draft-ietf-lsr-flex-algo

Peter Psenak (ppsenak@cisco.com)
Shraddha Hegde (shraddha@juniper.net)
Clarence Filsfils (cfilsfil@cisco.com)
Ketan Talaulikar (ketant@cisco.com)
Arkadiy Gulko (arkadiy.gulko@thomsonreuters.com)
Changes From Previous Version

• No functional changes
• “Operational considerations” section has been added based on request from Alvaro during IETF 107
  • Inter-area Considerations
  • Usage of SRLG Exclude Rule with Flex-Algorithm
  • Max-metric consideration
Inter-area Considerations

• Describes the FAD area scope
• FAD usage in a multi-area environment
• ISIS FAD re-generation at L1/L2 router from L2 to L1
• OSPF AS flooding scope for FAD
Usage of SRLG Exclude Rule with Flex-Algorithim

• Independent from the SRLG usage for backup path computation
• Creates disjoint sets of paths by pruning the links belonging to a specific SRLG
• Facilitates the usage of already deployed SRLG configurations
Max-metric Consideration

• Existing max-metric functionality is limited to IGP metric
  • Makes link either non-reachable or to serve as the link of last resort
• Similar mechanisms needed for other Flex-algo metric types
  • ASLA Min Unidirectional Link Delay
  • ASLA TE-metric
• Link can be made unreachable by by removing the Flex-Algorithm ASLA Min Unidirectional Link Delay from the link
  • for all Flex-Algorithms that use Min Unidirectional Link Delay as metric
• The link can be made unreachable by by removing the Flex-Algorithm ASLA TE-metric advertisement from the link
  • for all Flex-Algorithms that use TE metric
Max-metric Consideration (cont.)

• The link can be made the link of last resort by setting the value of ASLA Min Unidirectional Link Delay value to $2^{24} - 1$
  • for all Flex-Algorithms that use Min Unidirectional Link Delay as metric

• The link can be made the link of last resort by setting the value of ASLA TE-metric advertisement to the value of $2^{24} - 1$ in ISIS and $2^{32} - 1$ in OSPF
  • for all Flex-Algorithms that use TE metric
Next Steps ...

• Initial version introduced in July 2017
• WG adoption in May 2018
• Multiple implementations available
• Multi vendor interoperability tested
• Draft has been stable for some time
• We are asking for the WG LC
• The draft provides the Flex-algo framework
  • new additions are expected – can be done in separate documents.