Differentiated Services

- QoS Problem
- DiffServ Architecture
- Per hop behaviors
Problem: QoS

- Need a mechanism for QoS in the Internet
- Issues to be resolved:
  - Indication of desired service
  - Definition of available services
  - Enforcement of contracts (policing)
  - Providing of service
  - Billing for service
Existing mechanisms: TOS

Current IP TOS/Precendence Bits (RFC 791)

- Must be set by clients — doesn’t work with existing apps
- Defines service type (“control traffic”)
- Defines only a small set of simple services (“minimize delay”)
- Billing and network provisioning hard - ISP doesn’t know what will happen
Existing mechanisms: ATM VCs

- Provides some aggregation
- still requires E2E signaling
- State still a problem
Existing mechanisms: RSVP

- No billing defined - useless without
- end-to-end nature makes billing really hard – multilateral, with path changes
- no way to aggregate → number of reservations scales with link bandwidth
- message overload → e2e signaling bad
- useless path state if no reservations exist
- Existing apps must be changed (even OS!)
Differentiated services

- Provide QoS \(<\) RSVP, \(>\) best effort – “BBE”
  - No end-to-end signaling
  - Must work with existing applications
  - Move intelligence and service provisioning to edge
  - Simple, well specified behaviors in core
  - Core behavior based on aggregates
  - Aggregation between domains
  - Flexibility for a wide range of services
  - Separate service primitives from implementation
Differentiated service architecture (RFC 2475)

- Service Providers (ISP) define services
- Services are negotiated with customers in the form of potentially complex Service Level Agreements (SLA’s)
- Customers can be people or other ISP’s or network providers
- At the edge of network, boundary router takes packets and marks, drops, or shapes them based on SLA
- Within the core of the network, routers treat packets solely based on markings they have received
- Markings are in the DS field of IP header (formerly TOS byte)
SLAs

Two components:

Service Level Specification (SLS): service offered to a stream by DS domain
  - absolute/relative loss
  - absolute/relative delay
  - absolute/relative throughput

Traffic Conditioning Specification (TCS): traffic profile
  - time of day
  - locality — per source, per destination
• application based
• traffic contracts — leaky bucket
Scope of services

- Services are not defined end to end
- Services are bilateral agreements between peers
  - End user to local ISP
  - Local ISP to Backbone ISP
  - Campus Network to Backbone provider
- Simplifies billing issues
- Provides a clean architecture for provisioning and implementation
Instantiating SLA’s

- Two kinds of SLA - static and dynamic
- Static SLA’s are pre-provisioned, dynamic are signalled when needed

- Many ways to instantiate a static SLA in a DS Edge Router
  - **SNMP** Lacks asynchronous notification, replication of data
  - **LDAP** Complex descriptions possible, replication supported, but lacks server to client updates
  - **COPS** Integrates with RSVP policy
  - **DIAMETER** General policy mechanism, meets requirements well

- Instantiating a dynamic SLA is hard - RSVP?
DiffServ functional elements

**DS Edge Router:** Connects to an edge router in a neighboring domain

**DS Interior Router:** Router inside core of network

**DS Domain:** Set of routers under a single policy authority

**DS Region:** Set of contiguous DS Domains
Diffserv functional elements

BORDER ROUTER
MARKS PACKETS
BASED ON SLA

CUSTOMER SHAPES
TRAFFIC TO CONFORM TO PROFILE

CORE ROUTER
TREATS PACKET
BASED ON MARKING

BORDER ROUTER
MARKS PACKETS
BASED ON SLA

BORDER ROUTER
SHAPES/REMARKS
ACCORDING TO SLA
WITH NEIGHBOR ISP

DS DOMAIN

DS REGION

EDGE ROUTER

INTERIOR ROUTER

EDGE ROUTER

EDGE ROUTER

INTERIOR ROUTER

EDGE ROUTER

November 27, 2001
DiffServ edge routers

- Edge Router contains a number of elements
  
  **Classifier:** Looks at fields in the packets to determine what SLA/treatment to given them
  
  **Micro Flow Classifier:** Based on 5-tuple
  
  **Bandwidth Aggregate Classifier:** DS field
  
  **Meter:** Determines traffic char. of classified packets;
  
  **Marker:** Sets DS field based on meter & classifier;
  
  **Dropper:** drops out-of-bound packets (“policing”);
  
  **Shaper:** Shapes traffic by adding delay
DiffServ edge router

EDGE ROUTER ARCHITECTURE

CLASSIFIER

METER

MARKER

SHAPER

SLA DATABASE
Per-Hop Behaviors (PHB)

- DS field consists of a 6 bit codepoint \( \rightarrow \) a *Per Hop Behavior (PHB)*
  - defines packet treatment;
  - does not specify implementation, just service;
  - not dependent on 5-tuple.

- codepoints mapped to PHB’s via a lookup table, with standard values;

- some PHB’s standard, others experimental/proprietary

- PHB’s that only have meaning relative to others form a PHB Group
likely there will be a small number of PHB’s
**Per-Hop Behaviors**

- Example PHBs and PHB Groups

**Best Effort:** Current best effort service

**Assured (AF):** (RFC 2597) four classes, with three drop preferences per class

**Expedited (EF):** (RFC 2598) priority, with rate limit, “virtual leased line”

- Services are created by intelligently classifying, metering, and then assigning packets to a small number of PHB’s
Assured Forwarding (RFC 2597)

- four forwarding classes, each allocated buffer space and bandwidth
- no particular scheduling precedence between AF classes
- three drop preferences, with no specified algorithm for each preference
- but probabilities must be increasing
- could use RED for each precedence class
Assured Forwarding (RFC 2597): RIO

- mark packets as “in” or “out” based on traffic profile
- **RIO**: schedule based on variation of random early drop (RED):
  - if average queue size below $\text{min}_{th}$, don’t drop
  - if average between $\text{min}_{th}$ and $\text{max}_{th}$, drop with probability 0 to $\text{max}_p$
  - if above $\text{max}_{th}$, drop all
  - OUT packets: measure whole queue (IN + OUT)
  - IN packets: count only “in” packets
Premium, Expedited Forwarding (EF) (RFC 2598)

Packets should experience almost no queueing delays – “virtual leased line”

- priority service
- weighted round robin (WRR)
- class-based queueing (CBQ)

must *drop* excess packets
Formal definition of EF PHB

Configured rate $R$:

- arrival time $a_j$, with length $l_j$
- departure time $d_j \leq f_j + E_a \forall j > 0$
- $f_0 = 0, d_0 = 0$
- $f_j = \max(a_j, \min(d_{j-1}, f_{j-1})) + l_j/R$
- error term $E_a$ for aggregate (also $E_p$ for individual)
Receiver-Based

- DS mechanisms sender oriented
- Receiver oriented is much harder
- Requires reverse congestion notification
- Requires co-operating sources
- Exception - receiver policies for access links
The DS field (RFC 2474)

- **BA**: behavior aggregate (DS field)
- **MF**: multi-field (DS field + source, destination, ...)

Beyond the DSField itself, the IPv4 TOS byte or IPv6 traffic class octet may be rewritten in the network.

DSCP: Differentiated Services Code Point = value in DSField

- 101110 EF
- 12 values: AF
Interoperability with Intserv

- Three modes:
  - Parallel - both exist, no interaction
  - IntServ over Diffserv - Diffserv agreements purchase tunnels over which RSVP can be used to finely manage bandwidth
  - IntServ aggregation - use Intserv, but aggregate at edges into differnet serv
Admission control

**in-band:** reserve “pipes” or “trunks” via RSVP or specialized protocol

**out-of-band:** bandwidth broker; track usage within domain and needs to keep congestion map
Example SLAs with a single PHB

- Single PHB 1: packets receive almost no delay or loss

- SLA 1:
  - User can send up to 100 kb/s with no loss on Saturdays, 50 kb/s during week
  - Implementation: Classifier selects customers packets. On Saturdays, traffic is leaky bucket shaped to 100 kb/s, 50 kb/s during week. Packets leaving shaper have PHB of 1.
Example SLAs

Toll quality IP telephony:

- Implementation: Classifier detects customers IP telephony traffic based on port/protocol field in headers (not easy...). All telephony classified traffic is marked with PHB 1, else PHB 0 (best effort)
Alternative Best-Effort – ABE

- goal: avoid access control
- applications choose lower end-to-end delay or higher throughput
- “green” traffic for low delay, “blue” for high throughput
- no guarantees
- local transparency to blue: blue traffic delay is not increased; non-dropped blue packets in BE, also not dropped in ABE
- throughput transparency to blue: green traffic based on loss-throughput equation $\leq$ blue traffic
ABE: Duplicate Scheduling with Deadlines
**Enqueueing**

- send duplicate to virtual queue
- duplicate dropped in virtual queue?
  - drop packet
  - deadline = now + queueing time in virtual queue
  - green queue deadline $d = \text{now} + D$

**Dequeuing**

- drop green packets where $t < d$
  - no blue to serve or blue can wait?
    - serve green
    - serve blue