

Update on MRT-FRR related drafts in RTGWG

Fast reroute for IP/LDP traffic using maximally redundant trees

draft-bowers-rtgwg-mrt-applicability-to-8021qca-00 (8021qca)

draft-ietf-rtgwg-mrt-frr-architecture (arch)

draft-ietf-rtgwg-mrt-frr-algorithm (algo)

Alia Atlas akatlas@juniper.net (arch, algo)

Chris Bowers cbowers@juniper.net (arch, algo, 8021qca)

Andras Csaszar Andras.Csaszar@ericsson.com (arch, algo)

Gabor Sandor Enyedi Gabor.Sandor.Enyedi@ericsson.com (arch, algo)

Janos Farkas janos.farkas@ericsson.com (8021qca)

Abishek Gopalan abishek@ece.arizona.edu (algo)

Robert Kebler rkebler@juniper.net (arch)

Jeff Tantsura jeff.tantsura@ericsson.com (arch)

Russ White russw@riw.us (arch)

MRT-related draft update

- **draft-bowers-rtgwg-mrt-applicability-to-8021qca-01**
 - Added text to explain that 802.1Qca applies the 802.1aq Agreement Protocol as additional loop avoidance mechanism for recomputed MRTs.
- **draft-ietf-rtgwg-mrt-frr-architecture-06**
 - Added Topology-Independent LFA to section comparing IP/LDP FRR mechanisms.
- **draft-ietf-rtgwg-mrt-frr-algorithm-05**
 - More detail in next slides
- **draft-ietf-mpls-ldp-mrt-01**
- **draft-ietf-ospf-mrt-00**
- **draft-ietf-isis-mrt-00**
- <https://github.com/cbowers>

Clarifications to MRT algorithm

- Clarified usage of 'root' in algorithm pseudo-code
 - GADAG root
 - Block root / local root
 - SPF root
- Clarified in pseudo-code where interface ordering is important for correctness of algorithm vs. simply iterating over all interfaces on a node.
- Modified pseudo-code to handle adding parallel links to the GADAG more cleanly
 - Algorithm now makes all parallel links between blocks both INCOMING and OUTGOING.

Clarifications to MRT algorithm (2)

- Provided more systematic explanation of logic behind `Select_Alternate_Internal()`, and modified pseudo-code to follow that explanation
 - Result of algorithm is unchanged.

Python implementation of MRT algorithm

- Runs with Python 2.6 and 2.7.
 - Known incompatibility with Python 3.x due to compare function.
- Incorporated in text of draft as code component.

- Also posted on Github at:

https://github.com/cbowers/draft-ietf-rtgwg-mrt-frr-algorithm/blob/master/src/mrt_lowpoint_draft_text.py

- Should be able to copy/paste, clone, or download single .py file and run it. Example topology input file created by program itself.
- Output files for GADAG, red and blue next-hops, and alternates sorted in “canonical” format for comparison with

Make Python implementation normative ?

- Current draft includes Python code as extra information.
- The pseudo-code and accompanying text description is implicitly the normative version of the algorithm.
- Should we explicitly make the Python implementation the normative version of the algorithm?