RSVP Open Issues

- Open issues on someone else's plate
- Negotiation model
- Interface with Routing
- Access control
- Filter spec representation
- "Killer reservation"
- Firewall architecture
- Division of Labor between multicast routing and RSVP

Open Issues on Someone Else's Plate

- Flowspec definition
- Flowspec comparison
- Merging of reservations.
- QoS routing

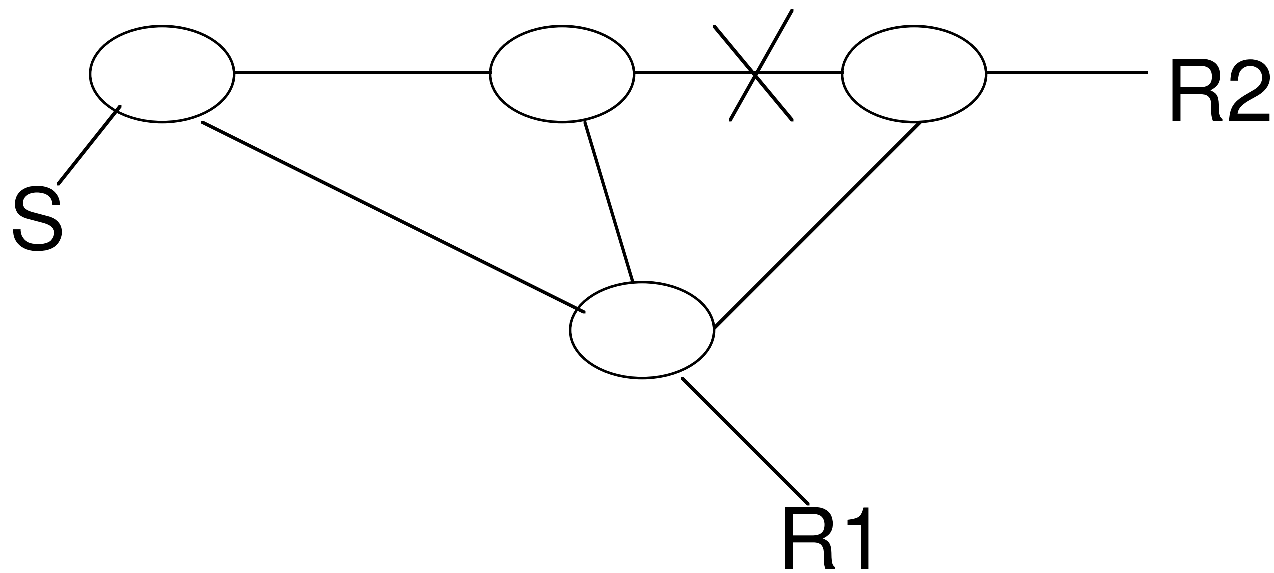
Negotiation Model

- **The current spec: one pass model**
  - Provide only negative feedback
  - Adequate? Inadequate?
- **Scott just presented a much richer model**
  - Too much? Just right? Or still not enough?
- **Any other proposals?**

Interface with Routing

- **Short term: do nothing more than what's in the spec. Consensus?**
- **Longer term: define the interface**
  - Mutual requirements and responsibilities
  - Reservation repair: attempt local repair, or inform the ends to refresh?
  - Smooth switchover to new routes: who want/ need it? Routing? RSVP? Both?

Reservation Repair: Local or from the Ends?



Topological changes may cause rerouting away from the original path.

Access Control

- **An Authentication/User ID field defined in RSVP messages**
- **Eventually: secure authentication, perhaps accounting too.**
- **What to do meanwhile?**
  - Some ad hoc fix needed?
  - If so, what?
    - » Manually configured access control list?
    - » Any other suggestions?

Filter Spec Representation

- **Currently we use bitmap masks.**
- **Question: Do we need something more general?**

  E.g. BPF like pseudo code?

“Killer Reservation”

- The issue: when a larger request merges with an existing reservation and fails adm control upstream, the original reservation should not be affected.
- The current implementation does not handle this situation properly.
- Additional state needed to solve this problem.

Firewall architecture

- We need one for RSVP
- How does it look like?

Division of labor between multicast routing and RSVP

- **Continued discussion from the last IETF: should we use RSVP filters or multiple mcast groups to differentiate substreams of the same source?**