FlowSpec MPLS Match

draft-yong-idr-flowspec-mpls-match-00

Lucy Yong, Sue Hares, Qiangdeng Liang, Yinjie You @huawei

April 2016, Buenos Aires

Why this proposal?

- MPLS is widely used
- For value added services, valuable to have BGP-FS policy filter that matches on the MPLS portion of a packet and take an action on matched packets

- Use cases: 1) matching n tuple is more complex than matching a label. Rate limiting on a flow, flow monitoring,

  2) label action (liang's label action)

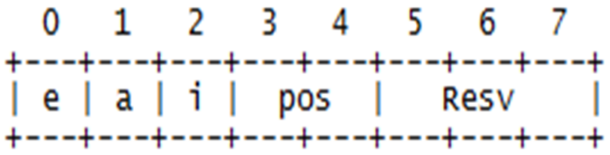
FlowSpec Encoding for MPLS Match

Function: The match1 applies to MPLS Label field on the label stack.

Encoding: <type(1 octet), length(1 octet), [operator,value]+>.

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

The operator byte is encoded as:



```
  0   1   2   3   4   5   6   7
+---+---+---+---+---+---+---+---+
| e | a | i |  pos  |    Resv   |
+---+---+---+---+---+---+---+---+
```

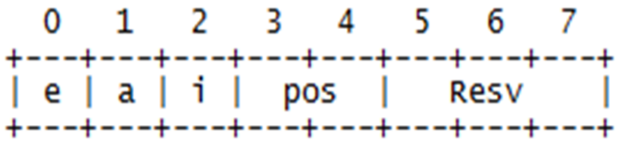
where:

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

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

i - before bit: If unset, apply matching filter before MPLS label data plane action; if set, apply matching filter afterMPLS label data plane action.

FlowSpec Encoding for MPLS Match



```
  0   1   2   3   4   5   6   7
+---+---+---+---+---+---+---+---+
| e | a | i |  pos  |    Resv   |
+---+---+---+---+---+---+---+---+
```

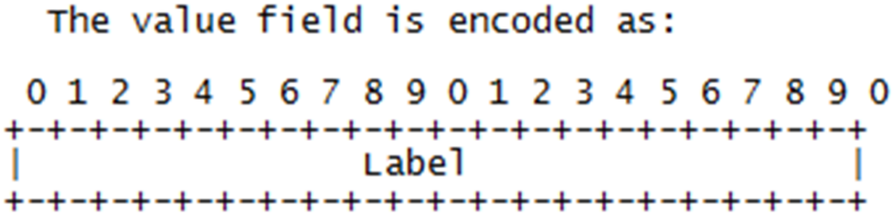
pos - the label position indication bits: where:

00:any position on the label stack - the presented label value is used to match any label on the label stack. When apply it, at least one label on the stack match the value

01: top label indication-   the presented label value MUST be used to match the top label on the label stack.

10: bottom label indication-   the presented label value MUST be used to match the bottom label on the label stack.

11:   (for reserved labels?)



```
 The value field is encoded as:

 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                 Label                   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

FlowSpec Encoding for MPLS Match

Type TBD2 - MPLS Match2

Function: MPLS Match2 applies to MPLS Label experiment bits (EXP) on the top label in the label stack.

Encoding: <type (1 octet), [op, value]+>

[op,value] - Defines a list of {operation, value} pairs used to match 3-bit TOS field on the top label of the stack [RFC3032].

Value:

```
  0   1   2   3   4   5   6   7
+---+---+---+---+---+---+---+---+
|  Reserved (Zero)|   TOS       |
+---+---+---+---+---+---+---+---+
```

Next Steps

- Welcome comments and suggestions
- Update the protocol specification

BGP Flow Specification MPLS Action

draft-liang-idr-flowspec-mpls-action-00

Qiandeng Liang (liangqiandeng@huawei.com)
Susan Hares (shares@ndzh.com)
Jianjie You (youjianjie@huawei.com)
Robert Raszuk (robert@raszuk.net)
Dan Ma (danma@cisco.com)

IETF95 Buenos Aires

Status of this I-D

◆ This draft originates from:
https://datatracker.ietf.org/doc/draft-liang-idr-bgp-flowspec-label/

◆ First presented at IETF 93, Prague meeting; presented again at IDR interim (10/26/2015) meeting

◆ The update compared to draft-liang-idr-bgp-flowspec-label-01

- Clarify the use case, and add example of use
- Define “order” in the label-action

FlowSpec Label Action

A new label-action is defined as BGP extended community value based on Section 7 of [RFC5575].

type	extended community	encoding
TBD1	label-action	MPLS tag

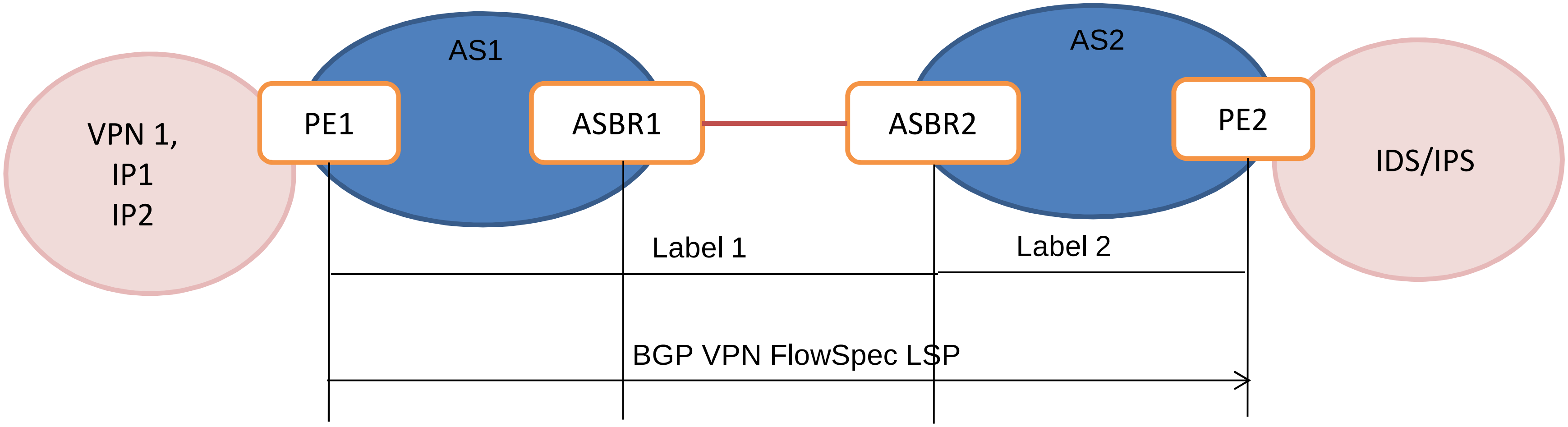
Label-action is described below:

```
 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  (TBD1)             |OpCode |Reserve|   order       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+ Label
|                Label                  | Exp |S|       TTL     | Stack
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+ Entry
```

- Type: indicates the label action
- OpCode: operation code; 0: Push; 1: Pop; 2: Swap; 3-15: Reserved
- Order: If multiple label-actions occur, this field gives the order of this action within that group.
- Label Stack Entry: the same as defined in RFC3032

Deployment Example 1

— MPLS Filter + MPLS Action

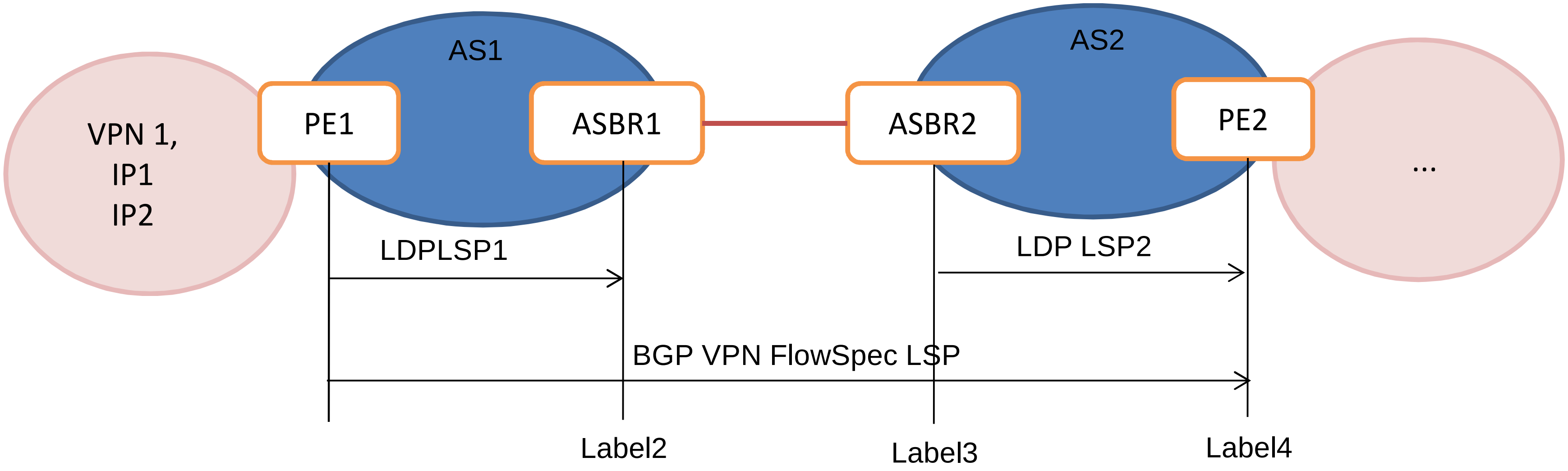


Forwarding information for the traffic
for source: IP2, Destination: IP1

Purpose of BGP-FS filters: send DDoS traffic to IDS/IPS server
PE1: in(<IP2,IP1>) --> out(Label1)
ASBR1: in(Label1) --> out(Label1)
ASBR2: in(Label1) --> out(Label2)
PE2: in(Label2) --> out(--)

Deployment Example 2

— IP Filter + MPLS Action



Forwarding information for the traffic from IP1 to IP2 in the Routers:

PE1: in(<IP2,IP1>) --> out(Label2)
ASBR1: in(Label2) --> out(Label3)
ASBR2: in(Label3) --> out(Label4)
PE2: in(Label4) --> out(--)

Labels allocated by Flow policy process

Label4 allocated by PE2
Label3 allocated by ASBR2
Label2 allocated by ASBR1

Next Step

- Accepted as WG doc?

Thank You!