IETF Integrated Services

- architecture for providing QoS guarantees in IP networks for individual application sessions
- resource reservation: routers maintain state info (a la VC) of allocated resources, QoS req's
- admit/deny new call setup requests:

**Question:** can newly arriving flow be admitted with performance guarantees while not violated QoS guarantees made to already admitted flows?

Call Admission

**Arriving session must:**
- declare its QoS requirement
  - **R-spec:** defines the QoS being requested
  - characterize traffic it will send into network
    - **T-spec:** defines traffic characteristics
  - signaling protocol: needed to carry R-spec and T-spec to routers (where reservation is required)
    - **RSVP**

Intserv: QoS guarantee scenario

- **Resource reservation**
  - call setup, signaling (RSVP)
  - traffic, QoS declaration
  - per-element admission control

Intserv QoS: Service models [rfc2211, rfc 2212]

**Guaranteed service:**
- worst case traffic arrival: leaky-bucket-policed source
- simple (mathematically provable) bound on delay [Parekh 1992, Cruz 1988]

**Controlled load service:**
- "a quality of service closely approximating the QoS that same flow would receive from an unloaded network element."

D\(_{max}\) = \(\frac{b}{R}\)
IETF Differentiated Services

Concerns with Intserv:
- **Scalability:** signaling, maintaining per-flow router state difficult with large number of flows
- **Flexible Service Models:** Intserv has only two classes. Also want "qualitative" service classes
  - "behaves like a wire"
  - relative service distinction: Platinum, Gold, Silver

Diffserv approach:
- simple functions in network core, relatively complex functions at edge routers (or hosts)
- Don't define service classes, provide functional components to build service classes

Edge-router Packet Marking
- **profile:** pre-negotiated rate $A$, bucket size $B$
- packet marking at edge based on per-flow profile

Possible usage of marking:
- class-based marking: packets of different classes marked differently
- intra-class marking: conforming portion of flow marked differently than non-conforming one

Classification and Conditioning
- Packet is marked in the Type of Service (TOS) in IPv4, and Traffic Class in IPv6
- 6 bits used for Differentiated Service Code Point (DSCP) and determine PHB that the packet will receive
- 2 bits are currently unused
Classification and Conditioning

may be desirable to limit traffic injection rate of some class:
- user declares traffic profile (e.g., rate, burst size)
- traffic metered, shaped if non-conforming

Forwarding (PHB)

- PHB result in a different observable (measurable) forwarding performance behavior
- PHB does not specify what mechanisms to use to ensure required PHB performance behavior
- Examples:
  - Class A gets x% of outgoing link bandwidth over time intervals of a specified length
  - Class A packets leave first before packets from class B

PHBs being developed:
- Expedited Forwarding: pkt departure rate of a class equals or exceeds specified rate
  - logical link with a minimum guaranteed rate
- Assured Forwarding: 4 classes of traffic
  - each guaranteed minimum amount of bandwidth
  - each with three drop preference partitions

Signaling in the Internet

connectionless (stateless) forwarding by IP routers + best effort service = no network signaling protocols in initial IP design

- New requirement: reserve resources along end-to-end path (end system, routers) for QoS for multimedia applications
- RSVP: Resource Reservation Protocol [RFC 2205]
  - "... allow users to communicate requirements to network in robust and efficient way," i.e., signaling!
- earlier Internet Signaling protocol: ST-II [RFC 1819]
RSVP Design Goals

1. accommodate heterogeneous receivers (different bandwidth along paths)
2. accommodate different applications with different resource requirements
3. make multicast a first class service, with adaptation to multicast group membership
4. leverage existing multicast/unicast routing, with adaptation to changes in underlying unicast, multicast routes
5. control protocol overhead to grow (at worst) linear in # receivers
6. modular design for heterogeneous underlying technologies

RSVP: overview of operation

- senders, receiver join a multicast group
  - done outside of RSVP
  - senders need not join group
- sender-to-network signaling
  - path message: make sender presence known to routers
  - path teardown: delete sender's path state from routers
- receiver-to-network signaling
  - reservation message: reserve resources from sender(s) to receiver
  - reservation teardown: remove receiver reservations
- network-to-end-system signaling
  - path error
  - reservation error

RSVP: does not...

- specify how resources are to be reserved
- rather: a mechanism for communicating needs
determine routes packets will take
- that's the job of routing protocols
- signaling decoupled from routing
- interact with forwarding of packets
- separation of control (signaling) and data (forwarding) planes

Path msgs: RSVP sender-to-network signaling

- path message contents:
  - address: unicast destination, or multicast group
  - flowspec: bandwidth requirements spec.
  - previous hop: upstream router/host ID
  - refresh time: time until this info times out
- path message: communicates sender info, and reverse-path-to-sender routing info
  - later upstream forwarding of receiver reservations
RSVP: simple audio conference

- H1, H2, H3, H4, H5 both senders and receivers
- multicast group m1
- no filtering: packets from any sender forwarded
- audio rate: b
- only one multicast routing tree possible

RSVP: building up path state

- H1, ..., H5 all send path messages on m1:
  (address=m1, Tspec=b, filter-spec=no-filter, refresh=100)
- Suppose H1 sends first path message

RSVP: building up path state

- next, H5 sends path message, creating more state in routers

RSVP: building up path state

- H2, H3, H5 send path msgs, completing path state tables
reservation msgs: receiver-to-network signaling

- reservation message contents:
  - desired bandwidth:
  - filter type:
    - no filter: any packets address to multicast group can use reservation
    - fixed filter: only packets from specific set of senders can use reservation
    - dynamic filter: senders who's packets can be forwarded across link will change (by receiver choice) over time.
  - filter spec

- reservations flow upstream from receiver-to-senders, reserving resources, creating additional, receiver-related state at routers.

RSVP: receiver reservation example 1

- H1 wants to receive audio from all other senders
- H1 reservation msg flows up tree to sources
- H1 only reserves enough bandwidth for 1 audio stream
- reservation is of type "no filter" - any sender can use reserved bandwidth

RSVP: receiver reservation example 1 (more)

- next, H2 makes no-filter reservation for bandwidth \( b \)
- H2 forwards to R1, R1 forwards to H1 and R2 (?)
- R2 takes no action, since \( b \) already reserved on L6
**RSVP: receiver reservation: issues**

*What if multiple senders (e.g., H3, H4, H5) over link (e.g., L6)?*
- arbitrary interleaving of packets
- L6 flow policed by leaky bucket: if H3+H4+H5 sending rate exceeds b, packet loss will occur

**RSVP: example 2**

- H1, H4 are only senders
  - send *path messages* as before, indicating filtered reservation
  - Routers store upstream senders for each upstream link
- H2 will want to receive from H4 (only)
**RSVP: soft-state**

- senders periodically resend path msgs to refresh (maintain) state
- receivers periodically resend resv msgs to refresh (maintain) state
- path and resv msgs have TTL field, specifying refresh interval

**RSVP: reflections**

- multicast as a "first class" service
- receiver-oriented reservations
- use of soft-state