IDR FSv2

IETF-121

IDR session 2: 11/8/2024

Why talk about FSv2

FSv2 work pending

- 15 IDR drafts in limbo
- 18 proposed drafts in the adoption process

FSv2 adoption + WG LC to start before IETF-122

- WG LC + Adoption for FS Extended Community Actions
- Basic direction for IP Extended Filters
  (draft-ietf-idr-fsv2-more-ip-filters)
- Additional Filters for FSv2 Extended Filters

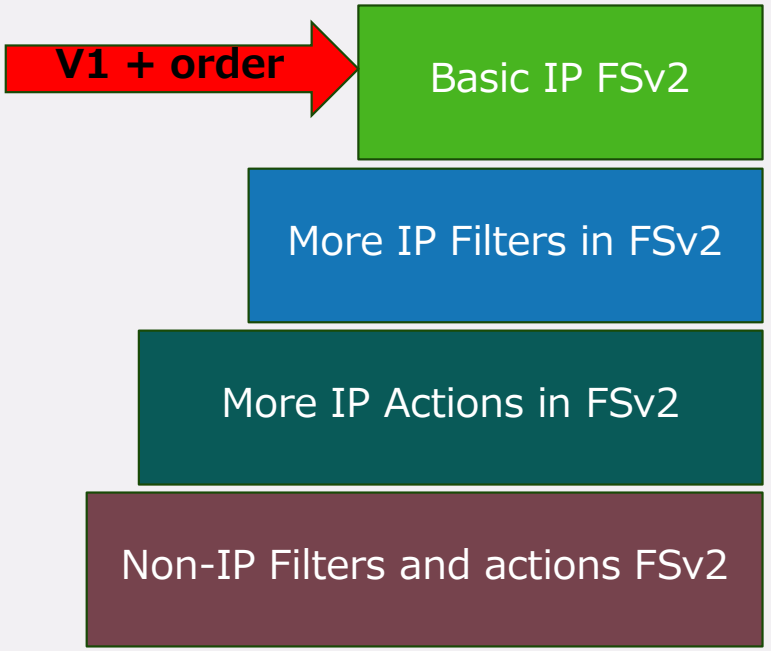
Goals of FSv2 Chunks: Simple + Complex

Simple DDOS Update

- User Ordering + FSv1 + Deterministic fixes

Platform for Complex Uses

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



Phased implementation

Basic IP FSv2: (draft-ietf-idr-fsv2-ip-basic)

Set minimal subset – V1 (actions + Filters) + User order

Extend IP Filters: (draft-ietf-hares-fsv2-more-ip-filters)

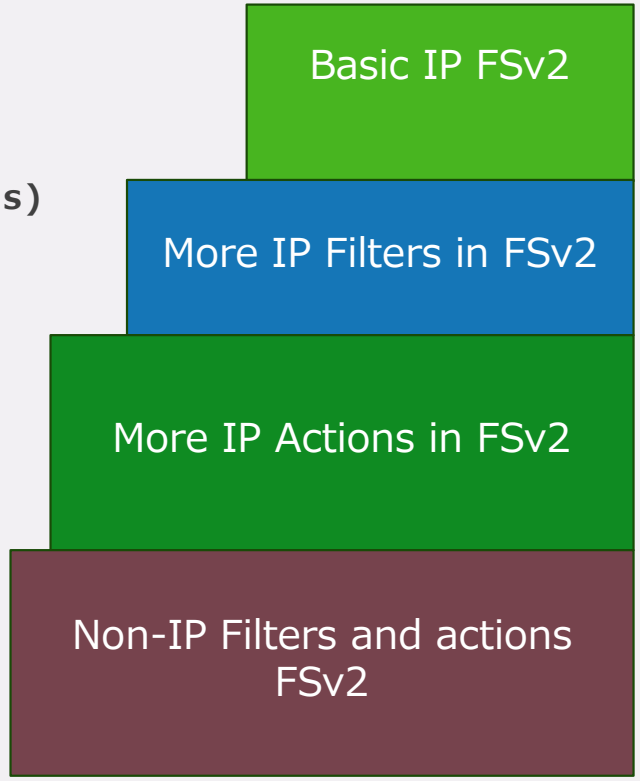
- (Optional) Extended IP Filters – in TLV
- (optional) Filter Chain = dependency between filters

FSv2 Actions in Community Path Attribute (draft-hares-idr-fsv2-more-ip-actions)

- (optional) Action chain = dependency between Actions

Team 4: Non-IP filters and actions FSv2

- TLVs for non-IP filters: L2, MPLS, SFC, Tunnel
- Optional techniques (filter chain + action chain)



3 problems with FSv1 Actions

- Lack of consistent ordering
- Lack of consistent action on multiple actions if one action fails?
- Lack of user ordering

How is this fixed in FSv1? Lots of configuration and constrained deployments

What about I-Ds proposing Flow Specification

FS Actions in Extended Community

- If deployed, append FSv2 section on action ordering.
- If not deployed, use FSv2 templates for drafts.

FSv2 Component IDs for Extended IP Filters

- Use v2 templates – for filters + actions

FSv2 Actions with Community Path Attribute

- Use v2 templates for actions

FSv2 Open issues

draft-ietf-idr-fsv2-ip-basic:

- Should open capability specify the TLV types supported?
- Transition from FSv1 to FSv2
- Filter prefix lists

Open issues – future work

- Filter dependency chain
- Action dependency chain

What feedback do the chairs need?

Should we ask for Early Allocation of values for FSv2

a) SAFIs: BGP FSv2 + BGP FSv2 VPN

b) Capability code for BGP FSv2 –

c) FSv2 Action Ordering Extended Community

Should we create Registries for FSv2 prior to 1st FSv2 draft being passed?

a) FSv2 TLV types

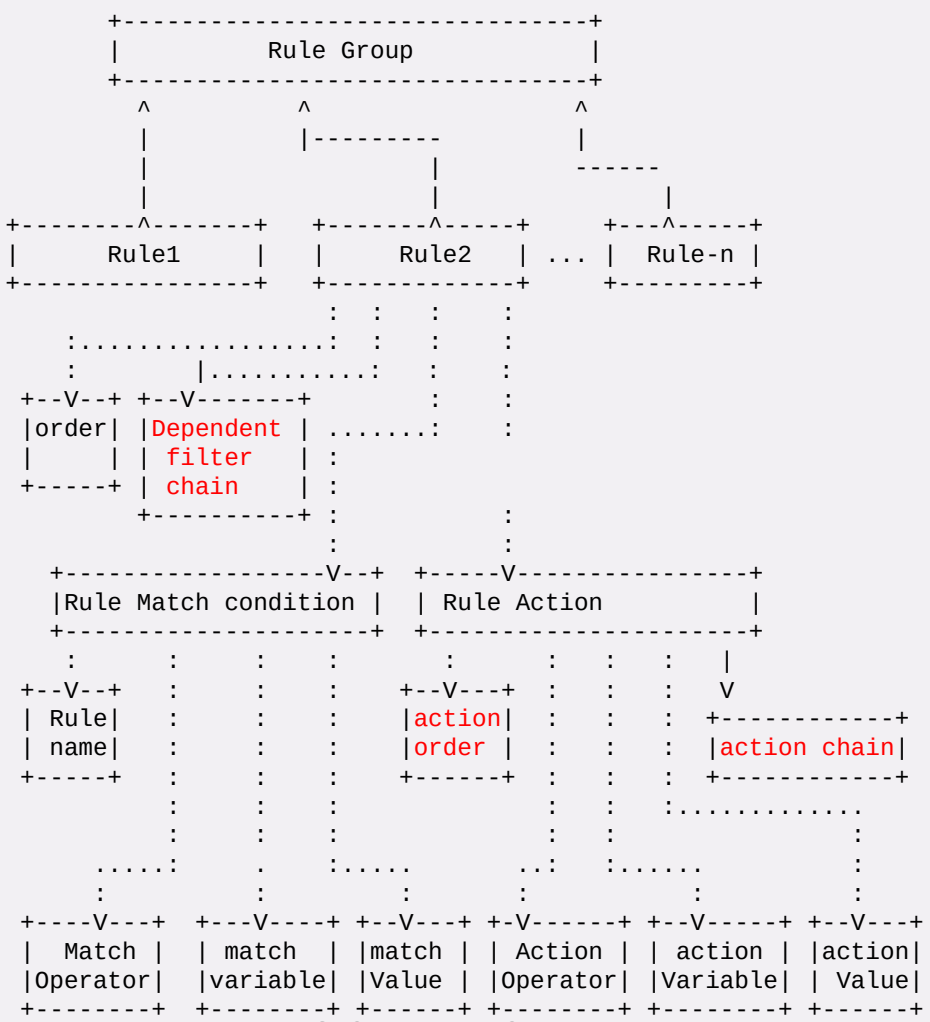
b) FSv2 Components

Requires Short RFC registries?

FSv2 for Basic IP Review

draft-ietf-idr-fsv2-ip-basic-04

Sue Hares



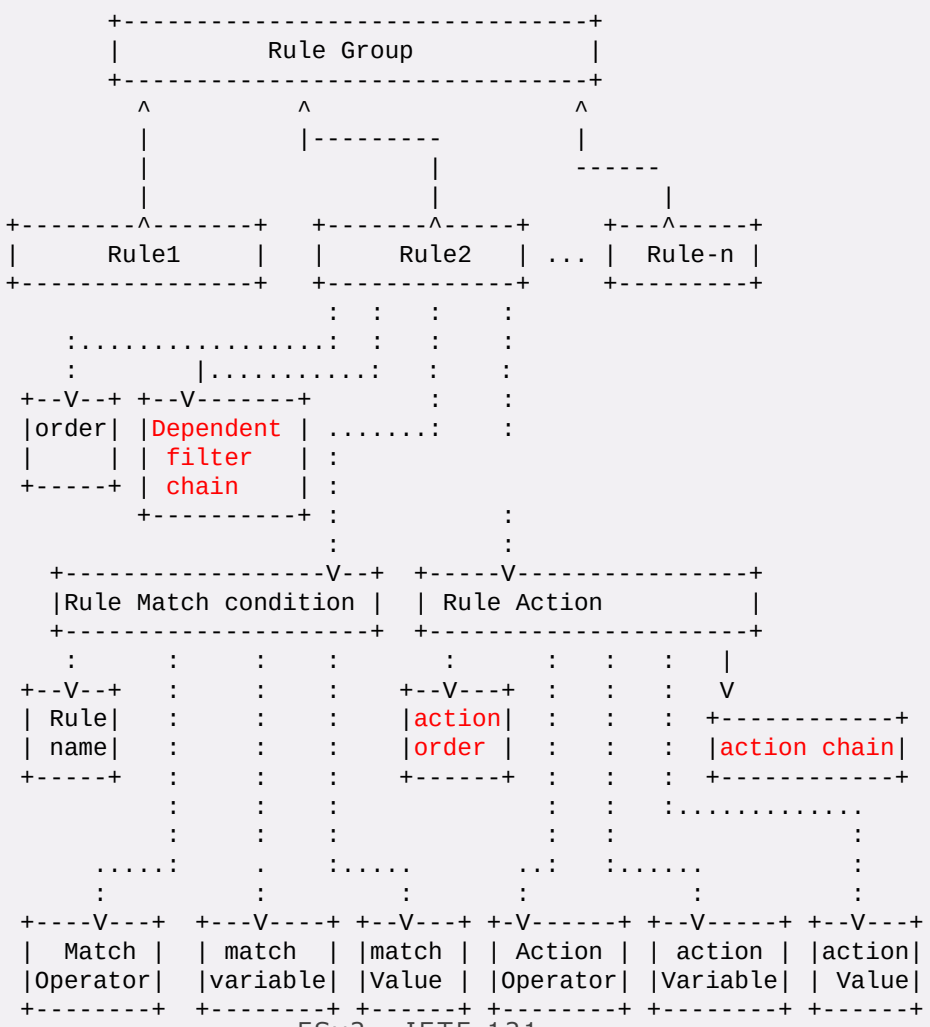
FSv2 Rules (see Rule 2) have

Filters –

1. User Order of Filters –
2. (optional) Dependency chain
3. Grouping of rules by TLV Component
   - match operator
   - match variables
   - match value

Order
Rule – 0 = permit all traffic
no actions

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



FSv2 Rules (see Rule 2) have

Actions -

1. Order
   if Extended Community – predefined
   if Community Path – user ordered

2. (optional) Dependency chain
3. Grouping of rules by TLV
   Component
   - match operator
   - match variables
   - match value

Order
Rule – 0 = permit all traffic
no actions

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

Order of FSv2 filters + FSv1 filters

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

  Rule 0 – permit all

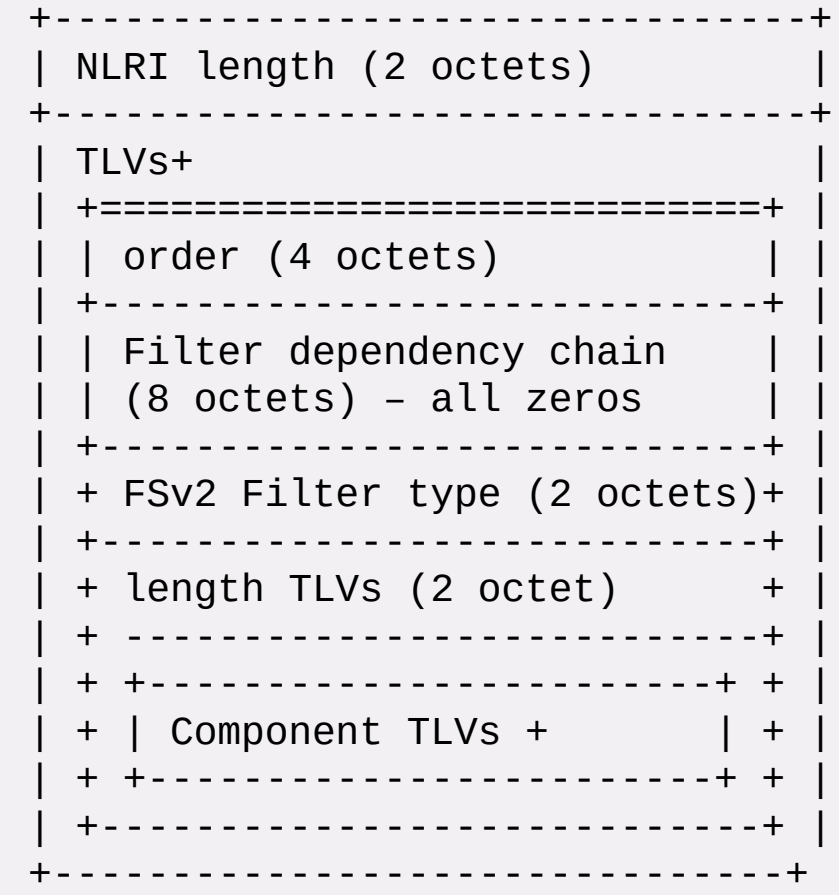
  Rule 1 to Rule N-1 – FSv2 with user order.

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

Defaults

- If same user order number, then order by component number.
- If same user order number + same component number,

then order the multiple components (with same user order + same component type) by rules defined in a component.



NLRI format for Basic IP Filters

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 Components in Filters

FSv1 IP Component Numbers

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

3 problems with FSv1 Actions

- Lack of consistent ordering
- Lack of consistent action on multiple actions if one action fails?
- Lack of user ordering

- Defined order for FSv2
- Action Chain Ordering FSv2
- Community Path Attribute

FSv2 – configuration knobs defined allow FSv1 work to upgrade actions

- New FS Action drafts – must define order within FS actions
- Ext Community – Action Chain Ordering – can be passed
- User ordering – needs thought.

IDR drafts: FSv2 actions in Ext. Community

Draft	redirect	Seq	copy	mark	Pkt
draft-ietf-idr-flowspec-redirect-ip	IPv4/v6		X		
draft-ietf-idr-flowspec-path-redirect	V4 to GID	X	X		
draft-ietf0-idr-srv6-flowspec-path-redirect	V6 to GID	X	X		
draft-ietf-idr-ts-flowspec-srv6-policy	SID Tunnel				
draft-ietf-idr-flowspec-network-slice-ts				nrp-id	Encap
draft-ietf-idr-flowspec-interfaset -					Install filters for subset of Int

Proposed drafts for Extended Community

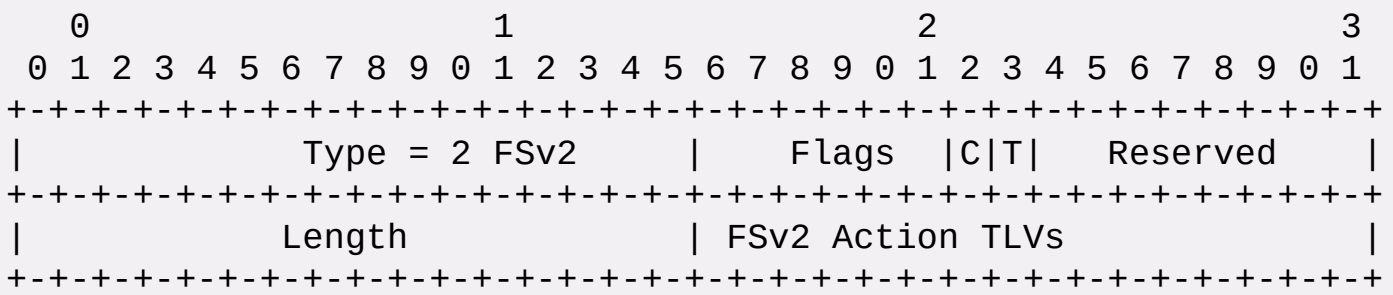
Draft for FSv2 actions	redirect	Seq	copy	mark	Pkt
draft-dmc-idr-flowspec-tn-aware-mobility	IPSec tunnel				
draft-lin-idr-cats-flowspec-ts				cat-id in IPv6	
draft-shen-idr-flowspec-traffic-compress-action					**Compress**
draft-peng-idr-apn-bgp-flowspec				apn-id	Sitch inherit
Draft-ietf0-idr-srv6-flowspec-path-redirect					

Filters + Actions for Non-IP

Draft	Filter	mark	Actions
draft-ietf-idr-flowspec-v2	MPLS label EXP bits		Push/pop/ swap labels
RFC9015	IP		SFC classifier (SPI, SI, SPT)
draft-ietf-idr-flowspec-l2vpn	L2 header (15 items)	Rewrite VLAN ID (inner or outer)	Push/pop/ swap VLAN
			Ma TPID action
draft-ietf-idr-ts-flowspec-srv6-policy	SID Tunnel		SFC classifier (SPI, SI, SPT)
draft-ietf-idr-flowspec-nv03	VLAN, GRE, L2TP	DSCP (outer)	None
draft-xiong-idr-detnet-flow-mapping	MAC + TSN info	Set latency TSN / Profile	Map flow to TSN stream (L2)

FSv2 in Community Attribute header

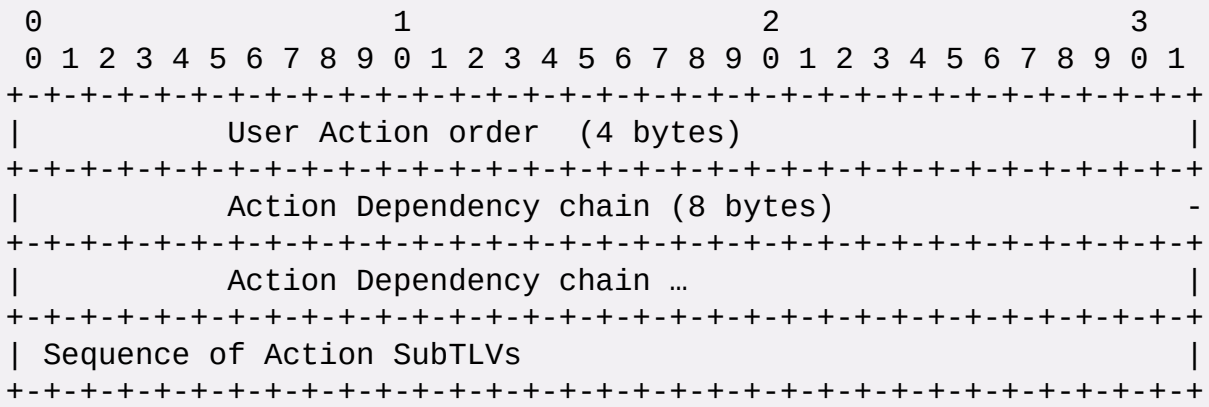
Community Path attribute common header (figure 2-1)



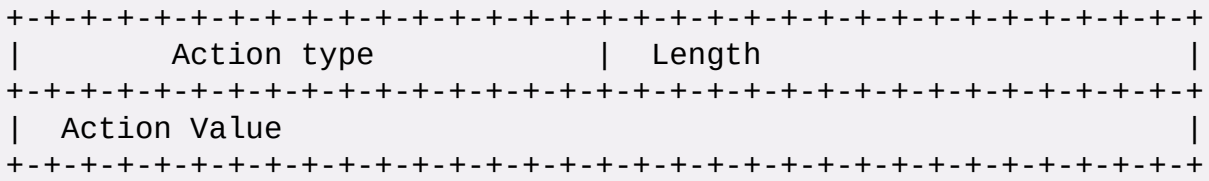
C = 1 - Transitive across Confederation boundaries
C = 0 - Non-transitive across Confederation boundaries
T = 1 – Transitive across AS boundaries
T = 0 – Non-Transitive across AS boundaries

Action TLV + SubTLV Format

Common Header for Action TLVs (figure 2-2)



Each Action SubTLV has the format:



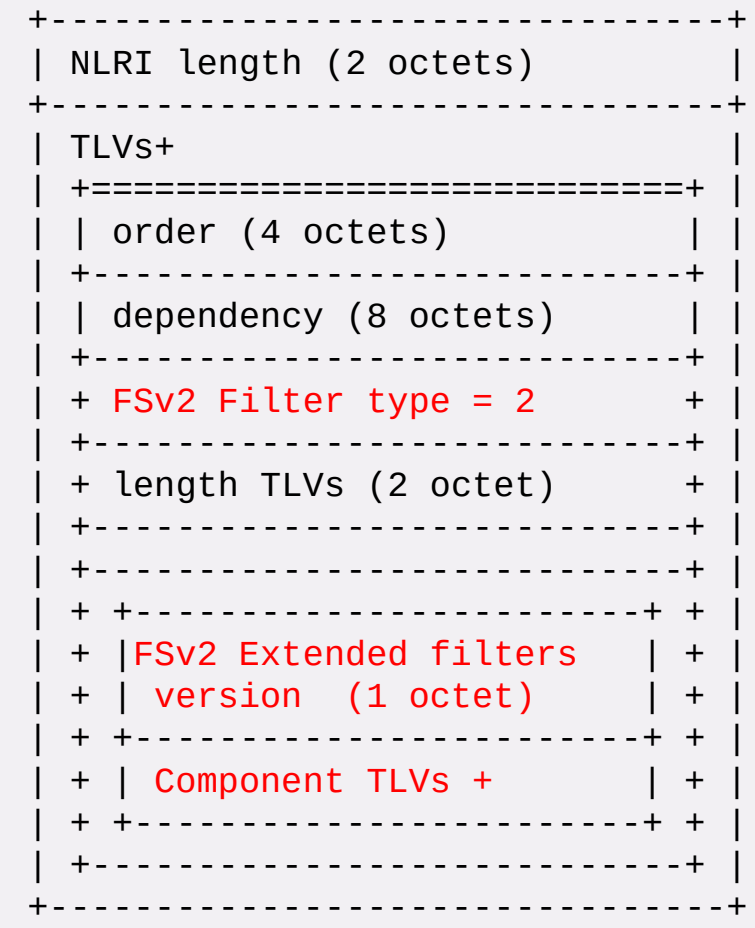
Action TLVs for Community Path Attribute

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)

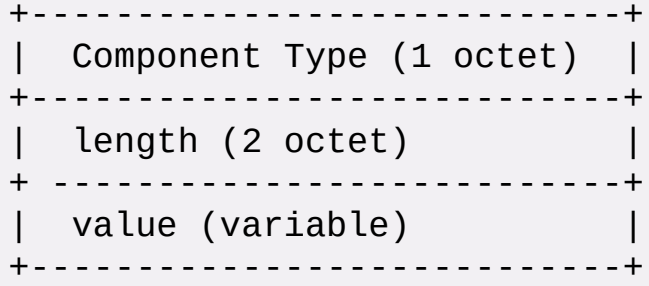
FSv2 for More IP Filters

draft-hares-idr-more-ip-filters-03



NLRI format for Extended IP Filters

Where the Component TLVs are:



- FSv2 Ext Filters Version – allows different formats of the components
  - (Why? Extensible)
- Component length – 2 octets

Additions to IP Basic

Required:

- New TLV format
- Component defined by individual drafts

Question: Should we define TTL as part of the base draft?

Optional:

Filter Dependency chain (zero for now)

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

L3 Components (Unique to each type)

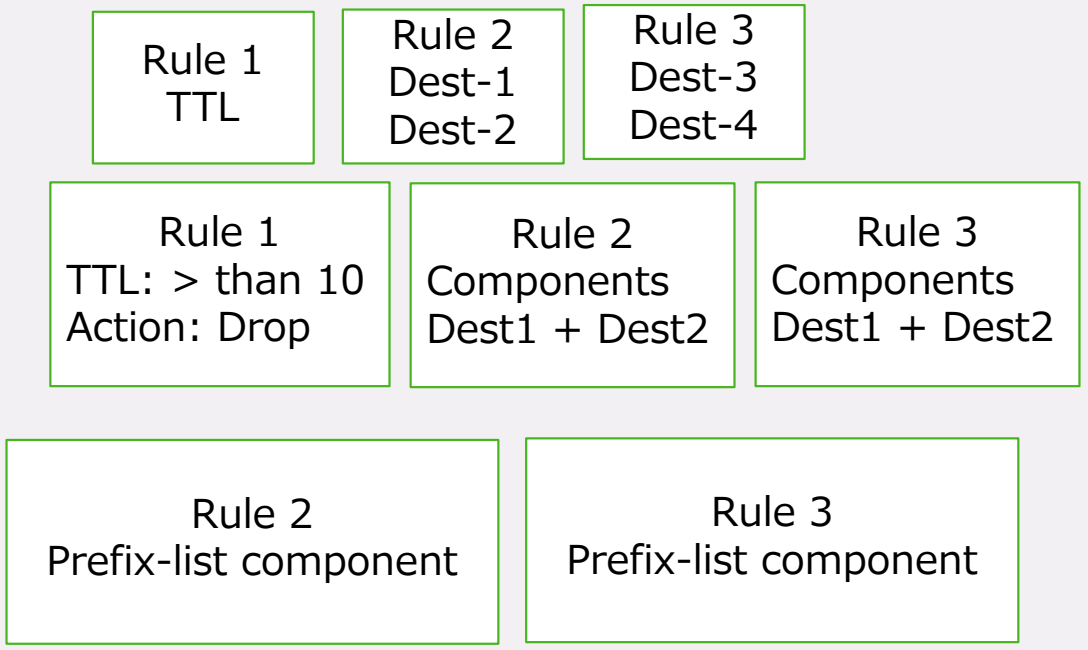
14 – TTL
15 – SID in IPv6 Routing header
16 – NRP in Hop-by-Hop IPv6 header
…
30 – Payload

Global Allocation of Component IDs (all standards action except FCFS)

14 - 63 Reserved for IP Extensions
64 - 150 Reserved for Non-IP (MPLS, L2, tunnel)
151 - 180 Associated data (interface, interface group, AS, Time, Color)
181 – 191 Reserved
192 - 240 FCFS
241 - 255 Reserved

Prefix lists Implementation

1) User order (1..n) – with current TLVs (destination)
2) One User order with Multiple IP destinations ordered by Prefix
3) New Component – that has prefix lists in a different format



Default Allocations for Proposed drafts

Existing 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

L2 Component Numbers [81-98]

1 - Ethernet type
2 - Source MAC
3 - Destination MAC
4 – DSAP in LLC
5 – SSAP in LLC
6 – control field in LLC
7 – SNAP
8 – VPAN ID
9 – VLAN PCP
10 – Inner VLAN ID
11 – Inner VLAN PCP
12 - VLAN DEI
13 - Inner VLAN DEI
14 – Src Mac Special bits
15 – Dst Mac Special bits
16 – RSN Mac Data unit
17 – Det. Latency Info

L3 Components (default order)

14 – TTL
15 – SID in IPv6 Routing header
16 – NRP in Hop-by-Hop IPv6 header
17 – CAT ID (IPv6 header (?))
30 – Payload

Linked data Components (151- 180)

151 interface or interface group
152 Color
153 Time (or times)
154 AS or Set of Ases
155 Group and Sub-group

MPLS Component Numbers [64-65)

01 (64) MPLS Label Match-1 (label)
02 (65) MPLS Label Match-2 (Exp bits)

Tunnel Component Numbers [131-142)

01 – VN ID
04 – Cookie
05 – Tunnel header flags
06 – L2TP control version
07 – L2TPv3 Control Connection ID
08 – L2TPv3 Ns
09 – L2TPv3 Nr
10 – Protocol type
11 – GRE Sequence

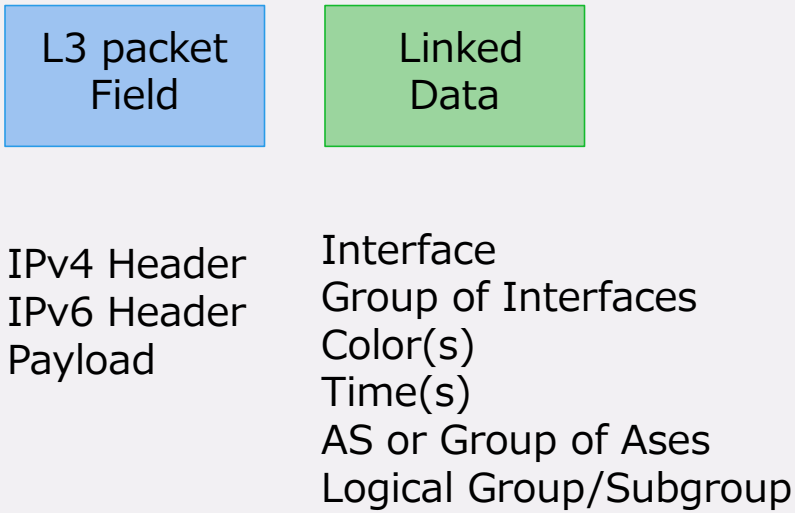
L3 Components as Firewall Rules

L3 Filters – L3 Packet field

- header IPv4 or IPv6
- Payload

Linked data

- smaller area in Firewalls
- Search packet + linked data



Summary of draft-hares-idr-fsv2-more-ip-filters

- Specifies the format of Extended IP TLV in FSV2 NLRIs
- Filter Dependency chain – left for future development.
- Authors request WG adoption call as direction

FSv2 User order Actions

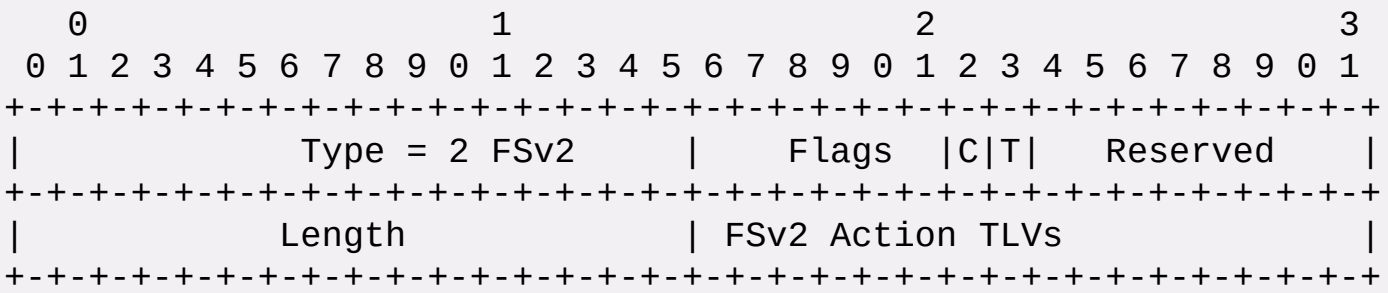
draft-ietf-hares-more-ip-actions-03

Draft defines

- Format of Action TLV within a Community Path Attribute
- Action Chain format
- Actions transitions to Action chain format – IP Redirect discussion on list

FSv2 in Community Attribute header

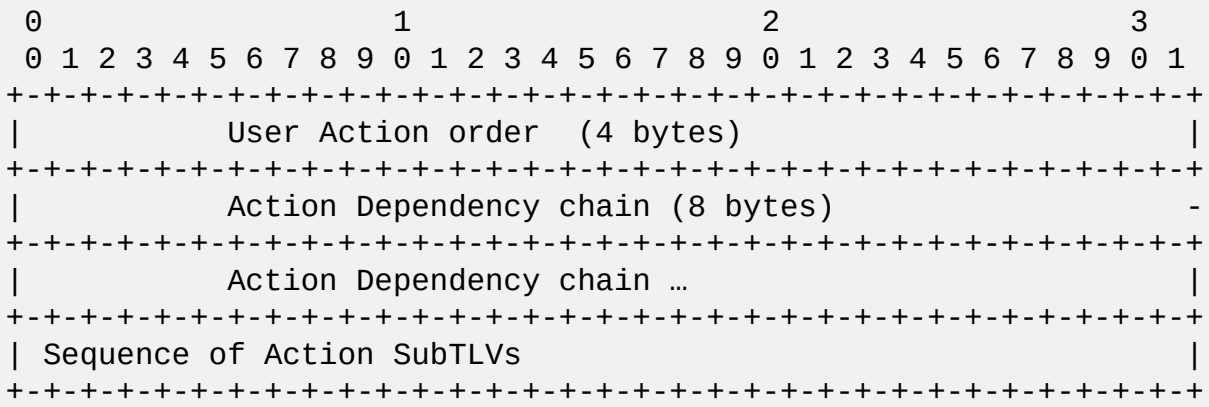
Community Path attribute common header (figure 2-1)



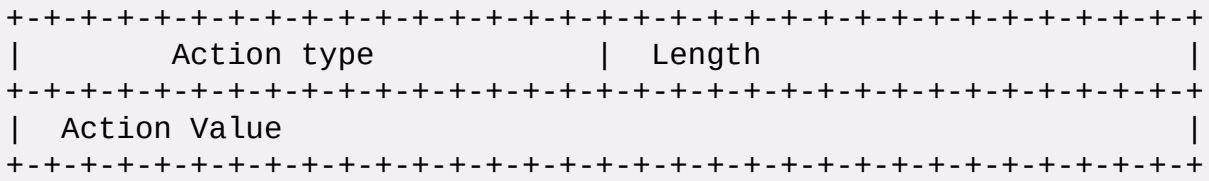
C = 1 - Transitive across Confederation boundaries
C = 0 - Non-transitive across Confederation boundaries
T = 1 – Transitive across AS boundaries
T = 0 – Non-Transitive across AS boundaries

Action TLV + SubTLV Format

Common Header for Action TLVs (figure 2-2)



Each Action SubTLV has the format:



Redirect Discussion (Table 2-1)

ID	FSv2H-L	Action	Abbrev.	Draft
8	0x80-08	Redirect in various forms	RD	[this document]
		to VRF (2 AS form)	RDIPvrf	RFC8955
8	0x81-08	to VRF (IPv4 form)	RDIPvrf	RFC8955
8	0x81-08	to VRF (4 AS form)	RDIPvrf	RFC8955
8	0x01-0C	to IPv4 / copy	RDIPv4C	RDIP
8	0x000C	to IPv6 / copy	RDIPv6C	RDIP
8	0x000D	to VRF (IPv6 form)	RDIP6vrf	RFC8956
8	0x09-xx	to Indirection ID	RGIDC	RGID

Action TLVs for Community Path Attribute

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

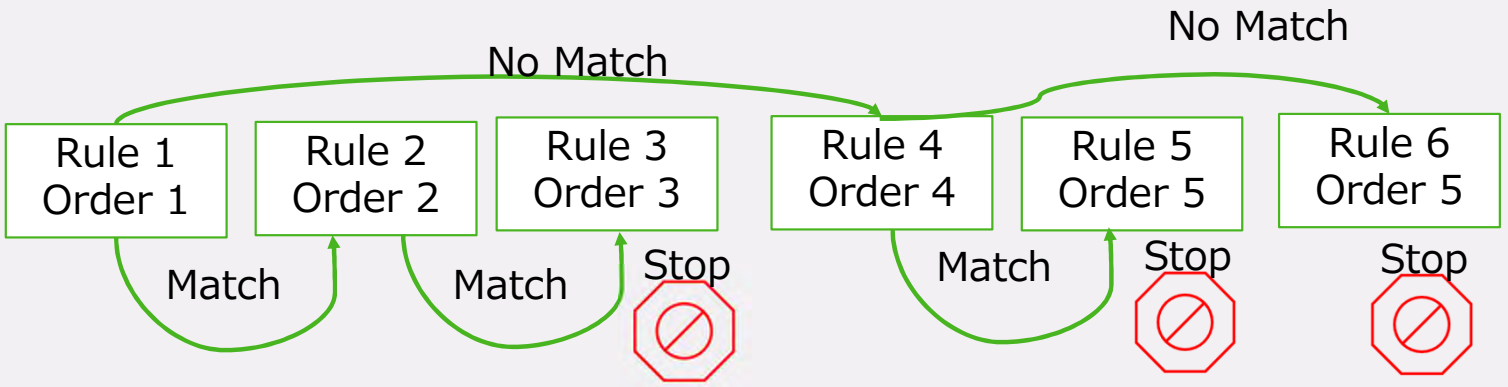
act-id	Name	Description	Document
TBD	MatchSet	Match and Set attribute	[IDR-rpd] (type = 03)
TBD	MatchNoA	Match and No Advertise	[IDR-rpd] (type = 04)
TBD	DetLat	Deterministic Latency action	[PD-detnet-flowmap] (type = 37)
TBD	TSNMap	Map flow to TSN stream	[PD-detnet-flowmap] (type = 38)

Extra slides

Filter dependency

Filter dependency chain logic

- Chain on Rule (User Order, simple)



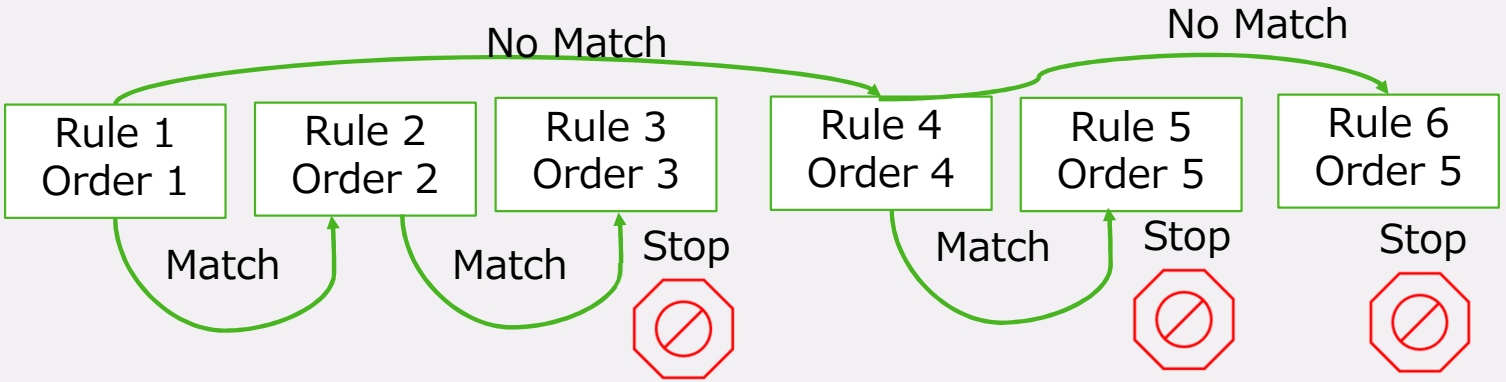
Complex rules: (wait until reason)

- User Order (chain) + Rule component chain (?)
- Conditional actions

Example 1: Packet rate limit (DDOS)

Rule 1: match Loc
Rule 2: match Destination Port 30
Rule 3: match Source Port 20
Action: Drop

Rule 4: match IP Dst-addr: 192.3.4/24
Rule 5: match Source Interfaces: if-2, if-3
Action: copy and drop
Rule 6: match IP Src-Addr: 192.5.2.1
Action: Drop



Example 2: 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**.

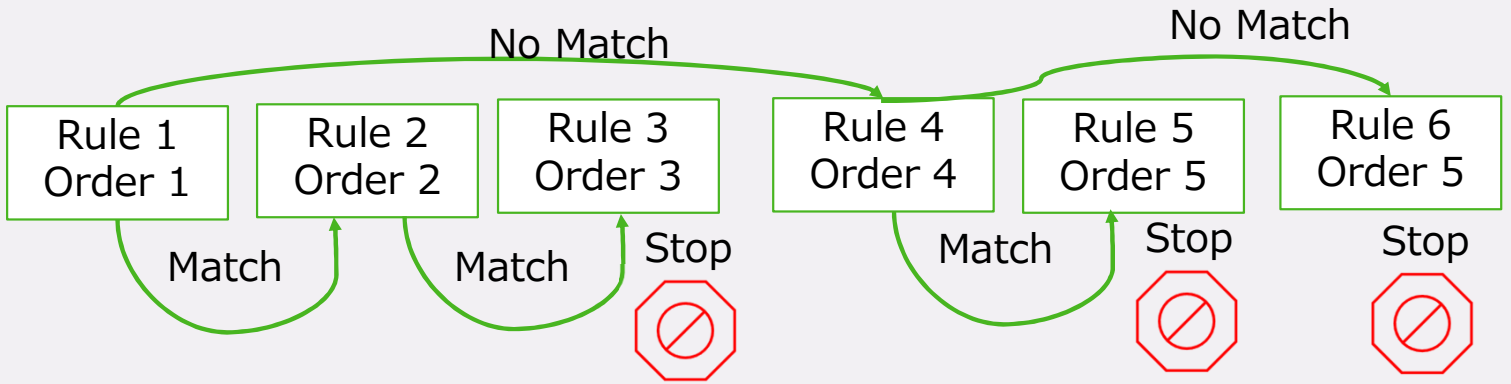
  Rule 400 will not match the unmodified packet, if we apply actions **after all rules**.

- Should we modify packets as soon as the match occurs?

Example 3: SR Header + NRP Example

Rule 1: Match Loc: SID1, FCN: End.X,
Arg: 128.2.1.1
Rule 2: match NRP-ID 10 in Hop by Hop
Rule 3: match Source Port 10
Action: Drop

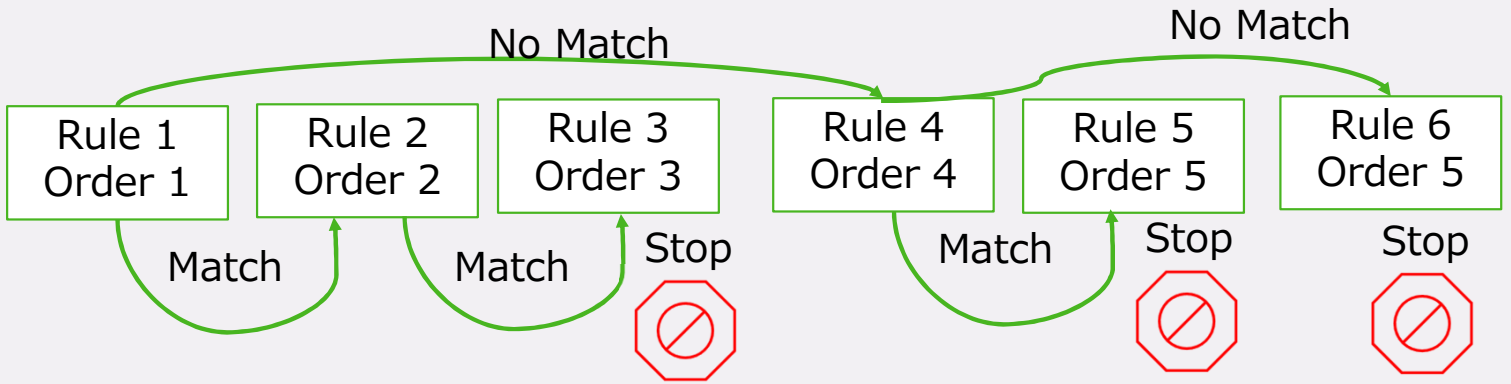
Rule 4: Match Loc: SID1, FCN: End.X,
Arg: 128.2.1.2
Rule 5: match NRP-ID 20 in Hop by Hop
Action: Copy and redirect to IP
Rule 6: Match IP Address 192.5.2.1
Action: drop



Example 4: Color + IP Prefix to place on Tunnel (SR or IP-sec)

Rule 1: Match Color: 20 (blue)
Rule 2: Match Dest Address: 128.2/16
Rule 3: Source Port 10
Action: redirect to 192.5.2.5

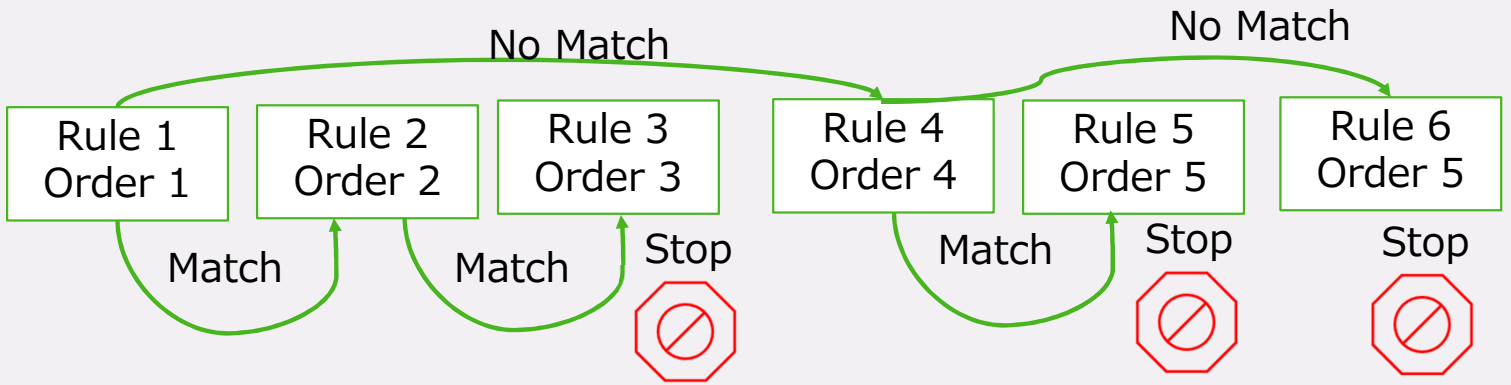
Rule 4: Match Color: 30 (gold)
Rule 5: Match Dest Address 129.4/16
Action: redirect to IP-sec tunnel
Rule 6: Match Color: 50 (red)
Action: drop



Example 5: Deep packet inspection

Rule 1: Match Destination Port (App-1)
Rule 2: Match Payload-1
Rule 3: Match Payload-2
  Action: copy and drop

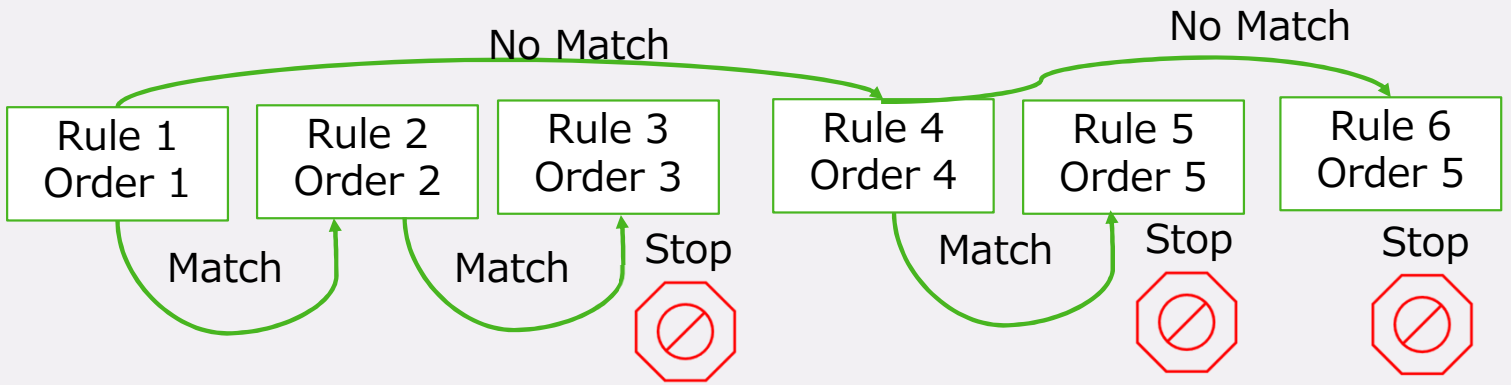
Rule 4: Match Source Port (attack-1)
  Action: rate limit
Rule 5: Match Payload-3
  Action: copy and drop
Rule 6: Match Source Port 2 (attack-2)
  Action: rate limit + forward



Example 6: Other data

Rule 1: Interface Group 1
Rule 2: AS 120
Rule 3: Mtach Prefix list
  Action: drop

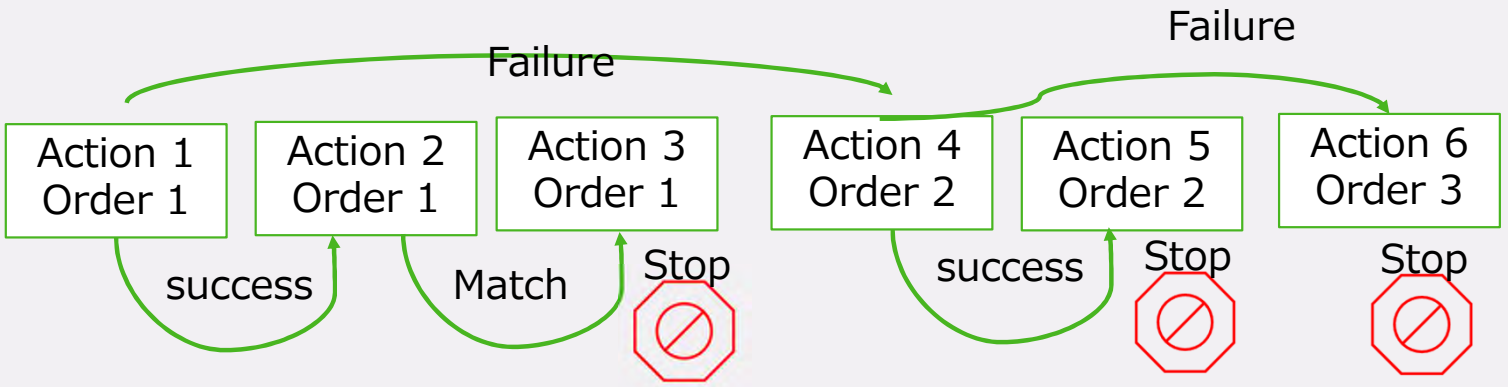
Rule 4: Match Group-ID 1
Rule 5: Match Subgroups 1-3
Action: rate limit, mark NRP-10, redirect to SID1
Rule 6: Match Group-ID 2
Action: mark NRP-20, redirect to SID2



Action dependency

Action chain logic

- Chain on Rule (User Order, simple)



Complex rules: (wait until reason)

- User Order (chain) + Rule component chain (?)
- Conditional actions

Questions