draft-kompella-mpls-larp-01

Kireeti Kompella

IETF 90
Toronto, July 2014

Changes from previous version (1)

- Added support for IPv6
  - What is needed here is simply the label to use to get to some remote IPv6 address, hence Neighbor Discovery would be an overkill
  - Solution: just use the “protocol type” ar$pro (0x86DD) to indicate that the address type is IPv6 (and of course, set the “protocol address length” ar$pln to 16)
  - This allows an L-ARP client to request either an IPv4 or IPv6 address by setting ar$pro to the appropriate value

Changes from previous version (2)

- Added a few words on Client-Server synchronization
  - What to do if, after a reply has been sent by the L-ARP server with some parameters, some of those parameters change?
  - How does the client find out?
  - How does the server rectify the situation?
  - What are the consequences meanwhile, on forwarding, on path optimality, on privacy/security, other factors?
  - How can these be mitigated?
- Some early thoughts on these subjects have been jotted down, more as a placeholder for further discussion

Changes from previous version (3)

- Added a metric field
  - for multi-connected hosts to make better decisions in choosing next hops
- … and Entropy Label Capability of the egress
  - This allows hosts that are capable and desirous of sending entropy labels to do so
  - The rationale is that good load balancing is important in Data Centers, so allow the ELC signaling to go all the way to the host
  - The L-ARP server essentially copies the ELC information (if any) from the tunnel LSP protocol to the L-ARP reply

Changes from previous version (4)

- Added a few words on Backward Compatibility
  - Essentially, having a new hardware type makes it easy for senders and receivers to easily separate and process L-ARP and E-ARP
- Again, these are just some early thoughts on this subject to serve as a placeholder for further discussion
  - It is not anticipated that this will be an issue

Further Work

- Mentioned the ARP-NAK opcode
  - May be helpful in avoiding timeouts
- ATMARP redefines packet format
  - Would this be a useful optimization for L-ARP?
- Offline discussions have asked for a number of further extensions
  - Some seem quite useful; others need discussion
  - These have to be balanced against keeping ARP simple

Next Steps

- The chairs have indicated that an open discussion on the mailing list is vital, so I will begin one, in particular on client-server synchronization and on extensions