RSVP Diagnostic Messages

why

- understand what is going on with the path or the reservation

how

- Collecting RSVP state at each hop after PATH state setup
  - if PATH state is not up, use traceroute to see if the routing is working right
  - once know the route, can check individual pieces to see where things went wrong

Design Goals and Non-Goals

- Information to collect
  - current reservation along the path
  - state of one's own reservation
  - reservation merging
- Collecting hop count for each non-RSVP clouds
- Simple, efficient
- No packet implosion/explosion

Non-goal

- collecting any information that is beyond RSVP's control
  - e. g. resource availability

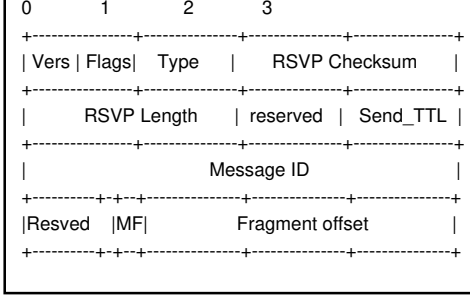
Overview

- Two types of packets, DREQ & DREP
- Anyone can send DREQ to any nodes along the path
  - allowing third party to invoke the diagnostic tool
- DREQ goes in the direction of receiver to sender
  - travel towards a single sender only
- At the point where DREQ cannot move forward further, convert to DREP and send back to the requester
  - reach the sender
  - reach TTL threshold
  - error prevents further forwarding
  - need to be fragmented

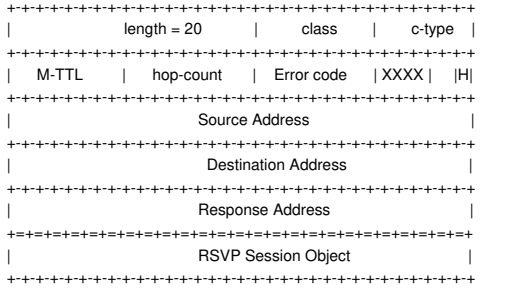
Overview (cont)

- Either unicast or hop-by-hop forward the results (DREP) to the requester
- keep packet size within a pre-defined MTU limit
  - trim off and return partial results to the requester when DREQ packet size gets too large
- build a reverse path for hop-by-hop return of DREP

RSVP Common Header



Diagnostic packet header object



Crossing Firewalls

- If the requester is a receiving host, should be no problem
- If the requester is a 3rd party host and is separated from the “destination address” by a firewall, we may try multicast
  - (doesn’t make sense but that’s beyond the scope here)
  - All DREP packets go back to “destination address” first
  - the destination then mcast DREP’s with a specified TTL limit