BGP MultiNexthop Attribute
- Status -

https://datatracker.ietf.org/doc/draft-ietf-idr-multinexthop-attribute/03/

IETF IDR Interim Meeting (interim-2024-idr-12)

Sep 23 2024

Kaliraj Vairavakkalai

Juniper Networks

(behalf of co-authors)

Agenda

- Background and Problem statement.
- MultiNextHop Attribute – quick recap.
- Changes to the draft – since IDR interim(Jan 29, 2024).
  - Interaction with Fwd info carried in Rest of update
  - Error Handling, M bit
  - Continuity Detection, C, E bits
  - Accumulated Metric
- Next steps.

Background: Expressing nexthops in BGP (1/2)

What is a Nexthop?

An "Instruction" on how to forward a payload to entities specified in BGP NLRI.

What comprises that Instruction?

- Endpoint Identifier
  - Where to forward?
- Encap to use
  - Label stack, SID, etc.
- Constraints:
  - Proximity check (Singlehop/Multihop).
  - Color of path to use.
- Endpoint Properties:
  - Bandwidth.
  - AIGP.

Background: Expressing nexthops in BGP (2/2)

How do we convey this Instruction in BGP Update/Route today?

- **Endpoint Identifier**
  - Nexthop attribute (code 3)
  - MP_REACH_NLRI attribute (code 14) : “Network Address of Next Hop”
  - Redirect to IP extended community attribute.
  - Tunnel Encap Attribute (Remote EP).
  - Color-only extended community attribute.
  - Redirect to VRF extended community attribute.
  - NHC Attribute (NNH)
- **Constraints:**
  - Proximity check: Singlehop/Multihop config
  - Color community or Mapping community attribute.

- **Encap to use:**
  - MP_REACH_NLRI attribute (code 14) : “Label in NLRI portion”
  - Prefix-SID attribute.
  - Tunnel Encap Attribute.
  - Repair-Label attribute.
  - Secondary-Label attribute.
  - FSv2 Redirect to * actions.
  - ELC attributes
  - NHC Attribute
- **Endpoint Properties:**
  - Link bandwidth extended community attribute.
  - AIGP attribute
  - NHC Attribute

Spread across multiple Attributes and Extended Communities.

Problems: current way of expressing nexthops in BGP

- Forwarding Info not scoped in one place.
- Inability to advertise more than one nexthop in a route.
- Not easily extensible to newer endpoint types, encapsulation types.
- Addpath unable to express relationship between different nexthops (active/backup, UCMP etc), and is Scaling Heavy.
- Inability to signal encap-information uniformly across families

  (e.g. cannot signal Labels for SAFI 1 routes or Flowspec route)
- Inability to signal additional label stack for repair path in a route.

  (helpful in some multihomed cases to avoid label oscillation, forwarding loop)

These problems are solved by MultiNexthop Attribute.

MNH: bird's eye view

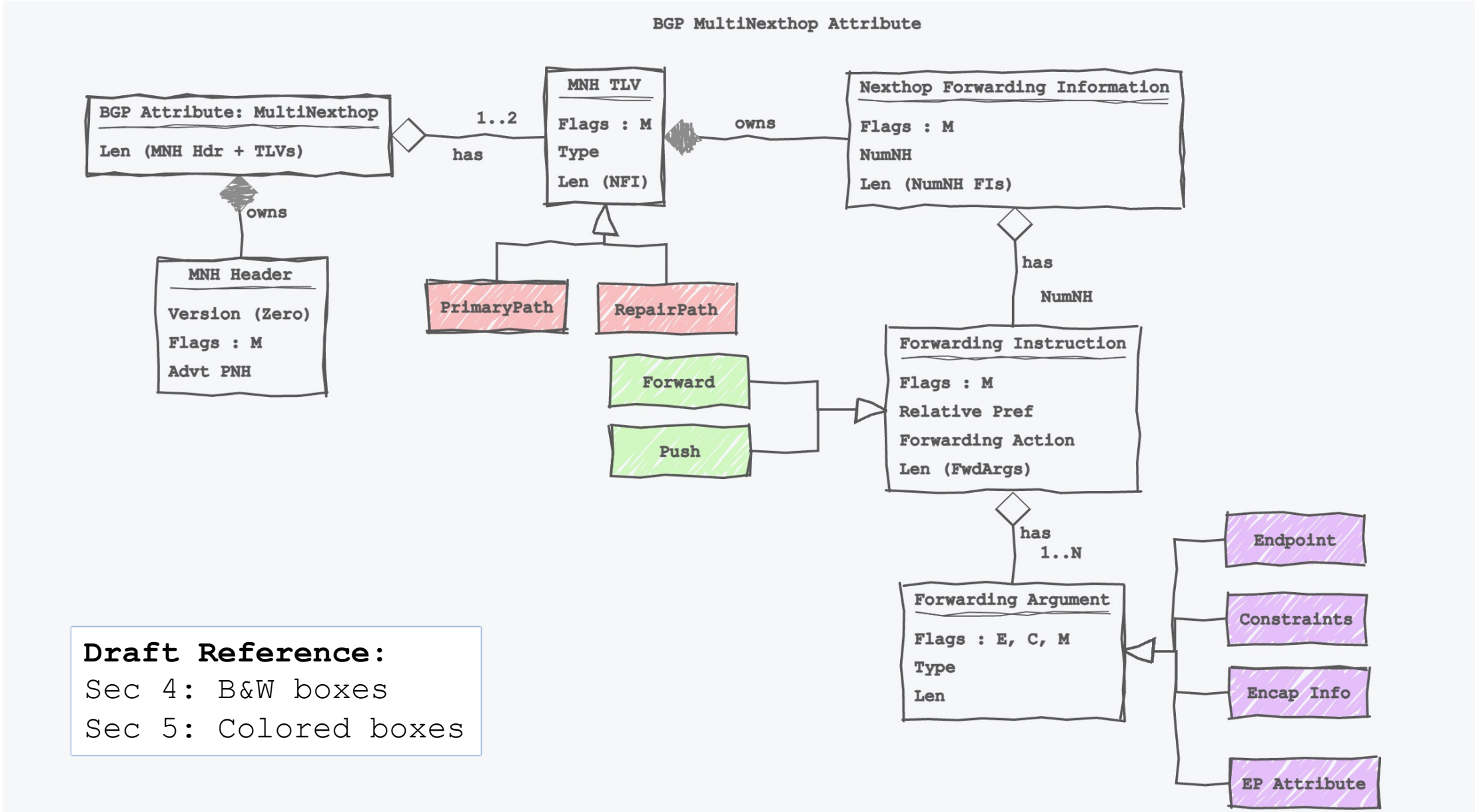
```
MNH Attribute: {
    PrimaryPath {
        [Forwarding Instruction 1],
         ..
        [Forwarding Instruction n]
    }
    RepairPath {
        [Forwarding Instruction 1],
         ..
        [Forwarding Instruction n]
    }
}
```

```
Forwarding Instruction : {
    FwdAction, FwdArguments
}
```

Changes to the draft – since IDR interim(Jan 29, 2024)

- Post Adoption, renamed draft from draft-kaliraj-* to draft-ietf-*
- Incorporated review comments received during Adoption call
- IDR GitHub Repo created for MNH
  - https://github.com/ietf-wg-idr/draft-ietf-idr-multinexthop-attribute
  - Uploaded draft, uml diagram and issues discussed during adoption-call to github
- Reorganization of draft :
  - Section 4 - Base Encoding and Protocol Procedures
  - Section 5 – Specific TLVs defined in this document.
  - Removed Label Descriptor TLV – will take it up in separate document, based on interest.
- Additions to draft:
  - MNH: interaction with Fwd info carried in Rest of update. (Sec 4.1.4)
  - Error handling, using M bit.
  - Procedures for continuity detection for any fwd-arg carried in MNH across NHS propagation. (Sec 4.5)
  - ‘Accumulated Metric’ as an endpoint attribute defined by this document (Sec 5.3.4.2).
  - Use-case A.8.2, another flavor of ‘multihomed-PEs protection each other’ problem, Per-table-label instead of per-nexthop-label.

MNH: UML view



Draft Reference:
Sec 4: B&W boxes
Sec 5: Colored boxes

MNH: interaction with Fwd info carried in Rest of update

- Forwarding Info or attribute carried outside the MNH is applicable to the Nexthop (in attr code 3, 14)

- Forwarding Info or attribute carried inside MNH is more-specific and applies to associated NH-Leg Endpoint.
  - So it overrides the info carried outside MNH in same Update message.
  - Exception: the MPLS Label in 8277 NLRI is used as inner-label in conjunction with MNH Encap info if-present used as outer encap.

MNH: Error Handling, using M bit

M bit – “Is Mandatory?”

0: element is optional, any errors are ignored , and continue processing.

1: element is mandatory, any errors are percolated up

Unrecognized type or Error in:	M bit value	Handling
MNH Attribute	1	If all MNH TLVs report error, Hide containing Route. RFC7606 “Treat-As-Withdrawal”.
	0	If all MNH TLVs report error, ignore attribute. RFC 7606 “Attribute Drop”
MNH TLV	1	If NHI report error, percolate up to ‘MNH Attr’
Nexthop Forwarding Info	1	If all FI TLVs report error, percolate up to ‘MNH TLV’
Forwarding Instruction	1	If any error while forming Nexthop Leg, percolate up to ‘Nexthop Forwarding Information’
Forwarding Argument	1	If arg is not recognized or an error encountered, Percolate up to ‘Forwarding Instruction’

MNH: Continuity Detection (1/2)

- Uses C, E bits in Forwarding Argument TLV (Sec 4.5).

Bit	Name	Procedure
C bit	Cumulative/Contiguous	• Set to 1 by node attaching the FA. • If set, intermediate nodes doing NHS accumulate value in re-advertised MNH. • By default Forwarding Arguments are not cumulative, C bit is 0 unless otherwise specified by the forwarding argument type.
E bit	Egress Node Attached	• This bit is maintained when C bit is set to 1. • E bit is set to 1 if a cumulative argument is being added to a route with empty AS-path.

MNH: Continuity Detection (2/2)

- F.A. type “Accumulated Metric” sets C bit to 1. E bit allows determining whether it was attached at Egress.

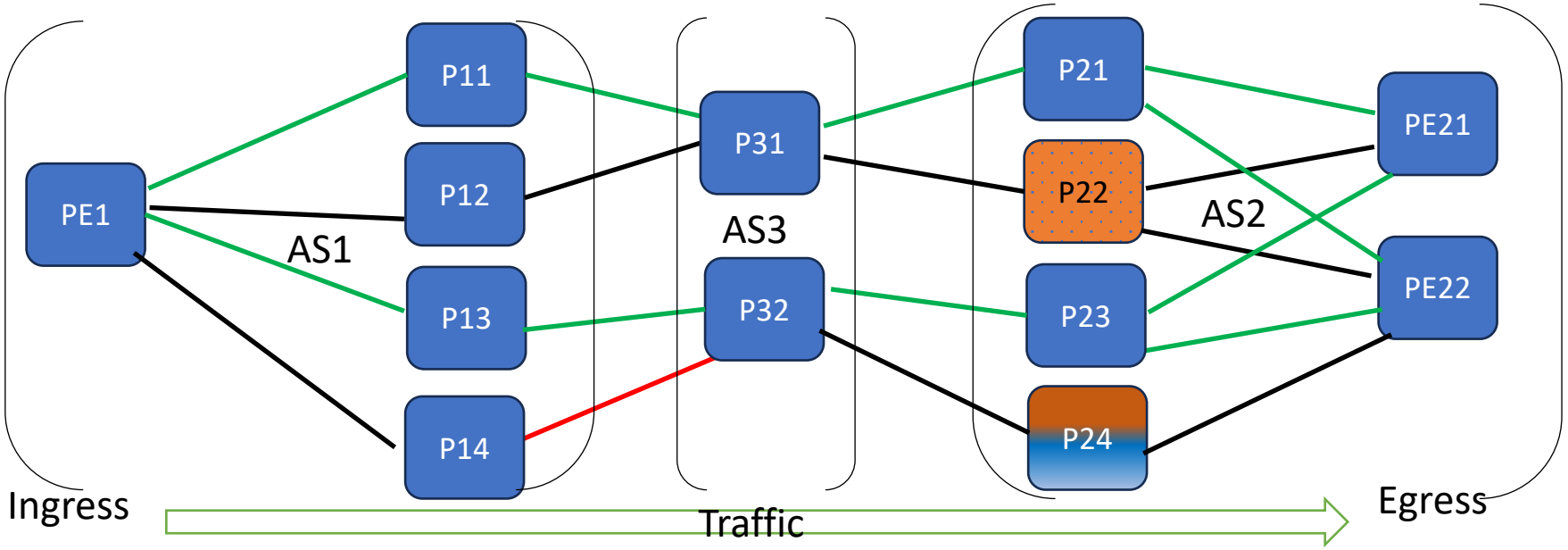
- Any other Forwarding Argument in future that needs continuity detection can use this MNH base infra without re-inventing it:

- F.A. type EP-Bandwidth(Sec 5.3.4.1) could potentially use C,E bits too. This can differentiate whether the bandwidth is upto Egress router or an intermediate hop router.
- F.A. type Path Constraints (Sec 5.3.2) can use C bit too, to hold the proximity/transport-class constrain thru out the route readvertisement path.

MNH: Accumulated Metric (Sec 5.3.4.2)

- Accumulated Metric in MNH allows subset of "IGP Metric-Type" registry:
  - Includes 'IGP cost' and 'Link Delay'
  - Does not include TE-metric and Bandwidth, since those interactions need to be carefully specified.
- Reason why MNH uses different encoding for Ametric, and does not reuse the NHC:Ametric-data one.
  - Above point about TE-metric, Bandwidth in 'IGP Metric Type' registry
  - NHC:Ametric has the continuity detection in the Ametric-data part, not the NHC part.
  - NHC:Ametric is not a TLV format (doesn't have length field). All MNH FAs use TLV format.
- Discussion point: Comment on MNH:Ametric divergence from "IGP Metric-Type" registry:
  - Received suggestion to create a separate registry, decoupled from IGP Metric-Type.
  - Pondering on whether F.A. type "Accumulated Metric" should be renamed to just "AIGP cost",
  - And a new F.A can be created for 'Latency/Link Delay'. Such that these remain top-level MNH Forwarding Arguments.

MNH: Accumulated Metric – continuity detection example



Scenario: SAFI-4 Route for PE21/32 originated at PE21 or P21. Similarly for PE22. NHS at each BGP hop.

- MNH:Ametric if added at P11 or P31 level will not have E bit set, as AS-path is not empty.
- MNH:Ametric if added at PE21 or P21 level will have E bit set, as AS-path is empty.
- P22, P24 are pruned from path because they don't understand MNH or MNH:Ametric. *Unknown FAs not propagated at NHS.*
- MNH received at PE1 via P14 will not have the Ametric, because P12 could not determine 'Latency' to P32.
- MNH:Ametric added at PE21 and received at PE1 via {P11, P31, P21} and {P13, P32, P23} will have both E bit and C bit set.
- *This way, PE1 discovers Two Paths to PE21 with known end-to-end latency.*

MNH Config Model - Example

Per Family config:

```
family inet unicast {
    mnh {          /* enable MNH Rx, Tx */
       dry-run;    /* send M bit = 0 */
    }
}
```

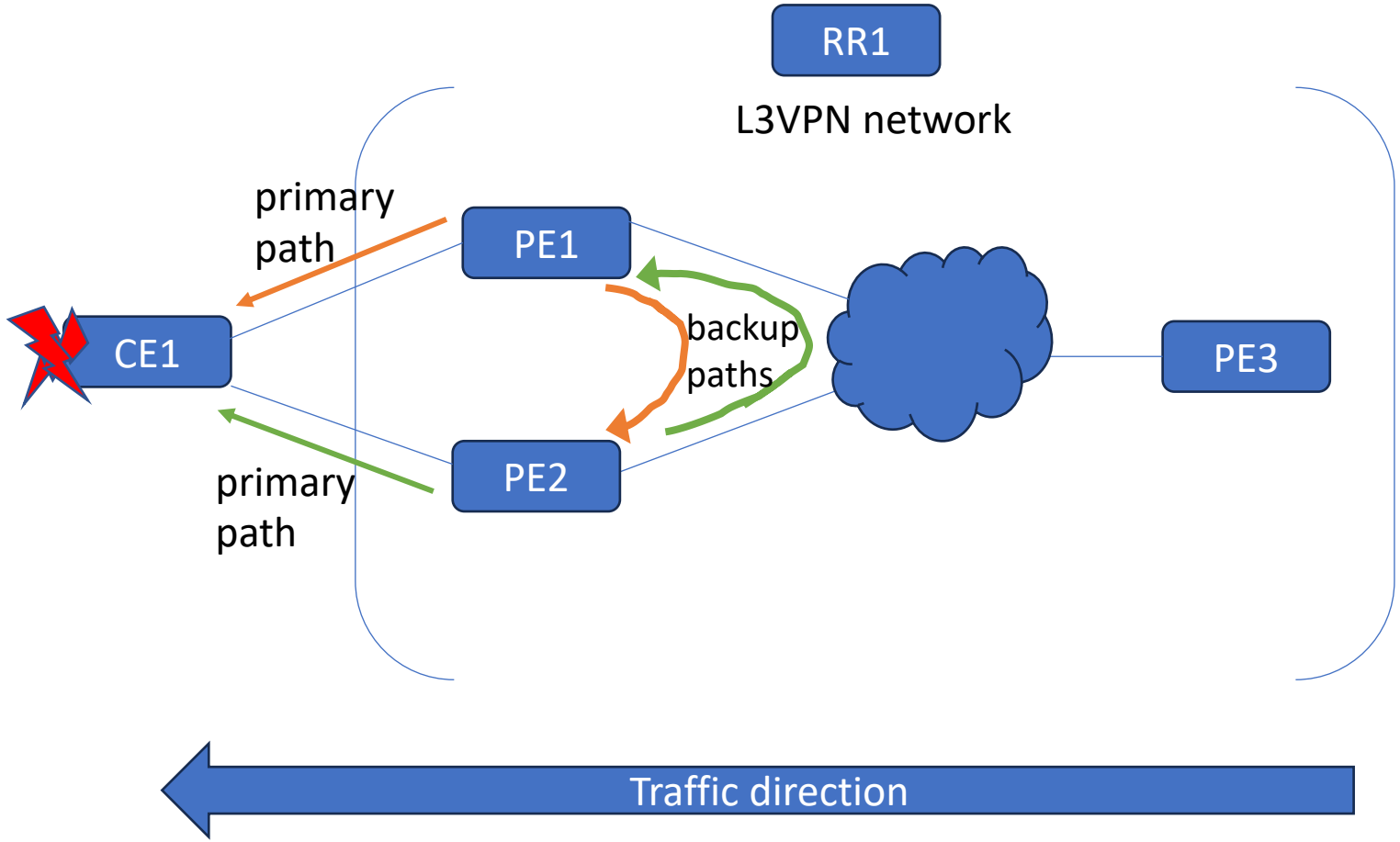
Policy Set Config:

```
    term T1{
        from family inet6;
        then {
           mnh {
                explicit-null;
                dry-run;
           }
         }
     }
```

Policy Match Config:

```
    term T2{
        from {
           mnh {
               end-point ip <a.b.c.d>;
               has-repair-path;
           }
        }
        then local-preference 200;
    }
```

MNH Use-case: Avoid Forward Loop between Multihomed PEs, Per Table Label (Sec A.8.2)



MNH Layout for Use-case A.8.2

PE1 MPLS FIB:

```
VL11: Pop, Fwd to CE1

VL12: Prim {IP-Lookup}
      Bkp {BackupPath fm PE2}
```

PE2 MPLS FIB:

```
VL21: Pop, Fwd to CE1

VL22: Prim {IP-Lookup}
      Bkp {BackupPath fm PE1}
```

PE1 advertised BGP route:

```
MNH Attribute: {
    PrimaryPath {
        [Push "VL12", "PE1"],
    }
    BackupPath {
        [Push "VL11", "PE1"],
    }
}
```

PE2 advertised BGP route:

```
MNH Attribute: {
    PrimaryPath {
        [Push "VL22", "PE2"],
    }
    BackupPath {
        [Push "VL21", "PE2"],
    }
}
```

- ***Avoids Forwarding Loop between the multihomed PEs.***
- ***BackupPath Label points to only primary CE paths. No IP lookup.***

Next Steps

- ❑ Work on Implementation.
- ❑ Try out with customers who are eager to experiment for certain use-cases
- ❑ Test and iterate
- ❑ Incorporate learnings and any changes to the draft.

Thank you.