IDR Working Group
Internet-Draft
Intended status: Standards Track
Expires: 4 September 2025

S. Hares
Hickory Hill Consulting
D. Eastlake
Independent
J. Dong
Huawei Technologies
C. Yadlapalli
ATT
S. Maduscke
Verizon

3 March 2025

BGP Flow Specification Version 2 - for Basic IP
draft-ietf-idr-fsv2-ip-basic-03

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 (basic, adding more filters, and adding more actions). This document specifies the basic FSv2 NLRI with user ordering of filters added to FSv1 IP Filters and FSv2 actions.

Status of This Memo

This Internet-Draft will expire on 4 September 2025.

Table of Contents

1. Introduction

Version 2 of BGP flow specification was original defined in [I-D.ietf-idr-flowspec-v2] (BGP FSv2).

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).

The BGP 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. The BGP FSv2 draft will remain a WG draft, but the content will be split out into a series of drafts (basic, adding more IP filters, adding more IP actions, and individual functions for TTL, MPLS and SRv6).

This draft (FSv2 Basic) provides the basic FSv2 framework specification for transmitting user-ordered IP filters in the FSV2 NLRI with Extended Ccommunity to specify actions.

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 "ships in the night" based on AFI/SAFI.

The remainder of section 1 provides background on why the FSv2 was necessary to fix problems with FSv1. Section 2 contains a Primer on FSv1 (section 2.1) and FSv2 (section 2.2). 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.

1.1. 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 "event-condition-action" (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.

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 α TLV encoding prevented extension of encodings (FSv1 uses TV (type-value),

* inability to allow user defined order for filtering rules,

* inability to order actions to provide deterministic interactions between actions during incremental deployments, and

* inability to allow users to define order for actions.

Networks currently cope with these issues above by constraining deployments or using topology/deployment specific workaround.

FSv1 is a critical component of deployed applications. Therefore, this specification defines how FSv2 will interact with BGP peers that support either FSv2, FSv1, or FSv2 and FSv1. It is expected that a transition to FSv2 will occur over time as new applications require FSv2 features.

1.2. Definitions and Acronyms

AFI- Address Family Identifier

AS - Autonomous System

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

BGP Commmunity Path Attribute - BGP Path attribute for Community defined by [I-D.hares-idr-bgp-community-attribute]

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

CPA - BGP Community Path Attribute

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)

FS - Flow Specification (either v1 or v2)

FS-EC - FS related Extended Community with FS actions

FSv1-EC - FSv1 related Extended Community with FS Actions supported by FSv1

FSv2-EC - FSv2 related Extended Community with FS Actions supported by FSv2

NETCONF - The Network Configuration Protocol [RFC6241].

RESTCONF - The RESTCONF configurati on Protocol [RFC8040]

RIB - Routing Information Base.

ROA - Route Origin Authentication [RFC9582]

RR - Route Reflector.

SAFI 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 a BGP Community 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.

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) SAFIs

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

Match conditions are ordered by component type in ascending order. If multiple component types filters exist, the ordering within a component type is defined by the component type. The FSv1 component format does not provide enough information to create a unique canonically sorted list for all implementations in all deployments.

The actions standardized for in [RFC8955] and [RFC8956] are:

* accept packet (default),
* traffic flow limitation by bytes (0x6),
* traffic-action (0x7),
* redirect traffic to VPN (0x8),
* mark traffic (0x9), and
* traffic flow rate limiting (12, 0xC)

An SFC action [RFC9015] defines an entry point into a specific SFP (Service Function Path)

While IDR has proposed other Extended Community Actions, no additional actions have completed the standardization process.

Additional proposals for FSv1 Actions exist, but these are still in the standardization process.

2.2. Transition to FSv2

The IDR WG draft [I-D.ietf-idr-flowspec-v2] contains a complete solution for FSv2. However, this complete solution makes implementation of these features a large task so this solution was revised to provide better incremental implementations and deployments.

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=TBD1),

* 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),

One question asked by developers is what AFI/SAFI is required for FSv2 IP Basic compliance. BGP negotiates support for each AFI/SAFI, so FSv2 IP Basic support for non-VPN could be as little as IPv4 FSv2 (AFI/SAFI: 1/TBD1) or IPv6 (AFI/SAFI: 2/TBD1).

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 require these to FSv2 TLV formats.

2.3. 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.

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.

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

1) Rule-0 (zero) is defined to be 0/0 with the “permit-all” 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 rules similar to FSv1 apply (order by component type and then by rules specified by component).

3) Importing FSv1 into a combined FSv2 and FSv2 rule set requires starting FSv2 with Rule 1 and adding FSv1 rules are added after FSv2 rules.

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

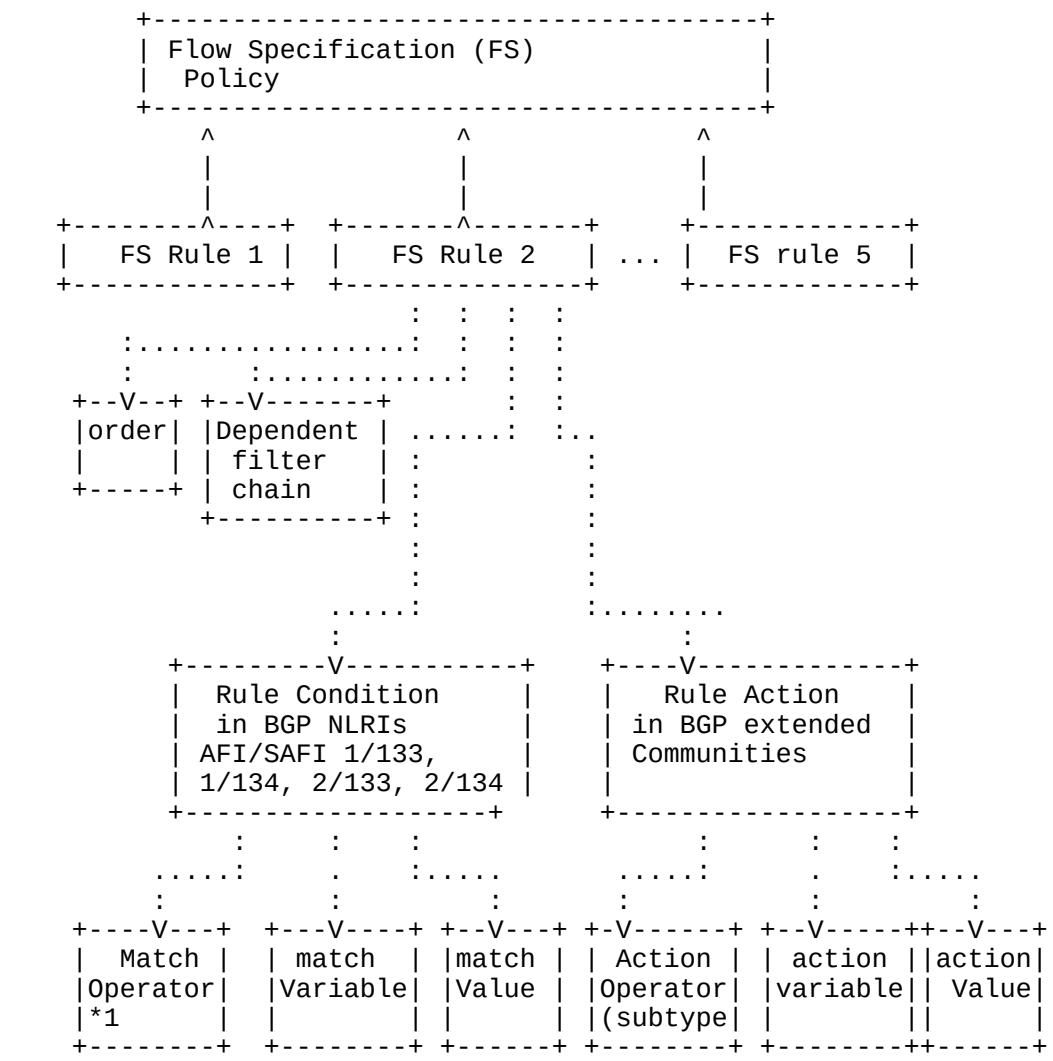
4) An FSv2 peer may receive BGP NLRI routes from a FSv1 peer or a BGP peer that does not support FSv1 or FSv2. The capabilities sent by a BGP peer indicate whether the AFI/SAFI can be received (FSv1 NLRI or FSv2 NLRI).

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

- An action chain of 1-n actions can be associated with a set of filter rules can via Extended Communities or a Community Path attribute with a FSv2 type. Only the Community Path 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.

Figure 1 shows a diagram of the FS (FSv1/FSv2) logical data structures with 5 rules. For easy of reading only Rule 2 shows the order and dependent filter chaing.



*1 match operator may be complex.

Figure 2-1: BGP Flow Specification v1 Policy

3. FSv2 NLRI Formats and Actions

3.1. FSv2 NLRI Format

The BGP FSv2 supports 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 (Flow Spec) and TBD2 (Flow Spec for VPNs) 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:

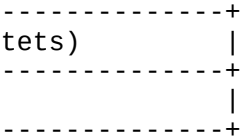


Figure 3-1 - NLRI format

where:

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

* TLV+ - indicates the repetition of the TLV field

Each each TLV has the Format:

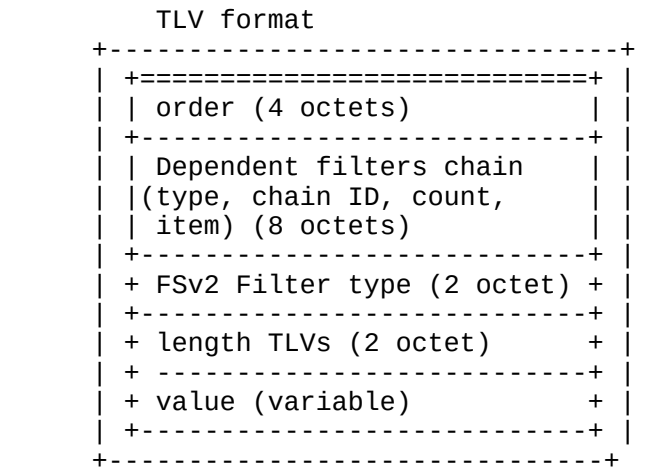


Figure 3-2 - TLV format within FSv2 NLRI

where:

* 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 specification 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, and future specifications will continue to ignore the field if the value is all zeros.

* FSv2 Filter type: contains a type for FSv2 TLV format of the NRLI (2 octets). This type specifies support for a set of components. [Editor’s note: We need to decide between Option 2 or Option 1. Option 2 makes provides a better default ordering for frame/packet filters.

  Option 1 ascending order (draft-ietf-idr-fsv2-ip-basic-02)

0 - reserved,

1 - IP Basic Filter rules

2 - IP Extended Filter Rules

3 - MPLS Filter Rules

4 - L2 traffic rules

5 - SFC Traffic rules

6 - Tunneled traffic

Option 2 ordering by frame/packet (new)

0 - reserved,

50 - MPLS Traffic Rules

100 - L2 traffic rules

150- SFC Traffic rules

200 - Tunneled traffic

256 - IP Basic Filter Rules (bit 1 of high bit)

280 - IP Extended Filter Rules

* 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.

This specification only defines operation of the IP Basic Filter Rules that all FSv2 must support.

3.1.1. Ordering of TLVs within the FSv2 NLRI

For ease of processing, the ordering within the FSV2 NLRI MUST be by order number. Within an order value (e.g. 20), the order MUST group the filters by the same filter type (e.g. 1 for IP Basic). The order within a filter type (e.g. IP Basic Filters) MUST be by the filter type. For IP Basic Filters, this ordering is by Component type.

3.1.2. Partial Deployments

Partial deployments can occur for two reasons:

* Only a portion of the nodes in a network with FSv2 support installing new FSv2 Filter types with new FSv2 components. Other nodes (such as RRs), check the syntax, but do not handle the semantic meaning.

* During upgrades, a portion of the nodes know about a new Filter type with the components, but other nodes do not.

Editor: Are there others?

3.2. FSv2 Basic IP Filters

3.2.1. Operators for comparison

3.2.1.1. Numeric Operator (numeric_op)

This operator is encoded as shown in Figure 3-3.

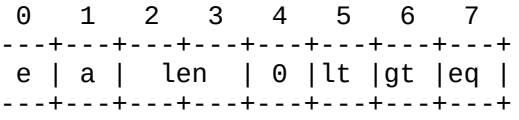


Figure 3-3: Numeric Operator (numeric_op)

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

a (AND bit): If unset, the result of the previous {op, value} pair is logically ORed with the current one. If set, the operation is a logical AND. In the first operator octet of a sequence, it MUST be encoded as unset and MUST be treated as always unset on decoding. The AND operator has higher priority than OR for the purposes of evaluating logical expressions.

len (length): The length of the value field for this operator given

as (1 << len). This encodes 1 (len=00), 2 (len=01), 4 (len=10), and 8 (len=11) octets.

0 MUST be set to 0 on NLRI encoding and MUST be ignored during decoding

lt less-than comparison between data and value

gt: greater-than comparison between data and value

eq: equality between data and value

The bits lt, gt, and eq can be combined to produce common relational operators, such as "less or equal", "greater or equal", and "not equal to", as shown in Table 3-1.

lt	gt	eq	Resulting operation
0	0	0	false (independent of the value)
0	0	1	== (equal)
0	1	0	> (greater than)
0	1	1	<= (greater than or equal)
1	0	0	< (less than)
1	0	1	<= (less than or equal)
1	1	0	!= (not equal value)
1	1	1	true (independent of the value)

Table 3-1: Comparison Operation Combinations

3.2.1.2. Bitmask Operator (bitmask_op)

This operator is encoded as shown in Figure 3-4.

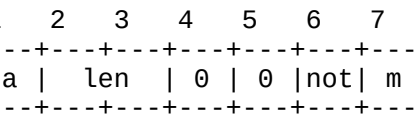


Figure 3-4 Bitmask Operator (bitmask_op)

Where:

e, a, len (end-of-list bit, AND bit, and length field): Most significant nibble; defined in the Numeric Operator format in section 3-x.

not (NOT bit): If set, logical negation of operation.

m (Match bit): If set, this is a bitwise match operation defined as "(data AND value) == value"; if unset, (data AND value) evaluates to TRUE if any of the bits in the value mask are set in the data.

0 (all 0 bits): MUST be set to 0 on NLRI encoding and MUST be ignored during decoding

3.2.2. IP Basic Filters (Filter type=1(0x01))

The format of the IP Basic TLV value field is shown in Figure 3-5. The IP header for the VPN case is specified in section 3.5.

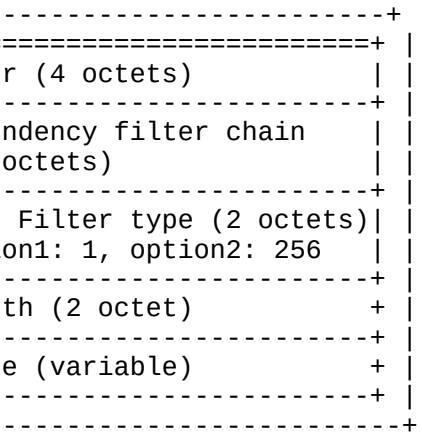


Figure 3-5 NLRI format for FSv2 IP Filter Type

Where:

order - is an 4 octet field with a value 1-N. The value 0 (zero) is invalid, and the TLV should be "treated-as-withdrawl".

dependency filter chain - is an 8 octet field which must be all zero for the IP Basic Filter rules.

length - is a 2 octet field indicating the length of the value field.

value - is a variable field comprised of a sequence of component TLVs:

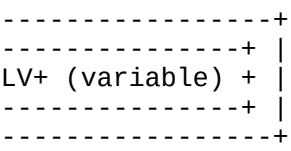


Figure 3-6 Value Field

Where the Component TLVs are:

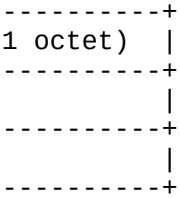


Figure 3-7 IP header Component TLVs

Where:

- Component type: component values are defined in the “Flow Specification Component types” 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). If the length of the component types does match the valid defined length(s) for the component, the component type is ignored and the Filter type TLV is "treated-as-withdrawl".

- 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 3-2 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 3-2 FSv2 IP Basic TLV Components

Sub-TLV	Definition
0 -	Reserved
1 -	IP Destination prefix
2 -	IP Source prefix
3	IPv4 Protocol / IPv6 Upper Layer Protocol
4	Port
5	Destination Port
6	Source Port
7	ICMPv4 type / ICMPv6 type
8	ICMPv4 code / ICPv6 code
9	TCP Flags
10	Packet length
11	DSCP
12	Fragment
13	Flow Label
14-255	Reserved

3.2.3. Ordering within the IP Basic Filter TLVs

The ordering of components within the value field of the IP Basic TLV follows the FSv1 rules. The following is a restatement of FSv1 rules in FSv2 terms.

* 1) order by component types (1-13).

* (2) If the components are the same, then the value fields are compared using mechanisms defined in [RFC8955] and [RFC8956] and MUST be in ascending order. NLRIs having component 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-withdrawl". [RFC7606].

See [RFC8955], [RFC8956], and for details on per component ordering.

3.2.4. FSv2 Components for IP Basic TLVs

3.2.4.1. 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.

3.2.4.2. 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.

3.2.4.3. 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.

3.2.4.4. 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.

3.2.4.5. 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]+

3.2.4.6. 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]+

3.2.4.7. 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]+

3.2.4.8. 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]+

3.2.4.9. 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

3.2.4.10. 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.

3.2.4.11. 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.

3.2.4.12. 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:

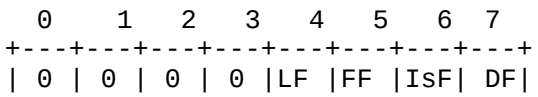


Figure 3-8

Where:

DF (don’t fragment): match If IP header flags bit 1 (DF) is 1.

IsF(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.

3.2.4.13. 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]+

3.3. FSv2 Actions for FSv2 IP Basic

The IP Basic FSv2 allows FSv2 actions to be sent in an Extended Community (FSv2-EC) for IPv4 and IPv6. The Extended Community encodes the Flow Specification actions in the Extended IPv4 Community format [RFC4360] or in the extended IPv6 Community format [RFC5701].

Often a Flow Spec filter match causes only one Flow Spec action. However, it is possible to define multiple Flow Spec Extended Community (FS-EC) Actions for FSv2 (FSv2-EC) or FSv1 (FSv1-EC). These actions cannot be ordered by the user and it is possible for some FS-EC actions to interact.

This section defines the FSv2-EC actions, interactions between actions, and one optional action to signal the ordering of actions. Section 3.3.1 contains the existing FSv2-EC action formats. Section 3.3.2 describes the interaction between FS-EC action, and ways to minimize interactions by use of action categories and default ordering of actions.

Section 3.3 defines an Action Chain Ordering (ACO) FSv2-EC. The ACO FSv2-EC provides two pieces of information about the originators expectation for the correct handling of multiple FSv2-EC per filter match. The first piece of information informs the BGP peer receiving the FSv2-EC whether the originator expects: a) the default ordering of multiple actions specified in FSv2 or b) implementation specific order. The second action field the originator sends in the ACO FSv2-EC is what happens with one of the multiple actions (associated with a single filter match) fails.

Note that FSv2 implementations that only associate 1 FSv2-EC per actions would never need the ACO FSV2-EC.

3.3.1. FSv2 Extended Community Actions

The FSv2 IP Basic uses FSv1 actions and defines for one one additional optional FSv2 specific FS-EC. This one optional action is the Action Chain Ordering (ACO) Extended Community (ACO-EC) which can pass around defaults currently only available by configuration in FSv1.

3.3.1.1. FSv2 Actions in Extended Community for IPv4

The format of the Extended Community for IPv4 defined in [RFC4360] is shown in Figure 3-9 with 2 octet type that is split into a high byte and low byte. The format of the IPv4 Extended Community is shown in Figure 3-10.

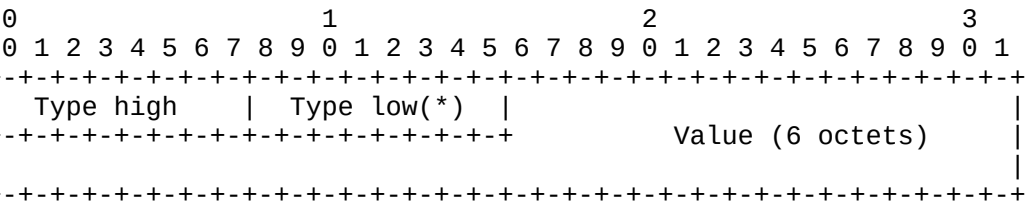


Figure 3-9

Table 3-3
FSv1 Transitive Extended Communities for IP Basic
High-Low byte of Transitive FS-EC

H-L	FSv1 Description	Short-ID	FS document
0x01-0C	Transitive IPv4	RDIPv4	RDIP
0x07-02	FSv1 for an Interface set	TAIS	ifset
0x09-xx	Redirect to Indirection ID	RGID	RGID
0x0b-00	SFC Reserved	SFC-R	RFC9015
0x0b-01	SFVC SFIR POOL Identifier	SFIR-PI	RFC9015
0x0b-02	SFC MPLS label stack Swapping or stacking labels	SFC-MPLS	RFC9015
0x80-06	Traffic rate limit by bytes	TRB	RFC8955
0x80-07	Traffic Action (sample, terminal)	TA	RFC8955
0x80-08	Redirection to VRF (2 AS form)	RDIP	RFC8955
0x80-09	Traffic mark DSCP	TM	RFC8955
0x80-0C	Traffic rate limit by packets	TRP	RFC8955
0x81-08	Redirect to VPN (IPv4 form)	RDIP	RFC8955
0x81-08	Redirect to VPN (4 AS form)	RDIP	RFC8955

References:

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

RDIP: [I-D.ietf-idr-flowspec-redirect-ip]

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

RFC9015: [RFC9015]

RFC8955: [RFC8955]

3.3.1.2. Flow Specification Actions in IPv6 forms

The Transitive IPv6-Address-Specific Extended Community encodes the Flow Specification actions in the Extended Community format specified in [RFC5701] shown in Figure 3-10. Table 3-3 lists the 4 octet format for high-byte and low-byte. Note that there are two allocations for redirect from IPv6.

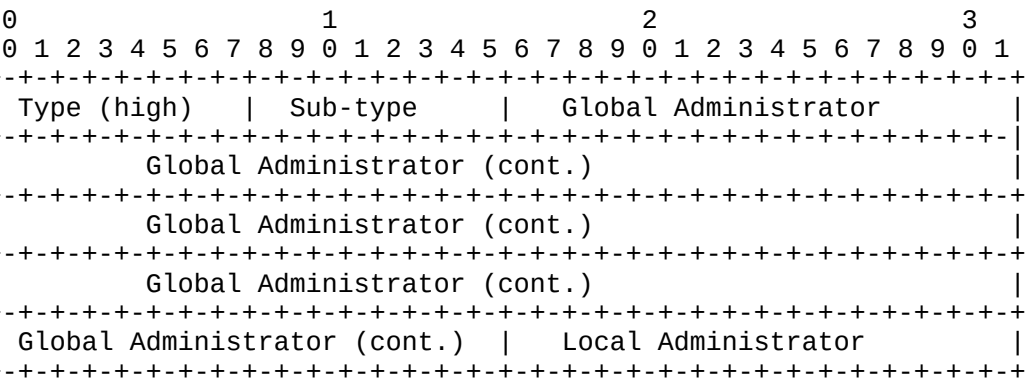


Figure 3-10

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 3-4
Transitive IPv6-Address-Specific Extended Community Types

H-L byte	FSv1 Description	Name	FS document
0x0000	Unassigned		
0x0001	Unassigned		
0x0002	Route Target	RT	No
0x0003	Route origin	ROrg	No (deprecated)
0x0004	OSPFv3 Route Attribute	OSPFv3	No (deprecated)
0x0005	FIT Tail Community	IFITv6	No
0x0006	Link Bandwidth	LBW	No
0x0008	Unassigned		
0x0009	Unassigned		
0x000A	Unassigned		
0x000B	VRF Route Import	VRP-I	No
0x000C	FS Redirect to IPv6	RDIPv6C	Yes: RDIP
0x000D	FS Redirect to IPv6	RDIPv6	Yes: RFC8956
0x000E	Unassigned		
Ox000F	Unassigned		
0x0010	Cisco VPN distinguisher	Cisco-VPN	No
0x0011	UUID-based Route Target	UUID-RT	No
0x0012	Inter-ARea P2MP S-NH	PM2P-NH	No
0x0013	Unassigned		
0x0014	VRF Recursive NH	VRF-RNH	no
0x0015	RT-derived-EC	RT-EC	no

3.3.2. Interactions between Flow Spec Extended Community Actions (FS-EC)

If multiple FS-EC actions are listed on a filter, these can interact within categories or if a previous FS-EC action fails.

3.3.2.1. Interactions Between Multiple Actions by Categories

The FSv1-EC actions and the actions proposed for FSv2-EC Action fall into the following categories:

1. limitation on filters

2. rate limiting (bytes (TRB) and packets (TRP)),

3. Set DSCP value in IP packet

4. redirect to IP paths (to VRF, to VPN, to Indirection-ID),

5. redirect to SFC path (set SFVC SFIR POOL Identifier, redirect to SFC MPLS label),

6. Sample packet (TAIS - sample)

7. Terminate action processing (TAIS terminate)

The order of the categories matters for FS-EC (FSv1 or FSv2) only if more than one action is attached to a filter-match. For many DDoS use cases, the single action per filter match is sufficient to handle the traffic.

The FS-EC Terminate action indicates that Flow Spec should stop processing filters and actions if the Terminate FS-EC action is set. This function allows the policy writer to match and stop processing. This provides a rudimentary control on a sequence of filters in a list with single actions.

Beyond this case, ordering multiple actions by categories (1-7) provides a default deterministic order for FSv2. Within the a category (such as redirect) the specification of the Extended Community, should specify the default interactions if multiple actions within the same category.

How is F2v2 different than FSv1 for multiple actions per filter?

FSv1 implementations that support multiple actions per filter process the actions in a manner configured by the implementation. Network operatores are responsible for configuring the boxs that load the filter/actions into the forwarding path to use the same processing of multiple actions.

In addition, FSv2 defines a default order by categories, but also allows implementations to configure the order (just like in FSv1). If the implementations configure FSv2-EC to operate in the same fashion, FSv2 with user ordered filters can handle multiple actions just like FSv1.

To future proof, the FSv2 design, FSv2 has defined an optional Action Chain Ordering (ACO) is defined in section 3.3.3 that allows the originate of the FSv2 NLRI/FSv2-Actions to indicate whether the default order is used for FS-EC or a configured order for multiple actions.

3.3.2.2. Failure of an FS-EC Action

If a single FS-EC action is attached to a Flow Spec filter, then if the action fails there are three options:

Option 1. Stop processing additional filters and (optionally) signal failure to the management process,

Option 2. Continue on processing in "best effort" for the next filters.

Option 3. Decide between 1 and 2 based on dependencies between filters and actions

Option 1 and 2 can be signaled by configuration within a Flow Specification implementation. Option 3 requires the encoding dependency lists in ordered filters and ordered actions. The FSv2 NLRI format has a field to carry filter dependency information, but these funtions are beyond the FSv2 Basic IP functions.

Suppose two multiple FS-EC actions are specified with redirect to VRF and terminate action (TAIS Terminal). Given the default category order the redirect actions would occur, and then the TAIS (terminal) action. If the redirect to VRF fails, the TAIS action would still occur. Therefore, TAIS-Terminate provides some protection but does not choose between option 1 and option 2.

To spread information about: a)configured/default action ordering, and b) failure actions, the optional Action Chain Ordering Extended Community is defined below. This FSV2-EC the originator of the FSv2-EC to express their preference for failure mode (option 1 or 2) if single action among multiple actions attached to a filter fails.

Editor’s comment: Part of our discussion is whether the optional Action Chain ordering should be moved to another specification.

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

This optiona; FSv2-EC action provides an alternate way to pass information regarding: FSv2-EC category ordering and FSv2-EC failure actions.

Summary: Action Chain Ordering sets the default order dependency and failure mode within the chain of actions engaged by a filter match.

Description: If a FSv2 implementation associates multiple Actions with a filter, then the following types of interactions between these multiple interactions can occur:

Two conflicting actions on traffic flow in a category If two

actions modify a single packet or a traffic flow, the order the action operate may be important. A deterministic order of action processing allows an implementation to be able to count on which actions occur first. The previous section of this document set default ordering actions based on FSv2-EC categories so that a FS implementation can count on which actions of multiple actions is enacted first.

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 Action Chain Ordering (FSV2-EC) AC-Failure value specifies what occurs when an action failes.

Encoding: The Generic Transitive encoding is shown in figure 3-11 with the field definitions below.

Generic Transitive Extended Community (IPv4)

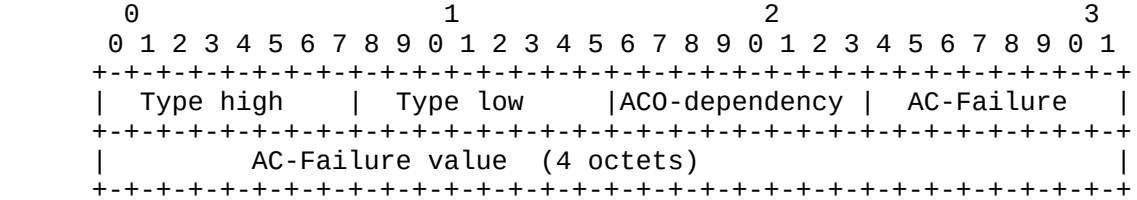


Figure 3-11

where:

Type high: This 1 octet field has a value of 0x80 For the Generic Transitive EC.

Type low: This one octet field identifies the ACO-Action. The value is TBD4.

ACO Dependency This field indicates whether the FS-EC category order is the pre-defined order or an implementation specific order.

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

1 = Implementation specific order and interaction. The

implementation specific order is outside the scope of this document

AC-failure-type 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 “stop on failure”.

where AC-Failure types are 0x00 stop on failure

0x01 continue on failure (best effort on actions)

AC-Failure value - Reserved for future use. Must be set to all zeros, and ignored upon reception.

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 “session reset” [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:

FSv2 received over SAFI=TBD1 FSv1 received over SAFI=133 are be validated against SAFI=1. Similarly, FSv2 routes received over SAFI=TBD1 will be validated against SAFI=1.

FSv2 received over SAFI=TBD2 FSv1 received over SAFI=134 are validated against SAFI=128, and FSv2 received over SAFI=TBD2 will be validated against SAFI=128.

FSv2 received with AFI = 31 The FSV2 routes received with (AFI=31, SAFI=TBD1) will be validated against SAFI=1. The FSv2 received with (AFI=31, SAFI=TBD2) will be validated against SAFI=128.

FSv2 L2 routes passed in (AFI=6, SAFI=TBD1) and L2VPN routes passed in (AFI=25, SAFI=TBD2). FSv2 L2 routes - validate (AFI=6, SAFI=TBD1) against (AFI=1, SAFI=1).

FSv2 L2VPN routes - validate (AFI=256, SAFI=TBD2) against (AFI=1, SAFI=128>

This is similar to FSv1. - The FSv1 L2 validated L2 routes passed in (AFI=6, SAFI=133) against (AFI=1, SAFI=1) and the L2VPN routes (AFI=25, SAFI=134) are validated against (AFI=1, 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-match 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 “more-specific” 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 “originator” 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 for IP Basic

FSv2 may be mapped to actions using Extended Communities for the IP Basic Functionality. The ordering of precedence for these actions in the precedence of the FSv2 NLRI action TLV values (lowest to highest).

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. This document defines the potential conflicts or duplications for existing FSv1 actions.

Specifications for new FSv2 actions outside of this specification MUST specify interactions or conflicts with any existing FSv2 actions

Well-formed syntactically correct actions defined in Extended Communities are linked to the filtering rules defined in the NLRI in UPDATE packet. Multiple syntactically correct FSv2 actions from Extended Communities can be linked to one filter rule. These actions will occur in the default FSv2 order if the ACO Extended Community with the "implementation specific" indicator is not attached.

If one action in the ordered list fails, the default FSv2 procedure is for the action process for this rule to stop and flag the error via system management. The action chain may continue if one of two things exist:

a) ACO community is attached to the FSv2 filter with an AC-Failure type of "continue on failure (0x01), or

b) local configuration that indicates a FSV2 action should 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, and "treated-as-withdrawl".

* 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, and treated "treated-as-withdrawl".

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 ‘session reset’ [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 FSv2 Filters and Actions

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 1 describes how a BGP peer that supports FSv1 and FSv2 should order the addition of FSv1 FS filters to a FSv2 FS filter list. A single list for all FS filter enable a single 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 (Extended Communities) provides a “ships-in-the-night” forwarding of based on different FSv1 versus FSv2 AFI/SAFIs. 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 “permit-all” action

- BGP peers which do not support flow specification permit traffic for routes received. Rule-0 is defined to be “permit-all” for 0/0 which is the normal case for filtering for routes received by BGP.

- By configuration option, the “permit-all” may be set to “deny-all” if traffic rules on routers used as BGP must have a “route” 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 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 such value ordering.

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

  o 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.

  o 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 for IP Basic

The FSv2 specification for IP Basic only allows for Extended Community actions. Ordering of Actions associated with an IP Basic filter is based on the Action type value (low byte) of the Extended Community. The action type values are listed in ascending numerical order in Table 3-11 for IPv4 and Table 3-12 for IPv6. Action type zero (0x00) is not valid.

The mixture of Extended Community action types and action types associated with a Community path attribute is outside the scope of this document.

4.3. Ordering of FS filters for BGP Peers which 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.

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 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. Generic Transitive Extended Community

IANA is requested to assign a type value from the "Generic Transitive Extended Community Sub-Types" registry at https://www.iana.org/assignments/bgp-extended-communities/bgp-extended-communities.xhtml

Table 7-3 - Generic Transitive Extended Community

Value	Description	Reference	Controller
TBD4	FSv2 Action Chain Ordering	[this document]	IETF

The requested value is "0x01".

7.4. FSv2 IP Filters Component Types

IANA is requested to create a new "BGP FSv2 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 7-5 - Flow Specification

Registry Name: BGP FSv2 Component 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]

7.5. FSV2 NLRI TLV Types

IANA is requested to create the a new registries on a new "Flow Specification v2 TLV Types” web page.

Table 7-6 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]
0x01-0x31	Unassigned	[this document]
0x32 (50)	MPLS Traffic Rules	[this document]
0x33-0x63	Unassigned	[this document]
0x64 (100)	L2 traffic rules	[this document]
0x65 (101)	Unassigned	[this document]
-0x95 (149)		
0x96 (150)	SFC Traffic rules	[this document]
0x97 (151)	Unassigned	[this document]
- 0xC7		
0xC8 (200)	Tunnel Traffic rules	[this document]
0xC9-0x99	Unassigned	[this document]
0x100 (256)	IP traffic rules	[this document]
0x101-0x10D	Unassigned	
0x10E (270)	Extended IP Rules	[this document]
0x10F-0x6000	Unassigned	[this document]
0x6000-0x7FFF	Vendor specific	[this document]
0x8000-0xFFFF	Reserved	[this document]

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-bgp-community-attribute]
Hares, S., "BGP Community Container Attribute", Work in Progress, Internet-Draft, draft-hares-idr-bgp-community-attribute-01, 14 October 2024, <https://datatracker.ietf.org/doc/html/draft-hares-idr-bgp-community-attribute-01>.

[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-24, 6 October 2024, <https://datatracker.ietf.org/doc/html/draft-ietf-idr-flowspec-l2vpn-24>.

[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-nvo3-21, 9 December 2024, <https://datatracker.ietf.org/doc/html/draft-ietf-idr-flowspec-nvo3-21>.

[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., akarch@cisco.com, Ray, S., Mohapatra, P., Henderickx, W., Simpson, A., and M. Texier, "BGP Flow-Spec Redirect-to-IP Action", Work in Progress, Internet-Draft, draft-ietf-idr-flowspec-redirect-ip-03, 8 September 2024, <https://datatracker.ietf.org/doc/html/draft-ietf-idr-flowspec-redirect-ip-03>.

[I-D.ietf-idr-flowspec-srv6]
Li, Z., Chen, H., Loibl, C., Mishra, G. S., Fan, Y., Zhu, Y., Liu, L., Liu, X., and S. Zhuang, "BGP Flow Specification for SRv6", Work in Progress, Internet-Draft, draft-ietf-idr-flowspec-srv6-06, 16 October 2024, <https://datatracker.ietf.org/doc/html/draft-ietf-idr-flowspec-srv6-06>.

[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>.

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

Jie Dong
Huawei Technologies
No. 156 Beiqing Road
Beijing
China
Email: jie.dong@huawei.com

Chaitanya Yadlapalli
ATT
United States of America
Email: cy098d@att.com

Sven Maduschke
Verizon
Germany
Email: sven.maduschke@de.verizon.com