86th IETF – Orlando, USA

draft-asghar-pim-explicit-rpf-vector-01

J. Asghar	jasghar@cisco.com
IJ. Wijnands	ice@cisco.com
S.Krishnaswawy	sowkrish@cisco.com
V. Arya	varya@directv.com

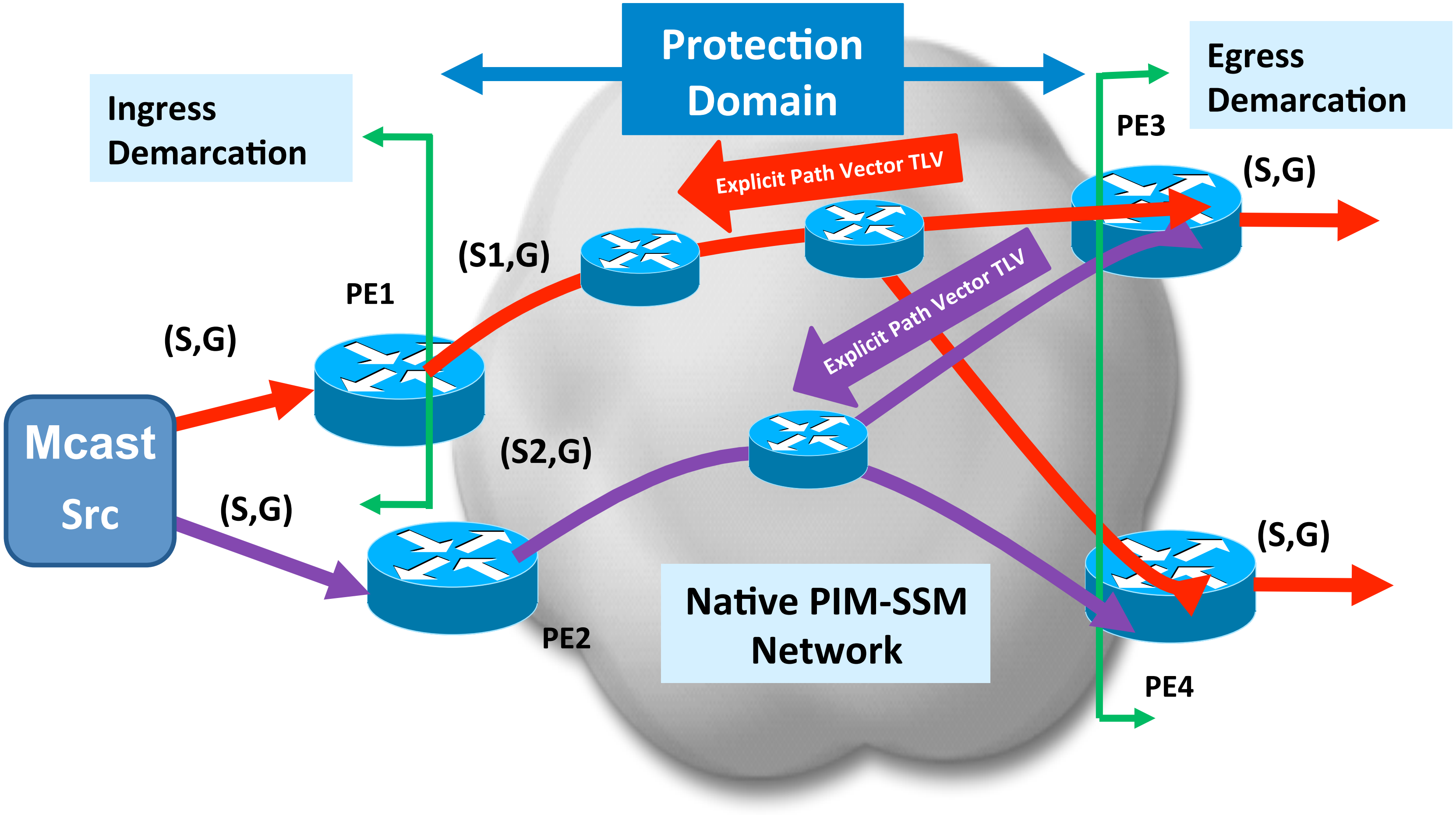
Editorial Changes 00 -> 01

- **Section 3.0** – (Update) Added more details on pim-path-vector-tlv requirement in Video Transport Network use case where live-live resiliency model utilized with multicast tree path diversity requirement
- **Section 4.0** – (Update) Condensed figure and text
- **Section 5.0** – (New) Added vector attribute value
- **Section 6.0** – (New) Added handling of conflicting RPF vectors (as in RFC5496)
- **Section 7.0** – (Update) Explicit RPF Vector Attribute TLV Format

Problem Statement

- This draft documents a solution to build multicast trees via an explicitly configured path sent in the PIM join
- Describes a special use of the Reverse Path Forwarding (RPF) Vector TLV as defined in [RPC 5496]

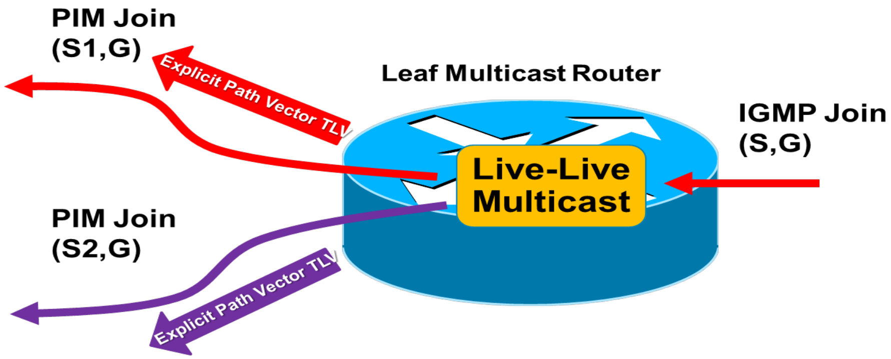
Solution Requirement:
Path Diversity in Live-Live Resiliency



Motivation behind this draft

- **A stack of RPF vectors can be specified to route PIM Joins semi-explicitly using the neighbor addresses:**
  1. However, upon a link/node failure the addresses within a stack of RPF vectors could be unreachable
  2. In this case, router will perform a RIB unicast source reachability lookup and route the PIM Join around the link/node failover and not use the desired RPF vector stack path
  3. In a live-live multicast network or Ring topology, both disjoint multicast trees could be routed along the same path, and not longer be disjoint
- Our draft addresses these issues by proposing a new encoding method that allows to explicitly route PIM Joins using Explicit RPF Vector TLV Stack:
  - **draft-asghar-pim-explicit-rpf-vector-01**

Maximally Redundant Trees

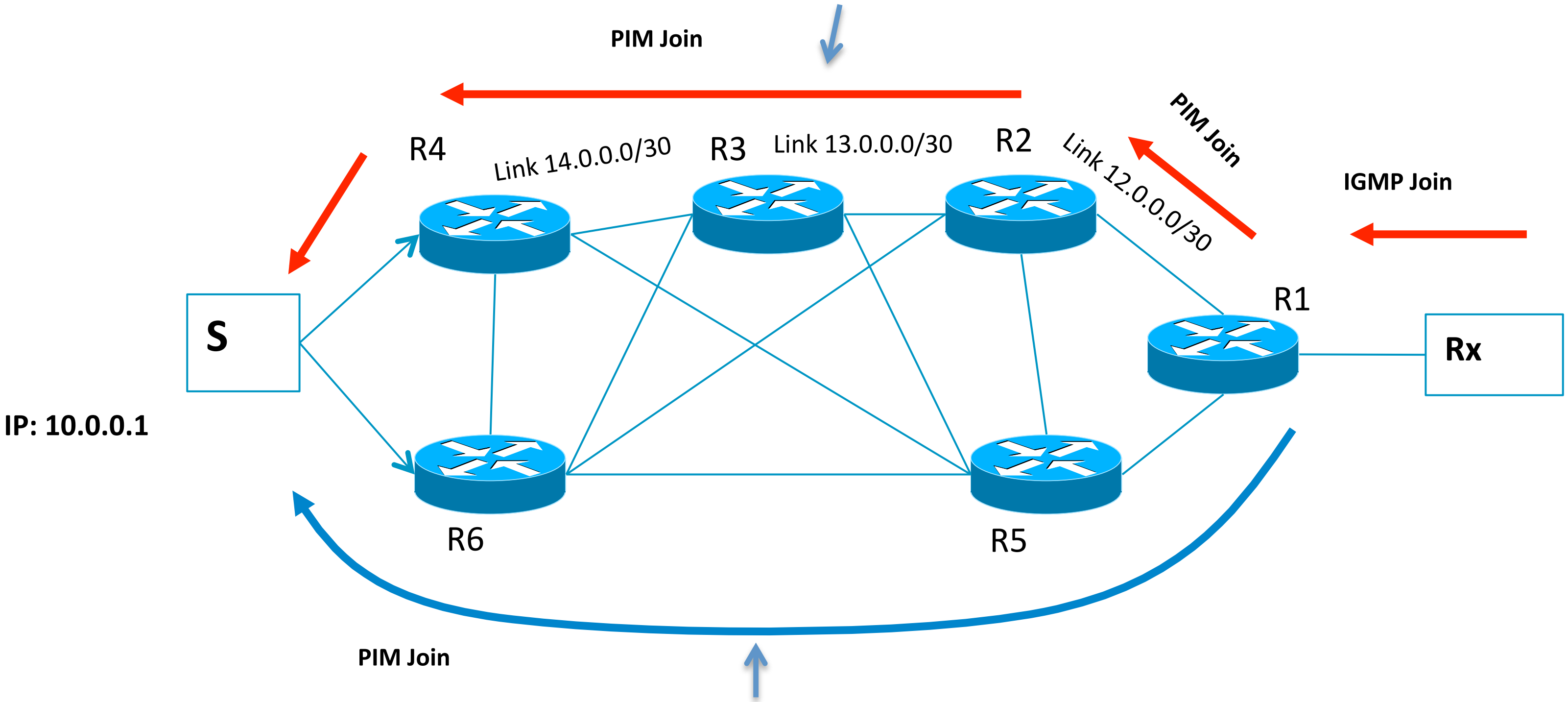


Solution Example (this draft):

Explicit Path Vector TLV Stack

1. Multicast Source IP: S = 10.0.0.1
   - R2: 12.0.0.1
   - R3: 13.0.0.1
   - R4: 14.0.0.1

Explicitly routed path for PIM Join using RPF vector TLV stack



RIB RPF computed path for PIM Join

Solution (this draft)

- Multicast join path R4->R3->R6->R5->R2->R1, where the multicast JOIN is explicitly routed to the source hop-by-hop using the explicit RPF vector list

```
[S]---(R1)--(R2)---(R3)--(R4)---[R]
       <---   |      |  ---
          |   |      |  |
          | (R5)---(R6) |
          - (S,G) Join -
```

Encoding

- RFC-5384 PIM Join Attributes
  - Established IANA registry for join attribute types

    0: RPF Vector TLV [RFC5496]

    “Loose path vector / ERO”

    1: MVPN Join Attribute [RFC6513]

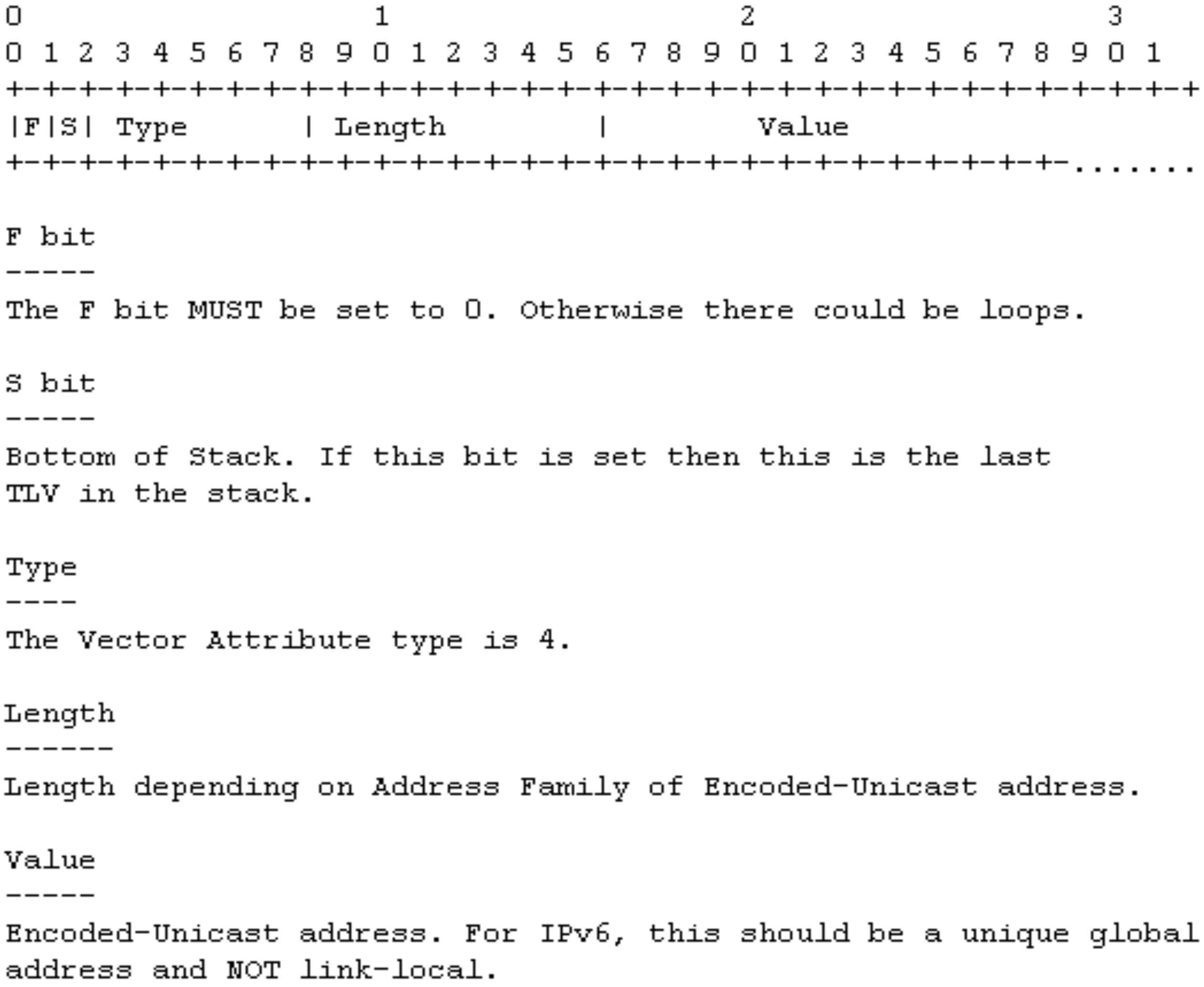
    2: MT-ID Join Attribute [RFC6420]

    3: Pop-Count [RFC6807]

    4 (tentative target): This draft

    “Strict path vector / ERO”

Explicit RPF Vector Attribute TLV 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|F|S| Type      | Length        |       Value
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-.......

F bit
-----
The F bit MUST be set to 0. Otherwise there could be loops.

S bit
-----
Bottom of Stack. If this bit is set then this is the last
TLV in the stack.

Type
----
The Vector Attribute type is 4.

Length
------
Length depending on Address Family of Encoded-Unicast address.

Value
-----
Encoded-Unicast address. For IPv6, this should be a unique global
address and NOT link-local.
```

Moving forward

- Looking for your feedback