traffic action (terminal/sample)	[RFC8955]
08	RDIP	Redirect IPv4	[IDR-FSv2-RDP-ip]
09	TM	Mark DSCP value	{RFC8955]
10	TBA	Unassigned	[this document]
11	TBA	Unassigned	[this document]
12	TRP	traffic rate limit by packet	[RFC8955]
13	TISFC	SFC Classifier	[RFC9015]
14	RDIID	redirect to Indirection-id	[IDR-FSv2-RDPIID] [PD-FSv2-mobility] move from 0x00)
15	TBA	Unassigned	[this document]
16	TBA	Unassigned	[this document]
17	NRP-ID	Encapsulate NRP-ID value	[IDR-net-slice-ts]
18	GrP-ID	Set Group ID	[PD-peng-group-sub]

Figure 3-15

Table 5-2 Short Names to IETF documents

Short-name	Filename
IDR-FSv2	draft-ietf-idr-flowspec-v2-04
IDR-FSv2-ifset	draft-ietf-idr-flowspec-interfaceset-05
IDR-FSv2-linkbw	draft-ietf-idr-linkbandwidth
IDR-FSv2-RDP-ip	draft-ietf-idr-flowspec-redirect-ip-02
IDR-FSv2-RDPIID	draft-ietf-idr-flowspec-redirect-path
IDR-net-slice-ts	draft-ietf-idr-flowspec-network-slide-ts-02
IDR-FSv2-L2VPN	draft-ietf-idr-flowspec-l2vpn-17
IDR-FSv2-mpls	draft-ietf-idr-fsv2-mpls-actions
IDR-FSv2-SR	draft-ietf-idr-fsv2-sr-actions
PD-FSv2-mobility	draft-dmc-flowspec-tn-aware-mobility
PD-peng-group-sub	draft-peng-idr-apn-bgp-flowspec-00
PD-redirect-IPv6	draft-ietf0-idr-path-redirect-12

Table 5-3 All Non-IP Actions in Extended Communities

act-id	Name	Description	Reference
31	TISFC	SFC classifier II	[RFC9015]- moved
32	MPLSLA	MPLS label action	[IDR-FSv2] [IDR-FSv2-mpls]
33	VLAN	VLAN-Action (0x16)	[ID-FSv2-L2VPN]
34	TPID	TPID-Action (0x17)	[ID-FSv2-LVPN]
24- 254	TBA	to be assigned	
255	reserved		

Figure 3-15

Table 5-4 IPv6 Extended Communities (Type 1)

Value	Description	Name	Reference
0x01	Flow Spec Action Chain	ACO	[This document]
0x0C	Flow Spec redirect-v6-flag	RD6F	[PD-redirect-IPv6]
0x0D	Flow Spec rt-redirect IPv6 format	RD6	[RFC8956]

5.2. Actions proposed for FSv2 Community Path Attribute

The FSv2 Community Path Attribute could inherits the FSv2 Extended Community actions (FSv2-EC) with the same action identifiers (act-id) numbering. For each of the actions, a new form of the FSv2-EC would need to be defined. The next section is set-aside for definition of the FSv1 based attributes standardized in [RFC8955] and [RFC8956]>, and deployed FSv1 actions. New FSv2-EC must define both an Extended Community form and a Community Path Attribute form.

5.2.1. FSv2 Community Path Attribute for FSv1 actions

The following FSv2 Communjity Path Attributes created from FSv1 actions will be defined in this section:

action chain order (ACO)(type = 01),

traffic actions per interface group (TAIS) (type = 02),

traffic rate limited by bytes (TRB (type = 06),

traffic actions (TA) (type = 07),

redirect IPv4 (RDIP) (type = 08),

mark DSCP valiue (TM) (type = 09),

traffic rate limit by packet (TRP)(type = 12),

SFC Classifier (TISFC) (type = 13),

Redirect to Indirection-ID (type=14),

Encapsulate in NRP-ID (type = 17)

Set Group ID (type = 18)

5.2.2. Current Proposals for FSv2 in Community Path Attribute

The current actions proposed for the FSv2 Community Path Attribute are are shown in Table 5-5. The

Table 5-5 All Actions Proposed for FSv2 Community Path Attribute

act-id	Name	Description	Document
TBD	MatchSet	Match and Set attribute	[IDR-rpd] (type = 03)
TBD	MatchNoA	Match and No Advertise	[IDR-rpd] (type = 04)
TBD	DetLat	Deterministic Latency action	[PD-detnet-flowmap] (type = 37)
TBD	TSNMap	Map flow to TSN stream	[PD-detnet-flowmap] (type = 38)

Table 5-6 File Names

IDR-rpd - [I-D.ietf-idr-rpd]

PD-detnet-flow-map -[I-D.xiong-idr-detnet-flow-mapping]

6. Validation and Ordering of Actions

The validation of FSv2 NLRI adheres to the combination of rules for general BGP FSv1 NLRI found in [RFC8955], [RFC8956], [RFC9117], and the specific additions made for SFC NLRI [RFC9015], and L2VPN NLRI [I-D.ietf-idr-flowspec-l2vpn].

The precedence for FSv2 actions are described in this section rather than simply referring to the relevant portions of these RFCs. Validation only occurs after BGP UPDATE message reception and the FSv2 NLRI and the path attributes relating to FSv2 (Extended community and Wide Community) have been determined to be well-formed. Any MALFORMED FSv2 NRLI is handled as a â\200\234TREAT as WITHDRAWâ\200\235 [RFC7606].

6.1. Validation of Flow Specification Actions

FSv2 actions may specify actions using Extended Communities or the path attribute Community with the FSv2 format. The FSv2 actions in Extended Communities and Wide communities can be associated with large number of NLRIs.

Actions may conflict, duplicate, or complement other actions. An example of conflict is the packet rate limiting by byte and by packet. An example of a duplicate is the request to copy or sample a packet under one of the redirect functions (RDIPv4, RDIPv6, RDIID,) Each FSv2 actions in this document defines the potential conflicts or duplications. Specifications for new FSv2 actions outside of this specification MUST specify interactions or conflicts with any FSv2 actions (that appear in this specification or subsequent specifications).

Well-formed syntactically correct actions should be linked to a filtering rule in the order the actions should be taken. If one action in the ordered list fails, the default procedure is for the action process for this rule to stop and flag the error via system management. By explicit configuration, the action processing may continue after errors.

Implementations MAY wish to log the actions taken by FS actions (FSv1 or FSv2).

6.2. Ordering of Actions

The ordering of precedence for these actions in the case are:

* Action user-order number zero is defined to have an Action type of â\200\234Set Action Chain operationâ\200\235 (ACO) that defines the default action chain process.

* user order (1 to N-1)

* Extended Community Actions {starting at N)

By default, extended community actions are associated with default order number 32768 [0x8000] or a specific configured value for the FSv2 domain.

Within the actions defined at the same order, the order is:

Action ID (lowest to highest)

If multiple Actions of the same value are define (e.g. Redirection to Indirection ID), the action must define the order.

All Extended Community actions and Path Community attributes should be ordered in the same IP space.

6.3. Summary of FSv2 ordering

Operators should use user-defined ordering to clearly specify the actions desired upon a match. The FSv2 actions default ordering is specified to provide deterministic order for actions which have the same user-defined order and same type.

FS Action (lowest value to highest)	Value Order (lowest to highest)
0x01: ACO: Action chain operation	Failure flag
0x02: TAIS:Traffic actions per Interface group	AS, then Group-ID, then Action ID
0x03-0x05 to be assigned	TBD
0x06: TRB: Traffic rate limit by bytes	AS, then float value
0x07: TA: Traffic Action	traffic action value
0x08: RDIP: Redirect to IP	AS, then IP Address, then ID
0x09: TM: Traffic Marking	DSCP value (lowest to highest)
0x0A: AL2: Associated L2 Info.	TBD
0x0B: AET: Associated E-tree Info.	TBD
0x0C: TRP: Traffic Rate limit by bytes	AS, then float value
0x0D: RDIPv6: Traffic Redirect to IPv6	AS, IPv6 value, then local Admin
0x0E: TISFC: Traffic insertion to SFC	SPI, then SI, the SFT
0xOF: Redirect to Indirection-ID	ID-type, then Generalized-ID
0x10: MPLSLA: MPLS Label stack	order, action, label, Exp
0x16 â\200\223 VLAN action	rewrite-actions, VALN1, VLAN2, PCP-DE1, PCP-DE2
0x17 â\200\223 TPID action	rewrite actions, TP-ID-1, TP-ID-2

Figure 6-1

7. Error handling

The following two error handling rules must be followed by all BGP speakers which support FSv2:

* FSv2 NLRI having TLVs which do not have the correct lengths or syntax must be considered MALFORMED.

* FSv2 NLRIs having TLVs which do not follow the above ordering rules described in section 4.1 MUST be considered as malformed by a BGP FSv2 propagator.

The above two rules prevent any ambiguity that arises from the multiple copies of the same NLRI from multiple BGP FSv2 propagators.

A BGP implementation SHOULD treat such malformed NLRIs as â\200\230Treat-as-withdrawâ\200\231 [RFC7606]

An implementation for a BGP speaker supporting both FSv1 and FSv2 MUST support the error handling for both FSv1 and FSv2.

8. Example for Ordering of the Actions

8.1. Example of Action Chain Operation (ACO)

The â\200\234Action Chain Operationâ\200\235 (ACO) changes the way the actions after
the current action in an action chain are handled after a failure. If no action chain operations are set, then the default action of â\200\234stop upon failureâ\200\235 (value 0x00) will be used for the chain.

8.1.1. Example 1 - Default ACO

Use Case 1: Rate limit to 600 packets per second

Description: The provider will support 600 packets per second All Packets sampled for reporting purposes and packet streams over 600 packets per second will be dropped.

Suppose BGP Peer A has a

* a Wide Community action with user-defined order 10 with Traffic Sampling

* a Wide Community action with user-defined order 11 from AS 2020 that limits packet-based rate limit of 600 packets per second.

* an Extended Community from AS 2020 that does limits packet-based rate limit of 50 packets per second.

The FSv2 data base would store the following action chain:

* at user-defined action order 10

  - A user action of type 7 (traffic action) with values of Sampling and logging.

* at user-defined action order 11

  - a user action type of 12 (packet-based rate limit) with values of AS 2020 and float value for 600 packets per second (pps)

* at user-defined action order 32768 (0x8000) with type 12 and values of A user action of type 12 with values of AS 2020 and float value of 50 packets/second.

Normal action:

The match on the traffic would cause a sample of the traffic (probably with packet rate saved in logging) followed by a rate limit to 600 pps. The Extended community action would further limit the rate to 50 packets per second.

When does the action chain stop?

The default process for the action chain is to stop on failure. If there is no failure, then all three actions would occur. This is probably not what the user wants.

If there is failure at action 10 (sample and log), then there would be no rate limiting per packet (actions 11 and action 32768).

If there is failure at action 11 (rate limit to packet 600), then there would be no rate limiting per packet (action 32768).

The different options for Action chain ordering (ACO) have been worked on with NETCONF/RESTCONF configuration and actions.

8.1.2. Example 2: Redirect traffic over limit to processing via SFC

Use case 2: Redirect traffic over limit to processing via SFC.

Description: The normal function is for traffic over the limit to be forwarded for offline processing and reporting to a customer.

Suppose we have the following 4 actions defined for a match:

* Sent Redirect to indirection ID (0x01) with user-defined match 2 attached in wide community,

* Traffic rate limit by bytes (0x07) with user-defined match 1 attached in wide community,

* Traffic sample (0x07) sent in extended community, and

* SF classifier Info (0x0E) sent in extended community.

These 4 filters rate limit a potential DDoS attack by: a) redirect the packet to indirection ID (for slower speed processing), sample to local hardware, and forward the attack traffic via a SFC to a data collection box.

The FSv2 action list for the match would look like this

Action 0: Operation of action chain (0x01) (stop upon failure)

Action 1: Traffic Rate limit by byte (0x07)

Action 2: Redirect to Redirection ID (0x0F)

Action 32768 (0x8000) Traffic Action (0x07) Sample

Action 32768 (0x8000) SFC Classifier: (0xE)

If the redirect to a redirection ID fails, then Traffic Sample and sending the data to an SFC classifier for forwarding via SFC will not happen. The traffic is limited, but not redirect away from the network and a sample sent to DDOS processing via a SFC classifier.

Suppose the following 5 actions were defined for a FSV2 filter:

* Set Action Chain Operation (ACO) (0x01) to continue on failure (ox01) at user-order 2 attached in wide community,

* redirect to indirection ID (0x0F) at user-order 2 attached in wide community,

* traffic rate limit by bytes (0x07)with user-order 1 attached in wide community,

* Traffic sample (0x07) attached via extended community, and

* SFC classifier Info (0x0E) attached in extended community.

The FSv2 action list for the match would look like this:

Action 00: Operation of action chain (0x01) (stop upon failure)

Action 01:Traffic Rate limit by byte (0x07)

Action 02:Set Action Chain Operation (ACO) (0x01) (continue on failure)

Action 02: Redirect to Redirection ID (0F)

Action 32768 (0x8000): Traffic Action (0x07) Sample

Action 32768 (0x8000): SFC classifier (0x0E) forward via SFC [to DDoS classifier]

If the redirect to a redirection ID fails, the action chain will continue on to sample the data and enact SFC classifier actions.

9. IANA Considerations

This section complies with [RFC7153].

9.1. FSV2 Action TLV Types

IANA is requested to create the following entries on a new "Flow Specification v2 Actionâ\200\235 web page.

Name: BGP FSv2 Action types
Reference: [this document]
Registration Procedure: 0x01-0x3FFF Standards Action.

Type	Use	Reference
0x00	Reserved	[this document]
0x01	ACO: Action Chain Operation	[this document]
0x02	TAIS: Traffic actions per interface group	[this document]
0x03	Unassigned	[this document]
0x04	Unassigned	[this document]
0x05	Unassigned	[this document]
0x06	TRB: traffic rate limited by bytes	[this document]
0x07	TA: Traffic action (terminal/sample)	[this document]
0x08	RDIPv4: redirect IPv4	[this document]
0x09	TM: traffic marking (DSCP)	[this document]
0x0A	AL2: associate L2 Information	[this document]
0x0B	AET: associate E-Tree information	[this document]
0x0C	TRP: traffic rate limited by packets	[this document]
0x0D	RDIPv6: Redirect to IPv6	[this document]
0x0E	TISFC: Traffic insertion to SFC	[this document]
0x0F	RDIID: Redirect to indirection-iD	[this document]
0x10	MPLS Label Action	[this document]
0x11	unassigned	[this document]
0x12	unassigned	[this document]
0x13	unassigned	[this document]
0x14	unassigned	[this document]
0x15	unassigned	[this document]
0x16	VLAN action	[this document]
0x17	TIPD action	[this document]
0x18-0x3ff	Unassigned	[this document]
0x4000-0x7fff	Vendor assigned	[this document]
0x8000-0xFFFF	Reserved	[this document]

9.2. Wide Community Assignments

IANA is requested to assign values from the Registered Type TBD4 BGP Wide Community Types:

Name	type Value
FSv2 Actions	TBD4

10. Security Considerations

The use of ROA improves on [RFC8955] by checking to see of the route origination. This check can improve the validation sequence for a multiple-AS environment.

>The use of BGPSEC [RFC8205] to secure the packet can increase security of BGP flow specification information sent in the packet.

The use of the reduced validation within an AS [RFC9117] can provide adequate validation for distribution of flow specification within a single autonomous system for prevention of DDoS.

Distribution of flow filters may provide insight into traffic being sent within an AS, but this information should be composite information that does not reveal the traffic patterns of individuals.

11. References

11.1. Normative References

[I-D.hares-idr-fsv2-ip-basic]
Hares, S., Eastlake, D. E., Yadlapalli, C., and S. Maduschke, "BGP Flow Specification Version 2 - for Basic IP", Work in Progress, Internet-Draft, draft-hares-idr-fsv2-ip-basic-02, 12 May 2024, <https://datatracker.ietf.org/doc/html/draft-hares-idr-fsv2-ip-basic-02>.

[I-D.hares-idr-fsv2-more-ip-actions]
Hares, S., "BGP Flow Specification Version 2 - More IP Actions", Work in Progress, Internet-Draft, draft-hares-idr-fsv2-more-ip-actions-00, 8 May 2024, <https://datatracker.ietf.org/doc/html/draft-hares-idr-fsv2-more-ip-actions-00>.

[I-D.hares-idr-fsv2-more-ip-filters]
Hares, S., "BGP Flow Specification Version 2 - More IP Filters", Work in Progress, Internet-Draft, draft-hares-idr-fsv2-more-ip-filters-01, 12 May 2024, <https://datatracker.ietf.org/doc/html/draft-hares-idr-fsv2-more-ip-filters-01>.

[I-D.ietf-idr-bgp-flowspec-label]
liangqiandeng, Hares, S., You, J., Raszuk, R., and D. Ma, "Carrying Label Information for BGP FlowSpec", Work in Progress, Internet-Draft, draft-ietf-idr-bgp-flowspec-label-02, 20 October 2022, <https://datatracker.ietf.org/doc/html/draft-ietf-idr-bgp-flowspec-label-02>.

[I-D.ietf-idr-flowspec-interfaceset]
Litkowski, S., Simpson, A., Patel, K., Haas, J., and L. Yong, "Applying BGP flowspec rules on a specific interface set", Work in Progress, Internet-Draft, draft-ietf-idr-flowspec-interfaceset-05, 18 November 2019, <https://datatracker.ietf.org/doc/html/draft-ietf-idr-flowspec-interfaceset-05>.

[I-D.ietf-idr-flowspec-l2vpn]
Weiguo, H., Eastlake, D. E., Litkowski, S., and S. Zhuang, "BGP Dissemination of L2 Flow Specification Rules", Work in Progress, Internet-Draft, draft-ietf-idr-flowspec-l2vpn-23, 15 April 2024, <https://datatracker.ietf.org/doc/html/draft-ietf-idr-flowspec-l2vpn-23>.

[I-D.ietf-idr-flowspec-mpls-match]
Yong, L., Hares, S., liangqiandeng, and J. You, "BGP Flow Specification Filter for MPLS Label", Work in Progress, Internet-Draft, draft-ietf-idr-flowspec-mpls-match-02, 20 October 2022, <https://datatracker.ietf.org/doc/html/draft-ietf-idr-flowspec-mpls-match-02>.

[I-D.ietf-idr-flowspec-nvo3]
Eastlake, D. E., Weiguo, H., Zhuang, S., Li, Z., and R. Gu, "BGP Dissemination of Flow Specification Rules for Tunneled Traffic", Work in Progress, Internet-Draft, draft-ietf-idr-flowspec-nvo3-19, 26 December 2023, <https://datatracker.ietf.org/doc/html/draft-ietf-idr-flowspec-nvo3-19>.

[I-D.ietf-idr-flowspec-path-redirect]
Van de Velde, G., Patel, K., and Z. Li, "Flowspec Indirection-id Redirect", Work in Progress, Internet-Draft, draft-ietf-idr-flowspec-path-redirect-12, 24 November 2022, <https://datatracker.ietf.org/doc/html/draft-ietf-idr-flowspec-path-redirect-12>.

[I-D.ietf-idr-flowspec-redirect-ip]
Uttaro, J., Haas, J., Texier, M., akarch@cisco.com, Ray, S., Simpson, A., and W. Henderickx, "BGP Flow-Spec Redirect to IP Action", Work in Progress, Internet-Draft, draft-ietf-idr-flowspec-redirect-ip-02, 5 February 2015, <https://datatracker.ietf.org/doc/html/draft-ietf-idr-flowspec-redirect-ip-02>.

[I-D.ietf-idr-flowspec-srv6]
Li, Z., Li, L., Chen, H., Loibl, C., Mishra, G. S., Fan, Y., Zhu, Y., Liu, L., and X. Liu, "BGP Flow Specification for SRv6", Work in Progress, Internet-Draft, draft-ietf-idr-flowspec-srv6-05, 29 March 2024, <https://datatracker.ietf.org/doc/html/draft-ietf-idr-flowspec-srv6-05>.

[I-D.ietf-idr-rpd]
Li, Z., Ou, L., Luo, Y., Mishra, G. S., Chen, H., and H. Wang, "BGP Extensions for Routing Policy Distribution (RPD)", Work in Progress, Internet-Draft, draft-ietf-idr-rpd-19, 28 March 2024, <https://datatracker.ietf.org/doc/html/draft-ietf-idr-rpd-19>.

[I-D.ietf-idr-wide-bgp-communities]
Raszuk, R., Haas, J., Lange, A., Decraene, B., Amante, S., and P. Jakma, "BGP Community Container Attribute", Work in Progress, Internet-Draft, draft-ietf-idr-wide-bgp-communities-11, 9 March 2023, <https://datatracker.ietf.org/doc/html/draft-ietf-idr-wide-bgp-communities-11>.

[I-D.xiong-idr-detnet-flow-mapping]
Xiong, Q., Wu, H., Zhao, J., and D. Yang, "BGP Flow Specification for DetNet and TSN Flow Mapping", Work in Progress, Internet-Draft, draft-xiong-idr-detnet-flow-mapping-05, 16 October 2023, <https://datatracker.ietf.org/doc/html/draft-xiong-idr-detnet-flow-mapping-05>.

[RFC0791] Postel, J., "Internet Protocol", STD 5, RFC 791, DOI 10.17487/RFC0791, September 1981, <https://www.rfc-editor.org/info/rfc791>.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[RFC3032] Rosen, E., Tappan, D., Fedorkow, G., Rekhter, Y., Farinacci, D., Li, T., and A. Conta, "MPLS Label Stack Encoding", RFC 3032, DOI 10.17487/RFC3032, January 2001, <https://www.rfc-editor.org/info/rfc3032>.

[RFC4271] Rekhter, Y., Ed., Li, T., Ed., and S. Hares, Ed., "A Border Gateway Protocol 4 (BGP-4)", RFC 4271, DOI 10.17487/RFC4271, January 2006, <https://www.rfc-editor.org/info/rfc4271>.

[RFC4360] Sangli, S., Tappan, D., and Y. Rekhter, "BGP Extended Communities Attribute", RFC 4360, DOI 10.17487/RFC4360, February 2006, <https://www.rfc-editor.org/info/rfc4360>.

[RFC4760] Bates, T., Chandra, R., Katz, D., and Y. Rekhter, "Multiprotocol Extensions for BGP-4", RFC 4760, DOI 10.17487/RFC4760, January 2007, <https://www.rfc-editor.org/info/rfc4760>.

[RFC5065] Traina, P., McPherson, D., and J. Scudder, "Autonomous System Confederations for BGP", RFC 5065, DOI 10.17487/RFC5065, August 2007, <https://www.rfc-editor.org/info/rfc5065>.

[RFC5701] Rekhter, Y., "IPv6 Address Specific BGP Extended Community Attribute", RFC 5701, DOI 10.17487/RFC5701, November 2009, <https://www.rfc-editor.org/info/rfc5701>.

[RFC6482] Lepinski, M., Kent, S., and D. Kong, "A Profile for Route Origin Authorizations (ROAs)", RFC 6482, DOI 10.17487/RFC6482, February 2012, <https://www.rfc-editor.org/info/rfc6482>.

[RFC7153] Rosen, E. and Y. Rekhter, "IANA Registries for BGP Extended Communities", RFC 7153, DOI 10.17487/RFC7153, March 2014, <https://www.rfc-editor.org/info/rfc7153>.

[RFC7606] Chen, E., Ed., Scudder, J., Ed., Mohapatra, P., and K. Patel, "Revised Error Handling for BGP UPDATE Messages", RFC 7606, DOI 10.17487/RFC7606, August 2015, <https://www.rfc-editor.org/info/rfc7606>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

[RFC8955] Loibl, C., Hares, S., Raszuk, R., McPherson, D., and M. Bacher, "Dissemination of Flow Specification Rules", RFC 8955, DOI 10.17487/RFC8955, December 2020, <https://www.rfc-editor.org/info/rfc8955>.

[RFC8956] Loibl, C., Ed., Raszuk, R., Ed., and S. Hares, Ed., "Dissemination of Flow Specification Rules for IPv6", RFC 8956, DOI 10.17487/RFC8956, December 2020, <https://www.rfc-editor.org/info/rfc8956>.

[RFC9015] Farrel, A., Drake, J., Rosen, E., Uttaro, J., and L. Jalil, "BGP Control Plane for the Network Service Header in Service Function Chaining", RFC 9015, DOI 10.17487/RFC9015, June 2021, <https://www.rfc-editor.org/info/rfc9015>.

[RFC9117] Uttaro, J., Alcaide, J., Filsfils, C., Smith, D., and P. Mohapatra, "Revised Validation Procedure for BGP Flow Specifications", RFC 9117, DOI 10.17487/RFC9117, August 2021, <https://www.rfc-editor.org/info/rfc9117>.

[RFC9184] Loibl, C., "BGP Extended Community Registries Update", RFC 9184, DOI 10.17487/RFC9184, January 2022, <https://www.rfc-editor.org/info/rfc9184>.

11.2. Informative References

[I-D.ietf-idr-flowspec-v2] Hares, S., Eastlake, D. E., Yadlapalli, C., and S. Maduschke, "BGP Flow Specification Version 2", Work in Progress, Internet-Draft, draft-ietf-idr-flowspec-v2-04, 28 April 2024, <https://datatracker.ietf.org/doc/html/draft-ietf-idr-flowspec-v2-04>.

[RFC6241] Enns, R., Ed., Bjorklund, M., Ed., Schoenwaelder, J., Ed., and A. Bierman, Ed., "Network Configuration Protocol (NETCONF)", RFC 6241, DOI 10.17487/RFC6241, June 2011, <https://www.rfc-editor.org/info/rfc6241>.

[RFC8040] Bierman, A., Bjorklund, M., and K. Watsen, "RESTCONF Protocol", RFC 8040, DOI 10.17487/RFC8040, January 2017, <https://www.rfc-editor.org/info/rfc8040>.

[RFC8205] Lepinski, M., Ed. and K. Sriram, Ed., "BGPsec Protocol Specification", RFC 8205, DOI 10.17487/RFC8205, September 2017, <https://www.rfc-editor.org/info/rfc8205>.

[RFC8206] George, W. and S. Murphy, "BGPsec Considerations for Autonomous System (AS) Migration", RFC 8206, DOI 10.17487/RFC8206, September 2017, <https://www.rfc-editor.org/info/rfc8206>.

Author's Address

Susan Hares
Hickory Hill Consulting
7453 Hickory Hill
Saline, MI 48176
United States of America
Phone: +1-734-604-0332
Email: shares@ndzh.com

IDR Working Group
Internet-Draft
Intended status: Standards Track
Expires: 23 January 2025

S. Hares
Hickory Hill Consulting

22 July 2024

BGP Flow Specification Version 2 - More IP Filters
draft-hares-idr-fsv2-more-ip-filters-02

Abstract

The BGP flow specification version 2 (FSv2) for Basic IP defines user ordering of filters along with FSv1 IP Filters and FSv1 actions. This draft suggests additional IP Filters for Flow Specification FSv2.

Status of This Memo

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

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at https://datatracker.ietf.org/drafts/current/.

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."

This Internet-Draft will expire on 23 January 2025.

Table of Contents

1. Introduction

Version 2 of BGP flow specification was original defined in [I-D.ietf-idr-flowspec-v2] (denoted FSv2). However, the full FSv2 specification contains more than initial implementers desired. Therefore, this original FSv2 draft remains an WG draft, but the content will be split out into functions that implementers can manage. Section 1.4 contains the list of documents intended to be the split of the original FSv2 documents.

FSv2 specifies new user-ordered filters that will be used with the IPv4 (AFI=1) and IPv6 (AFI=2) 2 new SAFIs (TBD1, TBD2) for FSv2 to be used with 5 AFIs (1, 2, 6, 25, and 31) to allow user-ordered lists of traffic match filters for user-ordered traffic match actions encoded in Communities (Wide or Extended).

This draft specifies defines extensions to the FSv2 Basic IP package [I-D.hares-idr-fsv2-ip-basic]to support additional IP filters for IP packet and payload. The filters are passed in the Extended IP Filters (type 2) of the subTLVs. This filter form contains a filter version number so filters can be added easily.

BGP Flow Specifiction version 1 (FSv1) as defined in [RFC8955], [RFC8956], and [RFC9117] specified 2 SAFIs (133, 134) to be used with IPv4 AFI (AFI = 1) and IPv6 AFI (AFI=2). FSV2 specifies 2 new SAFIs (TBD1, TBD2) for FSv2 to be used with 5 AFIs (1, 2, 6, 25, and 31) to allow user-ordered lists of traffic match filters for user-ordered traffic match actions encoded in Communities (Wide or Extended). The first SAFI (TBD1) will be used for IP forwarding, and the second SAFI (TBD2) will be used with VPNs. The supported AFI/SAFI combinations in FSV2 are:

* IPV4 (AFI=1, SAFI=TBD1),

* IPv6 (AFI=2, SAFI=TBD1),

* L2 (AFI=6, SAFI=TBD1),

* SFC (AFI=31, SAFI=TBD1),

* BGP/MPLS IPv4 VPN (AFI=1, SAFI=TBD2),

* BGP/MPLS IPV6 VPN (AFI=2, SAFI=TBD2),

* BGP/MPLS L2VPN (AFI=25, SAFI=TBD2), and

* SFC VPN (AFI=31, SAFI=TBD2)

FSv2 specifies new IP filter that will be used with the IPv4 (AFI=1) and IPv6 (AFI=2) 2 new SAFIs (TBD1, TBD2) for FSv2 to be used with 5 AFIs (1, 2, 6, 25, and 31) to allow user-ordered lists of traffic match filters for user-ordered traffic match actions encoded in Communities (Wide or Extended). This document specifies IP filters used with IPvr (AFI=1) and IPv6 (AFI=2).

FSv1 and FSv2 use different AFI/SAFIs to send flow specification filters. Since BGP route selection is performed per AFI/SAFI, this approach can be termed â\200\234ships in the nightâ\200\235 based on AFI/SAFI.

Section 2 contains a description of the format of the FSv2 NLRI for the the Extended IP Filters type (type 2). Section 3 provides three new Filters approved in IDR WG drafts. Section 4 provides potential filters from individual drafts.

1.1. Definitions and Acronyms

AFI - Address Family Identifier

AS - Autonomous System

BGPSEC - secure BGP [RFC8205] updated by [RFC8206]

BGP Session ephemeral state - state which does not survive the loss of BGP peer session.

Configuration state - state which persist across a reboot of software module within a routing system or a reboot of a hardware routing device.

DDOs - Distributed Denial of Service.

Ephemeral state - state which does not survive the reboot of a software module, or a hardware reboot. Ephemeral state can be ephemeral configuration state or operational state.

FSv1 - Flow Specification version 1 [RFC8955] [RFC8956]

FSv2 - Flow Specification version 2 (this document)

NETCONF - The Network Configuration Protocol [RFC6241].

RESTCONF - The RESTCONF configuration Protocol [RFC8040]

RIB - Routing Information Base.

ROA - Route Origin Authentication [RFC6482]

RR - Route Reflector.

SAFI â\200\223 Subsequent Address Family Identifier

1.2. RFC 2119 language

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 BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals as shown here.

1.3. FSv2 Refesher

Note from Editor: This review section is here for the initial drafts to help with interim. It will be deleted as it is in [I-D.hares-idr-fsv2-ip-basic].

A BGP Flow Specification (version 1 or version 2) is an n-tuple containing one or more match criteria that can be applied to IP traffic, traffic encapsulated in IP traffic or traffic associated

with IP traffic. The following are examples of such traffic: IP packet or an IP packet inside a L2 packet (Ethernet), an MPLS packet, and SFC flow.

Flow Specification NLRI may be associated with a set of path attributes depending on the particular application to determine what happens upon matching the data flow filter. FSv1 and FSv2 support specifying the Extended Community specify a set of actions with a default order and known interactions. FSv2 also supports the ability to have user ordered actions by using the FSv2 type of Community BGP Path Attribute.

A particular application is identified by a specific AFI/SAFI (Address Family Identifier/Subsequent Address Family Identifier) and corresponds to a distinct set of RIBs. Those RIBs should be treated independently of each other in order to assure noninterference between distinct applications. FSv1 data is sent in a different NLRI than FSv2 NLRI.

BGP processing treats the NLRI as a key to entries in AFI/SAFI BGP databases. Entries that are placed in the Loc-RIB are then associated with a given set of semantics which are application dependent. Standard BGP mechanisms such as update filtering by NLRI or by attributes such as AS_PATH or large communities apply to the BGP Flow Specification defined NLRI-types.

Network operators can control the propagation of BGP routes by enabling or disabling the exchange of routes for a particular AFI/ SAFI pair on a particular peering session. As such, the Flow Specification may be distributed to only a portion of the BGP infrastructure.

Flow Specification v2 allows the user to order the flow specification rules and the actions associated with a rule. Each FSv2 rule may have one or more match conditions and one or more associated actions. The IDR WG draft [I-D.ietf-idr-flowspec-v2] contains the complete solution for FSv2. However, this complete solution makes implementation of these features a large task so, please see the next section on how the complete solution is broken into a series of solutions. This section describres the complete solution.

This FSv2 specification supports the components and actions for the following:

* IPv4 (AFI=1, SAFI=TBD1) [defined in FSv2-DDOS],

* IPv6 (AFI=2, SAFI=TBD2) [defined in FSv2-DDOS],

* L2 (AFI=6, SAFI=TDB1) [defined in FSv2-L2],

* BGP/MPLS IPv4 VPN: (AFI=1, SAFI=TBD2),

* BGP/MPLS IPv6 VPN: (AFI=2, SAFI=TBD2),

* BGP/MPLS L2VPN (AFI=25, SAFI=TDB2) [defined in FSv2-L2],

* SFC: (AFI=31, SAFI=TBD1) [defined in FSv2-SFC], and

* SFC VPN (AFI=31, SAFI=TBD2) [defined in FSv2-SFC].

The FSv2 specification for tunnel traffic is outside the scope of this specification. The FSv1 specification for tunneled traffic is in [I-D.ietf-idr-flowspec-nvo3]. The FSv2 tunnel traffic for FSv2 will be added to this list.

FSv2 operates in the ships-in-the night model with FSv1 so network operators can manipulate which the distribution of FSv2 and FSv1 using configuration parameters. Since the lack of deterministic ordering was an FSv1 problem, this specification provides rules and protocol features to keep filters in a deterministic order between FSv1 and FSv2.

The basic principles regarding ordering of flow specification filter rules are:

1) Rule-0 (zero) is defined to be 0/0 with the â\200\234permit-allâ\200\235 action.

2) FSv2 rules are ordered based on user-specified order.

- The user-specified order is carried in the FSv2 NLRI and a numerical lower value takes precedence over a numerically higher value. For rules received with the same order value, the FSv1 rules apply (order by component type and then by value of the components).

3) FSv2 rules are added starting with Rule 1 and FSv1 rules are added after FSv2 rules

- For example, BGP Peer A has FSv2 data base with 10 FSv2 rules (1-10). FSv1 user number is configured to start at 301 so 10 FSv1 rules are added at 301-310.

4) An FSv2 peer may receive BGP NLRI routes from a FSv1 peer or a BGP peer that does not support FSv1 or FSv2. The capabilities sent by a BGP peer indicate whether the AFI/SAFI can be received (FSv1 NLRI or FSv2 NLRI).

5) Associate a chain of actions to rules based on user-defined action number (1-n). (optional)

- If no actions are associated with a filter rule, the default is to drop traffic the filter rules match

- An action chain of 1-n actions can be associated with a set of filter rules can via Extended Communities or Wide Communities. Only Wide Communities can associate a user-defined order for the actions. Extended Community actions occur after actions with a user specified order (see section 5.2 for details).

Figure 2-2 provides a logical diagram of the FSv2 structure

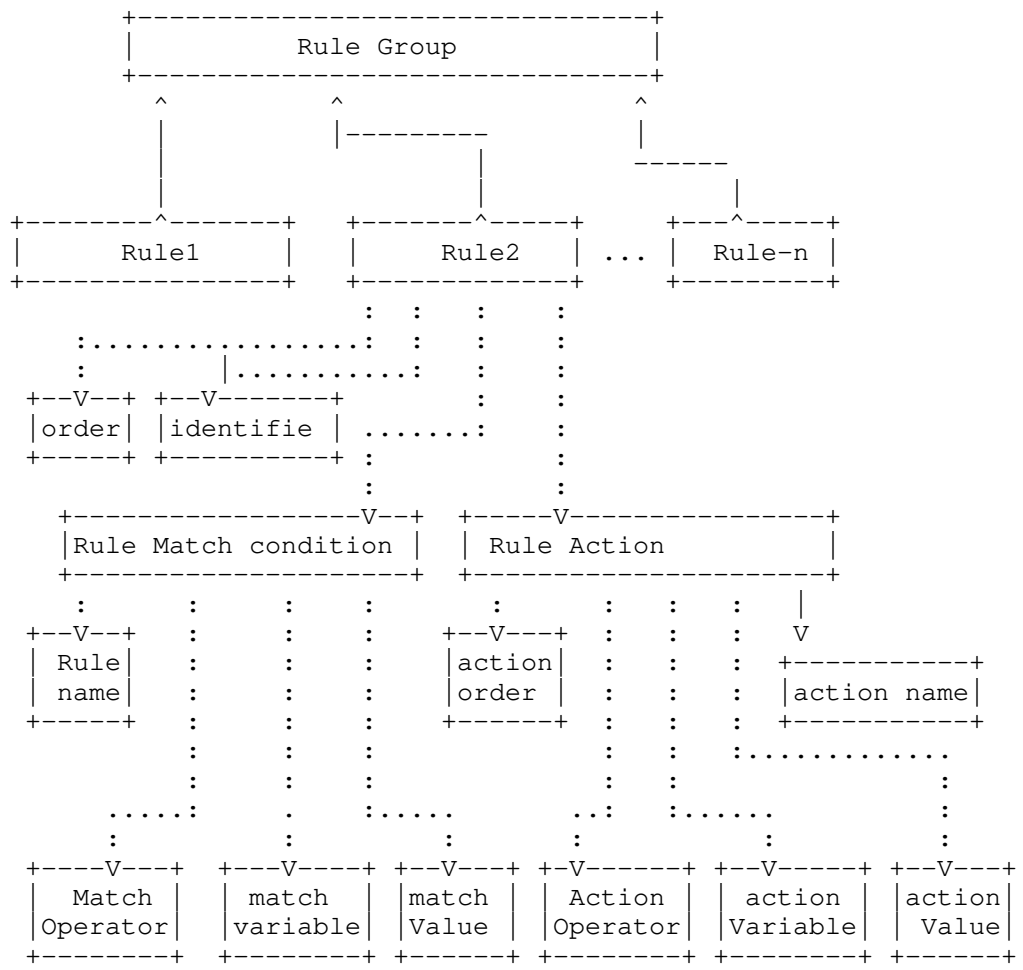


Figure 2-2: BGP FSv2 Data storage

1.4. FSv2 Series of Specifications

The full FSV2 information is contained in [I-D.ietf-idr-flowspec-v2].

Feedback from the implementers indicate that the Flow Specification v2 needs to broken into drafts based on the use cases the technology supports. These include IPv4/IPv6 IP Basic Filters for DDOS, IPv4/ IPv6 filters beyond DDOS, BGP/MPLS IPv4 VPN, BGP/MPLS IPv6 VPN, BGP/ MPLS L2VPN, Segment routing (SRMPLS, SRv6), SFC, SFC VPN, L2, L2 VPNs, and tunneled traffic (e.g., nv03 WG tunnels).

The following is the list of planned drafts:

FSv2 IP Basic: ([I-D.hares-idr-fsv2-ip-basic]) describes the minimal set of functions each FSv2 implementation must support.

FSv2 More IP Filters: (this document) defines a road map for additional IP filters. This base specification provides the format of the NLRI for IP Extended Filters" and gives templates for components used in the IP Extended Filters for FSv2. Temporarily, it list the IP Extended Filters - both filters approved by the IDR WG and the proposed filters.

FSv2 More IP Actions: ([I-D.hares-idr-fsv2-more-ip-actions]) This draft provides:

Template for additional FSv2 IP Actions in Extended Communities,

Template for additional FSv2 IP Actions in Community Path Attribute,

List of IDR WG approved IP Actions (in standardization process),

List of IP Actions proposed for IDR WG standarization)

FSv2 Non-IP Filters drafts: This group of drafts will include new proposals and revisions of the following existing IDR work:

FSv2 MPLS Filters and Actions:
((draft-hares-idr-fsv2-mpls-Filters) This base specification provides the format of the MPLS filters and actions.

FSv2 L2 Filters and Actions: Revision of [I-D.ietf-idr-flowspec-l2vpn] for FSv2.

FSv2 L2 Filters and Actions: Revision of [I-D.ietf-idr-flowspec-nvo3] for FSv2.

2. Extended IP Filters SubTLV

The format of the FSv2 NLRI field for IP Filters is defined in the original FSv2 draft [I-D.ietf-idr-flowspec-v2] and in the first of the FSv2 series drafts [I-D.hares-idr-fsv2-ip-basic]. As a review, the FSv2 NLRI with

The format of the NLRI for Basic IP Filters (type 1) is also defined in [I-D.hares-idr-fsv2-ip-basic]. This document defines the format of NLRI for the FSv2 Extended IP Filter type (type 2). Figure 3-1 provides the general header and Figure 3-2 provides the definition of the "value" portion. Figure 3-3 provides a diagram of the component types.

The key differences is that the extended IP filter types starts with a IP Filters identifier before SubTLVs with the filter components.

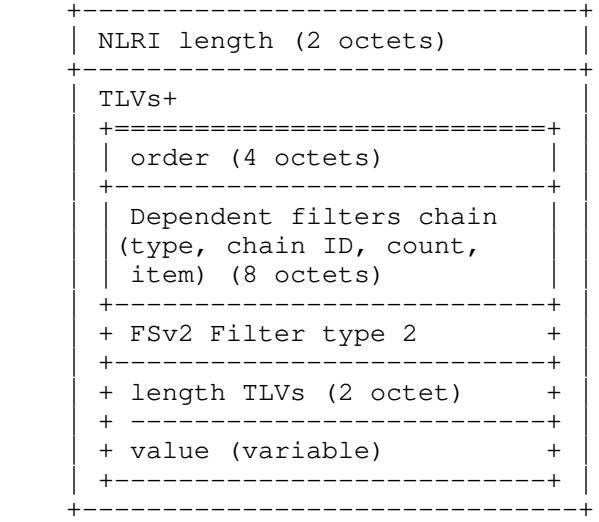


Figure 3-1 - FSv2 NLRI with Extended IP Filter type.

Where:

Dependent Filters Chain: 8 octets for identifying a chain of FSv2 filters that must be deployed at the same time.

Why needed in FSv2 filters: Flow spedcification filters distributed in BGP UPDATE packets may be broken into multiple packets. In FSv2, the dependent filter ID allows the filter chains to be identified across all user-defined or default filters. The rules can be installed from BGP into the firewall after all filters have been installed.

Components of the field The field has the following components:

version: (1 octet) identifies format of field (zero is reserved)

Chain ID: (3 octet): Identifier for filter chain (zero is reserved)

Count of items (2 octet): count of item on chains

item on chain: (2 octets): filter sequence number on chain

Where: the IP Filter type has a value field has a series of SubTLV as shown in figure 3-2.

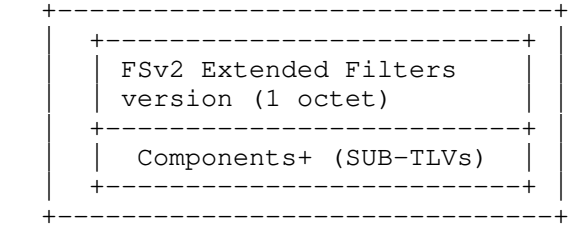


Figure 3-2 - FSv2 for Extended IP filters

Where: FSv2 Extended Filters version - gives a version number for the group of Extended IP Components supported. For example, the first version could support just the components listed in Table 3-1.

And Component SubTLV has the format of

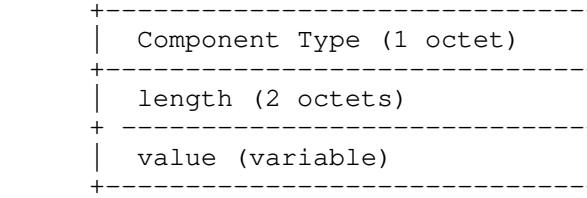


Figure 3-3 â\200\223 IP header SubTLV format

Where:

Component type: component values are defined in the â\200\234Flow Specification Component typesâ\200\235 registry for IPv4 and IPv6 by [RFC8955], [RFC8956], and [I-D.ietf-idr-flowspec-srv6]

length: length of SubTLV (varies depending on the component type)>

value: dependent on component type. The component types supported are based on the FSv2 filter version.

- The component types supported for FSv2 IP Extended Filters depends on the Extended Capability version.

- For descriptions of value portions for components 1-13 see [RFC8955] and [RFC8956]. Potential new filter components are listed in Table 3-3.

Table 3-1 Extended IP Filters Components

SubTLV -type	Definition
0 -	Reserved
1 -	IP Destination prefix
2 -	IP Source prefix
3 â\200\223	IPv4 Protocol / IPv6 Upper Layer Protocol
4 â\200\223	Port
5 â\200\223	Destination Port
6 â\200\223	Source Port
7 â\200\223	ICMPv4 type / ICMPv6 type
8 â\200\223	ICMPv4 code / ICPv6 code
9 â\200\223	TCP Flags
10 â\200\223	Packet length
11 â\200\223	DSCP
12 â\200\223	Fragment
13 â\200\223	Flow Label
14 -	TTL
15 -	Reserved
16 -	SID in Routing IPv6 Header
17 -	NRP-ID in Hop-by-Hop IPv6 Header
18 -	Payload component

Table 3-2 Extended IP Component Ranges (proposed)

Sub-TLV range	Definition
1-13	V1 filters
14-63	IP Extended Filters
64-150	Non-IP filters
151-180	Associated Data filters
181-191	Reserved
192-249	FCFS
250-255	Reserved

Ordering within the TLV in FSv2: The transmission of SubTLVs within a flow specification rule MUST be sent ascending order by SubTLV type. If the SubTLV types are the same, then the value fields are compared

using mechanisms defined in [RFC8955] and [RFC8956] and MUST be in ascending order. NLRIs having TLVs which do not follow the above ordering rules MUST be considered as malformed by a BGP FSv2 propagator. This rule prevents any ambiguities that arise from the multiple copies of the same NLRI from multiple BGP FSv2 propagators. A BGP implementation SHOULD treat such malformed NLRIs as "Treat-as-withdraw" [RFC7606].

See [RFC8955], [RFC8956], and [I-D.ietf-idr-flowspec-srv6]. for specific details.

3. Template for New FSv2 IP Filters

Summary: 1 line summary of function

Component ID: TBD-X1 (14)

Packet filtering:

This should be a choice of IPv4, IPv6, L2 Frame, MPLS frame, tunnel

What fitering in packet: specific field

Encoding: encoding of values in component. (below is example of v1 component)

<[numeric_op, value]+>

where:

- numeric_op - definition
- value - (give definition)

Ordering within component: by full value of number_op concatenated with value

dependency between components: no issues

conflict with other filters: none

reference document: [this document]

Examples in: in section (fill in section number)

4. Review of existing proposals (For review only)

This section provides examples of the use of templates for existing drafts. This section is for example only.

4.1. New Filter Components (IDR approved)

This approved Filters will be moved into individual drafts

4.1.1. TTL (type=TTL-Type (TBD1)

Summary: TTL filter defines matches for 8-bit TTL field in IP header

Component ID: TBD-X1

What packet filtering: IPv4

What fitering in packet: 8 bit TTL field

Encoding: <[numeric_op, value]+>

where:

- numeric_op - is defined by Flow Specification v1
- value is a 1 octet value for TTL.

ordering within component: by full value of number_op concatenated with value

dependency between components:

User ordering of filters can place this at any point in the filter chain.

Default component order: V1 ordering does not have TTL default need to be set by IDR WG. User ordering cn set this in order.

conflict: none

reference: draft-bergeon-flowspec-ttl-match-00.txt

examples in: section 6

4.1.2. Parts of SID (type=16 (0x40))

Summary: IPv6 Service Identifier (SRv6 SID) Matches ([I-D.ietf-idr-flowspec-srv6])

component ID: TBD-X2 (16)

What Packet filtering: IPv6

What filtering in IPv6 Packet: Segment Routing Header (SRH) ([RFC8402])

SID in SRH: [RFC8402] defines SRv6 Segment Identifier (SID) as an IPv6 address explicitly associated with the segment. [RFC8986] defines the SID format as: "LOC:FUNCT:ARG" where:

- locator (LOC) is encoded in the L most significant bits of the SID,
- followed by F bits of function (FUNCT), and
- A bits of arguments (ARG).

Encoding FSv2 Component: Parts of SID Filter: defines a list of match bit match criteria for some combinations of the LOC (location), FUNCT (function) and ARG (arguments) fields in the SID or or whole SID.

Length: variable

Component Value format: [type, LOC-Len, FUNCT-Len, ARG-Len, [op, value]+]

where:

- type (1 octet): This indicates the new component type (TBD1, which is to be assigned by IANA).
- LOC-Len (1 octet): This indicates the length in bits of LOC in SID.
- FUNCT-Len (1 octet): This indicates the length in bits of FUNCT in SID.
- ARG-Len (1 octet): This indicates the length in bits of ARG in SID.
- [op, value]+: This contains a list of {operator, value} pairs that are used to match some parts of SID.

The total of three lengths (i.e., LOC length + FUNCT length + ARG length) MUST NOT be greater than 128. If it is greater than 128, an error occurs and it is treated as a withdrawal [RFC7606] and [RFC4760].

The operator (op) byte is encoded as:

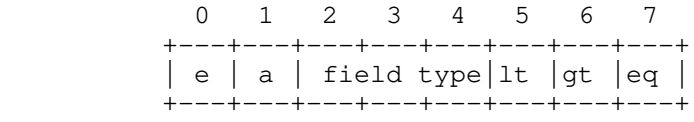


Figure 3-5

where:

- where the behavior of each operator bit has clear similarity with that of [RFC8955]'s Numeric Operator field.

- e (end-of-list bit): Set in the last {op, value} pair in the sequence.

- a - AND bit: If unset, the previous term is logically ORed with the current one. If set, the operation is a logical AND. It should be unset in the first operator byte of a sequence. The AND operator has higher priority than OR for the purposes of evaluating logical expressions.

- field type:

  - 000: SID's LOC

  - 001: SID's FUNCT

  - 010: SID's ARG

  - 011: SID's LOC:FUNCT (the concatenation of the LOC and FUNCTION fields)

  - 100: SID's FUNCT:ARG (the concatenation of the FUNCTION and ARG fields)

  - 101: SID's LOC:FUNCT:ARG (the concatenation of the FUNCTION and ARG fields)

- Note: For an unknown field type, Error Handling is to "treat as withdrawal" [RFC7606] and [RFC4760].

- lt: less than comparison between data' and value'.

- gt: greater than comparison between data' and value'.

- eq: equality between data' and value'.

The data' and value' used in lt, gt and eq are indicated by the field type in an operator and the value field following the operator.

The length of the value field depends on the field type and is the length of the SID parts being matched (see Table 3, Figure 3-6) in bytes, rounded up if that length is not a multiple of 8.

Table 3 - SID Parts fields

Field Type	Value
SID's LOC	value of LOC bits
SID's FUNCT	value of FUNCT bits
SID's ARG	value of ARG bits
SID's LOC:FUNCT	value of LOC:FUNCT bits
SID's FUNCT:ARG	value of FUNCT:ARG bits
SID's LOC:FUNCT:ARG	value of LOC:FUNCT:ARG bits

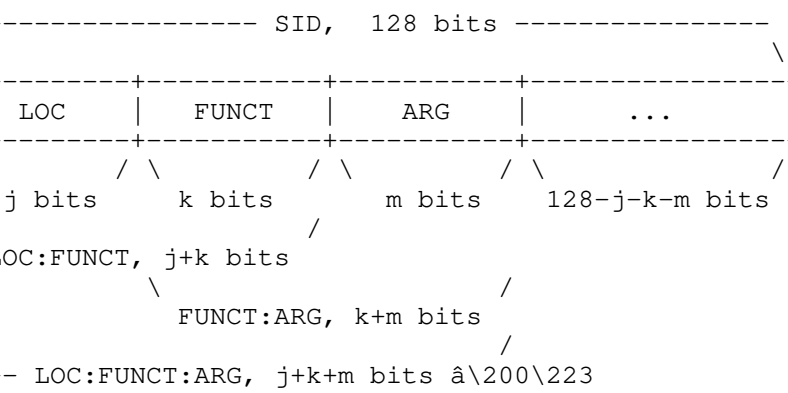


Figure 3-6

Dependency between components: TBD

conflicts between components: TBD

reference: [I-D.ietf-idr-flowspec-srv6]

Examples in: TBD

4.1.3. NRP ID Filter(type=17) (0x11)

Summary: Network Resource Partition ID Component

IP Packet filtering: IPv6

What filtering: IPv6 Hop-by-Hop Options Header ([RFC8402])

Description: Option in Next-Hop-Options header in IPv6 packet ([RFC8402], section 4). A Network Resource Partition (NRP) option carries around the network resource partition information (NRP) in the Hop-by-Hop options header ([I-D.ietf-6man-enhanced-vpn-vtn-id]). This IPv6 Extension head has:

Flags (flags): This is a 8 bit flag field in a single octet. One bit, "S" defined in most significant bit. The S stands for strict match of NRP ID field. The NRP Flags field is filtered for by the FSv2 component Flags field.

Context type (CT): - 1 octet field indicating the semantics and length of NRP-ID field. The value of CT=0 indicates a 4-octet NRP ID.

followed by F bits of function (FUNCT), and

A bits of arguments (ARG).

FSv2 NRP ID Component: Defines match for NRP ID in the NRP option of Hop-by-Hop Header. This FSv2 component has following format:

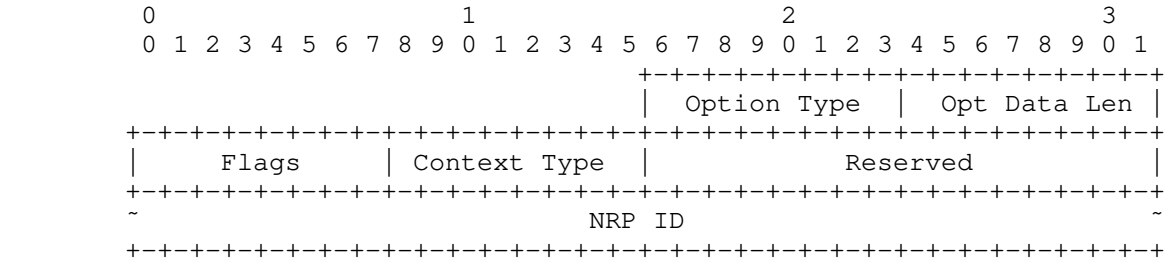


Figure: NRP FSv2 Component

Flags - This field is 2 octets with only the most signficant bit defined as Global Bit (g).

- Global bit (g): When set, it indicates the NRP ID to be matched with a globally unique NRP ID. Otherwise, the NRP-ID is to be a domain significant NRP ID. The global NRP ID has been coordinated among these domains.

Reserved: This a 2-octet field reserved for future use. It SHOULD be set to zero on transmission and MUST be ignored on receipt.

NRP ID: This is a 4-octet identifier which is used to identify an NRP

Interactions with: (TBD)

reference: [I-D.ietf-idr-flowspec-network-slice-ts]

4.2. Proposed Filter components

The documents in this section are proposed filters. Each of these proposals would be included in an individual draft.

4.2.1. IP Payloads Match type=18 (0x12))

Summary: IP Payload filter

IP Packet filtering: IPv4 or IPv6

What filtering in packet: Data in header or within the payload.

Encoding: The filter has an offset to filter data from the point specified in the "offset-type field" for using a filter of specific length (content-length) with a specific pattern (content). The type of packet IPv4 or IPv6 is specified in Type of IP packet.

The structure of the component is:

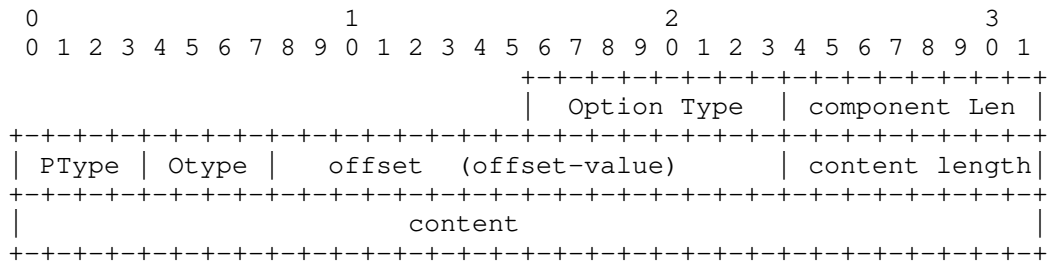


Figure 3-x: FSv2 IP Payload Match Component

Where the

- Ptype - 4 bit field indicating the packet type via AFI (IPv4 or IPv6)
  - IPv4 = 1
  - IPv6 = 2
- Otype - 4 bit field indicating the offset type where
  - 0 = IP header
  - 1 = IP header data
  - 2 = Data within TCP/UDP
- offset - is number of bytes to the payload from the point defined by Ptype and Otype.
- content length - length of the content.
- content - content filter field to match (significant field bit zero).

Ordering within component: (TBD)

interacts with components: (TBD)

reference: [I-D.cui-idr-content-filter-flowspec]

4.2.2. Group ID (type=19 (0x13))

Summary: Filter on Group ID

IP Packet filtering: IPv4 or IPv6

What filtering: Group ID specified sub-type

What Filtering in Packet: The filter looks for a specific type of group ID within either the IPv4 or IPv6 packet header.

Encoding of component: The structure of the component is the following

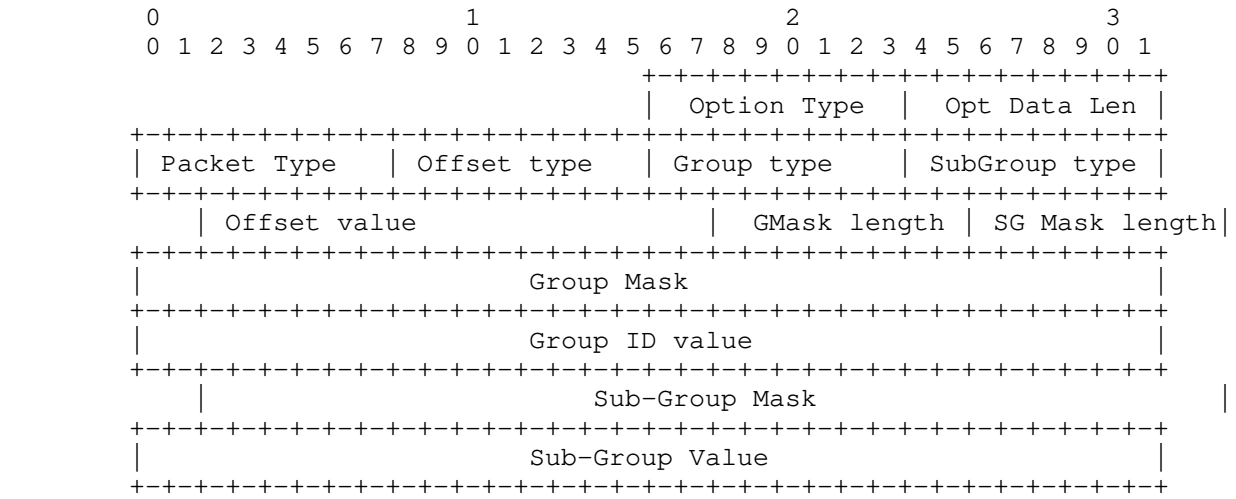


Figure 3-x: FSv2 IP Payload Match Component

Where the

- Packet type - 8 bit field indicating the packet type
  - o IPv4 = 1
  - o IPv6 = 2
- Offset type - 4 bit field indicating the offset type where
  - o 0 = IP header
  - o 1 = IP header data
  - o 2 = Data within TCP/UDP
- offset - is number of bytes to the payload from the point defined by Ptype and Otype.
- Group type - 1 octet field indicating the type of group ID
  - o 0 = Reserved
  - o 1 = Indirection ID
  - o 2 = Interface group
  - o 3 = CATS ID

- 4 = SAV ID
- 5 = APN ID

- Sub-Group type - Sub group within filters.
  - 0 = Reserved
  - 1 = data traffic (Inbound/outbound)
  - 1 = data traffic Inbound only
  - 2 = data traffic outbound only
- Group Mask - (variable) Group field mask
- Group ID value - (variable) Group ID value to match
- Sub Group Mask - (variable) Sub-Group Mask
- Sub-Group Value - (variable) Sub-Group value to match on

ordering within component: TBD

dependency between components: TBD

conflicts with other components: (TBD)

reference: TBD (this is just a sample).

5. IANA Considerations

This section complies with [RFC7153].

5.1. Filter IP Component types

IANA is requested to indicate [this draft] as a reference on the following assignments in the Flow Specification Component Types Registry:

ID	Name	Reference
14	TTL	[this document]
15	Partial SID	[draft-ietf-idr-flowspec-srv6] [this document]
16	NRP ID	[this document] [draft-ietf-idr-flowspec-network-slice-ts]
17	payload	[this document] [draft-cui-content-filter-flowspec-00]
18	Group ID	[this document] [draft-ietf-idr-flowspec-path-redirect] [draft-peng-idr-apn-bgp-flowspec] [draft-lin-idr-cats-flowspec-ts] [draft-geng-idr-flowspec-sav]

5.2. FSV2 Filter versions

IANA is requested to create the following three new egistries on a new "Flow Specification v2 Parametersâ\200\235 web page.

Name: BGP FSv2 Filter Version types
Reference: [this document]
Registration Procedures: 0x01-0x3F Standards Action.
0x40-0x6F FCFS
0x70-0xFF reserved

Type	Use	Reference
0x00	IP basic only	[this document] [FSv2 IP basic]
0x01	Extended IP Filters 1	[This document]

Figure 4-1

Name: BGP Group Types
Reference: [this document]
Registration Procedures: 0x01-0x3F Standards Action.
0x40-0x6F FCFS
0x70-0xFF reserved

Type	Use	Reference
0x00	reserved	[this document]
0x01	Indirection ID	[this document]
0x01	Interface Group	[this document]
0x02	CATs group	[this document]
0x03	SAVNet group	[this document]
0x04	APN group	[this document]

Figure 4-2 Groups

Name: BGP Sub Group Types
Reference: [this document]
Registration Procedures: 0x01-0x3F Standards Action.
0x40-0x5F FCFS
0x5F-0xFF reserved

Type	Use	Reference
0x00	Inbound/outbound	[this document]
0x01	Inbound on	[this document]
0x02	Outbound only	[this document]
0x03	SubGroup ID based	[this document]

figure 4-3 Sub-Group types

6. Security Considerations

The use of ROA improves on [RFC8955] by checking to see of the route origination. This check can improve the validation sequence for a multiple-AS environment.

>The use of BGPSEC [RFC8205] to secure the packet can increase security of BGP flow specification information sent in the packet.

The use of the reduced validation within an AS [RFC9117] can provide adequate validation for distribution of flow specification within a single autonomous system for prevention of DDoS.

Distribution of flow filters may provide insight into traffic being sent within an AS, but this information should be composite information that does not reveal the traffic patterns of individuals.

7. References

7.1. Normative References

[I-D.hares-idr-fsv2-ip-basic]
Hares, S., Eastlake, D., Yadlapalli, C., and S. Maduschke, "BGP Flow Specification Version 2 - for Basic IP", Work in Progress, Internet-Draft, draft-hares-idr-fsv2-ip-basic-03, 22 July 2024, <https://datatracker.ietf.org/api/v1/doc/document/draft-hares-idr-fsv2-ip-basic/>.

[I-D.hares-idr-fsv2-more-ip-actions]
Hares, S., "BGP Flow Specification Version 2 - More IP Actions", Work in Progress, Internet-Draft, draft-hares-idr-fsv2-more-ip-actions-01, 3 June 2024, <https://datatracker.ietf.org/doc/html/draft-hares-idr-fsv2-more-ip-actions-01>.

[I-D.hares-idr-fsv2-more-ip-filters]
Hares, S., "BGP Flow Specification Version 2 - More IP Filters", Work in Progress, Internet-Draft, draft-hares-idr-fsv2-more-ip-filters-01, 12 May 2024, <https://datatracker.ietf.org/doc/html/draft-hares-idr-fsv2-more-ip-filters-01>.

[I-D.ietf-6man-enhanced-vpn-vtn-id]
Dong, J., Li, Z., Xie, C., Ma, C., and G. S. Mishra, "Carrying Network Resource Partition (NRP) Information in IPv6 Extension Header", Work in Progress, Internet-Draft, draft-ietf-6man-enhanced-vpn-vtn-id-07, 8 July 2024, <https://datatracker.ietf.org/doc/html/draft-ietf-6man-enhanced-vpn-vtn-id-07>.

[I-D.ietf-idr-bgp-flowspec-label]
liangqiandeng, Hares, S., You, J., Raszuk, R., and D. Ma, "Carrying Label Information for BGP FlowSpec", Work in Progress, Internet-Draft, draft-ietf-idr-bgp-flowspec-label-02, 20 October 2022, <https://datatracker.ietf.org/doc/html/draft-ietf-idr-bgp-flowspec-label-02>.

[I-D.ietf-idr-flowspec-interfaceset]
Litkowski, S., Simpson, A., Patel, K., Haas, J., and L. Yong, "Applying BGP flowspec rules on a specific interface set", Work in Progress, Internet-Draft, draft-ietf-idr-flowspec-interfaceset-05, 18 November 2019, <https://datatracker.ietf.org/doc/html/draft-ietf-idr-flowspec-interfaceset-05>.

[I-D.ietf-idr-flowspec-l2vpn]
Weiguo, H., Eastlake, D. E., Litkowski, S., and S. Zhuang, "BGP Dissemination of L2 Flow Specification Rules", Work in Progress, Internet-Draft, draft-ietf-idr-flowspec-l2vpn-23, 15 April 2024, <https://datatracker.ietf.org/doc/html/draft-ietf-idr-flowspec-l2vpn-23>.

[I-D.ietf-idr-flowspec-mpls-match]
Yong, L., Hares, S., liangqiandeng, and J. You, "BGP Flow Specification Filter for MPLS Label", Work in Progress, Internet-Draft, draft-ietf-idr-flowspec-mpls-match-02, 20 October 2022, <https://datatracker.ietf.org/doc/html/draft-ietf-idr-flowspec-mpls-match-02>.

[I-D.ietf-idr-flowspec-network-slice-ts]
Dong, J., Chen, R., Wang, S., and J. Wenying, "BGP Flowspec for IETF Network Slice Traffic Steering", Work in Progress, Internet-Draft, draft-ietf-idr-flowspec-network-slice-ts-02, 4 March 2024, <https://datatracker.ietf.org/doc/html/draft-ietf-idr-flowspec-network-slice-ts-02>.

[I-D.ietf-idr-flowspec-nvo3]
Eastlake, D. E., Weiguo, H., Zhuang, S., Li, Z., and R. Gu, "BGP Dissemination of Flow Specification Rules for Tunneled Traffic", Work in Progress, Internet-Draft, draft-ietf-idr-flowspec-nvo3-20, 16 June 2024, <https://datatracker.ietf.org/doc/html/draft-ietf-idr-flowspec-nvo3-20>.

[I-D.ietf-idr-flowspec-path-redirect]
Van de Velde, G., Patel, K., and Z. Li, "Flowspec Indirection-id Redirect", Work in Progress, Internet-Draft, draft-ietf-idr-flowspec-path-redirect-12, 24 November 2022, <https://datatracker.ietf.org/doc/html/draft-ietf-idr-flowspec-path-redirect-12>.

[I-D.ietf-idr-flowspec-srv6]
Li, Z., Li, L., Chen, H., Loibl, C., Mishra, G. S., Fan, Y., Zhu, Y., Liu, L., and X. Liu, "BGP Flow Specification for SRv6", Work in Progress, Internet-Draft, draft-ietf-idr-flowspec-srv6-05, 29 March 2024, <https://datatracker.ietf.org/doc/html/draft-ietf-idr-flowspec-srv6-05>.

[I-D.ietf-idr-wide-bgp-communities]
Raszuk, R., Haas, J., Lange, A., Decraene, B., Amante, S., and P. Jakma, "BGP Community Container Attribute", Work in Progress, Internet-Draft, draft-ietf-idr-wide-bgp-communities-11, 9 March 2023, <https://datatracker.ietf.org/doc/html/draft-ietf-idr-wide-bgp-communities-11>.

[RFC0791] Postel, J., "Internet Protocol", STD 5, RFC 791, DOI 10.17487/RFC0791, September 1981, <https://www.rfc-editor.org/info/rfc791>.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[RFC3032] Rosen, E., Tappan, D., Fedorkow, G., Rekhter, Y., Farinacci, D., Li, T., and A. Conta, "MPLS Label Stack Encoding", RFC 3032, DOI 10.17487/RFC3032, January 2001, <https://www.rfc-editor.org/info/rfc3032>.

[RFC4271] Rekhter, Y., Ed., Li, T., Ed., and S. Hares, Ed., "A Border Gateway Protocol 4 (BGP-4)", RFC 4271, DOI 10.17487/RFC4271, January 2006, <https://www.rfc-editor.org/info/rfc4271>.

[RFC4360] Sangli, S., Tappan, D., and Y. Rekhter, "BGP Extended Communities Attribute", RFC 4360, DOI 10.17487/RFC4360, February 2006, <https://www.rfc-editor.org/info/rfc4360>.

[RFC4760] Bates, T., Chandra, R., Katz, D., and Y. Rekhter, "Multiprotocol Extensions for BGP-4", RFC 4760, DOI 10.17487/RFC4760, January 2007, <https://www.rfc-editor.org/info/rfc4760>.

[RFC5065] Traina, P., McPherson, D., and J. Scudder, "Autonomous System Confederations for BGP", RFC 5065, DOI 10.17487/RFC5065, August 2007, <https://www.rfc-editor.org/info/rfc5065>.

[RFC5701] Rekhter, Y., "IPv6 Address Specific BGP Extended Community Attribute", RFC 5701, DOI 10.17487/RFC5701, November 2009, <https://www.rfc-editor.org/info/rfc5701>.

[RFC6482] Lepinski, M., Kent, S., and D. Kong, "A Profile for Route Origin Authorizations (ROAs)", RFC 6482, DOI 10.17487/RFC6482, February 2012, <https://www.rfc-editor.org/info/rfc6482>.

[RFC7153] Rosen, E. and Y. Rekhter, "IANA Registries for BGP Extended Communities", RFC 7153, DOI 10.17487/RFC7153, March 2014, <https://www.rfc-editor.org/info/rfc7153>.

[RFC7606] Chen, E., Ed., Scudder, J., Ed., Mohapatra, P., and K. Patel, "Revised Error Handling for BGP UPDATE Messages", RFC 7606, DOI 10.17487/RFC7606, August 2015, <https://www.rfc-editor.org/info/rfc7606>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

[RFC8955] Loibl, C., Hares, S., Raszuk, R., McPherson, D., and M. Bacher, "Dissemination of Flow Specification Rules", RFC 8955, DOI 10.17487/RFC8955, December 2020, <https://www.rfc-editor.org/info/rfc8955>.

[RFC8956] Loibl, C., Ed., Raszuk, R., Ed., and S. Hares, Ed., "Dissemination of Flow Specification Rules for IPv6", RFC 8956, DOI 10.17487/RFC8956, December 2020, <https://www.rfc-editor.org/info/rfc8956>.

[RFC9015] Farrel, A., Drake, J., Rosen, E., Uttaro, J., and L. Jalil, "BGP Control Plane for the Network Service Header in Service Function Chaining", RFC 9015, DOI 10.17487/RFC9015, June 2021, <https://www.rfc-editor.org/info/rfc9015>.

[RFC9117] Uttaro, J., Alcaide, J., Filsfils, C., Smith, D., and P. Mohapatra, "Revised Validation Procedure for BGP Flow Specifications", RFC 9117, DOI 10.17487/RFC9117, August 2021, <https://www.rfc-editor.org/info/rfc9117>.

[RFC9184] Loibl, C., "BGP Extended Community Registries Update", RFC 9184, DOI 10.17487/RFC9184, January 2022, <https://www.rfc-editor.org/info/rfc9184>.

7.2. Informative References

[I-D.cui-idr-content-filter-flowspec]
Cui, Y., Gao, Y., and S. Hares, "Packet Content Filter for BGP FlowSpec", Work in Progress, Internet-Draft, draft-cui-idr-content-filter-flowspec-02, 19 July 2024, <https://datatracker.ietf.org/api/v1/doc/document/draft-cui-idr-content-filter-flowspec/>.

[I-D.ietf-idr-flowspec-v2]
Hares, S., Eastlake, D. E., Yadlapalli, C., and S. Maduschke, "BGP Flow Specification Version 2", Work in Progress, Internet-Draft, draft-ietf-idr-flowspec-v2-04, 28 April 2024, <https://datatracker.ietf.org/doc/html/draft-ietf-idr-flowspec-v2-04>.

[RFC6241] Enns, R., Ed., Bjorklund, M., Ed., Schoenwaelder, J., Ed., and A. Bierman, Ed., "Network Configuration Protocol (NETCONF)", RFC 6241, DOI 10.17487/RFC6241, June 2011, <https://www.rfc-editor.org/info/rfc6241>.

[RFC8040] Bierman, A., Bjorklund, M., and K. Watsen, "RESTCONF Protocol", RFC 8040, DOI 10.17487/RFC8040, January 2017, <https://www.rfc-editor.org/info/rfc8040>.

[RFC8205] Lepinski, M., Ed. and K. Sriram, Ed., "BGPsec Protocol Specification", RFC 8205, DOI 10.17487/RFC8205, September 2017, <https://www.rfc-editor.org/info/rfc8205>.

[RFC8206] George, W. and S. Murphy, "BGPsec Considerations for Autonomous System (AS) Migration", RFC 8206, DOI 10.17487/RFC8206, September 2017, <https://www.rfc-editor.org/info/rfc8206>.

[RFC8300] Quinn, P., Ed., Elzur, U., Ed., and C. Pignataro, Ed., "Network Service Header (NSH)", RFC 8300, DOI 10.17487/RFC8300, January 2018, <https://www.rfc-editor.org/info/rfc8300>.

[RFC8402] Filsfils, C., Ed., Previdi, S., Ed., Ginsberg, L., Decraene, B., Litkowski, S., and R. Shakir, "Segment Routing Architecture", RFC 8402, DOI 10.17487/RFC8402, July 2018, <https://www.rfc-editor.org/info/rfc8402>.

[RFC8986] Filsfils, C., Ed., Camarillo, P., Ed., Leddy, J., Voyer, D., Matsushima, S., and Z. Li, "Segment Routing over IPv6 (SRv6) Network Programming", RFC 8986, DOI 10.17487/RFC8986, February 2021, <https://www.rfc-editor.org/info/rfc8986>.

Author's Address

Susan Hares
Hickory Hill Consulting
7453 Hickory Hill
Saline, MI 48176
United States of America
Phone: +1-734-604-0332
Email: shares@ndzh.com

IDR Working Group
Internet-Draft
Intended status: Standards Track
Expires: 16 February 2025

S. Hares
Hickory Hill Consulting
D. Eastlake
Independent
C. Yadlapalli
ATT
S. Maduscke
Verizon

15 August 2024

BGP Flow Specification Version 2 - for Basic IP
draft-ietf-idr-fsv2-ip-basic-00

Abstract

BGP flow specification version 1 (FSv1), defined in RFC 8955, RFC 8956, and RFC 9117 describes the distribution of traffic filter policy (traffic filters and actions) distributed via BGP. During the deployment of BGP FSv1 a number of issues were detected, so version 2 of the BGP flow specification (FSv2) protocol addresses these features. In order to provide a clear demarcation between FSv1 and FSv2, a different NLRI encapsulates FSv2.

The IDR WG requires two implementation Implementers feedback on FSv2 was that FSv2 has a correct design, but that breaking FSv2 into a progression of documents would aid deployment of the draft. The IDR WG requires two implementation so This document is the first of the series of documents indicating the basic FSv2 with user ordering of filters added to FSv1 IP Filters and IP actions.

Status of This Memo

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

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at https://datatracker.ietf.org/drafts/current/.

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."

This Internet-Draft will expire on 16 February 2025.

Table of Contents

1. Introduction

Version 2 of BGP flow specification was original defined in [I-D.ietf-idr-flowspec-v2] (BGP FSv2). In this document it will be refered to as FSv2.

The FSv2 specification was consider technically correct, but it contains more than the initial implementers desired. Why? The IDR WG requires two implementations of any specification. Therefore, the original FSv2 draft will remain a WG draft, but the content will be split out into functions that implementers can incrementally deploy.

This draft provides the FSv2 specification for transmitting user-ordered Basic IP filters with FSv2 actions in set of Extended Communities. These extended communities have either the pre-defined set of ordering and interactions, or a implementation specific set ordering. FSv2 filters and actions are defined in setion 3.

FSv2 is an update to BGP Flow specification version 1 (BGP FSv1). BGP FSv1 as defined in [RFC8955], [RFC8956], and [RFC9117] specified 2 SAFIs (133, 134) to be used with IPv4 AFI (AFI = 1) and IPv6 AFI (AFI=2). In this document it will be refered to as FS

This document specifies 2 new SAFIs (TBD1, TBD2) for FSv2 to be used with 5 AFIs (1, 2, 6, 25, and 31) to allow user-ordered lists of traffic match filters for user-ordered traffic match actions encoded in Communities (Wide or Extended).

FSv1 and FSv2 use different AFI/SAFIs to send flow specification filters. Since BGP route selection is performed per AFI/SAFI, this approach can be termed â\200\234ships in the nightâ\200\235 based on AFI/SAFI.

1.1. Why Flow Specification v2

Modern IP routers have the capability to forward traffic and to classify, shape, rate limit, filter, or redirect packets based on administratively defined policies. These traffic policy mechanisms allow the operator to define match rules that operate on multiple fields within header of an IP data packet. The traffic policy allows actions to be taken upon a match to be associated with each match

rule. These rules can be more widely defined as â\200\234event-condition-actionâ\200\235 (ECA) rules where the event is always the reception of a packet.

BGP ([RFC4271]) flow specification as defined by [RFC8955], [RFC8956], [RFC9117] specifies the distribution of traffic filter policy (traffic filters and actions) via BGP to a mesh of BGP peers (IBGP and EBGP peers). The traffic filter policy is applied when packets are received on a router with the flow specification function turned on. The flow specification protocol defined in [RFC8955], [RFC8956], and [RFC9117] will be called BGP flow specification version 1 (BGP FSv1) in this draft.

Some modern IP routers also include the abilities of firewalls which can match on a sequence of packet events based on administrative policy. These firewall capabilities allow for user ordering of match rules and user ordering of actions per match.

Multiple deployed applications currently use BGP FSv1 to distribute traffic filter policy. These applications include: 1) mitigation of Denial of Service (DoS), 2) traffic filtering in BGP/MPLS VPNS, and 3) centralized traffic control for networks utilizing SDN control of router firewall functions, 4) classifiers for insertion in an SFC, and 5) filters for SRv6 (segment routing v6).

During the deployment of BGP flow specification v1, the following issues were detected:

* lack of consistent TLV encoding prevented extension of encodings,

* inability to allow user defined order for filtering rules,

* inability to order actions to provide deterministic interactions or to allow users to define order for actions, and

* no clearly defined mechanisms for BGP peers which do not support flow specification v1.

Networks currently cope with some of these issues by limiting the type of traffic filter policy sent in BGP. Current Networks do not have a good workaround/solution for applications that receive but do not understand FSv1 policies.

FSv1 is a critical component of deployed applications. Therefore, this specification defines how FSv2 will interact with BGP peers that support either FSv2, FSv1, FSv2 and FSv1,or neither of them. It is expected that a transition to FSv2 will occur over time as new applications require FSv2 extensibility and user-defined ordering for rules and actions or network operators tire of the restrictions of FSv1 such as error handling issues and restricted topologies.

Section 2 contains a Primer on FSv1, FSv2, and the FSv2 seriess of specifications. Section 3 contains the encoding rules for FSv2 and user-based encoding sent via BGP. Section 4 describes how to validate and order FSv2 NLRI. Sections 5-8 discusses scalability, optional security additions, security considerations, and IANA considerations.

1.2. Definitions and Acronyms

AFI - Address Family Identifier

AS - Autonomous System

BGPSEC - secure BGP [RFC8205] updated by [RFC8206]

BGP Session ephemeral state - state which does not survive the loss of BGP peer session.

Configuration state - state which persist across a reboot of software module within a routing system or a reboot of a hardware routing device.

DDOs - Distributed Denial of Service.

Ephemeral state - state which does not survive the reboot of a software module, or a hardware reboot. Ephemeral state can be ephemeral configuration state or operational state.

FSv1 - Flow Specification version 1 [RFC8955] [RFC8956]

FSv2 - Flow Specification version 2 (this document)

NETCONF - The Network Configuration Protocol [RFC6241].

RESTCONF - The RESTCONF configuration Protocol [RFC8040]

RIB - Routing Information Base.

ROA - Route Origin Authentication [RFC9582]

RR - Route Reflector.

SAFI â\200\223 Subsequent Address Family Identifier

1.3. RFC 2119 language

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 BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals as shown here.

2. Flow Specification Version 2 Primer

A BGP Flow Specification (v1 or v2) is an n-tuple containing one or more match criteria that can be applied to IP traffic, traffic encapsulated in IP traffic or traffic associated with IP traffic. The following are examples of such traffic: IP packet or an IP packet inside a L2 packet (Ethernet), an MPLS packet, and SFC flow.

A given Flow Specification NLRI may be associated with a set of path attributes depending on the particular application, and attributes within that set may or may not include reachability information (e.g., NEXT_HOP). FSv1 and FSv2-DDOS use only the Extended Community to encode a set of pre-determined actions. The full FSv2 uses either Extended Communities or Wide Communities to encode actions.

A particular application is identified by a specific AFI/SAFI (Address Family Identifier/Subsequent Address Family Identifier) and corresponds to a distinct set of RIBs. Those RIBs should be treated independently of each other in order to assure noninterference between distinct applications.

BGP processing treats the NLRI as a key to entries in AFI/SAFI BGP databases. Entries that are placed in the Loc-RIB are then associated with a given set of semantics which are application dependent. Standard BGP mechanisms such as update filtering by NLRI or by attributes such as AS_PATH or large communities apply to the BGP Flow Specification defined NLRI-types.

Network operators can control the propagation of BGP routes by enabling or disabling the exchange of routes for a particular AFI/SAFI pair on a particular peering session. As such, the Flow Specification may be distributed to only a portion of the BGP infrastructure.

2.1. Flow Specification v1 (FSv1) Overview

The FSv1 NLRI defined in [RFC8955] and [RFC8956] include 13 match conditions encoded for the following AFI/SAFIs:

* IPv4 traffic: AFI:1, SAFI:133

* IPv6 Traffic: AFI:2, SAFI:133

* BGP/MPLS IPv4 VPN: AFI:1, SAFI: 134

* BGP/MPLS IPv6 VPN: AFI:2, SAFI: 134

If one considers the reception of the packet as an event, then BGP FSv1 describes a set of Event-MatchCondition-Action (ECA) policies where:

* event is the reception of a packet,

* condition stands for â\200\234match conditionsâ\200\235 defined in the BGP NLRI as an n-tuple of component filters, and

* the action is either: the default condition (accept traffic), or a set of actions (1 or more) defined in Extended BGP Community values [RFC4360].

The flow specification conditions and actions combine to make up FSv1 specification rules. Each FSv1 NLRI must have a type 1 component (destination prefix). Extended Communities with FSv1 actions can be attached to a single NLRI or multiple NLRIs in a BGP message

Within an AFI/SAFI pair, FSv1 rules are ordered based on the components in the packet (types 1-13) ordered from left-most to right-most and within the component types by value of the component. Rules are inserted in the rule list by component-based order where an FSv1 rule with existing component type has higher precedence than one missing a specific component type,

Since FSv1 specifications ([RFC8955], [RFC8956], and [RFC9117]) specify that the FSv1 NLRI MUST have a destination prefix (as component type 1) embedded in the flow specification, the FSv1 rules with destination components are ordered by IP Prefix comparison rules for IPv4 ([RFC8955]) and IPv6 ([RFC8956]). [RFC8955] specifies that more specific prefixes (aka longest match) have higher precedence than that of less specific prefixes and that for prefixes of the same length the lower IP number is selected (lowest IP value). [RFC8955] specifies that if the offsets within component 1 are the same, then the longest match and lowest IP comparison rules from [RFC8955] apply. If the offsets are different, then the lower offset has precedence.

These rules provide a set of FSv1 rules ordered by IP Destination Prefix by longest match and lowest IP address. [RFC8955] also states that the requirement for a destination prefix component â\200\234MAY be relaxed by explicit configurationâ\200\235 Since the rule insertions are based on comparing component types between two rules in order, this means the rules without destination prefixes are inserted after all rules which contain destination prefix component.

The actions specified in FSv1 are:

* accept packet (default),

* traffic flow limitation by bytes (0x6),

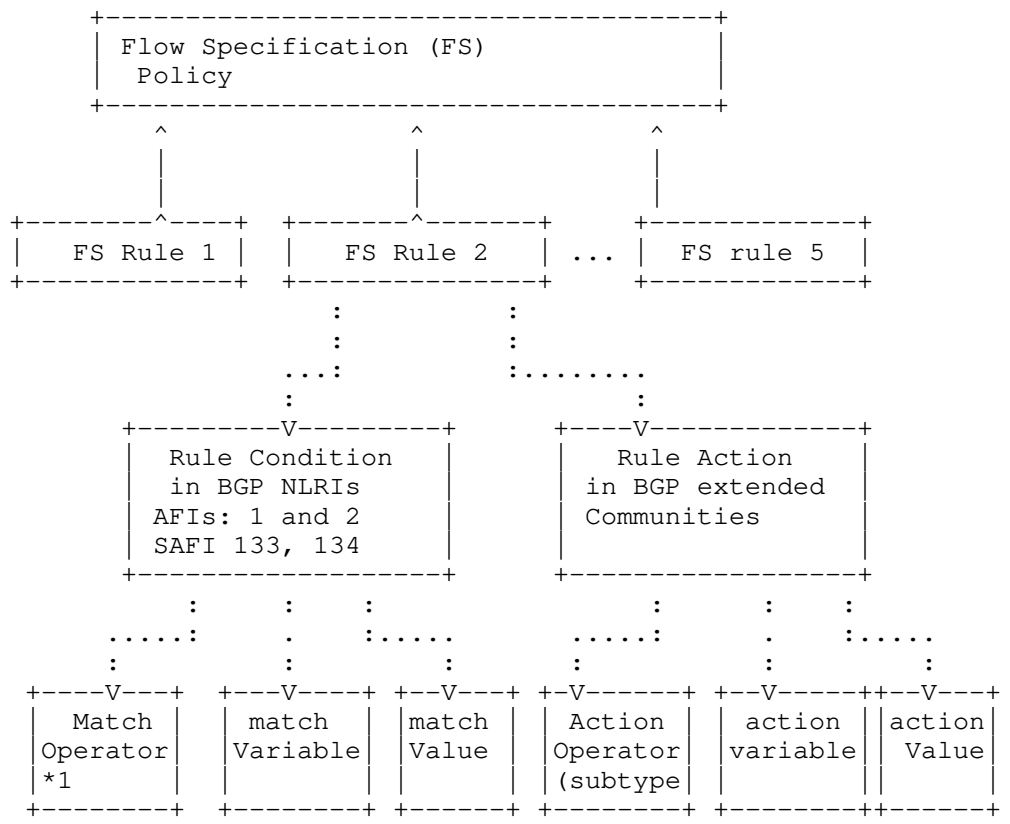
* traffic-action (0x7),

* redirect traffic (0x8),

* mark traffic (0x9), and

* traffic flow limitation by packets (12, 0xC)

Figure 1 shows a diagram of the FSv1 logical data structures with 5 rules. If FSv1 rules have destination prefix components (type=1) and FSv1 rule 5 does not have a destination prefix, then FSv1 rule 5 will be inserted in the policy after rules 1-4.



*1 match operator may be complex.

Figure 2-1: BGP Flow Specification v1 Policy

2.2. FSv2 Overview

FSv2 allows the user to order the flow specification rules and the actions associated with a rule. Each FSv2 rule may have one or more match conditions and one or more associated actions. The IDR WG draft [I-D.ietf-idr-flowspec-v2] contains the complete solution for FSv2. However, this complete solution makes implementation of these features a large task so, please see the next section on how the complete solution is broken into a series of solutions. This section describres the complete solution.

The original FSv2 specification [I-D.ietf-idr-flowspec-v2] supports the components and actions for the following:

* IPv4 (AFI=1, SAFI=TBD1),

* IPv6 (AFI=2, SAFI=TBD2),

* L2 (AFI=6, SAFI=TDB1) [described in [I-D.ietf-idr-flowspec-l2vpn]),

* BGP/MPLS IPv4 VPN: (AFI=1, SAFI=TBD2),

* BGP/MPLS IPv6 VPN: (AFI=2, SAFI=TBD2),

* BGP/MPLS L2VPN (AFI=25, SAFI=TDB2) [described in [I-D.ietf-idr-flowspec-l2vpn]),

* SFC: (AFI=31, SAFI=TBD1),

* SFC VPN (AFI=31, SAFI=TBD2),

The IDR specification for L2 VPN traffic was specified in [I-D.ietf-idr-flowspec-l2vpn]. An IDR specification for tunneled traffic is in [I-D.ietf-idr-flowspec-nvo3]. Both of these drafts were targeted for FSv1, but the WG decided to implement these as FSv2. The series of FSv2 support the same scope of functionality in a series of documents.

FSv2 operates in the ships-in-the night model with FSv1 so network operators can manipulate which the distribution of FSv2 and FSv1 using configuration parameters. Since the lack of deterministic ordering was an FSv1 problem, this specification provides rules and protocol features to keep filters in a deterministic order between FSv1 and FSv2.

The basic principles regarding ordering of flow specification filter rules are:

1) Rule-0 (zero) is defined to be 0/0 with the â\200\234permit-allâ\200\235 action.

2) FSv2 rules are ordered based on user-specified order.

- The user-specified order is carried in the FSv2 NLRI and a numerical lower value takes precedence over a numerically higher value. For rules received with the same order value, the FSv1 rules apply (order by component type and then by value of the components).

3) FSv2 rules are added starting with Rule 1 and FSv1 rules are added after FSv2 rules

- For example, BGP Peer A has FSv2 data base with 10 FSv2 rules (1-10). FSv1 user number is configured to start at 301 so 10 FSv1 rules are added at 301-310.

4) An FSv2 peer may receive BGP NLRI routes from a FSv1 peer or a BGP peer that does not support FSv1 or FSv2. The capabilities sent by a BGP peer indicate whether the AFI/SAFI can be received (FSv1 NLRI or FSv2 NLRI).

5) Associate a chain of actions to rules based on user-defined action number (1-n). (optional)

- If no actions are associated with a filter rule, the default is to drop traffic the filter rules match

- An action chain of 1-n actions can be associated with a set of filter rules can via Extended Communities or a Community attribute with a FSv2 type. Only the Community attribute allows for user-defined order for the actions. If an implementation allows for FSv2 actions with user-ordering and Extended Community actions, the by default the Extended Community are ordered after the user-ordered actions. This FSv2 action order default can be changed by the Action Chain Ordering FSv2 action.

Figure 2-2 provides a logical diagram of the FSv2 structure

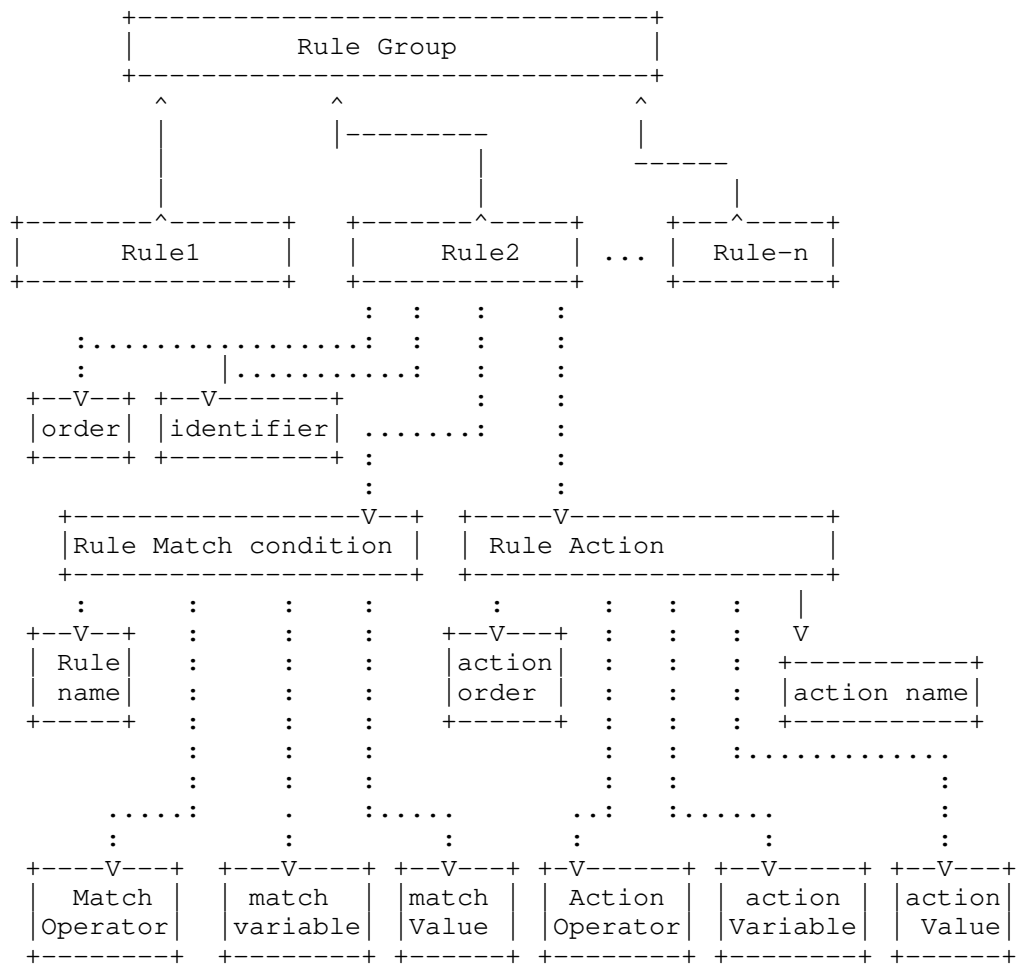


Figure 2-2: BGP FSv2 Data storage

2.3. Flow Specification v2 (FSv2) Series of Specifications

The full FSV2 information is contained in [I-D.ietf-idr-flowspec-v2].

Feedback from the implementers indicate that the Flow Specification v2 needs to broken into drafts based on the use cases the technology supports. These include IPv4/IPv6 IP Basic Filters for DDOS, IPv4/ IPv6 filters beyond DDOS, BGP/MPLS IPv4 VPN, BGP/MPLS IPv6 VPN, BGP/ MPLS L2VPN, Segment routing (SRMPLS, SRv6), SFC, SFC VPN, L2, L2 VPNs, and tunneled traffic (e.g., nv03 WG tunnels).

The following is the list of planned drafts:

FSv2 IP Basic: This document specified the minimal support for FSv2 that all other FSv2 specifications will extended It defines an NLRI format for the filters, Extended Community actions supported by [RFC8955] and [RFC8956], and user ordering of IP Filters. This FSv2 draft defines the order that these basic Extended Community actions defined in [RFC8955] and [RFC8956] are preformed. This specification also defines:

* how to handle a filter list with inter-filter dependencies,

* how to handle a chain of action with inter-dependencies,

FSv2 More IP Filters This group of specifications will extend FSv2 IP Basic to add more filters. A road map draft ([I-D.hares-idr-fsv2-more-ip-filters]) provides details on how to specify additional IP filters. This includes

* Format for Extended IP filters TLV,

* IDR Approved Filter Components (TTL, SID, NRP IP) that will be moved into individual specifications,

* Proposed IP Filter components (IP Payloads and Group ID) that are currently defined in individual specifications.

FSv2 More IP Actions This group of specification will extend the FSv2 IP actions in Extended Communities and the Community Path Attribute. A road map draft ([I-D.hares-idr-fsv2-more-ip-actions]) provides details on how to specify additional Actions, and a summary of current actions specified or proposed.

FSv2 Non-IP Filters and Actions This group of specifiations define non-IP Filters and non-IP Actions. These non-IP filter rules include the following filters and actions. The Filters are:

MPLS filters: This document contains MPLS component filters to match labels. Original IDR work is found in [I-D.ietf-idr-flowspec-v2] from [I-D.ietf-idr-flowspec-mpls-match]. Additional work from SR-MPLS is included in this category. A simple set of MPLS Label match components are provided in this draft.

FSv2 L2 filters: The current FSv2 work on L2 includes work on

L2VPNs ([I-D.ietf-idr-flowspec-l2vpn]). Other drafts have suggested extending this to cover the reduced latency L2 use case (detnet). This draft provides a discussion of how to integrate this work initially done for FSv1 into the FSv2 user-ordered filters.

FSV2 filters SFC direction: Network Service Header (NSH) is defined in [RFC8300]. Flow specification filters were not defined in [RFC9015], but the FSv2 provide a template for adding NSH filters.

Tunnels Defined by nv03 group An IDR draft was approved for FSv1 encoding of tunnel overlays (see [I-D.ietf-idr-flowspec-nvo3]). This draft contains a discussion of how to integrate this work initially done for FSv1 into the FSv2 user-ordered filters.

The actions are:

FSV2 for MPLS: MPLS actions to push, pop, swap labels. Original IDR work is found in [I-D.ietf-idr-flowspec-v2] from [I-D.ietf-idr-bgp-flowspec-label]. New MPLS actions for

FSV2 actions for SFC: SFC classifier actions based on Action with Service Path identifier (SPI), Service Index (SI), and Service function type (SFT). The original description of the action is in [RFC9015] in section 7.4.

FSv2 L2VPN actions: The L2 filters for packets in L2 or L2VPN Actions were defined for FSv1 in ([I-D.ietf-idr-flowspec-l2vpn]).

Tunnels actions The tunnel actions were defined for FSv1 in [I-D.ietf-idr-flowspec-nvo3].

3. FSv2 NLRI Formats and Actions

3.1. FSv2 NLRI Format

The BGP FSv2 uses an NRLI with the format for AFIs for IPv4 (AFI = 1), IPv6 (AFI = 2), L2 (AFI = 6), L2VPN (AFI=25), and SFC (AFI=31) with SAFIs TBD1 and TBD2 to support transmission of the flow specification which supports user ordering of traffic filters and actions for IP traffic and IP VPN traffic.

This NLRI information is encoded using MP_REACH_NLRI and MP_UNREACH_NLRI attributes defined in [RFC4760]. When advertising FSv2 NLRI, the length of the Next-Hop Network Address MUST be set to 0. Upon reception, the Network Address in the Next-Hop field MUST be ignored.

Implementations wishing to exchange flow specification rules MUST use BGP's Capability Advertisement facility to exchange the Multiprotocol Extension Capability Code (Code 1) as defined in [RFC4760], and indicate a capability for FSv1, FSv2 (Code TBD3), or both.

The AFI/SAFI NLRI for BGP Flow Specification version 2 (FSv2) has the format:

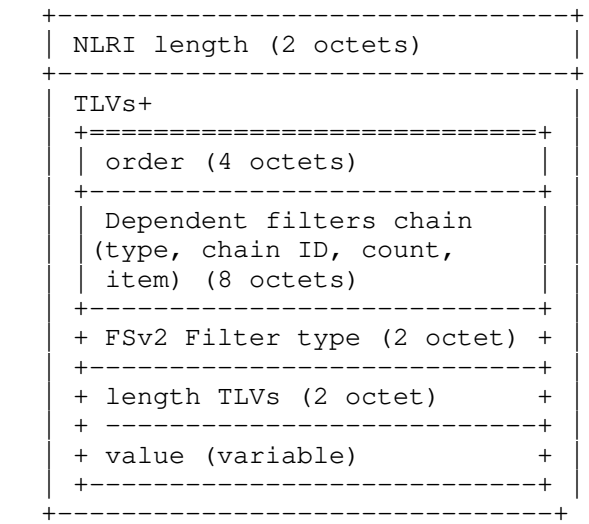


Figure 3-1 – NLRI format for FSv2

where:

* TLV+ – indicates the repetition of the TLV field

* NLRI length: length of field including all SubTLVs in octets.

* order: flow-specification global rule order number (4 octets).

* Dependent Filters Chain: 8 octets for identifying a chain of FSv2 filters that must be deployed at the same time.

  Why needed in FSv2: Flow spedcification filters distributed in

BGP UPDATE packets may be broken into multiple packets. In FSv2, the dependent filter ID allows the filter chains to be identified across all user-defined or default filters. The rules can be installed from BGP into the firewall after all filters have been installed.

For basic FSV2: This field is required to be set to all zero, and ignored upon reception.

For future FSV2: Future specifications will specify the use of this field. The chain will be designed so the "all zeros" value is ignored.

* FSv2 Filter type: contains a type for FSv2 TLV format of the NRLI (2 octets) which can be:
  - 0 - reserved,
  - 1 - IP Basic Filter Rules
  - 2 - Extended IP Filter rules
  - 3 - MPLS Traffic Rules
  - 4 - L2 traffic rules
  - 5- SFC Traffic rules
  - 6 - Tunneled traffic
* length-TLV: is the length of the value part of the Sub-TLV,
* value: value depends on the type of FSv2 Filter type.

All FSv2 function must recognize valid Filter Types, even if the handling of the Filter types are not supported by the implementation. The TLV allows all FSv2 Filter types to be passed, even if the Filter rules cannot be installed.

Note: This specification only defines the IP Basic Filter Rules that all FSv2 must support.

3.2. Basic IP Filters

3.2.1. IP header SubTLV (type=1(0x01))

The format of the IP header TLV value field is shown in figure 3-2. The IP header for the VPN case is specified in section 3.5.

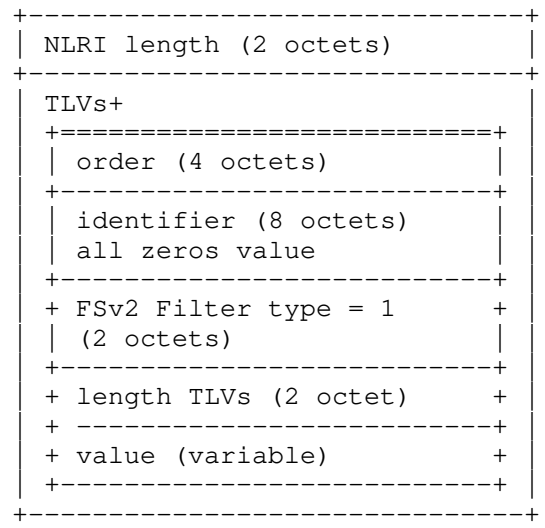
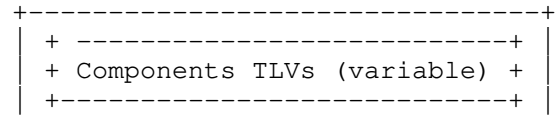


Figure 3-2 NLRI format for FSv2 IP Filter Type

Where: Each value field has the format:



Where the Component TLVs are:

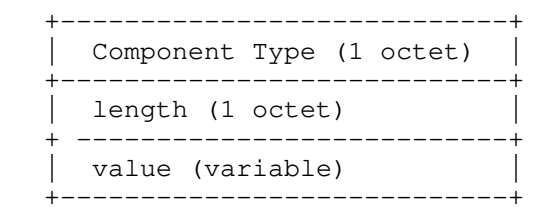


Figure 3-3 â\200\223 IP header Component TLVs

Where:

Component type: component values are defined in the â\200\234Flow Specification Component typesâ\200\235 registry for IPv4 and IPv6 by [RFC8955], [RFC8956], and [I-D.ietf-idr-flowspec-srv6]

length: length of SubTLV (varies depending on the component type)>

value: dependent on component type.

Many of the components use the operators [numeric_op] and [bitmask_op] defined in [RFC8955]

The list of valid SubTLV types appears in Table 3-1 for filter type of IP Filters (type=1). Other filters beyond these filters may be defined other filter types (e.g. IP Extended Filters).

Table 3-1 IP SubTLV Types for IP filters for IP Basic FSv2

Sub-TLV	Definition
1 -	IP Destination prefix
2 -	IP Source prefix
3 â\200\223	IPv4 Protocol / IPv6 Upper Layer Protocol
4 â\200\223	Port
5 â\200\223	Destination Port
6 â\200\223	Source Port
7 â\200\223	ICMPv4 type / ICMPv6 type
8 â\200\223	ICMPv4 code / ICPv6 code
9 â\200\223	TCP Flags
10 â\200\223	Packet length
11 â\200\223	DSCP
12 â\200\223	Fragment
13 â\200\223	Flow Label
14-63	Reserved for IP Filter Extensions
64-191	Reserved (Std Action)
192-249	FCFS
250-255	Reserved

Other FSv2 filter types (e.g. IP Extended Filters or L2 filters) may assign component types as specific to the filter type (e.g. 1-50) or utilize a global assignment component IDs. Table 3-2 below gives an example of a global definition of filters assignment.

Table 3-2 Possible Global Component Assignment

SubTLV	Filter type	Sub TLV definition
14-6	Extended IP	---
14	Extended IP	TTL in IP packet
15	Extended IP	SID in IPv6 header
16	Extended IP	NRP-ID in IPv6 header
17	Extended IP	CAT-ID in IPv6 Header
30	Exteded IP	flexible field in IPv4/IPv6 pkt
64-80	MPLS	---
64	MPLS	MPLS Label Match-1
65	MPLS	MPLS Label Match-2
81-120	L2	15 types specified) (81-95) in L2VPN document
86	L2	RSN MAC data unit
87	L2	Det. Latency
121-130	SFC	non-specified
131â\200\223150	Tunnel	11 specified in nvo3 document
151-180	Linked Data	interface, AS, Group, time
181 â\200\223 191	Reserved	
192 - 240	FCFS	
241 - 255	Reserved	

Ordering within the TLV in FSv2: The transmission of SubTLVs within a flow specification rule MUST be sent ascending order by SubTLV type. If the SubTLV types are the same, then the value fields are compared using mechanisms defined in [RFC8955] and [RFC8956] and MUST be in ascending order. NLRIs having TLVs which do not follow the above ordering rules MUST be considered as malformed by a BGP FSv2 propagator. This rule prevents any ambiguities that arise from the multiple copies of the same NLRI from multiple BGP FSv2 propagators. A BGP implementation SHOULD treat such malformed NLRIs as "Treat-as-withdraw" [RFC7606].

See [RFC8955], [RFC8956], and [I-D.ietf-idr-flowspec-srv6]. for specific details.

3.2.2. Components for FSv2 supporting IP Basic FSV2

3.2.2.1. IP Destination Prefix (type = 1)

IPv4 Name: IP Destination Prefix (reference: [RFC8955])

IPv6 Name: IPv6 Destination Prefix (reference: [RFC8956])

IPv4 length: Prefix length in bits

IPv4 value: IPv4 Prefix (variable length)

IPv6 length: length of value

IPv6 value: [offset (1 octet)] [pattern (variable)] [padding(variable)]

If IPv6 length = 0 and offset = 0, then component matches every address. Otherwise, length must be offset "less than" length "less than" 129 or component is malformed.

3.2.2.2. IP Source Prefix (type = 2)

IPv4 Name: IP Source Prefix (reference: [RFC8955])

IPv6 Name: IPv6 Source Prefix (reference: [RFC8956])

IPv4 length: Prefix length in bits

IPv4 value: Source IPv4 Prefix (variable length)

IPv6 length: length of value

IPv6 value: [offset (1 octet)] [pattern (variable)][padding(variable)]

If IPv6 length = 0 and offset = 0, then component matches every address. Otherwise, length must be offset < length < 129 or component is malformed.

3.2.2.3. IP Protocol (type = 3)

IPv4 Name: IP Protocol IP Source Prefix (reference: [RFC8955])

IPv6 Name: IPv6 Upper-Layer Protocol: (reference: [RFC8956])

IPv4 length: variable

IPv4 value: [numeric_op, value]+

IPv6 length: variable

IPv6 value: [numeric_op, value}+

where the value following each numeric_op is a single octet.

3.2.2.4. Port (type = 4)

IPv4/IPv6 Name: Port (reference: [RFC8955]), [RFC8956])

Filter defines: a set of port values to match either destination port or source port.

IPv4 length: variable

IPv4 value: [numeric_op, value]+

IPv6 length: variable

IPv6 value: [numeric_op, value]+

where the value following each numeric_op is a single octet.

Note-1: (from FSV1) In the presence of the port component (destination or source port), only a TCP (port 6) or UDP (port 17) packet can match the entire flow specification. If the packet is fragmented and this is not the first fragment, then the system may not be able to find the header. At this point, the FSv2 filter may fail to detect the correct flow. Similarly, if other IP options or the encapsulating security payload (ESP) is present, then the node may not be able to describe the transport header and the FSv2 filter may fail to detect the flow.

The restriction in note-1 comes from the inheritance of the FSv1 filter component for port. If better resolution is desired, a new FSv2 filter should be defined.

Note-2: FSv2 component only matches the first upper layer protocol value.

3.2.2.5. Destination Port (type = 5)

IPv4/IPv6 Name: Destination Port (reference: [RFC8955]), [RFC8956])

Filter defines: a list of match filters for destination port for TCP or UDP within a received packet

Length: variable

Component Value format: [numeric_op, value]+

3.2.2.6. Source Port (type = 6)

IPv4/IPv6 Name: Source Port (reference: [RFC8955]), [RFC8956])

Filter defines: a list of match filters for source port for TCP or UDP within a received packet

IPv4/IPv6 length: variable

IPv4/Ipv6 value: [numeric_op, value]+

3.2.2.7. ICMP Type (type = 7)

IPv4: ICMP Type (reference: [RFC8955])

Filter defines: Defines: a list of match criteria for ICMPv4 type

IPv6: ICMPv6 Type (reference: [RFC8956])

Filter defines: a list of match criteria for ICMPv6 type.

IPv4/IPv6 length: variable

IPv4/IPv6 value: [numeric_op, value]+

3.2.2.8. ICMP Code (type = 8)

IPv4: ICMP Type (reference: [RFC8955])

Filter defines: a list of match criteria for ICMPv4 code.

IPv6: ICMPv6 Type (reference: [RFC8956])

Filter defines: a list of match criteria for ICMPv6 code.

IPv4/IPv6 length: variable

IPv4/IPv6 value: [numeric_op, value]+

3.2.2.9. TCP Flags (type = 9)

IPv4/IPv6: TCP Flags Code (reference: [RFC8955])

Filter defines: a list of match criteria for TCP Control bits

IPv4/IPv6 length: variable

IPv4/IPv6 value: [bitmask_op, value]+

Note: a 2 octets bitmask match is always used for TCP-Flags

3.2.2.10. Packet length (type = 10 (0x0A))

IPv4/IPv6: Packet Length (reference: [RFC8955], [RFC8956])

Filter defines: a list of match criteria for length of packet (excluding L2 header but including IP header).

IPv4/IPv6 length: variable

IPv4/IPv6 value: [numeric_op, value]+

Note:[RFC8955] uses either 1 or 2 octet values.

3.2.2.11. DSCP (Differentiaed Services Code Point)(type = 11 (0x0B))

IPv4/IPv6: DSCP Code (reference: [RFC8955], [RFC8956])

Filter defines: a list of match criteria for DSCP code values to match the 6-bit DSCP field.

IPv4/IPv6 length: variable

IPv4/IPv6 value: [numeric_op, value]+

Note: This component uses the Numeric Operator (numeric_op) described in [RFC8955] in section 4.2.1.1. Type 11 component values MUST be encoded as single octet (numeric_op len=00).

The six least significant bits contain the DSCP value. All other bits SHOULD be treated as 0.

3.2.2.12. Fragment (type = 12 (0x0C))

IPv4/IPv6: Fragment (reference: [RFC8955], [RFC8956])

Filter defines: a list of match criteria for specific IP fragments.

Length: variable

Component Value format: [bitmask_op, value]+

Bitmask values are:

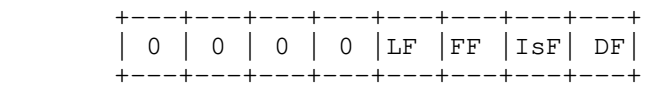


Figure 3-4

Where:

DF (donâ\200\231t fragment): match If IP header flags bit 1 (DF) is 1.

Is F(is a fragment other than first: match if IP header fragment offset is not 0.

FF (First Fragment): Match if [RFC0791] IP Header Fragment offset is zero and Flags Bit-2 (MF) is 1.

LF (last Fragment): Match if [RFC0791] IP header Fragment is not 0 And Flags bit-2 (MF) is 0

0: MUST be sent in NLRI encoding as 0, and MUST be ignored during reception.

3.2.2.13. Flow Label(type = 13 (0xOD))

IPv4/IPv6: Fragment (reference: [RFC8956])

Filter defines: a list of match criteria for 20-bit Flow Label in the IPv6 header field.

Length: variable

Component Value format: [numeric_op, value]+

3.2.3. FSv2 Actions for IP Basic

The full FSV2 [I-D.ietf-idr-flowspec-v2] specifies that FSv2 actions can be sent in Extended Communities or a Community attribute with the FSv2 community type. The IP Basic FSv2 only allows FSv2 actions to be sent in an Extended Community (FSv2-EC)

The Extended Community encodes the Flow Specification actions in the Extended IPv4 Community format [RFC4360] or in the extended IPv6 Community format [RFC5701]. The FSv2-EC actions cannot be ordered by the user and some FSv2-EC interact. . This section defines the FSv2-EC actions for FSv2 IP Basic by defining existing FSv2-EC action formats, the interaction between actions, and the default order of actions.

The FSv2 Action Chain Ordering Extended Community (AO-EC) signals if the defaults for the FSv2 Extended Community action ordering and interactions are being ignored, and an implementation specific ordering being used instead. This Action Chain Ordering Extended Community aids the transition between FSv1 actions which are ordered uniquely by each implementation, and the FSv2 actions which use a global default.

The implementer and the operator deploying need to be aware of default order of actions and the interactions between any set of FSv2 actions.

The Community attribute [I-D.ietf-idr-wide-bgp-communities] describes an attribute with flexible format for specifying community information. The flexible format defines a short common header followed by type-specific community. FSv2 [I-D.ietf-idr-flowspec-v2] defines a new type of Community denoted as a FSv2 Action for the Community Attribute (FSv2-CA) This FSv2 More IP Actions [I-D.hares-idr-fsv2-more-ip-actions]) defines the format of the FSv2-CA.

3.2.3.1. FSv2 Extended Community Actions inherited from FSv1

This section reviews FSv1 actions in Extended Communities (IPv4 and IPv6) and conflicts FSv1 actions. The FSv2 IP Basic uses these basic FSv1 with one addition Action Ordering Extended Community.

This section first describes the following Information elated to FSv2 Actions in Extended Communities:

* Generic Transitive Extended Communities for FSv2 Actions (FS-TG-EC) [RFC8955]

* Transitive Extended Communities for redirect. This includes:

  - (Generalized redirection ID with Sequencing and copy) [I-D.ietf-idr-flowspec-path-redirect]

  - Redirect plus Copy bit [I-D.ietf-idr-flowspec-redirect-ip]

  - Transitive IPv6-Address Extended Community formats for FSv2 actions [RFC8956]

3.2.3.1.1. Encoding FSv2 Actions in Generic Transitive Communities

The FSv2 actions encoded in Generic Transitive communities inherit the FSv1 actions in Generic Transitive communities.

The Extended Community encodes the Flow Specification actions in the Extended Community format as generic transitive extended communities per [RFC4360] per [RFC8955], [RFC9117], and [RFC9184].

The format of the these actions can be:

Generic Transitive Extended Community (0x80): where the Sub-Types are defined in the Generic Transitive Extended Community Sub-Types registry.

Generic Transitive Extended Community Part 2(0x81): where the Sub-Types are defined in the Generic Transitive Extended Community Part 2 Sub-Types registry.

Transitive Four-Octet AS-Specific Extended Communit(0x82): where the Sub-Types defined in the Generic Transitive Extended Community Part 3 Sub-Types registry.

Generic Transitive Extended Community Part 3 (0x83): where the Sub-Types defined in the Transitive Opaque Extended Community Sub-Types" registry.

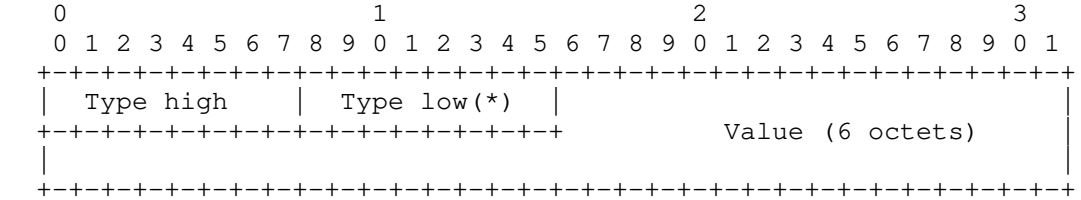


Figure 3-5

Table 3-3 Generic Transitive Extended Community
Part 1 - (0x80)

IPv4 Extended Communities (Type 0x80)

Value	Description	Name	Reference
0x01	FSv2 Action Chain Ordering	ACO	(This document)
0x06	FSv2 traffic-rate-byte	TRB	(RFC8955)
0x07	Flow spec traffic-action	TAIS	(RFC8955)
0x08	Flow spec rt-redirect AS-2 octet format	RDIP	(RFC8955)
0x09	Flow spec Remark DSCP	TMDS	(RFC8955)
0x0C	Flow Spec Traffic-rate-packets	TRP	(RFC8955)
0xOD	Flow Spec for SFC classifiers	SFCC	(RFC9015)

Table 3-4 Generic Transitive Extended Community
Part 2 (0x81)

IPv4 Extended Communities FSv2 action (Type 0x81)

Value	Description	Name	Reference
0x08	Flow spec rt-redirect	RDIP	(RFC8955)

Table 3-5 Generic Transitive Extended Community
Part 3 (Type 0x82)

Value	Description	Name	Reference
0x08	Flow spec rt-redirect AS-4 octet format	RDIP	(RFC8955)

Table 3-6: Traffic Action bits

Bit	Name	Name	Reference
47	Terminal Action	TAct	(RFC8955)
46	Sample	Samp	(RFC8955]
45	Copy	Copy	(this document)
44	Drop	drop	(this document)

Figure 3-13

3.2.3.1.2. Encoding Path Forwarding in IPv4 Transitive Extended Communities

FSv2 needs to refine the following Transitive Extended Communities that are not "Transitive Generic Communities" to a specific set of functions. These features provide overlapping functions. While some of these features are implemented, these actions should be be reviewed.

There are three types of functions:

* Active filters on interfaces in group for inbound or outbound data traffic

* Redirect to an IP address. Optionally perform a traffic action (copy)

* Redirect to an Indirection ID of a specific type. Optionally perform a traffic action (copy).

Table 3-7 Transitive Extended Community types (T-EC-types)

sub-type	FSv1 Description	Name
0x07	FS Interface set	Ifset
0x08	FS Redirect/Mirror	RIPv4
0x09	FS Redirect to Indirection ID	RGID

References:

ifset - [I-D.ietf-idr-flowspec-interfaceset]

RIPv4 - [I-D.ietf-idr-flowspec-redirect-ip]

RGID - [I-D.ietf-idr-flowspec-path-redirect]

3.2.3.1.3. Encoding FSv2 Actions in IPv6 Extended Community

The IPv6 Extended Community encodes the Flow Specification actions in the Extended Community format [RFC5701] per [RFC8956], [RFC9117], and [RFC9184] in the transitive opaque format.

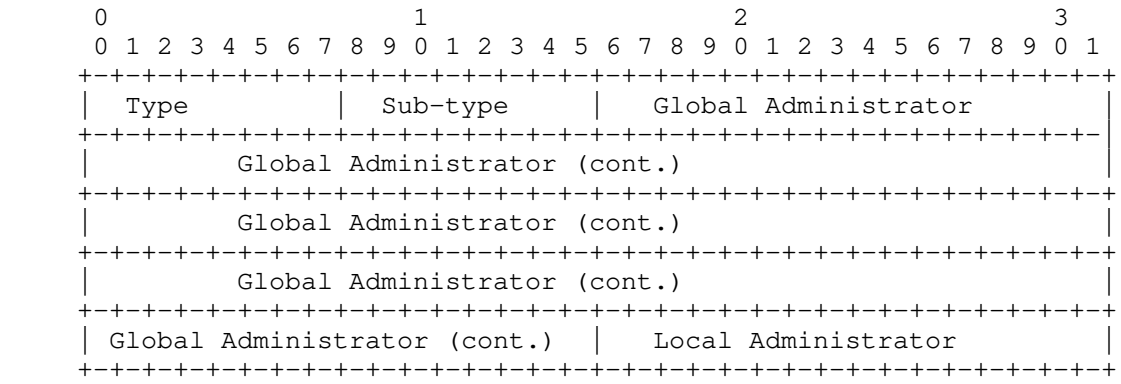


Figure 3-6

The 20 octets of value are given in the following format:

Global Administrator: IPv6 address assigned by Internet Registry
Local Administrator: 2 bytes of Local Administrator

Table 3-8 transitive IPv6-Address-Specific Actions

Value	Description	Name
0x01	Flow Spec Action Chain	ACO
0x0C	Flow Spec redirect-v6-flag	RD6F
0x0D	Flow Spec rt-redirect IPv6 format IPv6 format	RDv6

References:

ACO - This document

RD6F - [I-D.ietf-idr-flowspec-redirect-ip]

RDv6 - [RFC8956]

3.2.3.1.4. Conflicts between FSv2 actions inherited from FSv1 Actions

Table 3-9: Conflicts between FSv2 Transitive Generic IPv4 actions

IPv4 Extended Communities (Type 0x80)

Value	Name	Conflicts with
0x01	ACO	none
0x06	TRB	TRP
0x07	TAIS	duplication also done in RDIP, RIPv4, RGID
0x08	RDIP	redirection done in RIPv4, RGID copy done in TAIS
0x09	TMDS	none
0x0C	TRP	TRB
0xOD	SFCC	none

Table 3-10 Transitive IPv6-Address-Specific Actions

Value	Name	Conflicts with
0x01	ACO	none
0x0C	RD6F	RDv6
0x0D	RDv6	RD6F

3.2.3.2. Default Ordering for FSv2 Extended Community Actions

One of the issue that started the FSv2 work was the fact that actions interacted. These interactions might occur when both actions performed their duties which caused conflicting results. One example of a potentially unexpected interaction is when the FSv2 for rate limiting by packet (TRP) combines with the FSv2-EC action for rate limiting by byte (TRB).

The default order is the numerical order of the action type as shown in table x-x for IPv4 and table x-x for IPv6.

Table 3-11 Default Order of FSv2-EC IPv4 Actions

IPv4 Extended Communities (Type 0x80)		
Value	Description	Name
0x01	FSv2 Action Chain Ordering	ACO
0x06	FSv2 traffic-rate-byte	TRB
0x07	Flow spec traffic-action	TAIS
0x07	FS Interface set	
0x08	Flow spec rt-redirect	RDIP
0x08	FS Redirect/Mirror	RDIPv4
0x08	FS Redirect/Mirror	RDIPv4
0x09	FS Redirect to Path ID	RD
0x09	Flow spec Remark DSCP	TMDS
0x0C	Flow Spec Traffic-rate-packets	TRP
0xOD	Flow Spec for SFC classifiers	SFCC

Note: If FS Interface is widely deployed it would be good to move it to another type.

Table 3-12 default order for FSv2-EC IPv6 actions

Value	Name	Conflicts with
0x01	ACO	none
0x0C	RD6F	RDv6
0x0D	RDv6	RD6F

3.2.3.3. Action Chain Ordering FSv2 Extended Community (ACO FSv2-EC)

One of the issues with FSv1 is the lack of a clear definition on what happens if multiple actions interact. One way a FSv2 action can interact is if two actions try to do different things with the packet. A second way an FSv2 action can interact is if the first action fails. For example, if the first action was copy (via a mirror action) and the second action is the packet. If the first action fails, should the second action still occur? The correct answer depends on the FSv2 application. If the order of the two actions is drop the packet and then mirror, the mirror function would not copy any packets.

The default ordering of the FSv2-EC actions makes a default action chain for the FSv2 actions supported by the IP Basic. The addition of the FSv2-EC action For Action Chain ordering provides a deterministic way of determining what happens if an action fails.

The orginal specification FSv2 [I-D.ietf-idr-flowspec-v2] first defined the concept of an action chain to address the issues of interaction between user-order actions. A FSv2-CA action will be defined for FSv2 Action Chain Ordering (ACO). An implementation which implements both the the FSv2-CA ACO action the FSv2-EC ACO action, MUST give precedence to the ACO action AND provide a logging entry regarding any conflict between the two actions.

The FSv2-EC also provides a flag for "Implementation specific ordering." This flag is useful to aid transition between the FSv1 implementations and FSv2 implementations of IP Basic. In FSv1 implementations configurations or implementation defaults set the order for actions. In FSv2 there is a default order for actions and interactions. New FSv2-Action need to define

The AC-Failure types are:

* 0x00 â\200\223 default â\200\223 stop on failure

* 0x01 â\200\223 continue on failure (best effort on actions)

* 0x02 â\200\223 conditional stop on failure (depends on AC-Failure-value/ policy)

* 0x03 â\200\223 rollback do all or nothing (depends on AC-Failure-value/ policy)

Editors note: The following options for encoding ACO exist.

Option 1: redefine bits in Traffic Action subtype

Option 2: create a new Extended Community

3.2.3.3.1. FSV2 Basic DDOS Actions

3.2.3.3.1.1. New Actions for FSv2 DDOS

There are two options for encoding the Action chain.

3.2.3.3.1.1.1. Option 1: Action Chain operation IPv4 Extended (ACO)(1, 0x01)

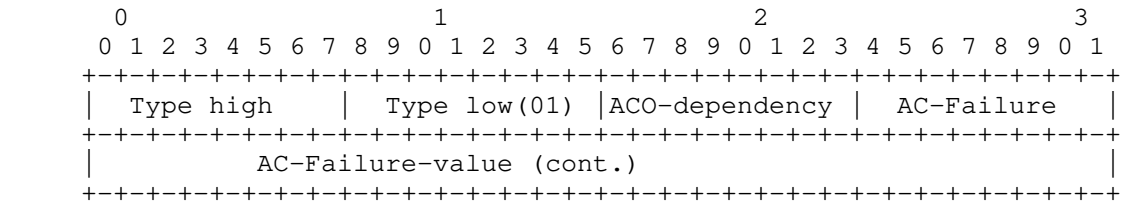


Figure 3-7

where:

ACO Dependency - The order dependency within the Action chain.

- 0 = default order and interaction. For FSv2-EC this means a pre-defined order and inter-dependency.

- 1 = Implementation specific order and interaction.

AC-failure-type â\200\223 1 octet byte that determines the action on failure

- Actions may succeed or fail and an Action chain must deal with it. The default value stored for an action chain that does not have this action chain is â\200\234stop on failureâ\200\235.

- where:

  o AC-Failure types are:

    + 0x00 â\200\223 default â\200\223 stop on failure

    + 0x01 â\200\223 continue on failure (best effort on actions)

    + 0x02 â\200\223 conditional stop on failure â\200\223 depending on AC-Failure-value

    + 0x03 â\200\223 rollback â\200\223 do all or nothing - depending in AC-Failure-value

AC-Failure values: TBD

3.2.3.3.1.1.2. Option 2: Action Chain operation encoded in IPv4 Traffic Action (0x07)

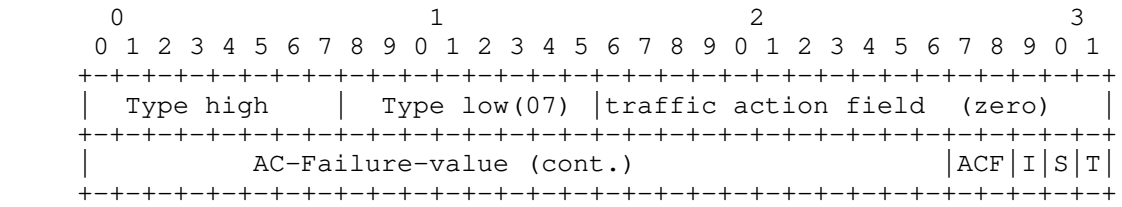


Figure 3-8

Where
ACO - is the Action Chain failure types (0x00 to 0x03)
00 - stop on failure
01 - continue on failure
02 - conditional stop on failure (by policy)
03 - rollback on failure (with policy)
I - Implementation ordering and interaction
0 - Default FSv2 ordering and interation
1 - Implementation defined user
S - Sample flag
T - Terminal action

3.2.3.3.1.2. Interactions between FSv2 DDOS actions

Table 3-13 - All FSv2 IPv4 Action types for IP DDOS

Action	Name	Description	May Interacts
01	ACO	Action Chain Operation	none
06	TRB	Traffic Rate limited by Bytes	TRP
07	TA	Traffic Action (terminal/sample/ACO)	none
08	RDIP	Redirect IPv4	none
09	TM	Mark DSCP value	none
12	TRP	Traffic Rate limited by Packets	TRB

Table 3-14 â\200\223 All FSv2 IPv6 Action types for IP DDOS

Action	Name	Description	May Interacts
01	ACO	Action Chain Operation	none
06	TRB	Traffic Rate limited by Bytes	TRP
07	TA	Traffic Action (terminal/sample/ACO)	none
08	RDIP	Redirect IPv4	none
09	TM	Mark DSCP value	none
12	TRP	Traffic Rate limited by Packets	TRB

3.2.3.3.2. Summary of all FSv2 Actions (informative only)

This table is informative only. Prior to Publication it will be moved to an appendix.

Table 3-15 - All IP Actions in Extended Communities

Action	Name: Description
00	reserved
01	ACO: action chain operation
02	reserved
03	TAIS: traffic actions per interface group
04	LkBW: Link bandwidth (draft-ietf-idr-linkbandwidth-07) [non-transitive] [juniper link bandwidth] [transitive]
06	TRB: traffic rate limited by bytes
07	TA: traffic action (terminal/sample)
08	RDIP: Redirect IPv4
09	TM: mark DSCP value
10	TBA (to be assigned)
11	TBA (to be assigned)
12	TRP: traffic rate limited by packets
13	TISFC: SFC Classifier
14	RDIID: redirect to Indirection-id (move from 0x00)
31	TISFC: SFC classifier II (this document)
32	MPLSLA: MPLS label action
33	VLAN: VLAN-Action (0x16)
34	TPID: TPID-Action (0x17)
24-254	TBA (to be assigned)
255	reserved

Table 3-16 IPv6 Extended Communities (Type 1)

Value	Description	Name	Reference
0x01	Flow Spec Action Chain	ACO	This document
0x0C	Flow Spec redirect-v6-flag	RD6F	ID-redirect-IP
0x0D	Flow Spec rt-redirect IPv6 format	RD6	RFC8956

ID-redirect-IP is [I-D.ietf-idr-flowspec-redirect-ip].

4. Validation and Ordering of NLRI

4.1. Validation of FSv2 NLRI

The validation of FSv2 NLRI adheres to the combination of rules for general BGP FSv1 NLRI found in [RFC8955], [RFC8956], [RFC9117], and the specific additions made for SFC NLRI [RFC9015], and L2VPN NLRI [I-D.ietf-idr-flowspec-l2vpn].

To provide clarity, the full validation process for flow specification routes (FSv1 or FSv2) is described in this section rather than simply referring to the relevant portions of these RFCs. Validation only occurs after BGP UPDATE message reception and the FSv2 NLRI and the path attributes relating to FSv2 (Extended community and Wide Community) have been determined to be well-formed. Any MALFORMED FSv2 NRLI is handled as a â\200\234TREAT as WITHDRAWâ\200\235 [RFC7606].

4.1.1. Validation of FS NLRI (FSv1 or FSv2)

Flow specifications received from a BGP peer that are accepted in the respective Adj-RIB-In are used as input to the route selection process. Although the forwarding attributes of the two routes for tbe same prefix may be the same, BGP is still required to perform its path selection algorithm in order to select the correct set of attributes to advertise.

The first step of the BGP Route selection procedure (section 9.1.2 of [RFC4271] is to exclude from the selection procedure routes that are considered unfeasible. In the context of IP routing information, this is used to validate that the NEXT_HOP Attribute of a given route is resolvable.

The concept can be extended in the case of the Flow Specification NLRI to allow other validation procedures.

The FSv2 validation process validates the FSv2 NLRI with following unicast routes received over the same AFI (1 or 2) but different SAFIs:

* Flow specification routes (FSv1 or FSv2) received over SAFI=133 will be validated against SAFI=1,

* Flow Specification routes (FSv1 or FSv2) received over SAFI=134 will be validated against SAFI=128, and

* Flow Specification routes (FSv1 or FSv2) [AFI =1, 2] received over SAFI=77 will be validated using only the Outer Flow Spec against SAFI = 133.

The FSv2 validates L2 FSv2 NLRI with the following L2 routes received over the same AFI (25), but a different SAFI:

* Flow specification routes (FSv1 or FSv2)received over SAFI=135 are validated against SAFI=128.

In the absence of explicit configuration, a Flow specification NLRI (FSv1 or FSv2) MUST be validated such that it is considered feasible if and only if all of the conditions are true:

a) A destination prefix component is embedded in the Flow Specification,

b) One of the following conditions holds true:

- 1. The originator of the Flow Specification matches the originator of the best-math unicast route for the destination prefix embedded in the flow specification (this is the unicast route with the longest possible prefix length covering the destination prefix embedded in the flow specification).

- 2. The AS_PATH attribute of the flow specification is empty or contains only an AS_CONFED_SEQUENCE segment [RFC5065].

  o 2a.This condition should be enabled by default.

  o 2b.This condition may be disabled by explicit configuration on a BGP Speaker,

  o 2c.As an extension to this rule, a given non-empty AS_PATH (besides AS_CONFED_SEQUENCE segments) MAY be permitted by policy].

c) There are no â\200\234more-specificâ\200\235 unicast routes when compared with the flow destination prefix that have been received from a different neighbor AS than the best-match unicast route, which has been determined in rule b.

However, part of rule a may be relaxed by explicit configuration, permitting Flow Specifications that include no destination prefix component. If such is the case, rules b and c are moot and MUST be disregarded.

By â\200\234originatorâ\200\235 of a BGP route, we mean either the address of the originator in the ORIGINATOR_ID Attribute [RFC4456] or the source address of the BGP peer, if this path attribute is not present.

A BGP implementation MUST enforce that the AS in the left-most position of the AS_PATH attribute of a Flow Specification Route (FSv1 or FSv2) received via the Exterior Border Gateway Protocol (eBGP) matches the AS in the left-most position of the AS_PATH attribute of the best-match unicast route for the destination prefix embedded in the Flow Specification (FSv1 or FSv2) NLRI.

The best-match unicast route may change over time independently of the Flow Specification NLRI (FSv1 or FSv2). Therefore, a revalidation of the Flow Specification MUST be performed whenever unicast routes change. Revalidation is defined as retesting rules a to c as described above.

4.1.2. Validation of Flow Specification Actions

Flow Specifications may be mapped to actions using Extended Communities or a Wide Communities. The FSv2 actions in Extended Communities and Wide communities can be associated with large number of NLRIs.

The ordering of precedence for these actions in the case when the user-defined order is the same follows the precedence of the FSv2 NLRI action TLV values (lowest to highest). User-defined order is the same when the order value for action is the same. All Extended Community actions MUST be translated to the user-defined order data format for internal comparison. By default, all Extended Community actions SHOULD be translated to a single value.

Actions may conflict, duplicate, or complement other actions. An example of conflict is the packet rate limiting by byte and by packet. An example of a duplicate is the request to copy or sample a packet under one of the redirect functions (RDIPv4, RDIPv6, RDIID,) Each FSv2 actions in this document defines the potential conflicts or duplications. Specifications for new FSv2 actions outside of this specification MUST specify interactions or conflicts with any FSv2 actions (that appear in this specification or subsequent specifications).

Well-formed syntactically correct actions should be linked to a filtering rule in the order the actions should be taken. If one action in the ordered list fails, the default procedure is for the action process for this rule to stop and flag the error via system management. By explicit configuration, the action processing may continue after errors.

Implementations MAY wish to log the actions taken by FS actions (FSv1 or FSv2).

4.1.3. Error handling and Validation

The following two error handling rules must be followed by all BGP speakers which support FSv2:

* FSv2 NLRI having TLVs which do not have the correct lengths or syntax must be considered MALFORMED.

* FSv2 NLRIs having TLVs which do not follow the above ordering rules described in section 4.1 MUST be considered as malformed by a BGP FSv2 propagator.

The above two rules prevent any ambiguity that arises from the multiple copies of the same NLRI from multiple BGP FSv2 propagators.

A BGP implementation SHOULD treat such malformed NLRIs as â\200\230Treat-as-withdrawâ\200\231 [RFC7606]

An implementation for a BGP speaker supporting both FSv1 and FSv2 MUST support the error handling for both FSv1 and FSv2.

4.2. Ordering for Flow Specification v2 (FSv2)

Flow Specification v2 allows the user to order flow specification rules and the actions associated with a rule. Each FSv2 rule has one or more match conditions and one or more actions associated with that match condition.

This section describes how to order FSv2 filters received from a peer prior to transmission to another peer. The same ordering should be used for the ordering of forwarding filtering installed based on only FSv2 filters.

Section 7.0 describes how a BGP peer that supports FSv1 and FSv2 should order the flow specification filters during the installation of these flow specification filters into FIBs or firewall engines in routers.

The BGP distribution of FSv1 NLRI and FSv2 NLRI and their associated path attributes for actions (Wide Communities and Extended Communities) is â\200\234ships-in-the-nightâ\200\235 forwarding of different AFI/SAFI information. This recommended ordering provides for deterministic ordering of filters sent by the BGP distribution.

4.2.1. Ordering of FSv2 NLRI Filters

The basic principles regarding ordering of rules are simple:

1) Rule-0 (zero) is defined to be 0/0 with the â\200\234permit-allâ\200\235 action

- BGP peers which do not support flow specification permit traffic for routes received. Rule-0 is defined to be â\200\234permit-allâ\200\235 for 0/0 which is the normal case for filtering for routes received by BGP.

- By configuration option, the â\200\234permit-allâ\200\235 may be set to â\200\234deny-allâ\200\235 if traffic rules on routers used as BGP must have a â\200\234routeâ\200\235 AND a firewall filter to allow traffic flow.

2) FSv2 rules are ordered based on the user-defined order numbers specified in the FSv2 NLRI (rules 1-n).

3) If multiple FSv2 NLRI have the same user-defined order, then the filters are ordered by type of FSv2 NRLI filters (see Table 1, section 4) with lowest numerical number have the best precedence.

- For the same user-defined order and the same value for the FSv2 filters type, then the filters are ordered by FSv2 the component type for that FSv2 filter type (see Tables 3-6) with the lowest number having the best precedence.

- For the same user-defined order, the same value of FSv2 Filter Type, and the same value for the component type, then the filters are ordered by value within the component type. Each component type defines value ordering.

- For component types inherited from the FSv1 component types, there are the following two types of comparisons:

  - FSv1 component value comparison for the IP prefix values, compares the length of the two prefixes. If the length is different, the longer prefix has precedence. If the length is the same, the lower IP number has precedence.

  - For all other FSv1 component types, unless specified, the component data is compared using the memcmp() function defined by [ISO_IEC_9899]. For strings with the same length, the lowest string memcmp() value has precedence. For strings of different lengths, the common prefix is compared. If the common string prefix is not equal, then the string with the lowest string prefix has higher precedence. If the common prefix is equal, the longest string is considered to have higher precedence

Notes:

* Since the user can define rules that re-order these value comparisons, this order is arbitrary and set to provide a deterministic default.

4.2.2. Ordering of the Actions

The FSv2 specification allows for actions to be associated by:

a) a Wide Community path attribute, or

b) an Extended Community path attribute.

Actions may be ordered by user-defined action order number from 1-n (where n is 2**16-2 and the value 2**16-1 is reserved.

Byy default, extended community actions are associated with default order number 32768 (0x8000) or a specific configured value for the FSv2 domain.

Action user-order number zero is defined to have an Action type of â\200\234Set Action Chain operationâ\200\235 (ACO) (value 0x01) that defines the default action chain process. For details on â\200\234set action chain operationâ\200\235 see section 3.2.1 or section 5.2.1 below.

If the user-defined action number for two actions are the same, then the actions are ordered by FSv2 action types (see Table 3 for a list of action types). If the user-defined action number and the FSv2 action types are the same, then the order must be defined by the FSv2 action.

4.2.2.1. Action Chain Operation (ACO)

The â\200\234Action Chain Operationâ\200\235 (ACO) changes the way the actions after the current action in an action chain are handled after a failure. If no action chain operations are set, then the default action of â\200\234stop upon failureâ\200\235 (value 0x00) will be used for the chain.

4.2.2.1.1. Example 1 - Default ACO

Use Case 1: Rate limit to 600 packets per second

Description: The provider will support 600 packets per second All Packets sampled for reporting purposes and packet streams over 600 packets per second will be dropped.

Suppose BGP Peer A has a

* a Wide Community action with user-defined order 10 with Traffic Sampling

* a Wide Community action with user-defined order 11 from AS 2020 that limits packet-based rate limit of 600 packets per second.

* an Extended Community from AS 2020 that does limits packet-based rate limit of 50 packets per second.

The FSv2 data base would store the following action chain:

* at user-defined action order 10

  - A user action of type 7 (traffic action) with values of Sampling and logging.

* at user-defined action order 11

  - a user action type of 12 (packet-based rate limit) with values of AS 2020 and float value for 600 packets per second (pps)

* at user-defined action order 32768 (0x8000) with type 12 and values of A user action of type 12 with values of AS 2020 and float value of 50 packets/second.

Normal action:

The match on the traffic would cause a sample of the traffic (probably with packet rate saved in logging) followed by a rate limit to 600 pps. The Extended community action would further limit the rate to 50 packets per second.

When does the action chain stop?

The default process for the action chain is to stop on failure. If there is no failure, then all three actions would occur. This is probably not what the user wants.

If there is failure at action 10 (sample and log), then there would be no rate limiting per packet (actions 11 and action 32768).

If there is failure at action 11 (rate limit to packet 600), then there would be no rate limiting per packet (action 32768).

The different options for Action chain ordering (ACO) have been worked on with NETCONF/RESTCONF configuration and actions.

4.2.2.1.2. Example 2: Redirect traffic over limit to processing via SFC

Use case 2: Redirect traffic over limit to processing via SFC.

Description: The normal function is for traffic over the limit to be forwarded for offline processing and reporting to a customer.

Suppose we have the following 4 actions defined for a match:

* Sent Redirect to indirection ID (0x01) with user-defined match 2 attached in wide community,

* Traffic rate limit by bytes (0x07) with user-defined match 1 attached in wide community,

* Traffic sample (0x07) sent in extended community, and

* SF classifier Info (0x0E) sent in extended community.

These 4 filters rate limit a potential DDoS attack by: a) redirect the packet to indirection ID (for slower speed processing), sample to local hardware, and forward the attack traffic via a SFC to a data collection box.

The FSv2 action list for the match would look like this

Action 0: Operation of action chain (0x01) (stop upon failure)

Action 1: Traffic Rate limit by byte (0x07)

Action 2: Redirect to Redirection ID (0x0F)

Action 32768 (0x8000) Traffic Action (0x07) Sample

Action 32768 (0x8000) SFC Classifier: (0xE)

If the redirect to a redirection ID fails, then Traffic Sample and sending the data to an SFC classifier for forwarding via SFC will not happen. The traffic is limited, but not redirect away from the network and a sample sent to DDOS processing via a SFC classifier.

Suppose the following 5 actions were defined for a FSV2 filter:

* Set Action Chain Operation (ACO) (0x01) to continue on failure (ox01) at user-order 2 attached in wide community,

* redirect to indirection ID (0x0F) at user-order 2 attached in wide community,

* traffic rate limit by bytes (0x07)with user-order 1 attached in wide community,

* Traffic sample (0x07) attached via extended community, and

* SFC classifier Info (0x0E) attached in extended community.

The FSv2 action list for the match would look like this:

Action 00: Operation of action chain (0x01) (stop upon failure)

Action 01:Traffic Rate limit by byte (0x07)

Action 02:Set Action Chain Operation (ACO) (0x01) (continue on failure)

Action 02: Redirect to Redirection ID (0F)

Action 32768 (0x8000): Traffic Action (0x07) Sample

Action 32768 (0x8000): SFC classifier (0x0E) forward via SFC [to DDoS classifier]

If the redirect to a redirection ID fails, the action chain will continue on to sample the data and enact SFC classifier actions.

4.3. Ordering of FS filters for BGP Peers support FSv1 and FSv2

FSv2 allows the user to order flow specification rules and the actions associated with a rule. Each FSv2 rule has one or more match conditions and one or more actions associated with each rule.

FSv1 and FSv2 filters are sent as different AFI/SAFI pairs so FSv1 and FSv2 operate as ships-in-the-night. Some BGP peers in an AS may support both FSv1 and FSv2. Other BGP peers may support FSv1 or FSv2. Some BGP will not support FSv1 or FSV2. A coherent flow specification technology must have consistent best practices for ordering the FSv1 and FSv2 filter rules.

One simple rule captures the best practice: Order the FSv1 filters after the FSv2 filter by placing the FSv1 filters after the FSv2 filters.

To operationally make this work, all flow specification filters should be included the same data base with the FSv1 filters being assigned a user- defined order beyond the normal size of FSv2 user-ordered values. A few examples, may help to illustrate this best practice.

Example 1: User ordered numbering - Suppose you might have 1,000 rules for the FSv2 filters. Assign all the FSv1 user defined rules to 1,001 (or better yet 2,000). The FSv1 rules will be ordered by the components and component values.

Example 2: Storage of actions - All FSv1 actions are defined ordered actions in FSv2. Translate your FSv1 actions into FSv2 ordered actions for storing in a common FSv1-FSv2 flow specification data base.

Example 3: Mixed Flow Specification Support -

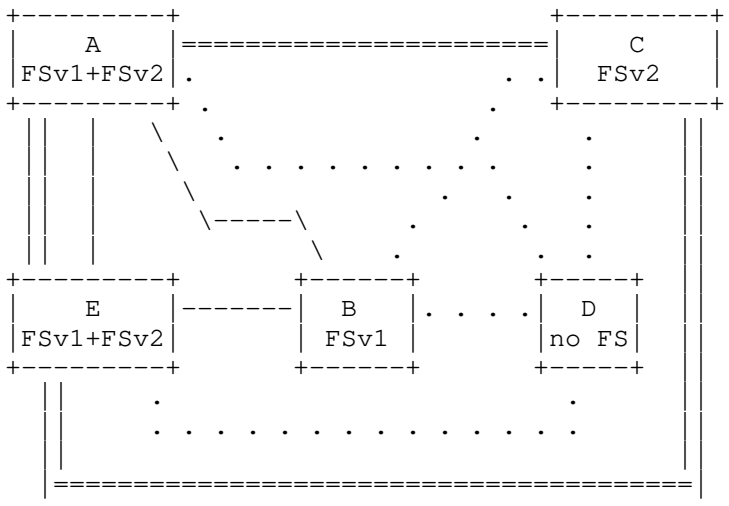
Suppose an FSv2 peer (BGP Peer A) has the capability to send either FSv1 or FSv2. BGP Peer A peers with BGP Peers B, C, D and E.

BGP Peer B can only send FSv1 routes (NLRI + Extended Community). BGP Peer C can send FSv2 routes (NLRI + path attributes (wide community or extended community or none)). BGP Peer D cannot send any FS routes. BGP E can send FSv2 and FSv1 routes

BGP Peer A sends FSv1 routes in its databases to BGP B. Since the FSv2 NLRI cannot be sent to the FSv1 peer, only the FSv1 NLRI is sent. BGP Peer A sends to BGP C the FSv2 routes in its database (configured or received).

BGP peer A would not send the FSv1 NLRI or FSv2 NLRI to BGP Peer D. The BGP Peer D does not support for these NLRI.

BGP Peer A sends the NLRI for both FSv1 and FSv2 to BGP Peer E.



Double line = FSv2
Single line = FSv1
Dotted line = BGP peering with no FlowSpec

Figure 4-1: FSv1 and FVs2 Peering

5. Scalability and Aspirations for FSv2

Operational issues drive the deployment of BGP flow specification as a quick and scalable way to distribute filters. The early operations accepted the fact validation of the distribution of filter needed to be done outside of the BGP distribution mechanism. Other mechanisms (NETCONF/RESTCONF or PCEP) have reply-request protocols.

These features within BGP have not changed. BGP still does not have an action-reply feature.

NETCONF/RESTCONF latest enhancements provide action/response features which scale. The combination of a quick distribution of filters via BGP and a long-term action in NETCONF/RESTCONF that ask for reporting of the installation of FSv2 filters may provide the best scalability.

The combination of NETCONF/RESTCONF network management protocols and BGP focuses each protocol on the strengths of scalability.

FSv2 will be deployed in webs of BGP peers which have some BGP peers passing FSv1, some BGP peers passing FSv2, some BGP peers passing FSv1 and FSv2, and some BGP peers not passing any routes.

The TLV encoding and deterministic behaviors of FSv2 will not deprecate the need for careful design of the distribution of flow specification filters in this mixed environment. The needs of networks for flow specification are different depending on the network topology and the deployment technology for BGP peers sending flow specification.

Suppose we have a centralized RR connected to DDoS processing sending out flow specification to a second tier of RR who distribute the information to targeted nodes. This type of distribution has one set of needs for FSv2 and the transition from FSv1 to FSv2

Suppose we have Data Center with a 3-tier backbone trying to distribute DDoS or other filters from the spine to combinational nodes, to the leaf BGP nodes. The BGP peers may use RR or normal BGP distribution. This deployment has another set of needs for FSv2 and the transition from FSv1 to FSV2.

Suppose we have a corporate network with a few AS sending DDoS filters using basic BGP from a variety of sites. Perhaps the corporate network will be satisfied with FSv1 for a long time.

These examples are given to indicate that BGP FSv2, like so many BGP protocols, needs to be carefully tuned to aid the mitigation services within the network. This protocol suite starts the migration toward better tools using FSv2, but it does not end it. With FSv2 TLVs and deterministic actions, new operational mechanisms can start to be understood and utilized.

This FSv2 specification is merely the start of a revolution of work â\200\223 not the end.

6. Optional Security Additions

This section discusses the optional BGP Security additions for BGP-FS v2 relating to BGPSEC [RFC8205] and ROA [RFC9582].

6.1. BGP FSv2 and BGPSEC

Flow specification v1 ([RFC8955] and [RFC8956]) do not comment on how BGP Flow specifications to be passed BGPSEC [RFC8205] BGP Flow Specification v2 can be passed in BGPSEC, but it is not required.

FSv1 and FSv2 may be sent via BGPSEC.

6.2. BGP FSv2 with ROA

BGP FSv2 can utilize ROAs in the validation. If BGP FSv2 is used with BGPSEC and ROA, the first thing is to validate the route within BGPSEC and second to utilize BGP ROA to validate the route origin.

The BGP-FS peers using both ROA and BGP-FS validation determine that a BGP Flow specification is valid if and only if one of the following cases:

* If the BGP Flow Specification NLRI has a IPv4 or IPv6 address in destination address match filter and the following is true:
  - A BGP ROA has been received to validate the originator, and
  - The route is the best-match unicast route for the destination prefix embedded in the match filter; or
* If a BGP ROA has not been received that matches the IPv4 or IPv6 destination address in the destination filter, the match filter must abide by the [RFC8955] and [RFC8956] validation rules as follows:
  - The originator match of the flow specification matches the originator of the best-match unicast route for the destination prefix filter embedded in the flow specification", and
  - No more specific unicast routes exist when compared with the flow destination prefix that have been received from a different neighboring AS than the best-match unicast route, which has been determined in step A.

The best match is defined to be the longest-match NLRI with the highest preference.

7. IANA Considerations

This section complies with [RFC7153].

7.1. Flow Specification V2 SAFIs

IANA is requested to assign two SAFI Values in the registry at https://www.iana.org/assignments/safi-namespace from the Standard Action Range as follows:

Table 7-1 SAFIs

Value	Description	Reference
TBD1	BGP FSv2	[this document]
TBD2	BGP FSv2 VPN	[this document]

7.2. BGP Capability Code

IANA is requested to assign a Capability Code from the registry at https://www.iana.org/assignments/capability-codes/ from the IETF Review range as follows:

Table 7-2 - Capability Code

Value	Description	Reference	Controller
TBD3	Flow Specification V2	[this document]	IETF

7.3. FSv2 IP Filters Component Types

IANA is requested to create a "FSv2 IP Filters Component Types" registry and indicate [this draft] as a reference. The following assignments in the FSv2 IP Filters Component Types Registry shold be made.

Table 7-3 - Flow Specification

Registry Name: BGP FSv2 TLV types
Reference: [this document]
Registration Procedures: 0x01-0x3FFF Standards Action.

Value	Description	Reference
1	Destination filter	[RFC8955][RFC8956][this document]
2	Source Prefix	[RFC8955][RFC8956][this document]
3	IP Protocol	[RFC8955][RFC8956][this document]
4	Port	[RFC8955][RFC8956][this document]
5	Destination Port	[RFC8955][RFC8956][this document]
6	Source Port	[RFC8955][RFC8956][this document]
7	ICMP Type [v4 or v6]	[RFC8955][RFC8956][this document]
8	ICMP Code [v4 or v6]	[RFC8955][RFC8956][this document]
9	TCP Flags [v4]	[RFC8955][RFC8956][this document]
10	Packet Length	[RFC8955][RFC8956][this document]
11	DSCP marking	[RFC8955][RFC8956][this document]
12	Fragment	[RFC8955][RFC8956][this document]
13	Flow Label	[RFC8956][this document]

7.4. FSV2 NLRI TLV Types

IANA is requested to create the a new registries on a new "Flow Specification v2 TLV Typesâ\200\235 web page.

Table 7-4 FSv2 TLV types

Registry Name: BGP FSv2 TLV types
Reference: [this document]
Registration Procedures: 0x01-0x3FFF Standards Action.

Type	Description	Reference
0x00	Reserved	[this document]
0x01	IP traffic rules	[this document]
0x02	Extended IP Rules	[this document]
0x03	MPLS Traffic Rules	[this document]
0x04	L2 Traffic rules	[this document]
0x05	SFC Traffic rules	[this document]
0x06	Tunneled traffic rules	[this document]
0x08-0x3FFF	Unassigned	[this document]
0x4000-0x7FFF	Vendor specific	[this document]
0x8000-0xFFFF	Reserved	[this document]

7.5. Community Container Type Assignments

IANA is requested to assign values from the BGP Community Container Types registry:

Tab;e 5 -

Name	type Value
FSv2 Actions	TBD4

8. Security Considerations

The use of ROA improves on [RFC8955] by checking to see of the route origination. This check can improve the validation sequence for a multiple-AS environment.

>The use of BGPSEC [RFC8205] to secure the packet can increase security of BGP flow specification information sent in the packet.

The use of the reduced validation within an AS [RFC9117] can provide adequate validation for distribution of flow specification within a single autonomous system for prevention of DDoS.

Distribution of flow filters may provide insight into traffic being sent within an AS, but this information should be composite information that does not reveal the traffic patterns of individuals.

9. References

9.1. Normative References

[I-D.hares-idr-fsv2-more-ip-actions]
Hares, S., "BGP Flow Specification Version 2 - More IP Actions", Work in Progress, Internet-Draft, draft-hares-idr-fsv2-more-ip-actions-01, 3 June 2024, <https://datatracker.ietf.org/doc/html/draft-hares-idr-fsv2-more-ip-actions-01>.

[I-D.hares-idr-fsv2-more-ip-filters]
Hares, S., "BGP Flow Specification Version 2 - More IP Filters", Work in Progress, Internet-Draft, draft-hares-idr-fsv2-more-ip-filters-02, 22 July 2024, <https://datatracker.ietf.org/doc/html/draft-hares-idr-fsv2-more-ip-filters-02>.

[I-D.ietf-idr-bgp-flowspec-label]
liangqiandeng, Hares, S., You, J., Raszuk, R., and D. Ma, "Carrying Label Information for BGP FlowSpec", Work in Progress, Internet-Draft, draft-ietf-idr-bgp-flowspec-label-02, 20 October 2022, <https://datatracker.ietf.org/doc/html/draft-ietf-idr-bgp-flowspec-label-02>.

[I-D.ietf-idr-flowspec-interfaceset]
Litkowski, S., Simpson, A., Patel, K., Haas, J., and L. Yong, "Applying BGP flowspec rules on a specific interface set", Work in Progress, Internet-Draft, draft-ietf-idr-flowspec-interfaceset-05, 18 November 2019, <https://datatracker.ietf.org/doc/html/draft-ietf-idr-flowspec-interfaceset-05>.

[I-D.ietf-idr-flowspec-l2vpn]
Weiguo, H., Eastlake, D. E., Litkowski, S., and S. Zhuang, "BGP Dissemination of L2 Flow Specification Rules", Work in Progress, Internet-Draft, draft-ietf-idr-flowspec-l2vpn-23, 15 April 2024, <https://datatracker.ietf.org/doc/html/draft-ietf-idr-flowspec-l2vpn-23>.

[I-D.ietf-idr-flowspec-mpls-match]
Yong, L., Hares, S., liangqiandeng, and J. You, "BGP Flow Specification Filter for MPLS Label", Work in Progress, Internet-Draft, draft-ietf-idr-flowspec-mpls-match-02, 20 October 2022, <https://datatracker.ietf.org/doc/html/draft-ietf-idr-flowspec-mpls-match-02>.

[I-D.ietf-idr-flowspec-nvo3]
Eastlake, D. E., Weiguo, H., Zhuang, S., Li, Z., and R. Gu, "BGP Dissemination of Flow Specification Rules for Tunneled Traffic", Work in Progress, Internet-Draft, draft-ietf-idr-flowspec-nvo3-20, 16 June 2024, <https://datatracker.ietf.org/doc/html/draft-ietf-idr-flowspec-nvo3-20>.

[I-D.ietf-idr-flowspec-path-redirect]
Van de Velde, G., Patel, K., and Z. Li, "Flowspec Indirection-id Redirect", Work in Progress, Internet-Draft, draft-ietf-idr-flowspec-path-redirect-12, 24 November 2022, <https://datatracker.ietf.org/doc/html/draft-ietf-idr-flowspec-path-redirect-12>.

[I-D.ietf-idr-flowspec-redirect-ip]
Uttaro, J., Haas, J., Texier, M., akarch@cisco.com, Ray, S., Simpson, A., and W. Henderickx, "BGP Flow-Spec Redirect to IP Action", Work in Progress, Internet-Draft, draft-ietf-idr-flowspec-redirect-ip-02, 5 February 2015, <https://datatracker.ietf.org/doc/html/draft-ietf-idr-flowspec-redirect-ip-02>.

[I-D.ietf-idr-flowspec-srv6]
Li, Z., Li, L., Chen, H., Loibl, C., Mishra, G. S., Fan, Y., Zhu, Y., Liu, L., and X. Liu, "BGP Flow Specification for SRv6", Work in Progress, Internet-Draft, draft-ietf-idr-flowspec-srv6-05, 29 March 2024, <https://datatracker.ietf.org/doc/html/draft-ietf-idr-flowspec-srv6-05>.

[I-D.ietf-idr-wide-bgp-communities]
Raszuk, R., Haas, J., Lange, A., Decraene, B., Amante, S., and P. Jakma, "BGP Community Container Attribute", Work in Progress, Internet-Draft, draft-ietf-idr-wide-bgp-communities-11, 9 March 2023, <https://datatracker.ietf.org/doc/html/draft-ietf-idr-wide-bgp-communities-11>.

[RFC0791] Postel, J., "Internet Protocol", STD 5, RFC 791, DOI 10.17487/RFC0791, September 1981, <https://www.rfc-editor.org/info/rfc791>.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[RFC4271] Rekhter, Y., Ed., Li, T., Ed., and S. Hares, Ed., "A Border Gateway Protocol 4 (BGP-4)", RFC 4271, DOI 10.17487/RFC4271, January 2006, <https://www.rfc-editor.org/info/rfc4271>.

[RFC4360] Sangli, S., Tappan, D., and Y. Rekhter, "BGP Extended Communities Attribute", RFC 4360, DOI 10.17487/RFC4360, February 2006, <https://www.rfc-editor.org/info/rfc4360>.

[RFC4456] Bates, T., Chen, E., and R. Chandra, "BGP Route Reflection: An Alternative to Full Mesh Internal BGP (IBGP)", RFC 4456, DOI 10.17487/RFC4456, April 2006, <https://www.rfc-editor.org/info/rfc4456>.

[RFC4760] Bates, T., Chandra, R., Katz, D., and Y. Rekhter, "Multiprotocol Extensions for BGP-4", RFC 4760, DOI 10.17487/RFC4760, January 2007, <https://www.rfc-editor.org/info/rfc4760>.

[RFC5065] Traina, P., McPherson, D., and J. Scudder, "Autonomous System Confederations for BGP", RFC 5065, DOI 10.17487/RFC5065, August 2007, <https://www.rfc-editor.org/info/rfc5065>.

[RFC5701] Rekhter, Y., "IPv6 Address Specific BGP Extended Community Attribute", RFC 5701, DOI 10.17487/RFC5701, November 2009, <https://www.rfc-editor.org/info/rfc5701>.

[RFC7153] Rosen, E. and Y. Rekhter, "IANA Registries for BGP Extended Communities", RFC 7153, DOI 10.17487/RFC7153, March 2014, <https://www.rfc-editor.org/info/rfc7153>.

[RFC7606] Chen, E., Ed., Scudder, J., Ed., Mohapatra, P., and K. Patel, "Revised Error Handling for BGP UPDATE Messages", RFC 7606, DOI 10.17487/RFC7606, August 2015, <https://www.rfc-editor.org/info/rfc7606>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

[RFC8955] Loibl, C., Hares, S., Raszuk, R., McPherson, D., and M. Bacher, "Dissemination of Flow Specification Rules", RFC 8955, DOI 10.17487/RFC8955, December 2020, <https://www.rfc-editor.org/info/rfc8955>.

[RFC8956] Loibl, C., Ed., Raszuk, R., Ed., and S. Hares, Ed., "Dissemination of Flow Specification Rules for IPv6", RFC 8956, DOI 10.17487/RFC8956, December 2020, <https://www.rfc-editor.org/info/rfc8956>.

[RFC9015] Farrel, A., Drake, J., Rosen, E., Uttaro, J., and L. Jalil, "BGP Control Plane for the Network Service Header in Service Function Chaining", RFC 9015, DOI 10.17487/RFC9015, June 2021, <https://www.rfc-editor.org/info/rfc9015>.

[RFC9117] Uttaro, J., Alcaide, J., Filsfils, C., Smith, D., and P. Mohapatra, "Revised Validation Procedure for BGP Flow Specifications", RFC 9117, DOI 10.17487/RFC9117, August 2021, <https://www.rfc-editor.org/info/rfc9117>.

[RFC9184] Loibl, C., "BGP Extended Community Registries Update", RFC 9184, DOI 10.17487/RFC9184, January 2022, <https://www.rfc-editor.org/info/rfc9184>.

[RFC9582] Snijders, J., Maddison, B., Lepinski, M., Kong, D., and S. Kent, "A Profile for Route Origin Authorizations (ROAs)", RFC 9582, DOI 10.17487/RFC9582, May 2024, <https://www.rfc-editor.org/info/rfc9582>.

9.2. Informative References

[I-D.ietf-idr-flowspec-v2] Hares, S., Eastlake, D. E., Yadlapalli, C., and S. Maduschke, "BGP Flow Specification Version 2", Work in Progress, Internet-Draft, draft-ietf-idr-flowspec-v2-04, 28 April 2024, <https://datatracker.ietf.org/doc/html/draft-ietf-idr-flowspec-v2-04>.

[RFC6241] Enns, R., Ed., Bjorklund, M., Ed., Schoenwaelder, J., Ed., and A. Bierman, Ed., "Network Configuration Protocol (NETCONF)", RFC 6241, DOI 10.17487/RFC6241, June 2011, <https://www.rfc-editor.org/info/rfc6241>.

[RFC8040] Bierman, A., Bjorklund, M., and K. Watsen, "RESTCONF Protocol", RFC 8040, DOI 10.17487/RFC8040, January 2017, <https://www.rfc-editor.org/info/rfc8040>.

[RFC8205] Lepinski, M., Ed. and K. Sriram, Ed., "BGPsec Protocol Specification", RFC 8205, DOI 10.17487/RFC8205, September 2017, <https://www.rfc-editor.org/info/rfc8205>.

[RFC8206] George, W. and S. Murphy, "BGPsec Considerations for Autonomous System (AS) Migration", RFC 8206, DOI 10.17487/RFC8206, September 2017, <https://www.rfc-editor.org/info/rfc8206>.

[RFC8300] Quinn, P., Ed., Elzur, U., Ed., and C. Pignataro, Ed., "Network Service Header (NSH)", RFC 8300, DOI 10.17487/RFC8300, January 2018, <https://www.rfc-editor.org/info/rfc8300>.

Authors' Addresses

Susan Hares
Hickory Hill Consulting
7453 Hickory Hill
Saline, MI 48176
United States of America
Phone: +1-734-604-0332
Email: shares@ndzh.com

Donald Eastlake
Independent
2386 Panoramic Circle
Apopka, FL 32703
United States of America
Phone: +1-508-333-2270
Email: d3e3e3@gmail.com

Chaitanya Yadlapalli
ATT
United States of America
Email: cy098d@att.com

Sven Maduschke
Verizon
Germany
Email: sven.maduschke@de.verizon.com

Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: 6 December 2024

L. Dunbar
Futurewei
K. Majumdar
Microsoft Azure
S. Hares
Hickory Hill Consulting
R. Raszuk
Arrcus
V. Kasiviswanathan
Arista

4 June 2024

BGP UPDATE for SD-WAN Edge Discovery
draft-ietf-idr-sdwan-edge-discovery-13

Abstract

The document describes the encoding of BGP UPDATE messages for the SD-WAN edge node property discovery.

In the context of this document, BGP Route Reflector (RR) is the component of the SD-WAN Controller that receives the BGP UPDATE from SD-WAN edges and in turns propagates the information to the intended peers that are authorized to communicate via the SD-WAN overlay network.

Requirements Language

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 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

Status of This Memo

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

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at https://datatracker.ietf.org/drafts/current/.

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."

This Internet-Draft will expire on 6 December 2024.

Table of Contents

1. Introduction

BGP [RFC4271] can be used as a control plane for a SD-WAN network. SD-WAN network refers to a policy-driven network over multiple heterogeneous underlay networks to get better WAN bandwidth management, visibility, and control.

The document describes BGP UPDATE messages for an SD-WAN edge node to advertise its properties to its RR which then propagates that information to the authorized peers.

2. Conventions used in this document

The following acronyms and terms are used in this document:

Cloud DC: Off-Premises Data Centers that usually host applications and workload owned by different organizations or tenants.

Color-EC: Color Extended Community defined in [RFC9012].

Controller: Used interchangeably with SD-WAN controller to manage SD-WAN overlay path creation/deletion and monitor the path conditions between sites.

CPE: Customer (Edge) Premises Equipment

CPE-Based VPN: Virtual Private Secure network formed among CPEs. This is to differentiate such VPNs from most commonly used PE-based VPNs discussed in [RFC4364].

Encaps-EC: Encapsulation Extended Community defined in [RFC9012].

MP-NLRI: Multi-Protocol Network Layer Reachability Information [MP_REACH_NLRI] Path Attribute defined in [RFC4760].

RR: Route Reflector

SD-WAN: An overlay connectivity service that optimizes transport of IP Packets over one or more Underlay Connectivity Services by recognizing applications (Application Flows) and determining forwarding behavior by applying Policies to them. [MEF-70.1]

SD-WAN Endpoint: can be the SD-WAN edge node address, a WAN port address (logical or physical) of a SD-WAN edge node, or a client port address.

SD-WAN Hybrid tunnel: A single logical tunnel that combines several links of different encapsulation iinto a single tunnel.

RT-EC: Route Target Extended Community [RFC4360]

TEA: Tunnel Encapsulation Path Attribute [RFC9012]

VPN: Virtual Private Network

VRF: VPN Routing and Forwarding instance

WAN: Wide Area Network

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC8174] when, and only when, they appear in all capitals, as shown here.

3. Framework of SD-WAN Edge Discovery

3.1. The Objectives of SD-WAN Edge Discovery

The objectives of SD-WAN edge discovery for an SD-WAN edge node are to discover its authorized BGP peers and each peer's associated properities in order to establish secure overlay tunnels [Net2Cloud]. The attributes to be propagated include:

* the SD-WAN (client) VPNs information,

* the attached routes under the SD-WAN VPNs, and

* the properties of the underlay networks over which the client routes can be carried.

Some SD-WAN peers are connected by both trusted VPNs and untrusted public networks. Some SD-WAN peers are connected only by untrusted public networks. For the traffic over untrusted networks, IPsec Security Associations (IPsec SA) must be established and maintained. For the trusted VPNs, IPsec Security associations may not be set-up. If an edge node has network ports behind a NAT, the NAT properties need to be discovered by the authorized SD-WAN peers.

Like any VPN networks, the attached client routes belonging to specific SD-WAN VPNs can only be exchanged with the SD-WAN peer nodes authorized to communicate.

3.2. Comparing with Pure IPsec VPN

A pure IPsec VPN has IPsec tunnels connecting all edge nodes over public networks. Therefore, it requires stringent authentication and authorization (i.e., IKE Phase 1) before other properties of IPsec SA can be exchanged. The IPsec Security Association (SA) between two untrusted nodes typically requires the following configurations and message exchanges:

IPsec IKEV2: messages sent to authenticate with each other.

Establish IPsec SA : requires the following set-up

- Local key configuration,

- Setting the Remote Peer address,

- Information from IKEv2 Proposal directly sent to peer (Encryption method, Integrity sha512, etc.), and

- Transform set.

Attached client prefixes discovery acchieved by:
- Running routing protocol within each IPsec SA.

- If multiple IPsec SAs between two peer nodes are established to achieve load sharing, each IPsec tunnel needs to run its own routing protocol to exchange client routes attached to the edges.

Set-up Access List or Traffic Selector: such as Permit Local-IP1, Remote-IP2, and other parameters.

In a BGP-controlled SD-WAN network over hybrid MPLS VPN and public internet underlay networks, all edge nodes and RRs are already connected by private secure paths. The RRs have the policies to manage the authentication of all peer nodes. More importantly, when an edge node needs to establish multiple IPsec tunnels to many edge nodes, all the management information can be multiplexed into the secure management tunnel between RR and the edge node operating as a BGP peer. Therefore, the amount of authentication in a BGP-Controlled SD-WAN network can be significantly reduced.

Client VPNs are configured via VRFs, just like the configuration of the existing MPLS VPN. The IPsec equivalent traffic selectors for local and remote routes are achieved by importing/exporting VPN Route Targets. The binding of client routes to IPsec SA is dictated by policies. As a result, the IPsec configuration for a BGP controlled SD-WAN (with mixed MPLS VPN) can be simplified in the following manner:

* The SD-WAN controller has the authority to authenticate edges and peers so the Remote Peer association is controlled by the SD-WAN Controller (RR).

* The IKEv2 proposals (including the IPsec Transform set) can be sent directly to peers, or incorporated in a BGP UPDATE.

* The BGP UPDATE announces the client route reachability through the SDWAN hybrid tunnels. A SDWAN hybrid tunnnel combines several other tunnels into a single logical tunnel. The SD-WAN Hybrid tunnel implementations insure that all tunnels within are either running over secure network links or secured by IPsec.

* Importing/exporting Route Targets under each client VPN (VRF) achieves the traffic selection (or permission) among clients' routes attached to multiple edge nodes.

Note that with this method there is no need to run multiple routing protocols in each IPsec tunnel.

3.3. Client Routes and SDWAN UPDATE

There are two different sequences of BGP UPDATE messages are used for SD-WAN Edge Discovery. The first associates Client routes BGP Updates with the SD-WAN Hybrid Tunnel. The second passes information regarding the SD-WAN Underlay between Egress routers and Ingress routers.

3.3.1. Client Routes

This section describes how client routes in BGP Updates are associated with the SD-WAN Hybrid Tunnel.

Client routes BGP UPDATE:

This BGP UPDATE message contains the client routes (NLRI), Next Hop, and the attributes which identify the Hybrid SD-WAN tunnel toward the Next Hop. The SD-WAN-Hybrid Tunnel BGP attributes are either passed as either:

* Encapsulation Extended Community [Encap-EC] which identifies the SD-WAN-hybrid tunnel with a Tunnel Egress End Point as NextHop in BGP Update [per RFC9012],

* Tunnel Encapsulation Attribute (TEA) which identifies the SD-WAN-Hybrid tunnel and uses the Tunnel Egress Endpoint SubTLV to identify the egress endpoint (see [RFC9012] section 3.1)

Ordering: If both the TEA and the Extended Community for tunnel information exists, the Extended Community is preferred (i.e. takes precedence.)

Sample Topology: For a Hybrid SD-WAN Tunnel

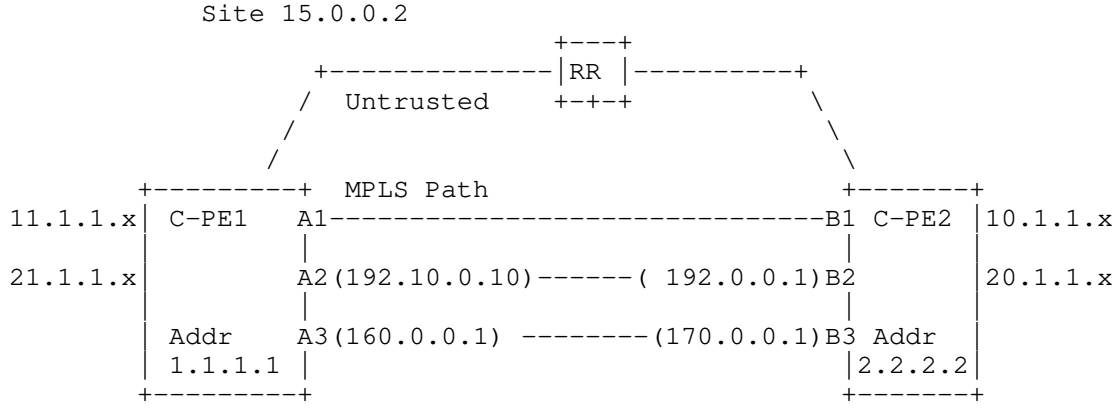


Figure 1: Hybrid SD-WAN

Example Client Routes: In figure 1, four overlay paths between C-PE1 and C-PE2 are established for illustration purpose. More overlay paths are possible. One physical port on C-PE2 can terminate multiple overlay paths from different ports on C-PE1.

a) MPLS-in-GRE path;

b) node-based IPsec tunnel [2.2.2.2 - 1.1.1.1]. As C-PE2 has two public internet facing WAN ports, either of those two WAN port IP addresses can be the outer destination address of the IPsec encapsulated data packets;

c) port-based IPsec tunnel [192.0.0.1 - 192.10.0.10]; and

d) port-based IPsec tunnel [172.0.0.1 - 160.0.0.1].

C-PE2 advertises the attached client routes in one of two forms below:

Client Route UPDATE - [RFC9012] "Barebones" (option 1)

```
NLRIs: AFI=IPv4 (1) and SAFI = VPN (128)
  Prefix: 10.1.1.x; 20.1.1.x
  NextHop: 2.2.2.2 (C-PE2)
Attributes:
  Extended community RT: (RT-EC) for SD-WAN VPN 1
  Encapsulation Extended Community (Encaps-EC)
        tunnel-type=SD-WAN-Hybrid
```

Figure 2: Client Routes Update Message

Client Route UPDATE [RFC9012] Tunnel Encaps Attribute (TEA) (option 2)

```
NLRIs: AFI=IPv4 (1) and SAFI = VPN (128)
  Prefix: 10.1.1.x; 20.1.1.x
      NextHop: 2.2.2.2 (C-PE2)
Attributes:
  Extended community RT: (RT-EC) for SD-WAN VPN 1
  Tunnel Encapsulation Attribute (TEA)
    Tunnel Egress Endpoint SubTLV: 2.2.2.2 (C-PE2)
```

Figure 3: Client Routes Update Message

3.3.2. SD-WAN Underlay UPDATE

Edges nodes use this BGP UPDATE to advertise the properties of directly attached underlay networks and IPsec SA attributes associated with an SD-WAN-Hybrid tunnel. The properties of underlay networks include encapsulation, NAT and underlay physical properties. The IPsec SA attributes passed include keys, nonce, and encryption algorithms, and other IP SEC attributes.

The security attributes are likely to change more rapidly than the physical attributes of links within the Hybrid SD-Wan Tunnel. Typically the attributes of the links are passed during initial set-up of Hybrid SD-WAN tunnels in the network.

Given the topology in figure 1, C-PE2 can send the SD-WAN NLRI in a BGP Update messages to advertise the properties of Internet facing ports 192.0.0.1 and 170.0.0.1, and their associated IPsec SA related parameters.

Example 1 below provides sample BGP Updates per port. This type of UPDATE packing provides poor packing of UPDATES, but it may occur. Example 2 provides a single BGP Update which passes the initial information in one update. In the update, Color-EC is Color Extended Community [RFC4360].

3.3.2.1. Example #1 SD-WAN Underlay Update Messages Used in Set-Up

Example #1 - BGP Updates per Port

```
# Update #1 for Port B2 on C-PE2
  SD-WAN NLRI (AFI=1/SAFI=SD-WAN)
    Port-id = Local port ID for WAN Port 192.0.0.1
    SD-WAN Color = Match to Color-EC of Client routes
    SD-WAN Node ID = 2.2.2.2 (C-PE2)
  Tunnel Encaps Attribute (TEA)
    Tunnel TLV: (type: Hybrid-SD-WAN)
      Tunnel Egress Endpoint SubTLV: 2.2.2.2
      Extended-Port Attribute SubTLV: Port B2 (192.0.0.1)
      IPsec SA Identifier SubTLV:  SA-1, SA-2

# Update #2 for Port B3 (IPsec) in Hybrid tunnel on PE2
  SD-WAN NLRI (AFI=1/SAFI=SD-WAN)
    Port-id = Local port ID for WAN Port 170.0.0.1
    SD-WAN Color = Match to Color-EC of Client Routes
    SD-WAN Node ID = 2.2.2.2 (C-PE2)
  Tunnel Encaps Attribute (TEA)
    Tunnel TLV: (type: SD-Wan-Hybrid)
      Tunnel Egress Endpoint SubTLV (SubTLV=6) = 2.2.2.2
          Extended Port Attribute SubTLV: Port B3 (170.0.0.1)
          Simplified IPSec SD-WAN SubTLV: SA-3, SA-4

See section 7.1 for Extended Port Attribute SubTLV definition.
See section 8.1 for IPsec SA Identifier SubTLV.
See section 8.5 for Simplified IPsec SD-WAN SubTlv.
```

Figure 4: SDWAN NLRI Update per Port

3.3.2.2. Example #2 IPsec terminated at Node with Hybrid Tunnel

```
Example #2 - IP Sec Terminated at Node C-PE2



# Ports B1 (MPLS), B2 (IPsec), B3 (IPsec)
# Update for C-PE2 (IPSec)
  SD-WAN NLRI (AFI=1 / SAFI = SD-WAN)
    SD-WAN Color = Match to Color-EC of Client routes
        Port ID = 0
        SD-WAN Node ID = 2.2.2.2
  Tunnel Encaps Attribute (TEA)
        Tunnel TLV (type: SD-Wan-Hybrid)
      Egress Endpoint = 2.2.2.2
      IPsec-SA-ID: SA-1, SA-2, SA-3, SA-4
```

Figure 5: SDWAN NLRI Update 3 ports

3.4. Edge Node Discovery

The basic scheme of SD-WAN edge node discovery using BGP consists of the following:

* Secure connection to a SD-WAN controller (BGP RR in this context):
  - For an SD-WAN edge with both MPLS and IPsec paths, the edge node should already have a secure connection to its controller (RR in this context). For an SD-WAN edge that is only accessible via Internet, the SD-WAN edge upon power-up establishes a secure tunnel (such as TLS or SSL) with the SD-WAN central controller whose address is preconfigured on the edge node. The central controller informs the edge node of its local RR. The edge node then establishes a transport layer secure session with the RR (such as TLS or SSL).
* The BGP Peer Edge node will advertise the properties of its Hybrid SD-WAN Tunnel to its designated RR via the secure connection.
* The RR propagates the received information to the authorized BGP peers.
* The authorized BGP peers can establish the secure data channels via Hybrid SD-WAN tunnel and exchange more information among each other.

For an SD-WAN deployment with multiple RRs, it is assumed that there are secure connections among those RRs. How secure connections are established among those RRs is out of the scope of this document. The existing BGP UPDATE propagation mechanisms control the edge properties propagation among the RRs.

For some environments where the communication to RR is highly secured, [RFC9016] IKE-less can be deployed to simplify IPsec SA establishment among edge nodes.

4. Constrained propagation of BGP UPDATE

4.1. SD-WAN Segmentation, Virtual Topology and Client VPN

In SD-WAN deployment, SD-WAN Segmentation is a frequently used term which refers to partitioning a network into multiple subnetworks, just like MPLS VPNs. SD-WAN Segmentation is achieved by creating SD-WAN virtual topologies and SD-WAN VPNs. An SD-WAN virtual topology consists of a set of edge nodes and the tunnels (a.k.a. underlay paths) interconnecting those edge nodes. These tunnels forming the underlay paths can be IPsec tunnels, or MPLS VPN tunnels, or other

tunnels.

An SD-WAN VPN is configured in the same way as the VRFs of an MPLS VPN. One SD-WAN client VPN can be mapped to multiple SD-WAN virtual topologies. SD-WAN Controller governs the policies of mapping a client VPN to SD-WAN virtual topologies.

Each SD-WAN edge node may need to support multiple VPNs. Route Target is used to differentiate the SD-WAN VPNs. For example, in the picture below, the Payment-Flow on C-PE2 is only mapped to the virtual topology of C-PEs to/from Payment Gateway, whereas other flows can be mapped to a multipoint-to-multipoint virtual topology.

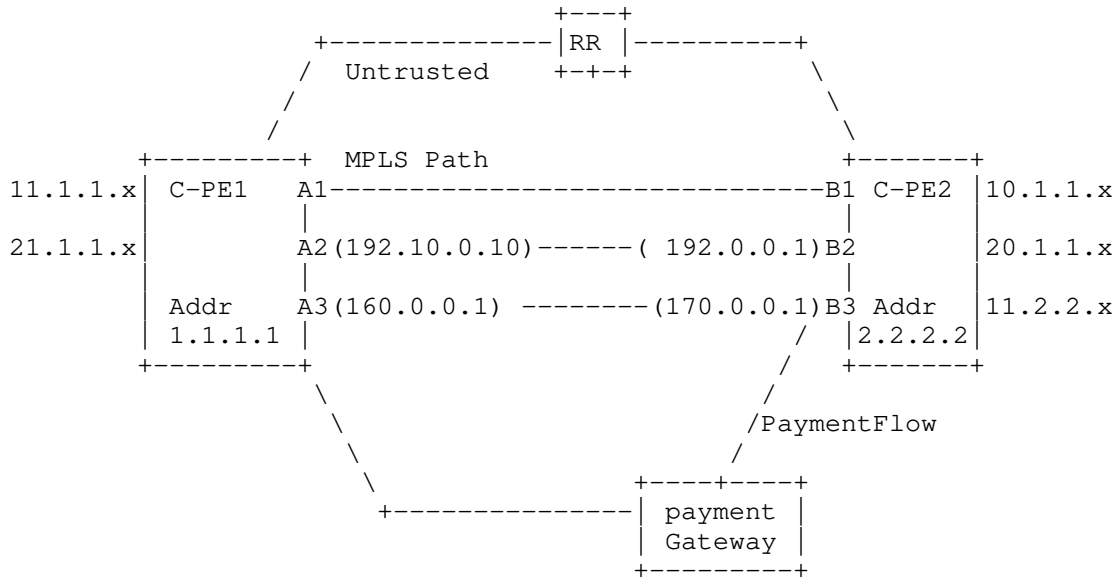


Figure 6: SD-WAN Virtual Topology and VPN

4.2. Constrained Propagation of Edge Capability

BGP Route Reflectors [RFC4456] may be configured to constrain the distribution of BGP informtion to specific BGP clients.

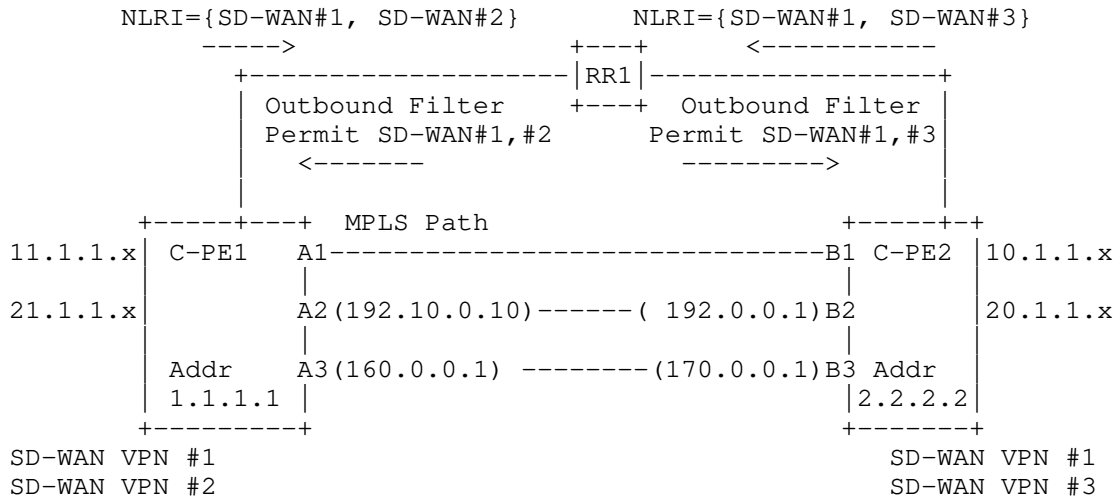


Figure 7: Constraint propagation of Edge Property

The RR is configured to speak to the BGP clients (CE-PE1 and CE-PE2) over secure virtual links (IPsec), and send only certain routes. The configuration on the RR and the BGP Peers sending SD-WAN routes forms a "walled garden" for the SD-WAN information.

It is out of the scope of this document on how RR is configured with the policies to filter out unauthorized nodes for specific SD-WAN VPNs.

5. Client Route UPDATE Procedures

The Tunnel Encapsulation Attribute for the SD-WAN Hybrid Tunnel Type may be associated with BGP UPDATE messages with NLRI with AFI/SAFI IPv4 Unicast (1/1), IPv4 with MPLS labels (1/4), IPVPN-IPv4 Label Unicast (1/128), IPv6 Unicast (2/1), IPv6 with MPLS labels (2/4), VPN-IPv6 Label Unicast (2/128), and EVPN (25/70).

When associated with any NLRI in this set, these routes are described as "Client Routes" in this document. Based on [RFC9012], there are two forms a Tunnel Encapsulation Attribute (TEA) can take: "Barebones" using the Encapsulation Extended Community (Encaps-EC) and a normal Tunnel Encapsulation form.

5.1. Tunnel Form 1: Encapsulation Extended Community (Encaps-EC)

The SD-WAN Client Route UPDATE message uses the Encapsulation Extended Community (Encap-EC) to identify the Hybrid SD-WAN tunnel and the Tunnel Egress Endpoint. Per [RFC9012], the Encapsulation Extended Community uses the NextHop Field in the BGP UPDATE as the Tunnel Egress EndPoint. The validation for the Tunnel Egress Endpoint uses the validation in section 6, 8, and 13 applied to the NextHop.

A Color Extended Community (Color-EC) or local policy applied to the client route directs the traffic for the client route to across appropriate interface within the Hybrid SD-WAN Tunnel to the Tunnel Egress Endpoint.

5.2. Tunnel Form 2: Tunnel Encapsulation Path Attribute (TEA)

The Client route with the Tunnel Encapsulation Path Attribute (TEA) with the Hybrid SD-WAN route TLV must have the Tunnel Egress Endpoint (SubTLV=6) and any of the SubTLVs found in [RFC9012]. The validation for the Tunnel Egress Endpoint uses the validation in section 6, 8, and 13 from [RFC9012].

5.3. Multiple tunnels attached to One Route

A single SD-WAN client route may be attached to multiple SD-WAN Hybrid tunnels. An Update with an SD-WAN client route may express these tunnels as an Encap-EC or a TEA. Each of these tunnel descriptions is treated as a unique Hybrid SD-WAN tunnel with a unique Egress Endpoint. Local Policy on the BGP Peer determines which tunnel the client data traffic will use.

5.4. SD-WAN VPN ID in Client Route Update

An SD-WAN VPN ID is same as a client VPN in a BGP controlled SD-WAN network. The Route Target Extended Community should be included in a Client Route UPDATE message to differentiate the client routes from routes belonging to other VPNs. Route Target value is taken as the VPN ID (for 1/1 and 2/1). For 1/128 and 2/128, the RD from the NLRI identifies the VPN ID. For EVPN, picking up the VPN-ID from EVPN SAFI.

5.5. SD-WAN VPN ID in Data Plane

SD-WAN edge node which can be reached by either an MPLS path or an IPsec path within the hybrid SD-WAN tunnel. If client packets are sent via a secure MPLS network within the Hybrid SD-WAN tunnel, then the data packets will have MPLS headers with the MPLS Labels based on the scheme specified by [RFC8277]. It is assumed the secure MPLS network assures the security outer MPLS Label header.

If the packets are sent via a link with IPsec outer encryption across a public network, the payload is still encrypted with GRE or VXLAN encryption. For GRE Encapsulation within an IPsec tunnel, the GRE key field can be used to carry the SD-WAN VPN ID. For network virtual overlay (VxLAN, GENEVE, etc.) encapsulation within the IPsec tunnel, the Virtual Network Identifier (VNI) field is used to carry the SD-WAN VPN ID.

6. SD-WAN Underlay UPDATE

The hybrid SD-WAN underlay tunnel UPDATE is to advertise the detailed properties associated with the public facing WAN ports and IPsec tunnels. The Edge BGP Peer will advertise the SD-WAN properties to its designated RR via the secure connection. The BGP Update message will contain the SD-WAN Underlay NLRI and a Tunnel Encapsulation Attribute (TEA) with SubTLVs for Extended Port attribute (see section 7) or IP Sec information (see section 8). The IPsec information subTLVs include: IPsec-SA-ID, IPsec SA Nonce, IPsec Public Key, IPsec SA Proposal, and Simplified IPsec SA. IP

6.1. NLRI for SD-WAN Underlay Tunnel Update

A new NLRI SAFI (SD-WAN SAFI=74) is introduced within the MP_REACH_NLRI Path Attribute of [RFC4760] for advertising the detailed properties of the SD-WAN tunnels terminated at the edge node:

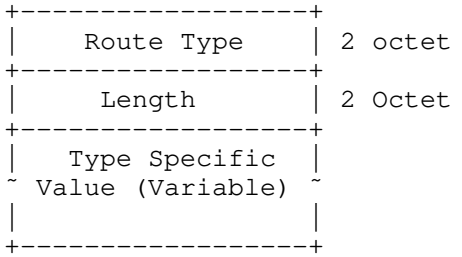


Figure 8: SD-WAN NLRI

Where

Route (NLRI) Type: 2 octet value to define the encoding of the rest of the SD-WAN the NLRI.

Length: 2 octets of length expressed in bits as defined in [RFC4760].

This document defines the following SD-WAN Route type:

NLRI Route-Type = 1: For advertising the detailed properties of the SD-WAN tunnels terminated at the edge, where the transport network port can be uniquely identified by a tuple of three values (Port-Local-ID, SD-WAN-Color, SD-WAN-Node-ID). The SD-WAN NLRI Route-Type =1 has the following encoding:

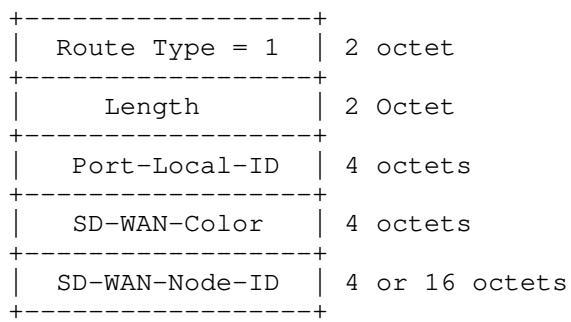


Figure 9: SD-WAN NLRI Route Type 1

Port-local-ID: SD-WAN edge node Port identifier, which is locally significant. If the SD-WAN NLRI applies to multiple WAN ports, this field is zero.

SD-WAN-Color: represents a group of tunnels, which correlate with the Color-Extended-community included in the client routes UPDATE. When a client route can be reached by multiple SD-WAN edges co-located at one site, the SD-WAN-Color can represent a group of tunnels terminated at those SD-WAN edges co-located at the site. If an SD-WAN-Color represents all the tunnels at a site, then the SD-WAN-Color effectively represents the site.

SD-WAN Node ID: The node's IPv4 or IPv6 address.

Route Type values outside of 1 are out of scope for this document.

6.2. SD-WAN-Hybrid Tunnel Encoding

A new BGP Tunnel-Type SD-WAN-Hybrid (code point 25) indicates hybrid underlay tunnels.

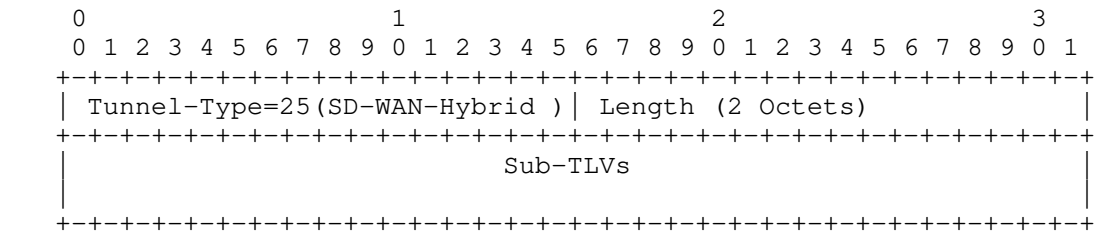


Figure 10: SD-WAN Hybrid Value Field

The valid SubTLVs for the Hybrid Tunnel type include subTLVs specified in [RFC9012] and the following SubTLVs specified in this document:

* Extended Port Attributes SubTLV
* IPsec SA-ID SubTLV
* IPSec SA Rekey SubTLV
* IPsec Public Key SubTLV
* IPsec SA Proposal SubTLV
* Simplified IPsec SA SubTLV

The Extended Port Attributes are described in section 7, and the IPsec related SubTLVs are described in section 8.

7. Extended Port Attribute Sub-TLV

The SD-WAN Underlay NLRI is sent with a Tunnel Encapsulation Attribute with the Extended Port Attribute Sub-TLV advertises the properties associated with a public Internet-facing WAN port that might be behind NAT. An SD-WAN edge node can query a STUN Server (Session Traversal of UDP through Network address translation [RFC8489]) to get the NAT properties, including the public IP address and the Public Port number, to pass to its peers.

The location of a NAT device can be:

* Only the initiator is behind a NAT device. Multiple initiators can be behind separate NAT devices. Initiators can also connect to the responder through multiple NAT devices.

* Only the responder is behind a NAT device.

* Both the initiator and the responder are behind a NAT device.

The initiator's address and/or responder's address can be dynamically assigned by an ISP or when their connection crosses a dynamic NAT device that allocates addresses from a dynamic address pool.

As one SD-WAN edge can connect to multiple peers, the pair-wise NAT exchange as IPsec's IKE[RFC7296] is not efficient. In the BGP Controlled SD-WAN, NAT properties for a WAN port are encoded in the Extended Port Attribute sub-TLV, which the following format:

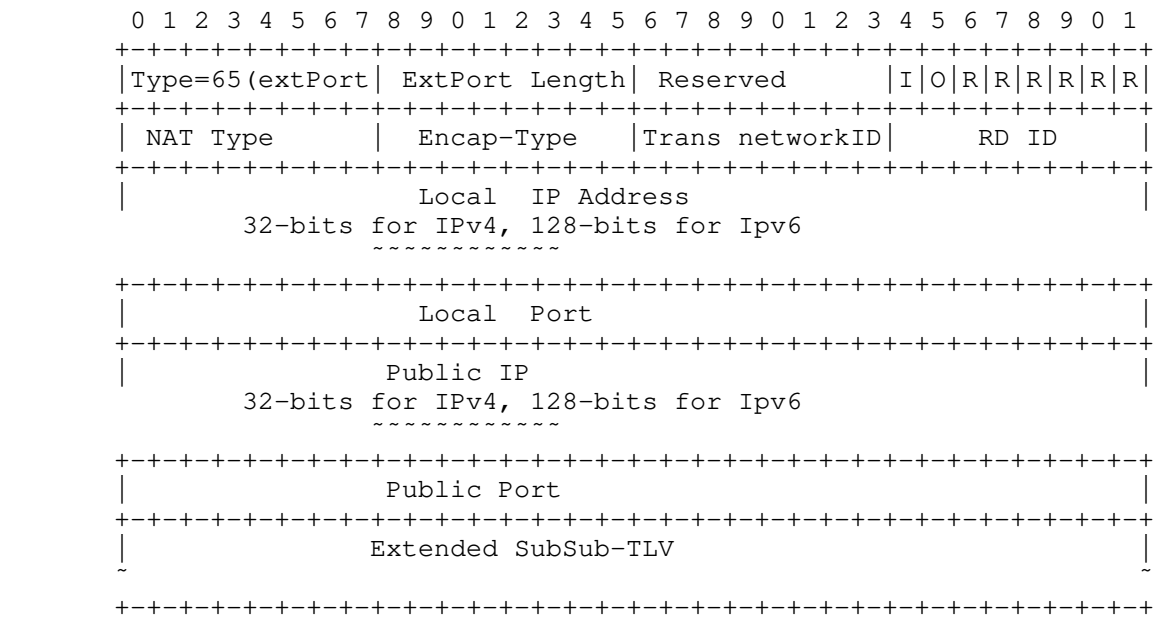


Figure 11: Extended Port Attribute Sub-TLV

Where:

* Extended Port Attribute Type (=65): indicating it is the Extended Port Attribute SubTLV

* ExtPort Length: the length of the subTLV in octets (variable).

* Flags:
  - I bit (CPE port address or Inner address scheme):
    o If set to 0, indicate the inner (private) address is IPv4.
    o If set to 1, indicates the inner address is IPv6.
  - O bit (Outer address scheme):
    o If set to 0, indicate the inner (private) address is IPv4.
    o If set to 1, indicates the inner address is IPv6.
  - R bits: reserved for future use. Must be set to 0, and ignored upon reception.
* NAT Type: the NAT type can be one of the following values:
  - 1: without NAT ;
  - 2: 1-to-1 static NAT;
  - 3: Full Cone;
  - 4: Restricted Cone;
  - 5: Port Restricted Cone;
  - 6: Symmetric; or
  - 7: Unknown (i.e. no response from the STUN server).

  NAT type values outside of 1-7 are invalid for this SubTLV.
* Encap-Type: the supported encapsulation types for the port.
  - Encap-Type=1: GRE;
  - Encap-Type=2: VxLAN;

  Notes:
  - The Encap-Type inside the Extended Port Attribute Sub-TLV is different from the RFC9012's BGP-Tunnel-Encapsulation type. The port can indicate the specific encapsulations, such as:

- If the IPsec-SA-ID subTLV or the IPsec SA detailed subTLVs (Nonce/publicKey/Proposal) are included in the SD-WAN-Hybrid tunnel, the Encap-Type indicates the encapsulation type within the IPsec payload.

- If the IPsec SA subTLVs are not included in the SD-WAN-Hybrid Tunnel, the Encap-Type indicates the encapsulation of the payload without IPsec encryption.

- Encapsulation types outside of GRE and VxLAN are outside of the scope of this specification.

* Transport Network ID: Central Controller assigns a global unique ID to each transport network. Any value in this octet is valid

* RD ID: Routing Domain ID, need to be globally unique. Any value in this octet is valid.

  - Some SD-WAN deployment might have multiple levels, zones, or regions that are represented as logical domains. Policies can govern if tunnels can be established across domains. For example, a hub node can establish tunnels with different logical domains but the spoke nodes cannot establish tunnels with nodes in different domains.

* Local IP: The local (or private) IP address of the WAN port.

* Local Port: used by Remote SD-WAN edge node for establishing IPsec to this specific port.

* Public IP: The IP address after the NAT. If NAT is not used, this field is set to all-zeros

* Public Port: The Port after the NAT. If NAT is not used, this field is set to all-zeros.

* Extended SubSub-TLV: for carrying additional information about the underlay networks.

7.1. Extended SubSub-TLV

One Extended SubSub-TLVs is specified in this document: Underlay Network Transport SubSub-TLV

7.1.1. Underlay Network Transport SubSub-TLV

The Underlay Network Transport SubSub-TLV is an optional Sub-TLV to carry the WAN port connection types and bandwidth, such as LTE, DSL, Ethernet, etc.

The format of this Sub-TLV is as follows:

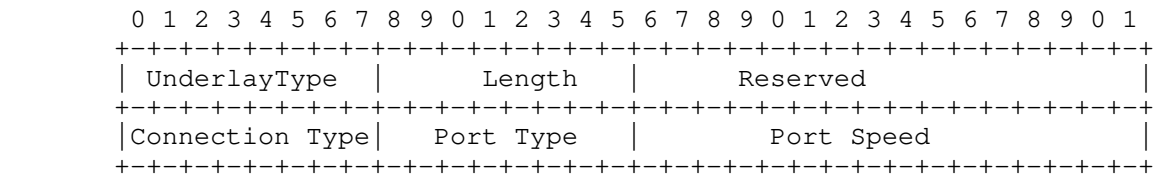


Figure 12: Underlay Network SubSub-TLV

Where:

Underlay Network Properties: sub Type=66

Length: always 6 bytes

Reserved: 2 octet of reserved bits. It SHOULD be set to zero on transmission and MUST be ignored on receipt.

Connection Type: are listed below as:

* 1 = Wired
* 2 = WIFI
* 3 = LTE
* 4 = 5G
* Any value outside of 1-4 is outside the scope of this specification.

Port Type: Port type define as follows:

* 1 = Ethernet
* 2 = Fiber Cable
* 3 = Coax Cable

* 4 = Cellular

* Any value outside of values 1-4 are outside the scope of this specification.

Port Speed: The port seed is defined as 2 octet value. The values are defined as Gigabit speed. For example, a value of 1 would mean 1 gigabit. The port speed of "0" is not valid.

The connection types of equipment and port types will continue to grow with technology change. Future specifications may specify additional connection types or port types.

8. IPsec SA Property Sub-TLVs

This section describes the SubTLVs that pass data regarding IPsec parameters for the Hybrid SD-WAN tunnel. During set-up of the Hybrid SD-WAN tunnels, the IPsec parameters need to be securely passed to set-up secure association. For hybrid SD-WAN tunnels, the IPsec security association for IPsec links may change to different security associations over time.

The IPSec subTLVs supported by the Hybrid Tunnel type are: IPSec-SA-ID, IPsec SA Nounce, IPsec Public Key, IPsec Proposal, and Simplified IPSec SA. The IPSec-SA-ID SubTLV provides a way to indicate the IPsec SA Identifiers (section 8.1) for pre-configured security association. The other four SubTLVs provide different ways to pass details regarding IPsec security associations. The IPsec SA Nounce passes Nounce and rekey counters for a Secure Association identified by IPsec SA Identifier (see section 8.2). The IPSec Public Key SubTLV passes IPsec Public Key data with a time duration (see section 8.3). The IPsec Proposal SubTLV provides Transform attributes and Transform IDs (see section 8.4). The Simplified IP SEC SA passes the information that identifies configuration for 2 keys (see section 8.5).

For a quick rotation between security associations, the SDWAN NLRI (port-id, color, node) can quickly distribute a switch to a set of new security association using the BGP Update message. In this case, the BGP UPDATE message would like figure 10

```
SDWAN NLRI
       Route-type: 1
       length: 12
       port-id - 0.0.0.0
       SD-WAN-Color - 0.0.0.1
       node-id - 2.2.2.2
TEA:
 Tunnel TLV: (type: SD-WAN Hybrid)
       Tunnel Egress Endpoint SubTLV: 2.2.2.2
       IPsec-SA-ID SubTLV: 20, 30
```

Figure 13: SD-WAN NLRI IPsec rotation in attack

8.1. IPsec-SA-ID Sub-TLV

IPsec-SA-ID Sub-TLV within the Hybrid Underlay Tunnel UPDATE indicates one or more pre-established IPsec SAs by using their identifiers, instead of listing all the detailed attributes of the IPsec SAs.

Using an IPsec-SA-ID Sub-TLV not only greatly reduces the size of BGP UPDATE messages, but also allows the pairwise IPsec rekeying process to be performed independently.

The following is the structure of the IPsec-SA-ID sub-TLV

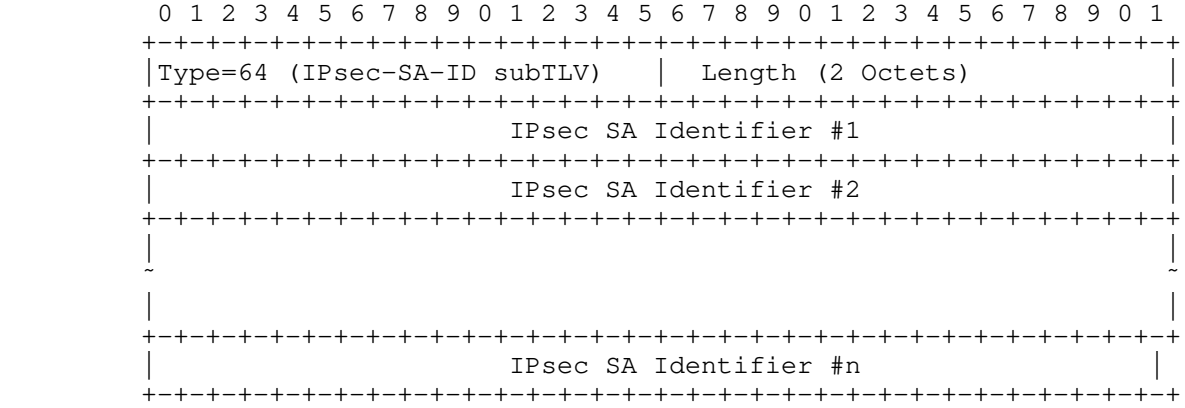


Figure 14: IPsec-SA-ID Sub-TLV

where:

length is the length of the subTLV in octets (must be on 4 octet boundary)

The length is followed by a sequence of IPsec SA Identifiers of length 4 octets.

- IPSec SA Identifier #1 - is a 4 octet identifier for a IP Security association.

- IPSec SA Identifier #2 - is a 4 octet identifier for a IP Security association.

- IPSec SA Identifier #n - is a 4 octet identifier for a IP Security association.

8.2. IPsec SA Rekey Counter Sub-TLV

The IPsec SA Rekey Counter Sub-TLv provides the rekey counter for a security association (identified by IPsec SA Identifier).

The format of this Sub-TLV is as follows:

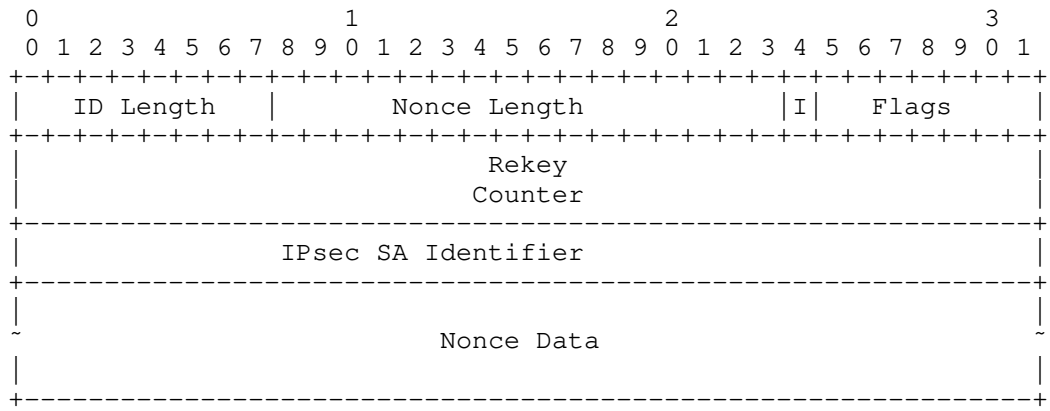


Figure 15: IPsec SA Rekey Counter Sub-TLV

where:

ID Length - is a 1 octet value indicating the length of SA-Identifer length. This length should be 4 octets.

Nonce length - is a 2 octet value indicate the length in octets of the Nonce Data.

Flags: - is 1 octet field with the following form [I-R-R-R-R-R-R]

- I - indicates status of initial contact (1 = initial, 0 = follow-on) in the left most bit.

- R - Reserved bits - which are ignored upon reception and set to zero upon transmission.

IPsec SA Identifier (IPSec-SA-ID) - identifies a specific IPsec SAs terminated at the edge. The length is 4 octets.

Nonce Data - a random or pseudo-random number for preventing replay attacks. Its length is a multiple of 32 bits[RFC7296].

Note:

* The IPsec-SA-ID may also refer to the values carried in the same TEA in the same Tunnel TLV (type SD-WAN Hybrid) as the IPsec SA Rekey SubTLV in either the IPsec Public Key SubTLV or the IPsec SA Proposal SubTLV. The IPsec SA Rekey Counter, IPsec Public Key, and IPsec SA Proposal SubTLVs work together to create security associations.

* The IPsec-SA-ID may refer to information in another Tunnel TLV in the same TEA associated wiht the same BGP UPDATE message as the IPsec SA Rekey Counter sub-TLV.

* The IPsec-SA-ID can be used in the IPsec-SA-ID subTLV of a different BGP UPDATE message.

8.3. IPsec Public Key Sub-TLV

The format of this Sub-TLV is as follows:

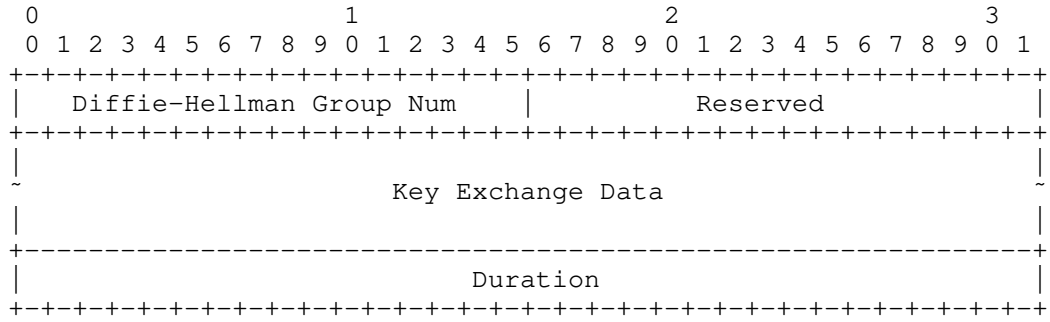


Figure 16: IPsec SA Public Key Sub-TLV

where:

Diffie-Hellman Group Num - is 2 octets long. It identifies the Diffie-Hellman group in the Key Exchange Data was computed. Diffie-Hellman group numbers are discussed in IKEv2 [RFC7296] Appendix B and [RFC5114].

The Key Exchange data - is constructed by copying one's Diffie-Hellman public value into the "Key Exchange Data" portion of the payload. The length of the Diffie-Hellman public value is described for MOPD groups in [RFC7296] and for ECP groups in [RFC5903].

Duration - is a 4-octet value specifying the life span of the Diffie-Hellman key usage. The units of the life span depends on the Diffie-Hellman group.

8.4. IPsec SA Proposal Sub-TLV

The IPsec SA Proposal Sub-TLV is to indicate the number of Transform Sub-TLVs.

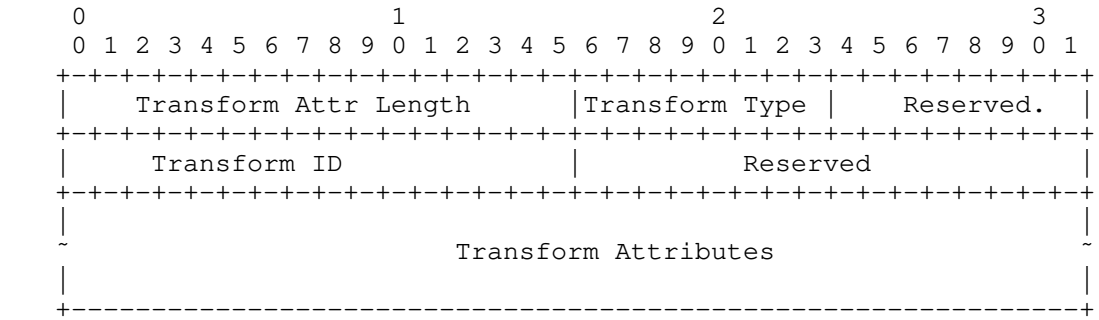


Figure 17: IPsec SA Proposal Sub-TLV

where:

Transform Attr Length is a 2 octet field indicating the length of the Transform Attributes field.

Transform Type is a 1 octet field. The Transform Type values are taken from is from Section 3.3.2 of [RFC7296] and [IKEV2IANA]. Only the values ENCR, INTEG, and ESN are allowed.

Reserved - is a 1 octet field reserved for future use. The Reserved os ignored upon receipt and set to zero upon transmission.

Transform ID - is a 2 octet identifer for the transform described by the transform attributes.

Transform Attributes Sub-SubTLV are taken from the section 3.3.5 of RFC7296.

8.5. Simplified IPsec SA sub-TLV

For a simple SD-WAN network with edge nodes supporting only a few pre-defined encryption algorithms, a simple IPsec sub-TLV can be used to encode the pre-defined algorithms, as below:

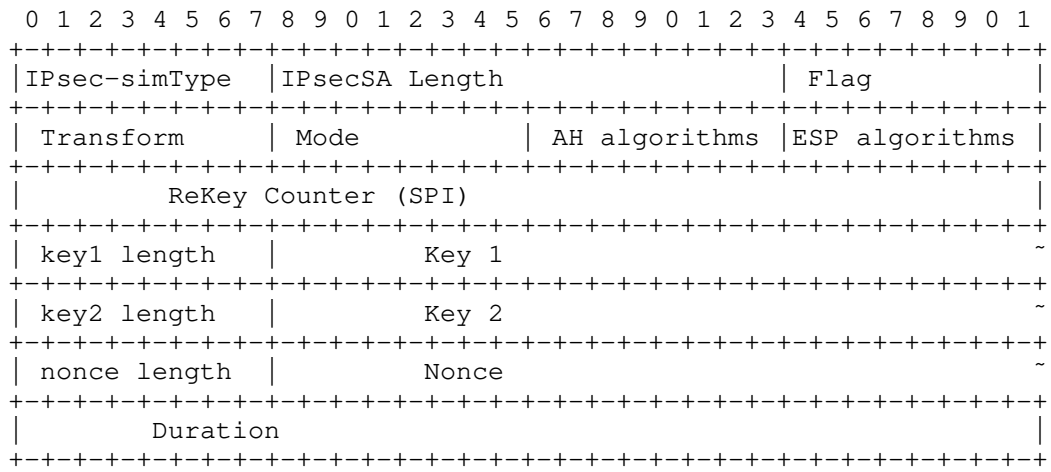


Figure 18: Siplified IPsec SA Sub-TLV

Where:

IPsec-SimType=70: indicate the simplified IPsec SA attributes.

IPsec-SA subTLV Length (2 Byte): variable (25 or longer).

Flags: 1 octet of flags. None are defined at this stage. Flags SHOULD be set to zero on transmission and MUST be ignored on receipt.

Transform (1 Byte):

* Transform = 1 means AH,
* Transform = 2 means ESP, or
* Transform = 3 means AH+ESP.

All other transform values are outside the scope of this document.

IPsec Mode (1 byte):

* Mode = 1 indicates that the Tunnel Mode is used.

* Mode = 2 indicates that the Transport mode is used.

All mode values besides 1 and 2 are outside the scope of this document.

AH algorithms (1 byte): AH authentication algorithms supported. The values are specified by [IANA-AH]. Each SD-WAN edge node can support multiple authentication algorithms; send to its peers to negotiate the strongest one.

ESP algorithms (1 byte): the ESP algorithms supported. Its values are specified by [IANA-ESP]. One SD-WAN edge node can support multiple ESP algorithms and send them to its peers to negotiate the strongest one. The default algorithm is AES-256.

When a node supports multiple authentication algorithms, the initial UPDATE needs to include the "Transform Sub-TLV"

Rekey Counter (Security Parameter Index)): 4 octet which indicates the count for rekeying.

key1 length: is a 1 octet value indicating the IPsec public key 1 length

Public Key 1: IPsec public key 1

key2 length: 1 octet value indicating the IPsec public key 2 length

Public Key 2: IPsec public key 2

none-length: 1 octet value indicating the Nonce key length

Nonce: IPsec Nonce

Duration: is a 4 octet value specifying the security association (SA) life span.

The units on duration are specified by the deployment of the security association. The operators managing these security association must have common units for Security Asssociation duration.

9. Error handling

The Error handling for SD-WAN VPN support has two components: error handling for Tunnel Encapsulation signaling (Encaps-EC and TEA) and the SD-WAN NLRI.

9.1. Error handling for the Tunnel Encapsulation Signaling

The error handling for the tunnel encapsulation signaling (Encaps-EC and TEA) adheres to the error handling and validation specified by [RFC9012].

The Tunnel encapsulation signaled with the client routes indicates the Egress endpoint via Next Hop in the Encaps-EC or the TEA SubTLV for Tunnel Egress Endpoints. As indicated in sections 5.1 and 5.2, the SD-WAN Hybrid tunnel follows the validation section 6, 8, and 13 from [RFC9012].

The SD-WAN client routes associate the same NLRIs that [RFC9012] associates with the Encaps-EC and the TEA using the validation specified in [RFC9012] in sections 6, 8, and 13. When the SD-WAN Hybrid Tunnel is associated with the SD-WAN NLRI, and all RFC9012 validation rules in section 6, 8, and 13 are extended to apply to the SD-WAN NLRI.

[RFC9012] contains the necessary detail to specify validation for the new SubTLVs present for the SD-WAN Tunnel type. However, to aid users of this document the following recap of validation of [RFC9012] is provided below.

The validation from section 13 of [RFC9012] includes:

* Invalid tunnel type must be treat if the TLV was not present.

* A malformed subTLVs must be treated as an unrecognized subTLV except for Tunnel Egress Endpoint. If Tunnel Egress Endpoint is malformed, the entire TLV must be ignored.

* Multiple incidents of Tunnel Egress Endpoint, Encapsulation, DS, UDP Destination Port, Embedded Label Handling, MPLS Label Stack, Prefixes-SID cause the first incident of these subTLVs to be utilized. Subsequent TLVs after the first one per type are ignored (per RFC9012), but propogated.

* If a subTLV is meaningless for a tunnel type, the subTLV is ignored, but the subTLV is not considered malformed or removed from the Tunnel Attribute propagated with the NLRI.

For SD-WAN client routes with a TEA with a SD-WAN Hybrid Tunnel type, the Extended Port subTLV and the IPSec SubTLVs (IPsec SA-ID, IPSec nonce, IPSec Public Key, IPsec Proposal, and Simplified IPsec SA) are meaningful, but may be rarely sent.

For SD-WAN NLRI underlay routes, the the Extended Port subTLV and the IPSec SubTLVs (IPsec SA-ID, IPSec nonce, IPSec Public Key, IPsec Proposal, and Simplified IPsec SA) are valid and meaningful. Incorrect fields within any of these 5 TLVs or subSubTLVs within the TLVs should cause the subTLV to be treated as malformed SubTLV. Per [RFC9012], a malformed subTLV is treated as an unrecognized subTLV. Multiple copies of each SubTLV may be included in a single TLV.

9.2. Error Handling for NLRI

The SD-WAN NLRI [AFI 1/SAFI = 74] utilizes a route type field to describe the format of the NLRI. This specification only allows an NLRI with a type value of 1. An NLRI with a type of field of another value is ignored and not processed. The implementation MAY log an error upon a reception of an type value outside of Route Type 1. Error handling for the SD-WAN NLRI also adheres to the BGP Update error handling specified in [RFC7606].

Section 6.1 specifies that Route Type 1 has a tuple of (Port-Local-ID, SD-WAN-Color, SD-WAN-Node-ID). Port-Local-ID may be zero if the NLRI applies to multiple ports. The BGP Peer receiving the NLRI must have pre-configured inbound filters to set the preference for the SD-WAN NLRI tuple.

Since a Port-Local-ID value of zero indicates the NLRI applies to multiple ports, it is possible to have the following NLRI within a packet (or received in multiple packets):

Port-Local-ID (0), SD-WAN-Color (10), SD-WAN-Node-ID (2.2.2.2),

Port-Local-ID (0), SD-WAN-Color (20), SD-WAN-Node-ID (2.2.2.2), and

Port-Local-ID (0), SD-WAN-Color (30), SD-WAN-Node-ID (2.2.2.2).

These NLRI may simply indicate that there are three groups of tunnels for SD-WAN-Node-ID (2.2.2.2) assigned three colors. For example, these tunnels could represent three types of gold, silver and bronze network service.

The local policy configuration in the BGP peer receiving this NLRI must determine the validity of the route based on policy. Local configuration and policy must be carefully constrain the SD-WAN-NLRI, tunnels, and IPsec security associations in to create a "walled garden".

In the future, other proposals for a SD-WAN NLRI may specify a different route type. Those proposals must specify the following:

validation for new Route Type in the SD-WAN-NLRI, and

how the new Route Type interacts with the Route Type 1.

10. Manageability Considerations

Unlike MPLS VPN whose PE nodes are all controlled by the network operators, SD-WAN edge nodes can be installed anywhere, in shopping malls, in 3rd party Cloud DCs, etc.

It is very important to ensure that client routes advertisement from an SD-WAN edge node are legitimate. The RR needs to ensure the SD-WAN Hybrid Tunnels and routes run over the appropriate Security associations.

10.1. Detecting Misaligned Tunnels

It is critical that the Hybrid SD-WAN Tunnel have correctly forward traffic based on the local policy on the client routes, the tunnel egress and tunnel ingress, and the security association. The RR reflector and the BGP peer must check that the client routes, tunnel egress, tunnel ingress, and security associations align with expected values for a tunnel.

10.2. IPsec Attributes Mismatch

Each BGP peer (e.g. a C-PE) advertises a SD-WAN SAFI Underlay NLRI to the other BGP peers via a BGP Route Reflector to establish pairwise IPsec Security Associations (SA) between itself and other remote BGP Peers. During the SD-WAN SAFI NLRI advertisement, the BGP Peer originating may pass information about security asssociation in one of three forms:

* an identifier for a pre-configured and established IPsec Security Association,

* a simplified set of security parameters for setting up a IPsec Security association (Transform, IPsec Mode, AH and ESP Algorithms, rekey counter, 2 public keys, nonce, and duration of security association), or

* a flexible set of security parameters where Nonce, Public Key, and SA Proposal are uniquely specified.

For existing IPsec Security associations, the receiving BGP peer can simply utilize one of these existing security associations to pass data. If multiple IPsec associations are pre-configured, the local policy on the SD-WAN Edge Node may may help select which security association is chosen for the SD-WAN Hybrid Tunnel.

If the receiving and originating BGP peer engage in a set-up for the IPsec security associations for the link within the SD-WAN Hybrid tunnel, IPsec mechansims require that there are matching IPsec transforms. Without common IPsec transforms, the IPsec set-up process cannot operate.

10.2.1. SD-WAN Hybrid Tunnel Mechanisms for Passing IPSec Security Association Info

The TEA passes in the Tunnel TLV for the SD-WAN Hybrid Tunnel these three sets of information in the following subTLVs:

IPsec-SA-ID: passes the previous configured (pre-configured or generated) IPsec SA identifiers.

Simplified IPsec SA SubTLV: specifies a simplified set of information upon which to set-up the IPsec security associations for the tunnel.

A sequence of the following SubTLVs: IPsec SA Rekey Counter SubTLV, IPsec Public Key SubTLV, and a IPSec Proposal SubTLV. Configuration on the local node uses this information and any information in the underlay to create security associations.

The BGP Peer’s need to send the IPSec SA attributes received on the SD-WAN NLRI in the TEA between the local and remote WAN ports. If there is a match on the SA Attributes between the two ports, the IPSec Tunnel is established. If there is a mismtach on the SA Attributes, no IPsec Tunnel is established.

The C-PE devices do not try to negotiate the base IPSec-SA parameters between the local and the remote ports in the case of simple IPSec SA exchange or the Transform sets between local and remote ports. If there is a mismatch in the IPsec SA, then no IPsec Tunnel is created. If there is a mismatch on the Transform sets in the case of full-set of IPSec SA Sub-TLVs, no tunnel is created.

10.2.2. Example creation of IPsec Security Association over SD-WAN Hybrid tunnel

This section provides one example of how IPsec Security associationes are created over the SD-WAN Hybrid tunnel. Figure 1 in Section 3 shows an establish an IPsec Tunnel being created between C-PE1 and C-PE2 WAN Ports A2 and B2 (A2: 192.10.0.10 - B2:192.0.0.1).

To create this tunnel C-PE1 needs to advertise the following attributes for establishing the IPsec SA:

* NextHop: 192.10.0.10
* SD-WAN Node ID: 1.1.1.1
* SD-WAN-Site-ID: 15.0.0.2
* Tunnel Encap Attr (Type=SD-WAN) -
  - Extended Port Attribute Sub-TLV containing
    - Transport SubSubTLV - with information on ISP3.
  - IPSec information for detailed information about the ISP3
  - IPsec SA Rekey Counter Sub-TLV,
  - IPsec SA Public Key Sub-TLV,
  - Proposal Sub-TLV (type = ENCR, transform ID = 1)
    - type: ENCR
    - Transform ID: 1
    - Tranform attributes = trans 1 [from RFC7296]

C-PE2 needs to advertise the following attributes for establishing IPsec SA:

Next Hop: 192.0.0.1

SD-WAN Node ID: 2.2.2.2

SD-WAN-Site-ID: 15.0.0.2

Tunnel Encap Attr (Type=SD-WAN)
* Extended Port Attribute SubTLV
  - Transport SubSubTLV - with information on ISP3.
* IPsec SA Rekey Counter Sub-TLV,
* IPsec SA Public Key Sub-TLV,
* IPSec Proposal Sub-TLV with
  - transform type: ENCR
  - Transform ID = 1
  - Transform attributes = trans 2

As there is no matching transform between the WAN ports A2 and B2 in C-PE1 and C-PE2, respectively, no IPsec Tunnel will be established.

11. Security Considerations

The document describes the encoding for SD-WAN edge nodes to advertise its properties to their peers to its RR, which propagates to the intended peers via untrusted networks.

The secure propagation is achieved by secure channels, such as TLS, SSL, or IPsec, between the SD-WAN edge nodes and the local controller RR.

SD-WAN edge nodes might not have secure channels with the RR. In this case, BGP connection has be established over IPsec or TLS.

This document describes the encoding for SD-WAN edge nodes to advertise their properties to their peers via their respective Route Reflector (RR), which then propagates the information to the intended peers. SD-WAN edge nodes to advertise its properties to their peers via a secure connection (TLS, SSL, or IPsec) to the RR which propagates to the intended peers over a secure connection (TLS, SSL, or IPsec).

In a walled garden SD-WAN deployment where all SD-WAN edges and the central controller are under one administrative control and the network operates within a closed environment, the threat model is

primarily on internal threats, misconfigurations, and localized physical risks. Unauthorized physical access to SD-WAN edge devices in remote locations is a concern. Such access might allow attackers to compromise the edge devices and potentially manipulate the advertised Client prefixes with VPN IDs (or Route Targets) that do not belong to them. This can lead to unauthorized data interception and traffic redirection.

Therefore, it is necessary to ensure physical security controls are in place at remote locations, including locks, surveillance, and access controls. Additionally, the RR needs to verify the BGP advertisements from each SD-WAN edge to ensure that their advertised VPN IDs (or Route Targets) are truly theirs. This verification helps prevent unauthorized advertisement of prefixes and ensures the integrity of the routing information within the SD-WAN environment. Ensuring secure communication between SD-WAN edge nodes and the central controller within a walled garden deployment is crucial. It is essential to utilize secure communication channels such as TLS or IPsec for all communications between edge nodes and the controller.

12. IANA Considerations

12.1. Hybrid (SD-WAN) Overlay SAFI

IANA has assigned SAFI = 74 as the Hybrid (SD-WAN) SAFI.

12.2. Tunnel Encapsulation Attribute Type

IANA is requested to assign a type from the BGP Tunnel Encapsulation Attribute Tunnel Types as follows [RFC8126]:

Value	Description	Reference
25	SD-WAN-Hybrid	(this document)

12.3. Tunnel Encapsulation Attribute Sub-TLV Types

IANA is requested to assign the following sub-Types in the BGP Tunnel Encapsulation Attribute Sub-TLVs registry:

Value	Type Description	Reference	Section
64	IPSEC-SA-ID Sub-TLV	This document	8.1
65	Extended Port Property Sub-TLV	This document	7.0
66	Underlay Transport Sub-TLV	This document	7.1
67	IPsec SA Rekey Counter Sub-TLV	This document	8.2
68	IPsec Public Key Sub-TLV	This document	8.3
69	IPsec SA Proposal Sub-TLV	This document	8.4
70	Simplified IPsec SA sub-TLV	This document	8.5

13. References

13.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[RFC4271] Rekhter, Y., Ed., Li, T., Ed., and S. Hares, Ed., "A Border Gateway Protocol 4 (BGP-4)", RFC 4271, DOI 10.17487/RFC4271, January 2006, <https://www.rfc-editor.org/info/rfc4271>.

[RFC4360] Sangli, S., Tappan, D., and Y. Rekhter, "BGP Extended Communities Attribute", RFC 4360, DOI 10.17487/RFC4360, February 2006, <https://www.rfc-editor.org/info/rfc4360>.

[RFC4364] Rosen, E. and Y. Rekhter, "BGP/MPLS IP Virtual Private Networks (VPNs)", RFC 4364, DOI 10.17487/RFC4364, February 2006, <https://www.rfc-editor.org/info/rfc4364>.

[RFC4456] Bates, T., Chen, E., and R. Chandra, "BGP Route Reflection: An Alternative to Full Mesh Internal BGP (IBGP)", RFC 4456, DOI 10.17487/RFC4456, April 2006, <https://www.rfc-editor.org/info/rfc4456>.

[RFC4760] Bates, T., Chandra, R., Katz, D., and Y. Rekhter, "Multiprotocol Extensions for BGP-4", RFC 4760, DOI 10.17487/RFC4760, January 2007, <https://www.rfc-editor.org/info/rfc4760>.

[RFC7296] Kaufman, C., Hoffman, P., Nir, Y., Eronen, P., and T. Kivinen, "Internet Key Exchange Protocol Version 2 (IKEv2)", STD 79, RFC 7296, DOI 10.17487/RFC7296, October 2014, <https://www.rfc-editor.org/info/rfc7296>.

[RFC7606] Chen, E., Ed., Scudder, J., Ed., Mohapatra, P., and K. Patel, "Revised Error Handling for BGP UPDATE Messages", RFC 7606, DOI 10.17487/RFC7606, August 2015, <https://www.rfc-editor.org/info/rfc7606>.

[RFC8126] Cotton, M., Leiba, B., and T. Narten, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 8126, DOI 10.17487/RFC8126, June 2017, <https://www.rfc-editor.org/info/rfc8126>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

[RFC8277] Rosen, E., "Using BGP to Bind MPLS Labels to Address Prefixes", RFC 8277, DOI 10.17487/RFC8277, October 2017, <https://www.rfc-editor.org/info/rfc8277>.

[RFC8489] Petit-Huguenin, M., Salgueiro, G., Rosenberg, J., Wing, D., Mahy, R., and P. Matthews, "Session Traversal Utilities for NAT (STUN)", RFC 8489, DOI 10.17487/RFC8489, February 2020, <https://www.rfc-editor.org/info/rfc8489>.

[RFC9012] Patel, K., Van de Velde, G., Sangli, S., and J. Scudder, "The BGP Tunnel Encapsulation Attribute", RFC 9012, DOI 10.17487/RFC9012, April 2021, <https://www.rfc-editor.org/info/rfc9012>.

13.2. Informative References

[IANA-AH] IANA, "IANA-AH", <https://www.iana.org/assignments/isakmp-registry/isakmp-registry.xhtml#isakmp-registry-9>.

[IANA-ESP] IANA, "IANA-ESP", <https://www.iana.org/assignments/isakmp-registry/isakmp-registry.xhtml#isakmp-registry-9>.

[Net2Cloud] L. Dunbar, A Malis, C. Jacquenet, M. Toy and K. Majumdar, "Dynamic Networks to Hybrid Cloud DCs: Problem Statement and Mitigation Practice", September 2023, <https://datatracker.ietf.org/doc/draft-ietf-rtgwg-net2cloud-problem-statement/>.

[RFC5114] Lepinski, M. and S. Kent, "Additional Diffie-Hellman Groups for Use with IETF Standards", RFC 5114, DOI 10.17487/RFC5114, January 2008, <https://www.rfc-editor.org/info/rfc5114>.

[RFC5903] Fu, D. and J. Solinas, "Elliptic Curve Groups modulo a Prime (ECP Groups) for IKE and IKEv2", RFC 5903, DOI 10.17487/RFC5903, June 2010, <https://www.rfc-editor.org/info/rfc5903>.

[RFC9016] Varga, B., Farkas, J., Cummings, R., Jiang, Y., and D. Fedyk, "Flow and Service Information Model for Deterministic Networking (DetNet)", RFC 9016, DOI 10.17487/RFC9016, March 2021, <https://www.rfc-editor.org/info/rfc9016>.

Appendix A. Acknowledgments

Acknowledgements to Wang Haibo, Shunwan Zhuang, Hao Weiguo, and ShengCheng for implementation contribution. Many thanks to Yoav Nir, Graham Bartlett, Jim Guichard, John Scudder, and Donald Eastlake for their review and suggestions.

Contributors

Below is a list of other contributing authors:

* Gyan Mishra,
* Shunwan Zhuang,
* Sheng Cheng, and
* Donald Eastlake.

Authors' Addresses

Linda Dunbar
Futurewei
Dallas, TX,
United States of America
Email: ldunbar@futurewei.com

Kausik Majumdar
Microsoft Azure
California,
United States of America
Email: kmajumdar@microsoft.com

Susan Hares
Hickory Hill Consulting
United States of America
Email: shares@ndzh.com

Robert Raszuk
Arrcus
United States of America
Email: robert@raszuk.net

Venkit Kasiviswanathan
Arista
United States of America
Email: venkit@arista.com

BESS WG
Internet-Draft
Intended status: Standards Track
Expires: 7 January 2025

S. Dikshit
T R. Gadekal
Aruba, HPE

6 July 2024

Secure IP Binding Synchronization via BGP EVPN
draft-saumthimma-evpn-ip-binding-sync-04

Abstract

The distribution of clients of L2 domain across extended, networks leveraging overlay fabric, needs to deal with synchronizing the Client Binding Database. The 'Client IP Binding' indicates the IP, MAC and VLAN details of the clients that are learnt by security protocols. Since learning 'Client IP Binding database' is last mile solution, this information stays local to the end point switch, to which clients are connected. When networks are extended across geographies, that is, both layer2 and layer3, the 'Client IP Binding Database' in end point of switches of remote fabrics should be in sync. This literature intends to align the synchronization of 'Client IP Binding Database" through an extension to BGP control plane constructs and as BGP is a typical control plane protocol configured to communicate across network boundries.

Status of This Memo

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

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at https://datatracker.ietf.org/drafts/current/.

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."

This Internet-Draft will expire on 7 January 2025.

Copyright Notice

Table of Contents

1. Important Terms

BGP: Border Gateway Protocol

VTEP: Virtual Tunnel End Point or Vxlan Tunnel End Point

RD: Route Distinguisher

RT: Route Target

NLRI: Network Layer Reachability Information

EVPN: Etherenet Virtual Private Network

2. Introduction

The distribution of clients of L2 domain across extended, networks leveraging overlay fabric, needs to deal with synchronizing the Client Binding Database. The 'Client IP Binding' indicates the IP, MAC and VLAN details of the clients that are learnt by security protocols. Since learning 'Client IP Binding database' is last mile solution, this information stays local to the end point switch, to which clients are connected. When networks are extended across geographies, that is, both layer2 and layer3, the 'Client IP Binding Database' in end point of switches of remote fabrics should be in sync. This literature intends to align the synchronization of 'Client IP Binding Database" through an extension to BGP control plane constructs and as BGP is a typical control plane protocol configured to communicate across network boundries.

3. Requirements Language

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 [RFC2119].

When used in lowercase, these words convey their typical use in common language, and they are not to be interpreted as described in [RFC2119].

4. Problem Description

Access Switches, build trusted database of L3 clients by snooping into DHCP/ND packets in the client VLAN. This database can be leveraged by security and analytical features. Security features help in protecting the network from unauthorized/untrusted users while analytical features/application gauge the nature of access with the telemetry information for the designated hosts. This information help keeping the network health in a secure and informed way

* Prevent ARP cache depletion attacks

* Allow traffic only from known clients on access ports

* Denial of service and Man in the middle attacks

In Campus networks of Colleges, Branch office, Headquarters and DataCenter DC networks, it is a very common deployment, for networks, to extend Layer-2 across sites and geographies over WAN, Wide Area Network, via well defined overlay fabric solution like EVPN and constructs like Vxlan, GPE, GUE, NVGRE, with Vxlan being the most deployed. The solution for Campus and DC can be combined for enterprise solutions as well.

The campus networks are extended between headquarter and branch offices where as DCs are extended for load sharing, resiliency, hierarchical availability of data across geographies and region. In such deployments, the client database ends up being local to the first-hop access switch or gateway, the client is connected to. Client connected to the first hop access switch or gateway. The other set of access switches within or remotely placed in the extended fabric, treat the client as untruste or unauthorized on the client VLAN. This would result in switches to deny services to legitimate clients.

The following diagram shows the topology of disparate fabrics.

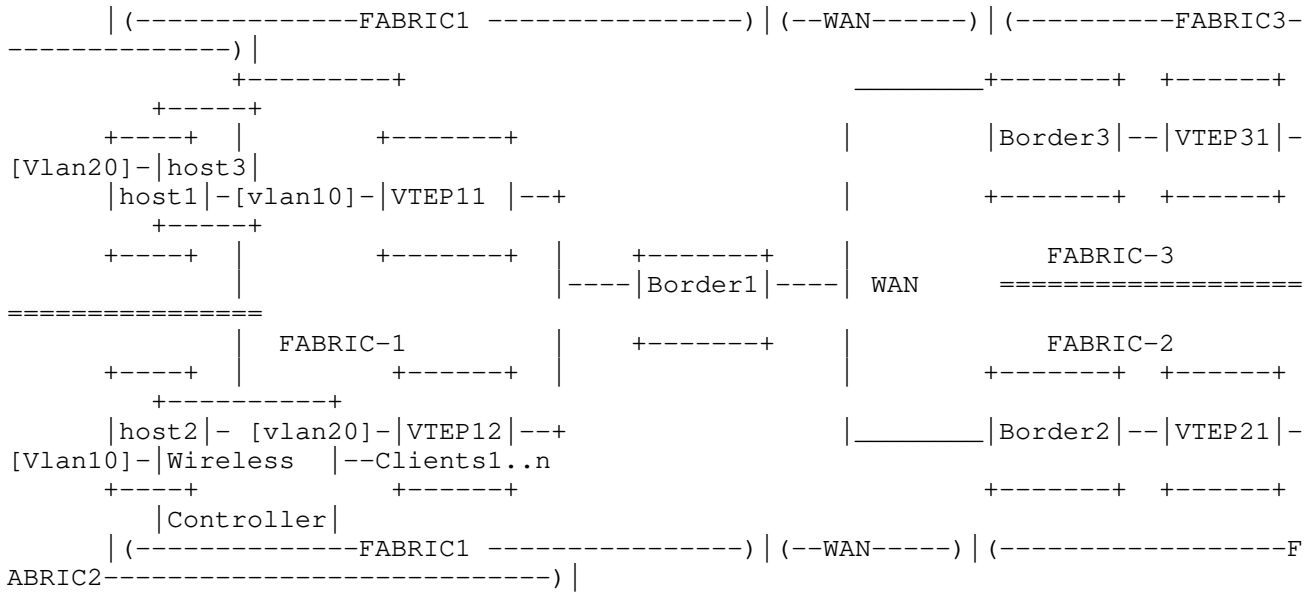


Figure 1: Figure 2: Multifabric Overlay Topology

In the above mentioned topology diagram,

* the Client IP Binding Database is local to client connected to the switches within the fabric. Without explicit solution in place, these details are not known to other switches within and outside the fabric, unless and until explicitly communicated. For example, the Access Vtep11 and Access Vtep12, though in the same fabric, can snoop in to DHCP packets originating or destined from the locally attached hosts, that is, Host1 and Host2 respectively, but not into the other Access hosts. These packets are typically unicasted to and from the DHCP server located in a centralized location. The packets are not leaked to remote fabrics as well.

The DHCP packets generated from Host1 or Host2 are typically not leaked to remote fabrics, Fabric2 or Fabric3 in the above example. Hence this information about host1 and host2 cannot be snooped in by remote Access devices Vtep12 and Vtep13 in the remote fabrics.

* The typical example shown in the above diagram, elaborates on the problem

* There are no known standard techniques to achieve this synchronization between switches, within or across extended network, via an overlay network.

* Security policies that needs knowledge about the connected clients in a VLAN across fabrics, will not work as expected in this model.

For example, to protect from neighbor cache depletion attacks, a switch can be configured to perform resolution only for the known clients in the network. This is not possible if Client IP Binding Database is not synced. NOTE, the above diagram, leverages BGP EVPN provisioned overlays over Vxlan fabric as the network extension instrument. Though the example can be generalized for any BGP provisioned overlay network like VPLS, VPWS, L3VPN etc.

One possible option is to build the client database by sniffing into data plane packets. This class of solution is ingrained with problems.

4.1. Broadcast Over WAN

This solution requires learnings over broadcast packets like ND and ARP. The bridge-domain extension over fabrics, will require the broadcasts to travel over the WAN pipe, and needs to be refreshed periodically as and when there is a request for host discovery. With reference to the above diagram, Gratuitous ARP(GARP), generated by Host2, will be required to be flooded to remote fabrics (fabric 2 and 3), in a typical data plane learning. But with EVPN control plane and arp-suppression techniques, there is minimal leak of arp broadcasts in the EVPN fabric, and thus the WAN. Hosts attached to access switches may GARP frequently (too many and too often), thus aggravating the broadcast leaks over WAN. As mentioned in the topology description, DCHP Snooping will not help here, as DHCP packets will not make it to remote fabrics. Even in DCHP Relay based deployments, the Relay configuration is local to a fabric and cannot be leaked to remote fabrics.

4.2. Same Subnet across fabrics over different Vlans

There is a possibility that same subnet allocation for ip address happens across disparate Vlans within or across the fabrics. Vlan 10 in fabric1 and Vlan20 in fabric 2 are mapped to same subnet gateway, lets say, 10.0.0.0/24. The IP binding database learnt via DHCP or ND snooping (hosted on access Vtep11 and Vtep13) needs to be synchronized as the broadcast domain is extended over a different VNI, lets say, 100 and 200 (mapped to vlan 10 and 20 respectively). Thus the problem at hand is that the allocations (IP bindings) are to be synchronized for same or different subnet, but across the Vlans in disparate fabrics. This is not possible with following deployment

* Native Vlan extension and vlan translation based fabric extensions

* Static Vxlan based network extensions.

Note that, In BGP Control plane, Route Targets help go around this anomaly with dataplane learnings.

4.3. Unwarranted and Insecure Information leak

With dataplane solution (via GARP or ND), the flooding over WAN to all remote fabrics will lead to unwarranted learning in fabrics over which there is no operator control.

* Thus potentially leaking client subnet specific information to the untargeted fabrics, creating an information leak and security risk.

* The source fabrics do not have control over the flooding in WAN (as its a flood in Vlan bridge domain).

* Hence there is a necessity to selectively leak or restrict the synchronization between specific fabrics.

In cases where DHCP Relay is configured,

* DHCP exchange happens between the client connected to Access Switch and the server through unicast channel. This is a predominant usecase but inherently prohibits other Access Switches snooping into the client DHCP packets.

* In the reference example, the Access Switches are also overlay tunnel end points, VTEP.

For example, in above diagram, the host1 GARP is flooded over WAN to both fabric2 and fabric3, even though the intention, is to sync the information to fabric2 only.

5. Security of Fast Roaming Clients

If a wireless client, lets say, host1, moves in a lift or elevator across floors and hence across gateways, the security policy synchronization is a must between the Vteps sitting behind the wireless controllers. Rather than waiting for client to speak up behind every wireless controller, the first controller and corresponding vtep can publish the security clearance or security risk of the host to all remote vteps.

6. Solution(s)

This document proposes a solution for synchronizing the IP Bindings between the Access Switches by leveraging the BGP EVPN control plane constructs called Route Type 2 NLRIs (EVPN NLRI, MAC Advertisement Route). The proposal defines a new extended community, which indicates the IP Binding synchronization, to be carried, in BGP Update message carrying the Route Type 2 NLRI (Network Layer Reachability Information). As EVPN is the at the core of fabric deployments to support multihoming and multitenancy, this control plane extension alleviates the Synchronization of IPs which is specific to tenants and applicable to all or selective hosts attached to the Access Vteps. The host can be attached to more than one Vtep (indicating multihoming) over a segment (called Ethernet Segment). BGP EVPN is a goto solution for Vxlan fabric extension across the WAN, as its also easy to realize the native transport (underlay) encapsulation as IP based.

For example, in the above diagram Vtep11 (also hosting the IP Binding database), publishes BGP update messages, carrying EVPN Route Type 2 for all the local MAC,IP learnings to the remote BGP peers (within or across the fabrics). These MAC,IP learnings are typically learnt via local dynamic learnings that is, host1 sends a GARP towards Vtep11 and is received on Vtep11 over the client (tenant) facing gateway interface (interface vlan 10). This interface can be a Vlan interface mapped to the subnet, for example, interface vlan 10 on Vtep11. All hosts attached over vlan 10 to Vtep11 are allocated the same subnet IP address and monitored by the IP Binding (or security validation) application hosted on Vtep11. Note that the similar validation is performed on all the access switches in a secure network. The dynamic learnings are validated against the IP Binding database (learnt via typically) as indicated in the EVPN Route Type 2. Note that the process of learning the IP Bindings (via snooping of DHCP or ARP) is outside the purview of this document.

6.1. Client IP Binding Sync Extended Communities

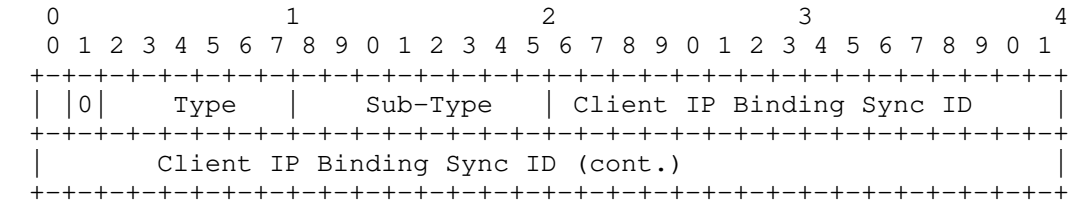
A new extended community Path attribute called Client-IP Binding Sync Extended Communities is defined. This attribute is an optional attribute and also transitive in nature and can be relayed in the control plane path. This attribute, if carried along with BGP update message with MAC, IP bindings (with EVPN NLRI, MAC Advertisement Route), indicates the following to the receiving BGP peer

* MAC, IP data can ALSO be leveraged for Client IP Binding synchronization.

* The data is to be handed over to the Security entity or application for validating the IP allocation

* The handover procedure is implementation specific and outside the purview of this invention.

The following diagram shows the Client IP Binding Sync Extended Communities



Field	Description
Type	1 Octet size, Type field can be newly defined as a proprietary one. Its a transitive Attribute, hence 1 bit in this octet is 0.
Sub-Type	1 Octet size. The value of sub-type will be allocated as per IANA (TBD) For experimental purposes, it can be used as any unallocated value within the free range. This field when set to appropriate value indicates to the receiver about processing the NLRI for IP-Binding synchronization.
Client IP Binding Sync ID	6 Octets size, This field carries the Identifier configured to categorize the IP Binding instance. The field is applicable to all data carried in MAC IP NLRIs in the same update message. The instance is configurable and value is implementation dependent.

Figure 2: Figure 2 Client-IP Binding Sync Extended Communities

the following section gives a detailed flow of the send and receive side processing the new PDU.

6.1.1. Processing of Client IP Binding

The following set of bullets describe the 'Send Side Processing' flow

(1) the Access Vtep hosting the Client IP Binding Database can be triggered to carry the new extended communities, via a knob .

(2) The configuration can be an explicit 'CLIENT IP BINDING SYNC ID' and carried inside the Path attribute which can be configured on Access Vteps. The guidance for configuration of this ID is that, it indicates a Group of Access Switch spread across fabrics, that intend to share the IP Binding data as they might be part of same client or group of clients. All, catering to IP allocation of same tenant or group of tenants over same or disparate vlans and subnets. The configured value is ALSO leveraged to match the received value in the Route Type 2 Update message.

(3) The configuration scope of the 'CLIENT IP BINDING SYNC ID' can be global for all BGP updates or specific to vlans for which MAC,IP is being published in the UPDATE message. This is implementation specific and based on the 'IP BINDING SYNC IDs' scope within the 'Group of Access Switch'. If Vlan based, then 'CLIENT IP BINDING SYNC IDs' can be inherited or derived from 'Route Target' extended communities, configured on the 'Group of Access Switches' .

(4) The Withdraw of routes, Route Type 2, MAY also carry the extended community to indicate to the BGP peers configured on Group of Access Switches to convey the INVALIDITY of the MAC,IP bindings, published earlier.

(5) The Access Vteps (or BGP peer) on the send side MUST NOT insert the new extended communities in the withdraw message, if the corresponding IP Bindings are still valid.

the following bullets describe the 'Receive Side Processing'

(1) The same 'CLIENT IP SYNC ID' should be configured on the receiving Access Switch (BGP peer) to process or absorb the MAC, IP information within the IP Binding Context. For example, It can be a security application, which validates the IP Bindings.

(2) If the receiving BGP peer DOES NOT SUPPORTS the 'Client IP Binding Sync' Extended Communities in the received Route Type 2,it ignores the processing of this extended community while processing other attributes.Thus, no implication on any 'Client IP Binding' application, if hosted on this BGP Peer. It MUST respond to unsupported attribute as defined in [RFC4379].

(3) If the receiving BGP peer SUPPORTs the Client IP Binding Sync Extended Communities in the received Route Type 2, It decodes the Extended Communities and extracts the "Client IP SYNC ID" value. It matches the value with the locally configured one.

* If match is through, then it imports the MAC, IP NLRIs carried in the same update message to the Security application, which validates the MAC,IP against the security policies andi absorbs or drops after comparing it against the local IP Binding database.

* If nomatch then it MUST NOT import the NLRIs into local IP Binding Database.

(4) In case there are other BGP peers to whom the Update message is to be relayed, then the Client IP Binding Sync Extended Communities are also relayed without any updates in the values as its a transitive attribute. For example, Border1 receives an update message with the Client IP Binding Sync Extended Communities from Vtep11. It processes it as mentioned in above bullets and relays it to its eBGP (external BGP peers), Border2 and Border3 in remote fabrics fabric2 and fabric3 respectively as its an optional transitive attribute. Note that the above example quotes eBGP (different Autonomous System) across fabrics, but same logic will apply when Route Reflector reflects routes between two iBGP peers.

The augmented control plane feature, helps all Access swtiches to be synchronized with respect to allowed or validated client credentials, thus preventing rogue traffic or DoS attacks 'originating in' or 'destined to' the local network.

7. Backward Compatibility

Backward Compatibility for non-support nodes is as per the following standards already defined in [RFC7606], that, BGP speaker should discard the unsupported TLV types

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997, <http://www.rfc-editor.org/rfc/rfc2119.txt>.

[RFC4379] Kompella, K., "Detecting Multi-Protocol Label Switched (MPLS) Data Plane Failures", RFC 4379, February 2006, <https://www.rfc-editor.org/rfc/rfc4379.html>.

[RFC7153] Rosen, E., "IANA Registries for BGP Extended Communities", RFC 7153, March 2014, <https://www.rfc-editor.org/rfc/rfc7153.html>.

[RFC7348] Mahalingam, M., "Virtual eXtensible Local Area Network (VXLAN): A Framework for Overlaying Virtualized Layer 2 Networks over Layer 3 Networks", RFC 7348, August 2014, <http://www.rfc-editor.org/rfc/rfc7348.txt>.

[RFC7432] Sajassi, A., "BGP MPLS-Based Ethernet VPN", RFC 7432, February 2015, <http://www.rfc-editor.org/rfc/rfc7432.txt>.

[RFC7606] Chen, E., "Revised Error Handling for BGP UPDATE Messages", RFC 7606, August 2015, <https://www.rfc-editor.org/rfc/rfc7606.html>.

[RFC9014] Rabadan, J., "Interconnect Solution for Ethernet VPN (EVPN) Overlay Networks", RFC 9014, May 2021, <http://www.rfc-editor.org/rfc/rfc9014.txt>.

Saumya Dikshit
Aruba Networks, HPE
Mahadevpura
Bangalore 560 048
Karnataka
India
Email: saumya.dikshit@hpe.com

Gadekal, Thimma Reddy
Aruba Networks, HPE
Mahadevpura
Bangalore 560 048
Karnataka
India
Email: tgadekal@hpe.com

Inter-Domain Routing
Internet-Draft
Intended status: Standards Track
Expires: 1 September 2024

C. Sheng
H. Shi, Ed.
Huawei
L. Dunbar
Futurewei
G. Mishra
Verizon

29 February 2024

Associated Gateway Exchange in Multi-segment SD-WAN
draft-sheng-idr-gw-exchange-in-sd-wan-02

Abstract

The document describes the control plane enhancement for multi-segment SD-WAN to exchange the associated GW information between edges.

Discussion Venues

This note is to be removed before publishing as an RFC.

Discussion of this document takes place on the Inter-Domain Routing Working Group mailing list (idr@ietf.org), which is archived at https://mailarchive.ietf.org/arch/browse/idr/.

Source for this draft and an issue tracker can be found at https://github.com/VMatrix1900/draft-sheng-idr-gw-exchange-in-sd-wan.

Status of This Memo

Table of Contents

1. Introduction

[I-D.draft-dmk-rtgwg-multisegment-sdwan] describes how enterprises leverage Cloud Providersâ\200\231 backbone infrastructure to interconnect their branch offices. As illustrated in Figure 1, CPE-1 and CPE-2 establish connections to their respective Cloud Gateways (GW) in distinct regions. CPE-1 and CPE-2 maintain the pairwise IPsec Security Associations (SAs). The IPsec encrypted traffic from CPE-1 to CPE-2 is encapsulated by the GENEVE header [RFC8926], with the outer destination address being the GW1.

[I-D.draft-dmk-rtgwg-multisegment-sdwan] specifies a set of sub-TLVs to convey information about the GWs associated with the destination branches, such as GW3 for CPE-2, along with additional attributes. To accomplish this, CPE-1 must be aware of the associated GW addresses of their peers. This document proposes a BGP extension, building upon [I-D.draft-ietf-idr-sdwan-edge-discovery], enabling a CPE to advertise its directly connected GW address to other CPEs .

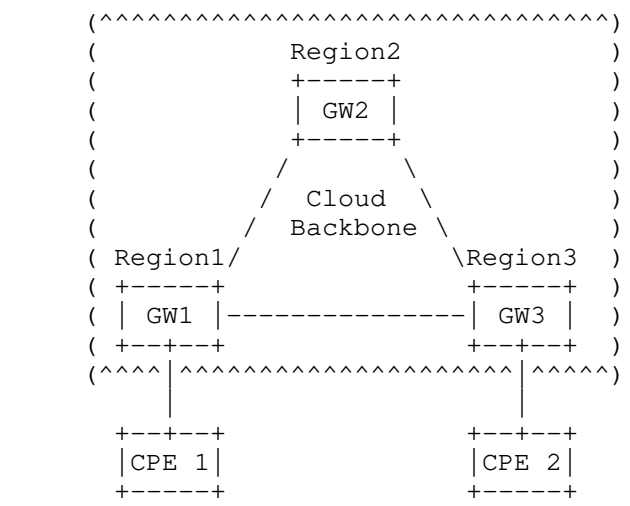


Figure 1: Multi-segment SD-WAN

2. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

The following acronyms and terms are used in this document:

* Cloud DC: Off-Premises Data Center, managed by 3rd party, that hosts applications, services, and workload for different organizations or tenants.

* CPE: Customer (Edge) Premises Equipment.

* OnPrem: On Premises data centers and branch offices.

* RR Route Reflector.

* SD-WAN: Software Defined Wide Area Network. In this document, â\200\234SD-WANâ\200\235 refers to a policy-driven transporting of IP packets over multiple underlay networks for better WAN bandwidth management, visibility, and control.

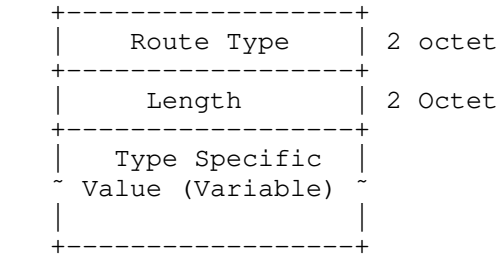
* VPN Virtual Private Network.

3. Extension to SD-WAN Underlay UPDATE for Associated GWs

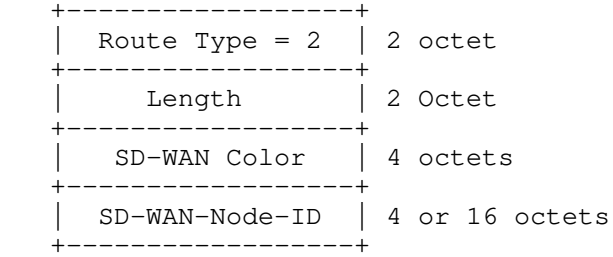
The Client Routes Update is the same as described in Section 5 of [I-D.draft-ietf-idr-sdwan-edge-discovery].

3.1. NLRI SD-WAN SAFI Route Type For GW

Adding a new attribute (Associated-Gateway Sub-TLV) to the SD-WAN-Hybrid Tunnel Encoding which is included in the SD-WAN SAFI (=74) Underlay Tunnel Update:



NLRI Route-Type = 2: For advertising the detailed properties of the transit gateways for the edge. The SD-WAN NLRI Route-Type =2 has the following encoding:



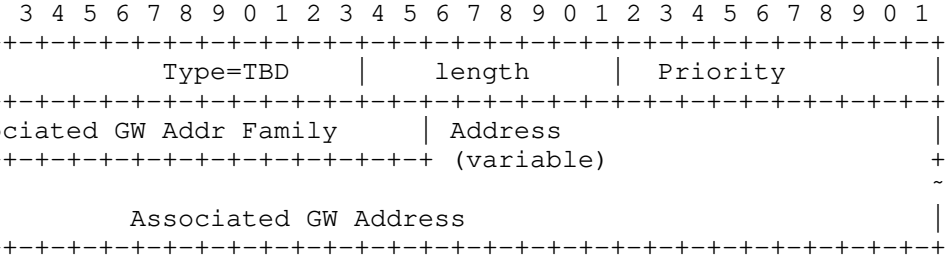
SD-WAN-Color: To represent a group of tunnels that correlate with the Color-Extended-community included in a client route UPDATE. When multiple SD-WAN edges can reach a client route co-located at one site, the SD-WAN- Color can represent a group of tunnels terminated at those SD-WAN edges co-located at the site, which effectively represents the site.

SD-WAN Node ID: The node’s IPv4 or IPv6 address.

3.2. Associated GW Sub-TLV

The Associated GW Sub-TLV, within the SD-WAN-Hybrid Tunnel TLV (code point 25), carries the associated GW address(es) with which the SD-WAN edge is associated.

The following is the structure of the associated GW Sub-TLV:



Priority (1-255) indicate the preference of the GW. The higher the value, the more preference is the GW.

Priority = 0 represents that the connection exists but request Cloud Backbone not to choose the GW if possible.

4. Manageability Considerations

Effective management of SD-WAN edge nodes and the exchange of associated cloud gateway information are critical aspects in ensuring a robust and scalable SD-WAN deployment.

5. Security Considerations

This document does not introduce any new security considerations.

6. IANA Considerations

Need IANA to assign a new Sub-TLV Type under the SD-WAN-Hybrid Tunnel TLV.

- SD-WAN Associated GW Sub-TLV.

7. Normative References

[I-D.draft-dmk-rtgwg-multisegment-sdwan]
Majumdar, K., Dunbar, L., Kasiviswanathan, V., and A. Ramchandra, "Multi-segment SD-WAN via Cloud DCs", Work in Progress, Internet-Draft, draft-dmk-rtgwg-multisegment-sdwan-07, 31 May 2023, <https://datatracker.ietf.org/doc/html/draft-dmk-rtgwg-multisegment-sdwan-07>.

[RFC8926] Gross, J., Ed., Ganga, I., Ed., and T. Sridhar, Ed., "Geneve: Generic Network Virtualization Encapsulation", RFC 8926, DOI 10.17487/RFC8926, November 2020, <https://www.rfc-editor.org/rfc/rfc8926>.

[I-D.draft-ietf-idr-sdwan-edge-discovery]
Dunbar, L., Hares, S., Raszuk, R., Majumdar, K., Mishra, G. S., and V. Kasiviswanathan, "BGP UPDATE for SD-WAN Edge Discovery", Work in Progress, Internet-Draft, draft-ietf-idr-sdwan-edge-discovery-12, 23 June 2023, <https://datatracker.ietf.org/doc/html/draft-ietf-idr-sdwan-edge-discovery-12>.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/rfc/rfc2119>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/rfc/rfc8174>.

Authors' Addresses

Cheng Sheng
Huawei
Beiqing Road
Beijing
Email: shengcheng@huawei.com

Hang Shi (editor)
Huawei
Beiqing Road
Beijing
China
Email: shihang9@huawei.com

Linda Dunbar
Futurewei
Email: linda.dunbar@futurewei.com

Gyan Mishra
Verizon
Email: gyan.s.mishra@verizon.com