IDR FSv2 Interim Design Team 1 + 2

6/10/2024

Time: 10-11:30am

Agenda for Design Team 1 (DT1)
FSv2 for Basic IP (30 minutes]

0) Agenda Bashing + Introductions [5 minutes]

1) Short overview of FSv2 for Basic IP [30 minutes]

   draft-hares-idr-fsv2-ip-basic (Sue Hares]

2) Design team 1 Discussion - 7 questions [30 minutes]

- Adoption call – confirm subset
- Comments on draft-hares-idr-fsv2-more-ip-filters
- Comments on draft-hares-idr-fsv2-more-ip-actions

FSv2 – IDR Interims operating as Open Parallel design teams to break FSv2 into “chunks”

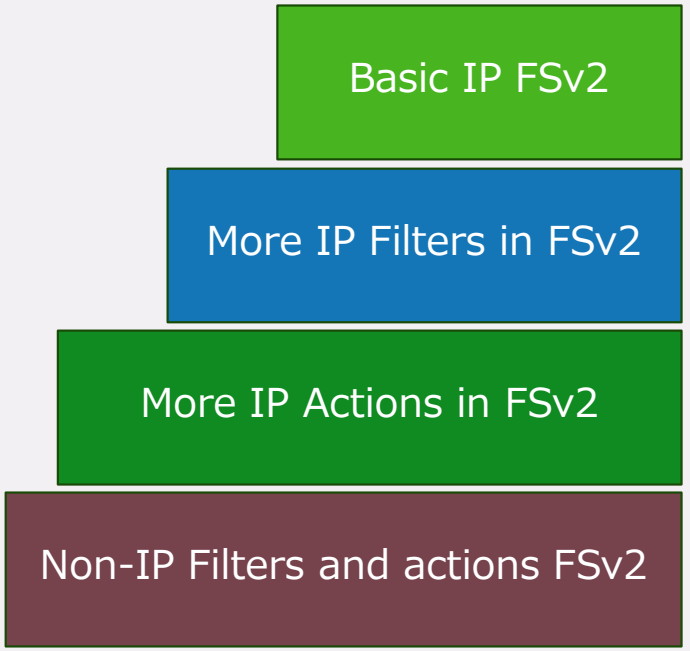
Team 1: Basic IP FSv2: Current IPv4/v6 filters + current actions + order

Team 2: More IP Filters FSv2 - Defining more IP filters to add to IP

Team 3: More IP actions FSv2 – Defining more actions + action sequences to add to

Team 4: Non-IP filters and actions FSv2

Ask to join a team!



Interims for FSv2 Open Design Teams

Phase 2 – June – Getting ready for IETF-120

- 6/03 – Design Team 1 – Review of FSv2 Basic IP

  draft-ietf-hares-fsv2-ip-basic-02 is in adoption phase
- 6/10 – Design Team 1 + 2 – More IP Filters
- 6/17 – Design Team 3 + 4 – IP Actions + Non-IP Work

IDR FSv2 Interim Design Team 1

5/6/2024

Time: 10-11:30am

FSv2 for Basic IP Review

draft-hares-idr-fsv2-ip-basic-02

Sue Hares

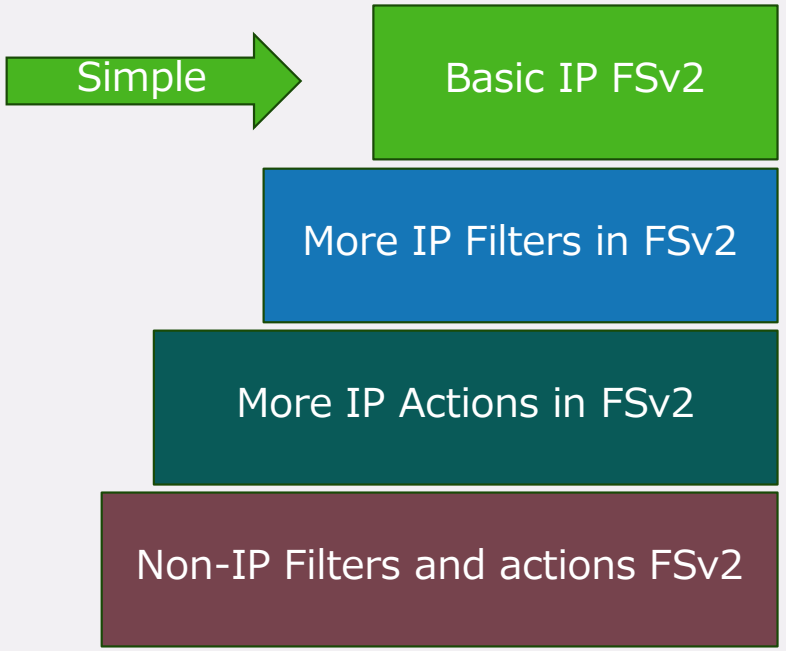
Goals of FSv2 Chunks: Simple + Complex

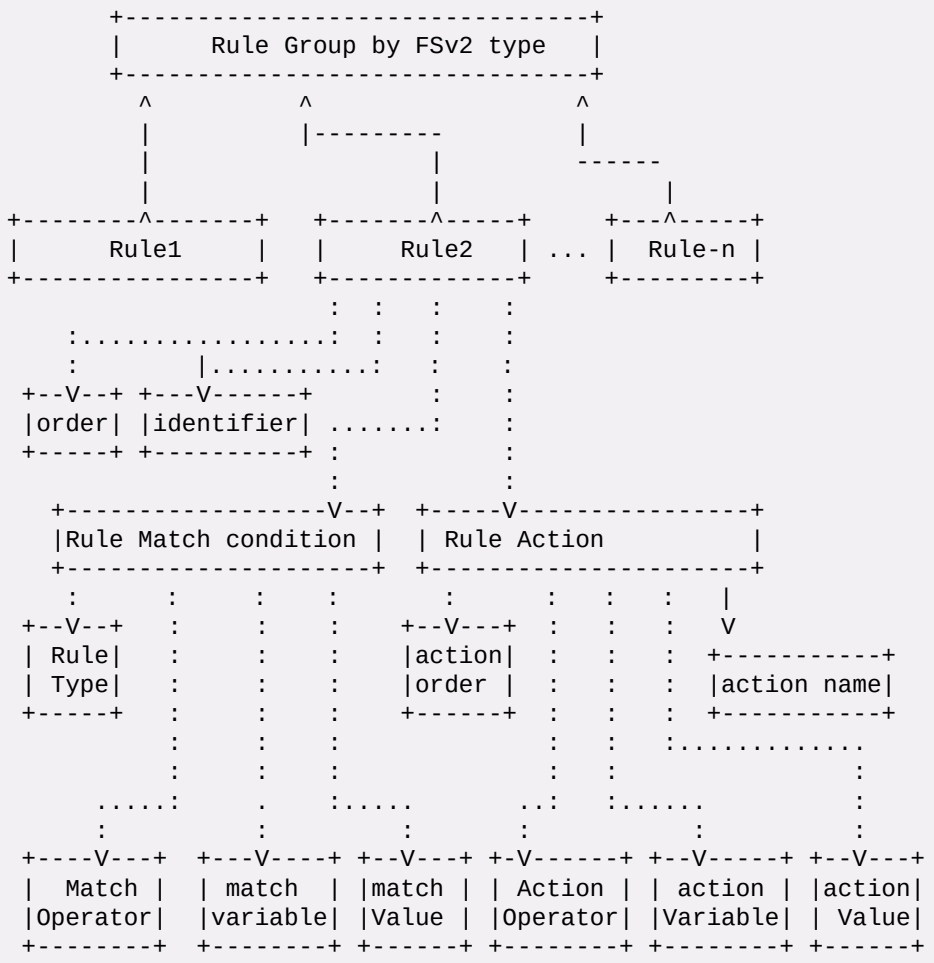
Simple DDOS Update

- User Ordering + FSv1 + Deterministic fixes

Platform for Complex Uses

- More Filters
- More Actions without conflict
- Non-IP Filters





FSv2 Rules (see Rule 2) have

1. User Order – User assigned number
2. Identifier – logging identifier
3. Rule type
   - match operator
   - match variables
   - match value

Possible to have multiple filter conditions before an action

Order
Rule – 0 = permit all traffic
no actions

Rule 1-N – Filter traffic to take an action on.

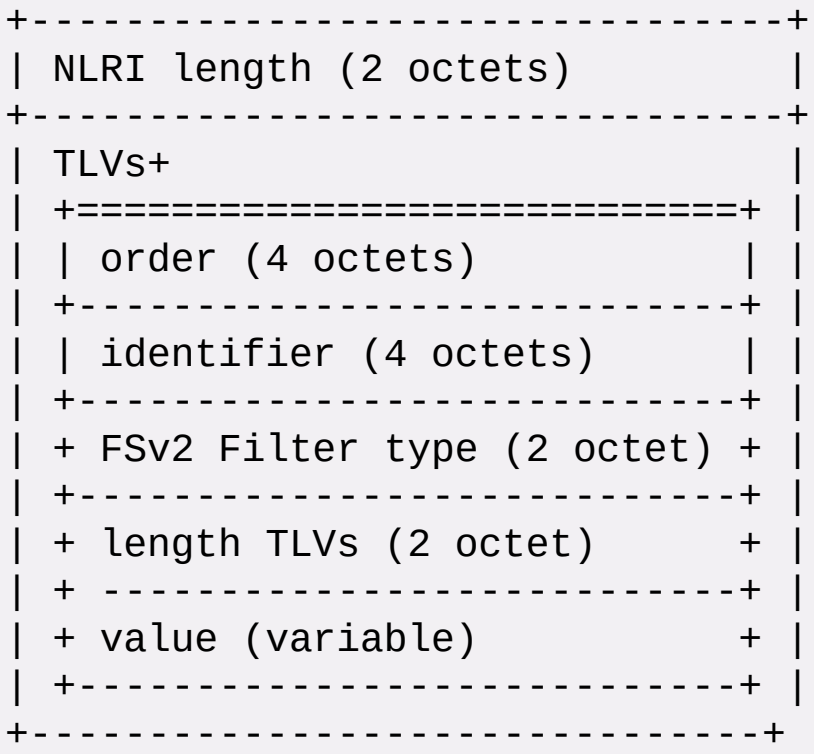
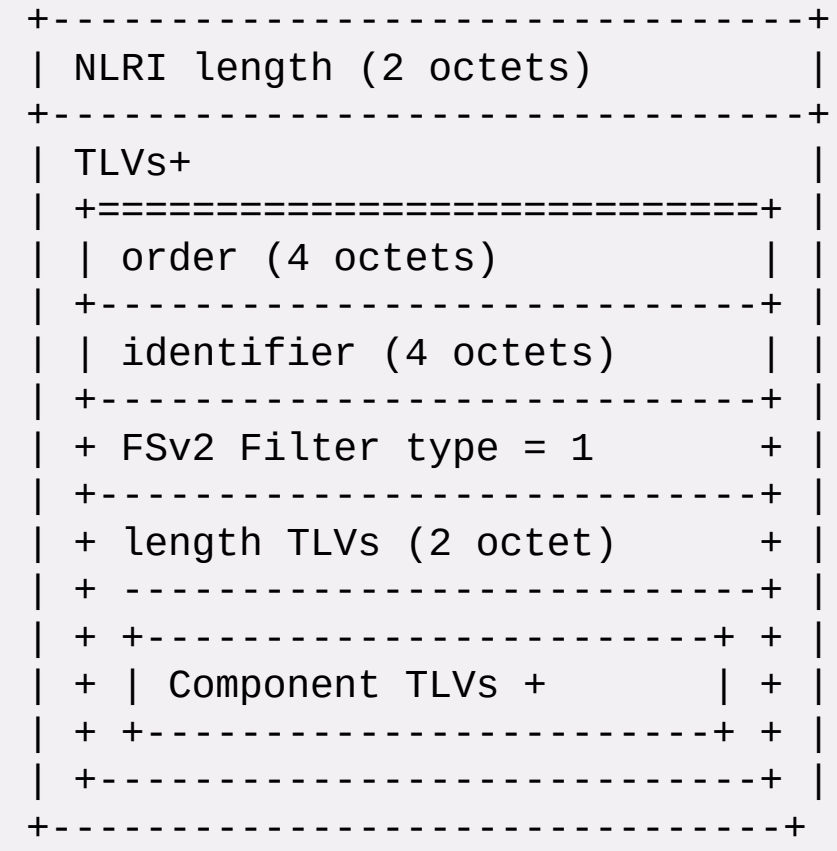


Figure 3-1 - NLRI format for FSv2

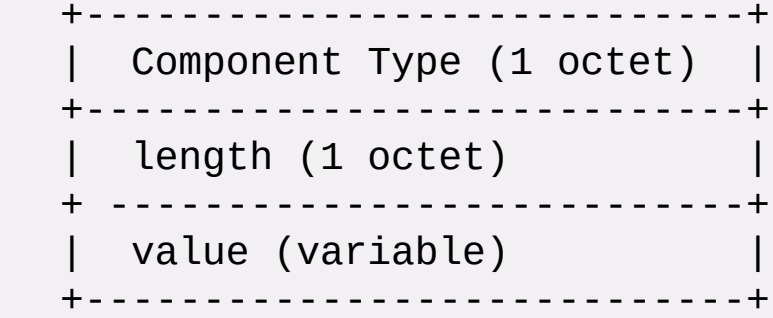
FSv2 Filter types
0 - reserved,
1 - IP Basic Filter Rules
2 - Extended IP Filter rules
3 – MPLS traffic rules
4 – L2 Traffic rules
5 – SFC traffic rules
6 – Tunnel traffic rules

IP Basic only has FSv1 Filters



NLRI format for Basic IP Filters

Where the Component TLVs are:



Components are
FSv1 components (IPv4, IPv6)
+ TTL field (value 14)

Current feedback:
move TTL into extended IP Filters.

FSv1 IP Component Numbers

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

IP Component numbers (in Extended IP Filters)

0 – TTL (option 2)
14 – TTL (option 1)
15 – SID in IPv6 Routing header
16 – NRP in Hop-by-Hop IPv6 header

Non-IP Component Numbers

64 MPLS Label Match-1
65 MPLS Label Match-2

Allocation of Component IDs

15-63 Reserved for IP Extensions (standards action)
64-127 Reserved for Non-IP Filters (standard action)
128-191 Reserved for Standard Action
192-249 FCFS
250-255 Reserved

FSv2 filters + FSv1 filters

- **FSv2 and FSv1 are Ships in the night** (two NLRIs)
- **Ordering**

  Rule 0 – permit all

  Rule 1 to Rule N-1 – FSv2 with user order.

  Rule N to end - FSv1 rules at a single user order

- If same user order, then order by component number.
- If same user order + component number, then order of multiple components using rules defined in a component.

Two types of Actions

Extended Communities (EC) – FSv2-EC Actions in

- Generic transitive EC
- IPv4 Transitive Extended Communities
- Transitive IPv6-Address-Specific Actions

Community Attribute Actions – User Ordered Actions

- Community Attribute with a FSv2 Community type

Ordering + Failure of FSv2 Actions

Ordering of FSv2 Actions

- First: User-Ordered [Community Path Attribute]
- Second: Pre-defined Extended Community – ordered by Type

Transition from FSv1 Extended Community

- Configuration knobs to Allow FSv1 Actions
- Configuration knobs must be consistent within AS or a Group of AS

What happens on Failure

- 4 Cases in NETCONF Configuration – Stop (Terminate), Go on (Best effort), Conditional Go on, Rollback changes

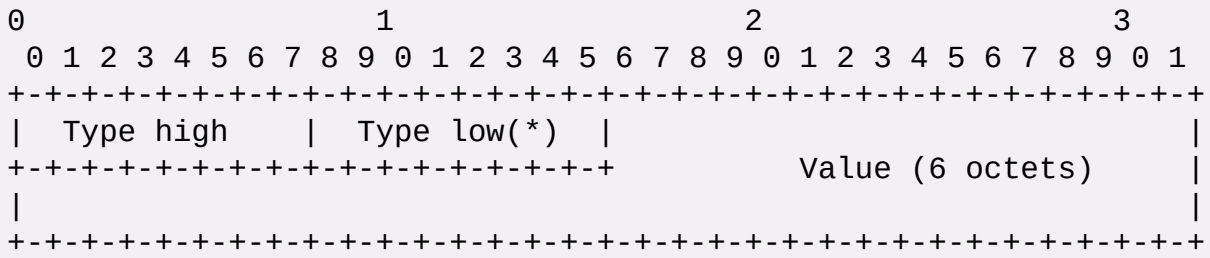


Figure 3-5 – Generic Transitive Extended Community

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

IPv4 Extended Communities (Type 0x80)

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

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

IPv4 Extended Communities FSv2 action (Type 0x81)

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

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

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

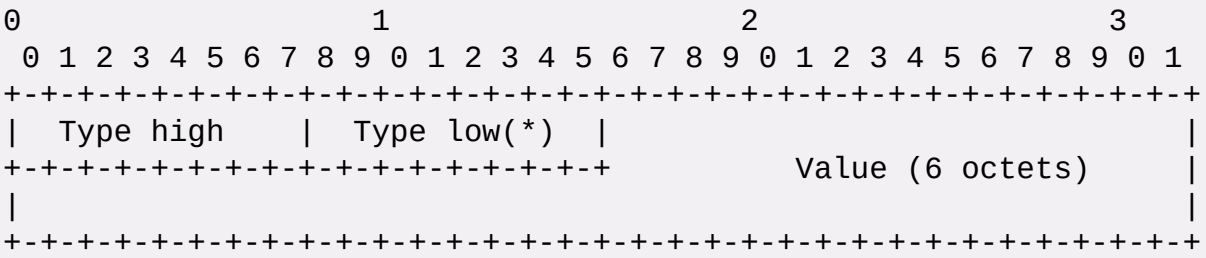


Figure 3-5 – Generic Transitive Extended Community

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

sub-type	FSv1 Description	Name	Reference
0x07	FS Interface set	Ifset	draft-ietf-idr-flowspec-interfaceset
0x08	FS Redirect/Mirror	RIPv4	draft-ietf-idr-flowspec-redirect-ip
0x09	FS Redirect to Indirection ID	RGID	draft-ietf-idr-flowspec-path-redirect

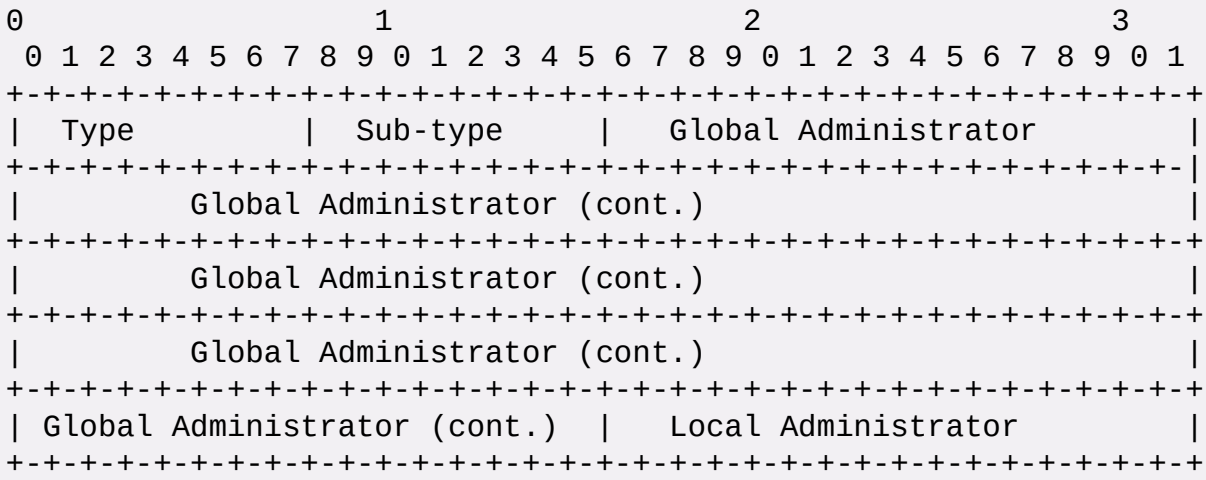


Figure 3-6 Transitive IPv6-Address-Specific-Actions

Table 3-8 Transitive IPv6-Address-Specific Actions

Value	Description	Name	
0x01	**Flow Spec Action Chain**	**ACO**	**[This document]**
0x0C	Flow Spec redirect-v6-flag	RD6F	draft-ietf-idr-flowspec-redirect-ip
0x0D	Flow Spec rt-redirect IPv6 format IPv6 format	RDv6	RFC8956

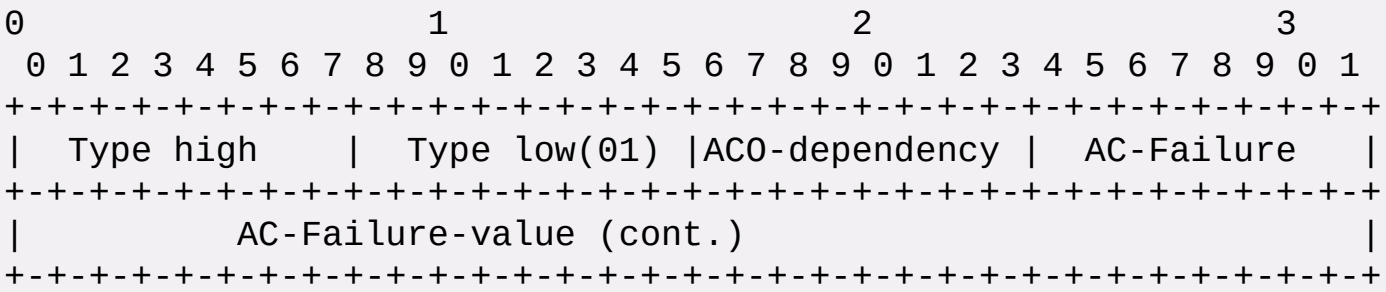


Figure 3-7

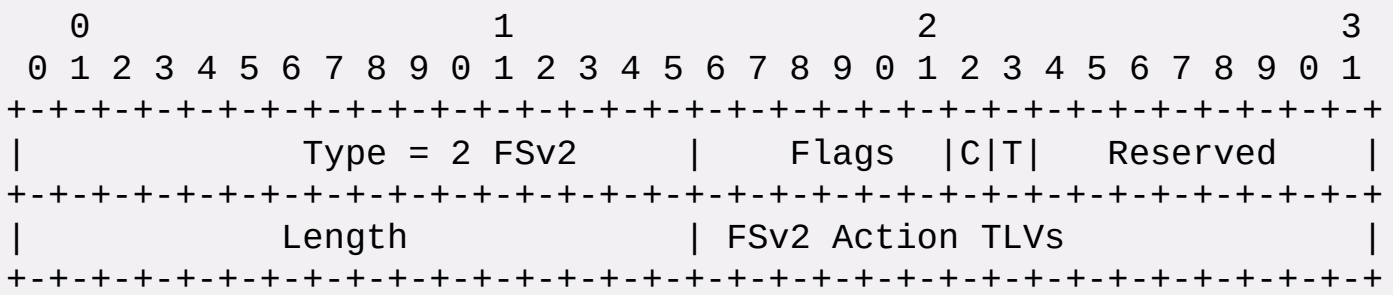
ACO Dependency - The order dependency within the Action chain.
where: 0 = default order and interactions (from this specification).
1 = Implementation specific ordering

AC Failure: Action chain action when an individual action fails
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 on AC-Failure-value

Yang configuration uses some of these modes on action failure.

FSv2 in Community path header

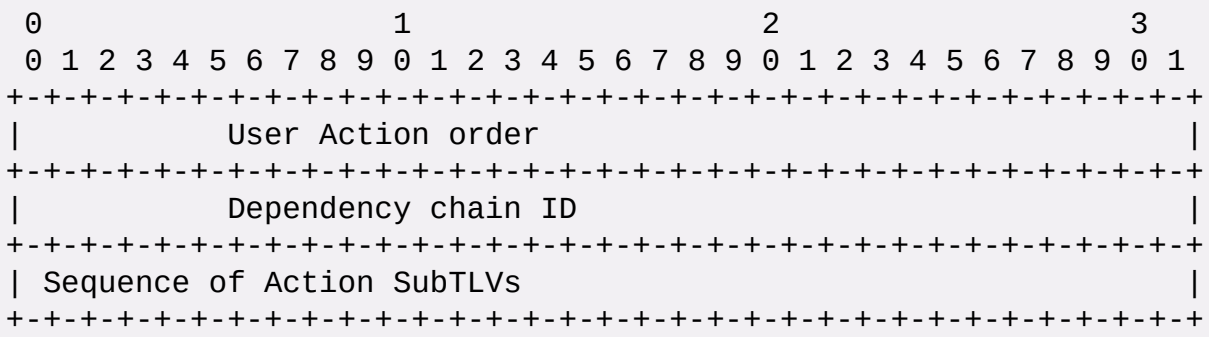
Community Path attribute common header (figure 2-3)



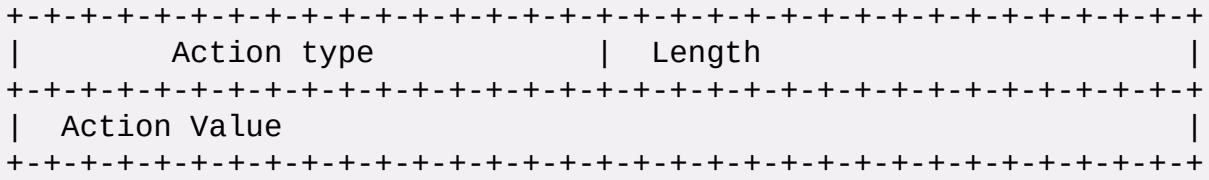
C = 1 - Transitive across Confederation boundaries
C = 0 - Non-transitive across Confederation boundaries
T = 1 – Transitive across AS boundaries
T = 0 – Non-Transitive across AS boundaries

Action TLV + SubTLV Format

Common Header for Action TLVs (figure 2-5)



Each Action SubTLV has the format:



Action TLVs for Community Path Attribute

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

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

Action TLVs

Type 1 – Use Extended Community Actions Types + Values

- Allows Extended Community Actions with user-defined order and dependency Change

Challenge:

- What FSv2 dependency chains should there be in FSv2 actions ?
- One type of action is based on failure/success
- One type of dependency is based on a modification of packet

Nat Kao's example of modifying packet

A packet with DSCP 0 hits Rule 100.

Rule 100 has actions <Set DSCP 4, GOTO Rule 400>.

Rule 400 is matching against DSCP 4.

Will that packet be considered a match for Rule 400?

-Rule 400 will match the modified packet if we apply actions after each rule.

-Rule 400 will not match the unmodified packet, if we apply actions after all rules.

Should we modify packets as soon as the match occurs?

Design Team 1 Questions for Open Issues

Question 1: Does the User Ordering in NLRI support use cases for current DDOS scenarios?

Action items:

- Design Team 1 will review the use cases for FSv2 and determine if the user ordering can support the use cases.

Where to find Use cases

- Use cases

2. What happens if multiple FSv2 filters are received with the same user order?

FSv2 currently orders filters with the same user order by FSv1 component types (using FSv1 component logic)

- Should Component types be common in IPv4 and IPv6?
- How should we order component types for upper layers?
- Does this scheme break down?

3. Does the NLRI format for user-ordering allow for additions?

- Can New IP Filters for payload be added?
- Can Non-IP Filters (such as L2 + Time) be added?
- How can dependency between filters be expressed?

  If filter-1 matches, can we skip to filter-2.
- Can we handle creating group tags and then testing?
- Can we handle groups + interfaces?
- Can we handle quick addition of new filters (DDOS)?

4. What happens if errors are detected in the parsing of the filters?

- If a filter format is broken, how does it impact past filters?

  IDR RFC7606 has a concept of “treat as withdraw”
- What types of errors are possible with user-ordered filters?

5. What happens if errors are detected when filters are used?

- What happens if the BGP parsing and validation for a filter is correct, but the filter fails to install on an IP firewall portion of a machine?
- What might cause this problem?
- Should BGP FSv2 flag this problem?

6. Are the Validity Checks from FSv1 sufficient

FSv1 – RFC8955 (IPv4), RFC8956, and RFC9117 focused on check that the FSv1 filters were valid.

- Are these validity checks enough for this work?

Thanks for all your work!