Workgroup: IDR Working Group
Internet-Draft:
draft-hares-idr-fsv2-ip-basic-00
Published: 17 April 2024
Intended Status: Standards Track
Expires: 19 October 2024
Authors: S. Hares — Hickory Hill Consulting
D. Eastlake — Futurewei Technologies
C. Yadlapalli — ATT
S. Maduscke — Verizon

BGP Flow Specification Version 2 - IP Basic

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.

IDR requires two implementations prior to standardization. Implementers feedback on FSv2 was that the complete FSv2 has the contains the correct information, but that breaking FSv2 into a progression of documents would be helpful. The first priority in this progression is expanded IP DDOS capabilities. This document contains original FSv2 IP DDOS capabilities in FlowSpec v2 using just the extended communities to define actions.

Status of This Memo

Copyright Notice

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, the original FSv2 draft will remain a WG draft, but the content will be split out into functions that implementers can manage.

This draft provides the FSv2 specification for DDOS for IP filtering using Extended communities with the following action features:

*Minimal ordering of actions (4 actions)

*Failure instructions (continue or stop)

This flowspecification will be denoted as FSv2-DDOS. As

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).

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.

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.

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 the definition of Flow specification, a short review of FSv1 and an overview of FSv2. Section 3 contains the encoding rules for FSv2 and user-based encoding sent via BGP. Section

4 describes how to validate FSv2 NLRI. Section 5 discusses how to order FSV2 rules. Section 6 covers combining FSv2 user-ordered match rules and FSv1 rules. Section 6 also discusses how to combine user-ordered actions, FSv1 actions, and default actions. Sections 7-10 address an alternate security mechanism, considerations for IANA, security in deployments, and scalability aspirations.

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 [RFC6482]

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 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 “match conditions” 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 “MAY be relaxed by explicit configuration” 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.

```
     +--------------------------------------+
     | Flow Specification (FS)              |
     |  Policy                              |
     +--------------------------------------+
         ^               ^              ^
         |               |              |
         |               |              |
+--------^----+  +-------^-------+     +-------------+
|   FS Rule 1 |  |   FS Rule 2   | ... |  FS rule 5  |
+-------------+  +---------------+     +-------------+
                    :          :
                    :          :
                 ...:          :........
                 :                     :
       +---------V---------+      +----V-------------+
       |  Rule Condition   |      |   Rule Action    |
       |  in BGP NLRIs     |      | in BGP extended  |
       | AFIs: 1 and 2     |      | Communities      |
       | SAFI 133, 134     |      |                  |
       +-------------------+      +------------------+
           :     :    :                 :      :    :
      .....:     .    :.....       .....:      .    :.....
      :          :         :       :           :         :
 +----V---+  +---V----+ +--V---+ +-V------+ +--V-----++--V---+
 |  Match |  | match  | |match | | Action | | action ||action|
 |Operator|  |Variable| |Value | |Operator| |variable|| Value|
 |*1      |  |        | |      | |(subtype| |        ||      |
 +--------+  +--------+ +------+ +--------+ +--------++------+
```

*1 match operator may be complex.

Figure 2-1: BGP Flow Specification v1 Policy

2.2. Flow Specification v2 (FSv2) Overview

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 describbes 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 will be defined in [FSv2-Tunnel].

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 “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 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

```
       +--------------------------------+
       |          Rule Group            |
       +--------------------------------+
         ^          ^                  ^
         |          |---------         |
         |                   |         ------
         |                   |               |
+--------^-------+   +-------^-----+     +---^-----+
|      Rule1     |   |     Rule2   | ... |  Rule-n |
+----------------+   +-------------+     +---------+
                      :  :   :    :
    :.................:  :   :    :
    :          |.........:   :    :
 +--V--+    +--V--+          :    :
 | name|    |order| .........:    :.....
 +-----+    +-----+ :                  :
                    :                  :
   +----------------V----+  +-----V----------------+
   |Rule Match condition |  | Rule Action          |
   +---------------------+  +----------------------+
    :      :     :    :       :      :   :   :   |
 +--V--+   :     :    :    +--V---+  :   :   :   V
 | Rule|   :     :    :    |action|  :   :   :  +-----------+
 | name|   :     :    :    |order |  :   :   :  |action name|
 +-----+   :     :    :    +------+  :   :   :  +-----------+
           :     :    :              :   :   :.............
           :     :    :              :   :                :
      .....:     .    :.....       ..:   :......          :
      :          :         :       :           :          :
 +----V---+  +---V----+ +--V---+ +-V------+ +--V-----+ +--V---+
 |  Match |  | match  | |match | | Action | | action | |action|
 |Operator|  |variable| |Value | |Operator| |Variable| | Value|
 +--------+  +--------+ +------+ +--------+ +--------+ +------+
```

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 DDOS, IPv4/IPv6 filters beyond DDOS, BGP/MPLS IPv4 VPN, BGP/MPLS IPv6 VPN, BGP/MPLS L2VPN, Segment routing (SRMPLS, SRv6), SFC, SFC VPN, and tunnel.

The following is the list of planned drafts:

FSv2 IP DDOS (FSv2 IP DDOS): The first draft will support IP filter functions (Type 1) and Extended Community actions supported by [RFC8955] and [RFC8956] with additions to provide the following:

- *user ordering of IP filters
- *new filter for filtering of unknown FSv2 types and filters (type = 250)
- *no support for user ordering of actions
- *a new action (Action Chain Operation) for handling failures when multiple actions are associated with a set of filters.

This draft provides the basic functions all other FSv2 drafts will extend.

FSv2 IP Extensions (FSv2 IP DDOS Extra): This draft provides extensions for proposed filters for IP FSv2.

FSv2 Non-IP (FSv2 Non-IP): This draft provides Non-IP Filters filters for BGP/MPLS IPv4 VPNS, BGP/MPLS IPv6 VPNs, Segment Routing (SRMPLS and SRV6).

FSv2 L2VPN: [I-D.ietf-idr-flowspec-l2vpn] provides user ordered filters for L2VPNs.

FSv2 Additional Extended Community Actions (FSv2 EC Actions Extra) : This draft provides additional Flow Specification Actions that can be implemented safely with action chain failures, but without user ordering.

FSv2 Wide Community Actions in Type 1 (FSv2 Wide Action T1): This draft provides the Wide Community actions in the Type 1 format of Wide communities.

FSv2 Wide Community Actions in Type 2 (FSv2 Wide Action T2): This draft provides Wide Community actions in the type 2 format of Wide communities.

3. FSv2 Filters and Actions

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:

```
+-------------------------------+
|length (2 octets)              |
+-------------------------------+
| Sub-TLVs (variable)           |
| +===========================+ |
| | order (4 octets)          | |
| +---------------------------+ |
| | identifier (4 octets)     | |
| +---------------------------+ |
| | type (2 octets)           | |
| +---------------------------+ |
| | length-Subtlv (2 octets)  | |
| +---------------------------+ |
| | value (variable)          | |
| +===========================+ |
+-------------------------------+
```

Figure 3-1: FSv2 format

where:

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

-The combined lengths of any FSv2 NLRI in the MP_REACH_NLRI or MP_UNREACH_NLRI. The BGP NLRI length must be less than the packet size minus the other fields (BGP header, BGP Path Attributes, and NLRI).

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

*identifier: identifier for the rule (used for NM/Logging) (4 octets) *type: contains a type for FSv2 TLV format of the NRLI (2 octets) which can be:

-0 - reserved,

-1 - IP Traffic Rules

-2 - Extended IP rules

-3- L2 traffic rules

-4- SFC Traffic rules

-5- SFC VPN Traffic rules

-6 - BGP/MPLS VPN IP Traffic Rules

-7 - BGP/MPLS VPN L2 Traffic Rules

-8 - Tunneled traffic

*length-Subtlv: is the length of the value part of the Sub-TLV,

*value: value depends on the subTLV (see sections below).

The FSv2 Basic DDOS function must recognize that the defined types are valid even if the implementation does not support anything except

3.1. Basic IP Filters

3.1.1. 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.

```
+--------------------------------+
| +--------------------------+   |
| | (subTLVs)+               |   |
| +==========================+   |
+--------------------------------+
```

Figure 3-2 - IP Header TLV

Where: Each SubTLV has the format:

```
+-------------------------------+
|  SubTLV type (1 octet)        |
+-------------------------------+
|  length (1 octet)             |
+ ------------------------------+
|  value (variable)             |
+-------------------------------+
```

Figure 3-3 – IP header SubTLV format

Where:

SubTLV 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 SubTLV type).

value: dependent on the subTLV

- For descriptions of value portions for components 1-13 see [RFC8955] and [RFC8956]. For component 14 see [I-D.ietf-idr-flowspec-srv6].

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

The list of valid SubTLV types appears in Table 2.

```
Table 2 IP SubTLV Types for IP Filters
        DDOS support
SubTLV
-type     Definition
======    ============
   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 -    TTL
  15-63   reserved for IP Extensions
          (standards action)

Table 2b IP SubTLV types for non-IP Filters
SubTLV   IP SubTLV types
-type    Definition
======   ===========
  64      Parts of SID
  65      MPLS LAbel Match-1
  66      MPLS Label Match-2
  67-127  Match reserved for Non-IP
  128-191 reserved (standards action)
  192-249 FCFS
  250-    FSv2 Filter Error handling
 251-255  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.

3.1.1.1. 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.

3.1.1.2. 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.

3.1.1.3. 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.

3.1.1.4. 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.

3.1.1.5. 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]+

3.1.1.6. 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]+

3.1.1.7. 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]+

3.1.1.8. 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]+

3.1.1.9. 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

3.1.1.10. 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.

3.1.1.11. 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.

3.1.1.12. 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:

```
   0   1   2   3   4   5   6  7
 +---+---+---+---+---+---+---+---+
 | 0 | 0 | 0 | 0 |LF |FF |IsF| DF|
 +---+---+---+---+---+---+---+---+
              Figure 3-4
```

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 [RFC7091] 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.

3.1.1.13. 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]+

3.1.1.14. TTL (type=14 (0x0E))

TTL: Defines matches for 8-bit TTL field in IP header

Encoding: <[numeric_op, value]+>

where: value is a 1 octet value for TTL.

ordering: by full value of number_op concatenated with value

conflict: none

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

3.1.2. FS Filter Error handling (type=250(0xFA))

Editor note: This filters FSv2 information. Is it useful? If not, it can be moved to the non-IP section.

Type Filter Error handling(0xFA)

Function: This function suggests additional for unknown types and missing fields in the FSv2 NLRI

reference: none

Encoding: <type(1 octet), length(1 octet), T-Err (1 octet), (M-Err (1 octet)

It contains a set of {operator, value} pairs that are used for the matching filter.

T-Err - specifies handling of unknown type. The values for this type are:

Disable AFI/SAFI

Treat as withdrawl

Ignore MPREACH_NLRI attribute

Ignore filter component (Sub-TLV)

M-Error - specifies the handling of a missing field with values of: (TBD).

```
      0   1   2   3   4   5   6   7   8       9      10  11  12  13
 +---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
 |            T-Err               |  M-error                        |
 +---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
              Figure 3-8b
```

3.2. Encoding of FSV2 Actions for Basic DDOS

The long-term goal of the FSv2 actions is to allow user ordering of the flow specification actions. Only Wide Communities provide 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. To provide an easy migration between FSv1 and FSv2, a sequence of drafts will break allow adding of additional actions to the basic IP DDOS actions in the Extended Community.

The Basic IP DDOS FSv2 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 existance of the Action chain ordering action enforces that the actions will have a deterministic outcome during failures.

The AC-Failure types are:

* 0x00 – default – stop on failure

* 0x01 – continue on failure (best effort on actions)

* 0x02 – conditional stop on failure (depends on AC-Failure-value/ policy)

* 0x03 – 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

3.2.1. FSV2 Basic DDOS Actions

3.2.1.1. Encoding FSv2 Actions in IPv4 Extended 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.

```
  0                   1                   2                   3
  0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |  Type high    |  Type low(*)  |                               |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+          Value (6 octets)     |
 |                                                               |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 3-9

Table x-x

IPv4 Extended Communities (Type 0x80) 2 byte AS,

Value	Description	Name	Reference
=====	=======================	=====	==========
0x01	Flow Spec Action Chain	ACO	[This document]
0x06	Flow spec 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 traffic-remarking	TM	[RFC8955]
0x0C	Flow Spec Traffic-rate-packets	TRP	[RFC8955]

Table x-x

IPv4 Extended Communities FSv2 action (Type 0x81)

Value	Description	Name	Reference
=====	=======================	=====	==========
0x08	Flow spec rt-redirect IPv4 octet format	RDIP	[RFC8955] [this document]

Table x-x

IPv4 Extended Communities (Type 0x82)

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

3.2.1.2. 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.

```
  0                   1                   2                   3
  0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |  Type         |  Sub-type     |   Global Administrator        |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-|
 |         Global Administrator (cont.)                          |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |         Global Administrator (cont.)                          |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |         Global Administrator (cont.)                          |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 | Global Administrator (cont.)  |   Local Administrator         |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

figure 3-10

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

Type = 0x00 (transitive)
Sub-Type = see table x-x

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

Table x-x

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]

3.2.1.3. New Actions for FSv2 DDOS

There are two options for encoding the Action chain.

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

```
   0                   1                   2                   3
  0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |  Type high    |  Type low(01) |AC-failure-type|  AC-Failure   |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |          AC-Failure-value (cont.)                             |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

SubTLV: 0x01

Length: 6 octets

Value:

AC-dependence - 1 octet byte of flag regarding dependency

AC-failure-type – 1 octet byte that determines the action on failure

AC-failure-value – 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 “stop on failure”.

where:

AC-Failure types are:

-0x00 – default – stop on failure

-0x01 – continue on failure (best effort on actions)

-0x02 – conditional stop on failure – depending on AC-Failure-value

-0x03 – rollback – 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

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

```
  0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  Type high    |  Type low(07) |traffic action field  (zero)   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|          AC-Failure-value (cont.)                   | ACO |S|T|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

```
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
```

3.2.1.4. Interactions between FSv2 DDOS actions

Table 5 – All FSv2 IPv4 Action types for IP DDOS

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

Figure 3-15

Table 5 – 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

Figure 3-15

3.2.2. Summary of all FSv2 Actions (informative only)

Table 4 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)[draft-ietf-idr-flowspec-l2vpn-17]
34	TPID: TPID-Action (0x17)[draft-ietf-idr-flowspec-l2vpn-17]
24-254	TBA (to be assigned)
255	reserved

Figure 3-15

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]

4. Validation and Ordering

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 “TREAT as WITHDRAW” [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].

o2a.This condition should be enabled by default.

o2b.This condition may be disabled by explicit configuration on a BGP Speaker,

o2c.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

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 ‘Treat-as-withdraw’ [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 “ships-in-the-night” 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 “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 (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:

  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

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 “Set Action Chain operation” (ACO) (value 0x01) that defines the default action chain process. For details on “set action chain operation” 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 “Action Chain Operation” (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 “stop upon failure” (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.

4.2.2.2. 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                          Value Order
(lowest value to highest)          (lowest to highest)
================================   ==============================
0x01: ACO: Action chain operation  Failure flag
0x02: TAIS:Traffic actions per     AS, then Group-ID, then Action ID
       Interface group

0x03-0x05 to be assigned           TBD
0x06: TRB: Traffic rate limit      AS, then float value
      by bytes
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      AS, then float value
         by bytes
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 – VLAN action                 rewrite-actions, VALN1, VLAN2,
                                   PCP-DE1, PCP-DE2
0x17 – TPID action                 rewrite actions, TP-ID-1, TP-ID-2

                 Figure 6-1
```

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.

```
+---------+                       +---------+
|    A    |=======================|    C    |
|FSv1+FSv2|.                   . .|  FSv2   |
+---------+ .                 .   +---------+
 ||  |   \   .               .      .     ||
 ||  |    \   . . . . . . . . .     .     ||
 ||  |     \               .   .    .     ||
 ||  |      \-----\      .      .   .     ||
 ||  |             \    .        .  .     ||
+---------+       +------+       +-----+  ||
|    E    |-------|  B   |. . . .|  D  |  ||
|FSv1+FSv2|       | FSv1 |       |no FS|  ||
+---------+       +------+       +-----+  ||
  ||     .                         .      ||
  ||     . . . . . . . . . . . . . .      ||
  ||                                      ||
  |========================================|

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

Figure 6-2: 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 – 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 [RFC6482].

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:

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:

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

7.3. 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:

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]
14	TTL	[this document]
15	Partial SID	[draft-ietf-idr-flowspec-srv6] [this document]
16	MPLS Label Match 1	[this document] [draft-ietf-idr-flowspec-mpls-match]
17	MPLS Label Match 2	[this document] [draft-ietf-idr-flowspec-mpls-match]

7.4. FSV2 NLRI TLV Types

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

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

Type	Use	Reference
0x00	Reserved	[this document]
0x01	IP traffic rules	[this document]
0x02	FSv2 Actions	[this document]
0x03	L2 traffic rules	[this document]
0x04	tunnel traffic rules	[this document]
0x05	SFC AFI filter rules	[this document]
0x06	BGP/MPLS VPN IP traffic rules	[this document]
0x07	BGP/MPLS VPN L2 traffic rules	[this document]
0x08-0x3FFF	Unassigned	[this document]
0x4000-0x7FFF	Vendor specific	[this document]
0x8000-0xFFFF	Reserved	[this document]

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]

7.5. Wide Community Assignments

IANA is requested to assign values in the BGP Community Container Atom Type Registry

Name	Type value
FSv2 action atom	TBD5

IANA is requested to assign values from the Registered Type 1 BGP Wide Community Types:

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.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-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>.

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-flowspecv2-03, 23 October 2023, <https://datatracker.ietf.org/doc/html/draft-ietf-idr-flowspec-v2-03>.

[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
Futurewei Technologies
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