RSVP Diagnostic Messages

- information collection, e.g.
  - state for an existing reservation
  - information for a future request
- independent from all other RSVP messages
  - not an error report
  - do not change any state at any router
- originated from a receiving host, forwarded hop-by-hop towards the source, then the response goes back to the originating host

Why need diagnostic messages

- check state for an existing reservation
  - collect information at every hop
- learn about how one's reservation merged with others
- find out about non-RSVP clouds

- checking resource availability for future reservation requests (?)

How it works

- originated from one receiver, destinated to one source
  - eliminate implosion
- forwarded hop-by-hop
  - all existing multicast routing protocols know the previous hop towards a given source host
  - each hop attaches a response block to the msg and forwards it
- the source host changes the msg type from "diagnosis" to "reply" and sends it back to the originating receiver host
- the msg may also be sent back by an intermediate router
  - if an error along the way prevents further forwarding
  - if a specified hop/scope limit has been reached

How it works (II)

information needed in a diagnostic msg

- originating host
- session ID
- source host

(minimal set) information to be collected

- time of arrival
- incoming and outgoing interface addresses
- previous routing hop address
- reservation state (at the output interface), or absence of it
  - Tspec, Rspec that are in place
  - timer values
  - "merge" flag

Design Issues

- the scope of information to be collected
  - any information one can collect from RSVP state
  - packet counts too?
    - » e.g. total packet count at the output interface
  - any information from resource management module?
- response message: unitcast directly to the originator, or forward hop-by-hop, hoping to get around firewalls?
  - other RSVP messages that face the same question
    - » PATH error messages
    - » RESV ACK messages (generated at the first merging point)
  - FYI:: multicast trace route proposes to multicast the response instead

Design Issues (II)

- should we allow a third party host to initiate a diagnosis message from a chosen receiver?
- response fragmentation: follow mtrace way, or our way?
- relation with multicast trace route?
  - partial overlap in the information collected by each
    - » incoming and outgoing interfaces at each node
      - mcast hops vs RSVP hops
    - » packet counts
      - they collect the total mcast packets count, we might collect per session count
  - lots of other things they don't do
    - » RSVP state
    - » info for unicast reservations