Status

• Reviewed by two NVO3 Interim meetings
• Received a lot of comments with regard to how NVE expressing interested VNss.
• A new subTLV (Enabled-VN TLV) under the IS-IS Router Capability TLV [RFC4971] is specified here for NVE to indicate all its interested VNss in the IS-IS LSP message
• Comparing with OVSDB (Open vSwitch DB Management) mechanism
• removing the IS-IS portion out of the NVA-NVE mapping control plane,
• make the draft specifically focus on the actual data (TLV based data models) and handshakes to be exchanged between NVA and NVE
NVA-NVE Mapping distribution: Push Model

- **Requesting Push Service:**
  - Push NVAs use VN scoped messages to announce their availability to push mapping information.
  - NVEs use VN scoped reliable messages to announce all the Virtual Networks in which they are participating.
  - Whenever, there are changes in the mapping entries, NVA only sends the changed portion of the entries.

- **Policies:** When ingress edge can’t find entries for the incoming data frame:
  - Simply drop the data frame,
  - Flood it to all other edges that are in the same VN, or
  - Start the “pull” process to get information from Pull NVA.
bitMap to express interested VNs subTLV

<table>
<thead>
<tr>
<th>INT-VN-TYPE-1</th>
<th>(1 byte)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Length</td>
<td>(1 byte)</td>
</tr>
<tr>
<td>Start VN ID</td>
<td>(4 bytes)</td>
</tr>
<tr>
<td>VNID bit-map..</td>
<td></td>
</tr>
</tbody>
</table>

Figure 2. Enabled-VN TLV using bit map
Range to express interested VN
t

```
+----------------------------+
|   INT-VN-TYPE-2            |
+----------------------------+
|       |  Start VN ID          |
+----------------------------+
|       |  End VN ID            |
```

(1 byte)

(1 byte)

(4 bytes)

(4 byptes)

Figure 3. Enabled-VN TLV using Range
List to express interested VNs

```
+---------------------------+   +---------------------------+
|   INT-VN-TYPE-3           |   |   Length                  |
+---------------------------+   +---------------------------+
| VN ID                     |   | VN ID                     |
+---------------------------+   +---------------------------+
| VN ID                     |   | VN ID                     |
+---------------------------+   +---------------------------+
| VN ID                     |   | VN ID                     |
+---------------------------+   +---------------------------+
                        . . .
```

Figure 4. Enabled-VN TLV using list
Incremental Push service

- A new TLV is needed for to carry NAMD timeout value and a flag for NVA to indicate it has completed all updates.

Figure 3. NAMD Complete TLV

Flags: A byte of flags defined as follows:

```
+---+---+---+---+---+---+---+---+
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
+---+---+---+---+---+---+---+---+
```
Reachable Interface Addresses (IA) TLV

- To advertise a set of addresses within a VN being attached to (or reachable by) a specific NVE
- These addresses can be in different address families. For example, it can be used to declare that a particular interface with specified IPv4, IPv6, and 48-bit MAC addresses in some particular VN is reachable from a particular NVE.
Pull Query Format

- PULL NVA announce its supported VNs
- Pull Requests for the interested VNs or TSs are sent to one specific NVA instance that has the needed information
  - Triggered by:
    - An NVE receives an ingress data frame with a destination whose egress NVE is unknown, or
    - An NVE receives an ingress ARP/ND request for a target whose link address (MAC) or egress edge NVE is unknown.
- Pull Response with instruction on how long entries can be kept by NVE, actions to take if no match is found

![Figure 4. Pull Query TLV](image-url)
PULL Responses

- When the mapping entry is available in the NVA
  - Valid Response
- When the mapping is not available:
  - “drop” or “native-forward” (i.e. flooding)
- cache timer
What if removing the sequence number?
**Pull Response**

RESPONSE: Each RESPONSE record within a Pull NVA Response message is formatted as follows:

```
  0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
 |   SIZE   |   OV   |   RESV   |   Index   |   ...   +---+---+---+---+---+---+---+---+---+---+---+---+---+---+
 |           |         |          |           |         |   Lifetime  |   ...   +---+---+---+---+---+---+---+---+---+---+---+---+---+---+
 |           |         |          |           |         |   Response Data  ...   +---+---+---+---+---+---+---+---+---+---+---+---+---+---+
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
```

---
Push-Pull Hybrid Model

• Push model are used for some VNs, and pull model are used for other VNs.
  – It can be operator’s decision (i.e. by configuration) on which VNs’ mapping entries are pushed down from NVA (e.g. frequently used) and which VNs’ mapping entries are pulled (e.g. rarely used).
  – Useful for Gateway nodes where great number of VNs are enabled.

• Or, a portion of hosts in a VN is pushed, other portion has to be pulled.
Next Step

• NVO3 needs at least one NVA-NVE Control Plane solution:
  – NVO3 charter: Oct 2015 NVE - NVA Control Plane Solution submitted for IESG review
  – NVO3 shouldn’t wait

• Suggest adopt the current draft to NVO3 WG
BACKGROUND INFORMATION
Various ways of NVAs connected to NVEs

**Locations:**
- Embedded in routers/switches in the core, or as standalone servers attached to them.
- Standalone servers or VMs connected to Edges via the client side port

**Contents:**
- Centralized NVA
- Distributed NVA:
  - Each NVA has mapping for a subset of VNs
  - multiple NVAs have mapping entries for a VN