Constrain Attribute announcement within BGP

draft-keyupate-idr-bgp-attribute-announcement-00

Keyur Patel, Jim Uttaro, Bruno Decraene, Wim Henderickx

IETF 94, November 2015, Yokohama, Japan
Motivation

- Currently there is no mechanism to scope the announcements of optional attributes

- The only possible way to filter attributes within BGP are:
  - Unrecognized Optional non-transitive attributes
  - Error handling filters malformed attributes
  - Attribute Specific rules to ensure their scope (Local Pref)

- Need for scoping attributes (atleast) at:
  - Confed boundary
  - AS boundary
  - At Multi-AS administration boundary
Use Case

- **BGP Tunnel Encap attribute**
  - Defined in ietf-idr-tunnel-encaps
  - Scope the Tunnel attribute announcements

- **BGP Nexthop Capabilities Attribute**
  - Defined in draft-decraene-idr-next-hop-capability-01
  - Optional Non Transitive Attribute defines Nexthop’s capabilities

- **BGP Timestamp Attribute**
  - Defined in draft-litkowski-idr-bgp-timestamp-02
  - Carries Timestamps for a given NLRI for each BGP speaker the NLRI traverses

- Any new attributes defined in future…..
Solution

- No use of Capability
  - Adds complexity to protocol

- Define 2 unused bits of Attribute flags:
  - O Optional or a Well-known as defined in [RFC4271] 1st bit
  - T Transitive or Non-Transitive as defined in [RFC4271] 2nd bit
  - P Partial as defined in [RFC4271] 3rd bit
  - E Extended Length type as defined in [RFC4271] 4th bit
  - A AS Wide Scope 5th bit
  - C Member-AS in Confederation Scope 6th bit
  - M Multi-AS Scope 5th and 6th bit

- In order to preserve the bits Multi-AS scope is enabled when 5th and 6th bits are both turned on!
Solution - Rules

- A, C OR M Bits require O bit to be set
- Filtering based on bits must be enforced when a BGP speaker receives or originates a route
- Requires implementation to enforce Enhance Error handling rules for attributes
  - Malformed attributes having impact on route selection or route installation should enforce “treat-as-withdraw” procedure
  - Other Malformed attributes should enforce “attribute-discard” procedure
Alternate Solution 1

- Reserve first 4 bytes of attribute data field for all newly allocated attributes
  - Mark them as flags field
- Defined the scope bits from the reserved flag fields
- Reserve IANA space for new attributes so that implementations modify the attribute code to reserve first 4 bytes as flags field

Only makes sense if more scoping modes are needed
Alternate Solution 2

- Define new attribute for scoping attributes
- Attribute consist of one or more TLVs
  - TLV contains, Attribute type value and its scope
- Modify the code to setup the dependency for attributes

Sets up a dependency with actual attributes! Complicates the code!
Questions?