RFC 7752bis
(an update to BGP-LS specification)

Ketan Talaulikar, Jan Medved (Cisco Systems)
Hannes Gredler (RtBrick)
Stefano Previdi,
Adrian Farrel (Juniper)
Saikat Ray
Why the update?

- Introduction of terminologies to describe the role and processing of BGP-LS information
- To clarify on error handling and fault management aspects
- Clarification on various TLVs, their mandatory/optional nature, verification, etc.
- Clarification on use of Instance ID in BGP-LS
- Description of certain aspects with interpretation and handling of BGP-LS information
- Handling the growth of BGP-LS Attribute
- Introduction of Private Use TLVs
- Other minor editorial changes and updates
BGP Speaker Roles for BGP-LS

• BGP-LS Producer
  • Originates information into BGP-LS
  • R1, R2, Rn

• BGP-LS Consumer
  • Application/process that uses the BGP-LS information; not the BGP Speaker
  • Outside the scope of BGP-LS

• BGP-LS Propagator
  • BGP processing related to propagation of BGP-LS information between BGP routers and performing BGP best path calculation
  • RRm is a pure BGP-LS Propagator

Roles not mutually exclusive; same BGP Speaker can originate, propagate information as well as handoff to a consumer
Link-State NLRIs

• RFC7752 defines 3 types; more introduced by other drafts
• Implementation MUST be able to handle unknown NLRI types in an opaque manner – process and propagate
  • Enables introduction of new extensions
  • Does not require BGP infrastructure (e.g. RR) to be upgraded
• TLVs within NLRI are handled opaquely
  • No semantic validation; only syntactic validation – i.e. length checks
  • Accept, preserve and propagate – unknown, unsupported or “unexpected” TLVs within NLRI
  • Only rule – MUST be organized as ascending order of TLV types (as per RFC7752) and for same type ascending order of value (interpreted as hex string); if not consider malformed
• Semantic checks only by the consumer application; outside the scope of BGP-LS
BGP-LS Attribute

• TLVs within BGP-LS Attribute are handled opaquely
  • No semantic validation; only syntactic validation – i.e. length checks
  • Accept, preserve and propagate – unknown, unsupported or “unexpected” TLVs
  • Need not be ordered by TLV type

• May grow to large size and cause BGP update with single NLRI to exceed 4K
  • Use of BGP extended messages is one option
  • Producer can put limit/check to ensure does not go beyond limit
  • If limit gets exceeded at Propagator then consider malformed and do ‘attribute discard’

• Semantic checks only by the consumer application; outside the scope of BGP-LS
IGP Instance Identification in BGP-LS

• 64 bit Identifier in NLRI is the BGP-LS Instance Identifier
  • Applicable for multiple IGP instances/process on a router
  • Also applicable for multiple IGP instances over the same link
  • Configured at IGP instance level on the Producer
  • Same value needs to be consistently used on all Producers originating from the same IGP routing instance
  • Unique values need to be assigned for different IGP routing instances/domains

• BGP-LS Identifier (TLV 513) is proposed to be deprecated
  • Confusion regarding its use as instance identifier vis-à-vis the 64 bit Identifier field
  • Need feedback from implementors on this
  • Backward compatibility aspects; recommend default value of 0
Fault Management – Overview

• BGP-LS provides transport for link-state information in opaque manner

• BGP implementation is to do
  • Only syntactic checking - e.g. length validations
  • No semantic checking – e.g. existence of TLVs, missing TLVs, unexpected TLVs, etc.
  • No checking for known TLVs as well – e.g. validity of field values, size being correct, etc.

• Consumer of BGP-LS information
  • May do semantic checking and interpretation of TLVs in both NLRI and BGP-LS Attribute
  • Application specific handling of errors (when detected) and outside the scope of BGP-LS
Fault Management – Link-State NLRI

• When error affects the parsing/processing of the rest of the update message then
  • If BGP-LS session isolation is used then session reset, else,
  • If session is used for other AFI/SAFI perform AFI disable (else session reset if disable not supported)

• When error affects only specific NLRI (or group of all Link-State NLRIs) then
  • Process as ‘treat as withdraw’ or ignore/discard when it cannot be parsed
Fault Management – BGP-LS Attribute

• When error affects the parsing/processing of the rest of the update message then
  • If BGP-LS session isolation is used then session reset, else,
  • If session is used for other AFI/SAFI perform AFI disable (else session reset if disable not supported)

• When error affects only BGP-LS Attribute then
  • Process as ‘attribute discard’
  • Continue to propagate the Link-State NLRI without BGP-LS attribute so the consumer detects that there might have been an error (i.e. not to misinterpret that the object does not exist anymore)
  • Enables diagnostic to detect and identify faults

• Entire BGP-LS attribute is discarded; not individual TLVs
Handling Unreachable IGP Nodes

• When BGP-LS Producers continue to advertise link-state objects based on stale LSA/LSPs of unreachable nodes, then a BGP-LS Consumer may get a wrong or inconsistent topology view.

• BGP-LS propagation happens based on BGP best path algorithm which can result in NLRI with stale information being preferred over another with newer and consistent information.

• BGP-LS producer should withdraw link-state objects when the associated node becomes unreachable in IGP processing on the producer node.
Other changes (refer draft for details)

- Ambiguity on MT-ID TLV and it’s usage is clarified
- OSPF Route Type is mandatory TLV for Prefix NLRI from OSPF
- OSPF support for Node Name is updated
- Recommendations on session isolation for BGP-LS
- Introduction of Private Use NLRI and TLV space for vendor specific extensions without resulting in conflicts

And other editorial changes and clarifications ... please run rfcdiff against RFC7752 to get all changes
Next Steps

• Feedback from existing implementations and deployments
• Review, discuss on IDR mailing list
• Feedback/inputs also welcome during IETF in Prague