ALBERT-LUDWIGS-
UNIVERSITÄT FREIBURG

Communication Systems

RTP-QoS

University of Freiburg
Computer Science
Computer Networks and Telematics
Prof. Christian Schindelhauer

CoNe
Freiburg

IIF
INSTITUT FÜR
INFORMATIK
FREIBURG

Organization

- I. Data and voice communication in IP networks
- II. Security issues in networking
- **III. Digital telephony networks and voice over IP**

Plan

- Voice over IP and other multimedia applications demand more bandwidth and realtime
- Introduction of special multimedia protocols
  - RTP (Real Time Transport Protocol)
  - RTCP (RTP Control Protocol)
  - RSVP (Resource Reservation Protocol)
- Problems of RSVP and multimedia challenges
- Bandwidth management and Quality of Services
- Provide QoS control in IP networks, i.e., going beyond best effort to provide some assurance for QoS
- Later on switch to Internet telephony, introduction to SIP and H.323.

Real Time Services

- Requirements toward networks for real-time audio and video at least
  - short delay (delay is composed from several parameters)
  - enough bandwidth: normally available in backbone networks
- But more problematic the (private) end user over low bandwidth connections
- During maturing of the Internet bandwidth was often scarce and expensive
  - many solutions to bandwidth management addressed the whole end-to-end system connection
  - but most concepts (e.g. the ToS flag in IP header) are not really used
- By now: It is often cheaper to add bandwidth than operating sophisticated bandwidth management
- But there are scenarios where quality of service (QoS) may improve the whole networks usability ...

Requirements Towards Network

- Voice over IP and Quality of Service:
- Major challenges: delay and delay variation (jitter)
  - Delay jitter is the variability of source-to-destination delays of packets within the same packet stream
  - Voice applications are usually interactive
  - Delay requirement for a telephone system: 150ms-250ms
- The group of Schneider identified the sources of delay in a voice over IP system:
  - OS delay: 10s-100s milliseconds (digitisazion of data, compression and inter software data handling) ...

Requirements Towards Network

- Source jitter:
  - Network: network conditions vary at different times.
  - Non-real time OS: samples processed at different time
- Jitter control - buffering at the destination – task of the application used
- QoS parameters which should be taken into account:
  - Accuracy, latency
  - Jitter and codec quality
- Depending on codec used a data stream of e.g. ~80kbit/s is generated for each direction (64kbit/s of ISDN PCM plus IP and UDP header)

Real Time Transport Protocol (RTP)

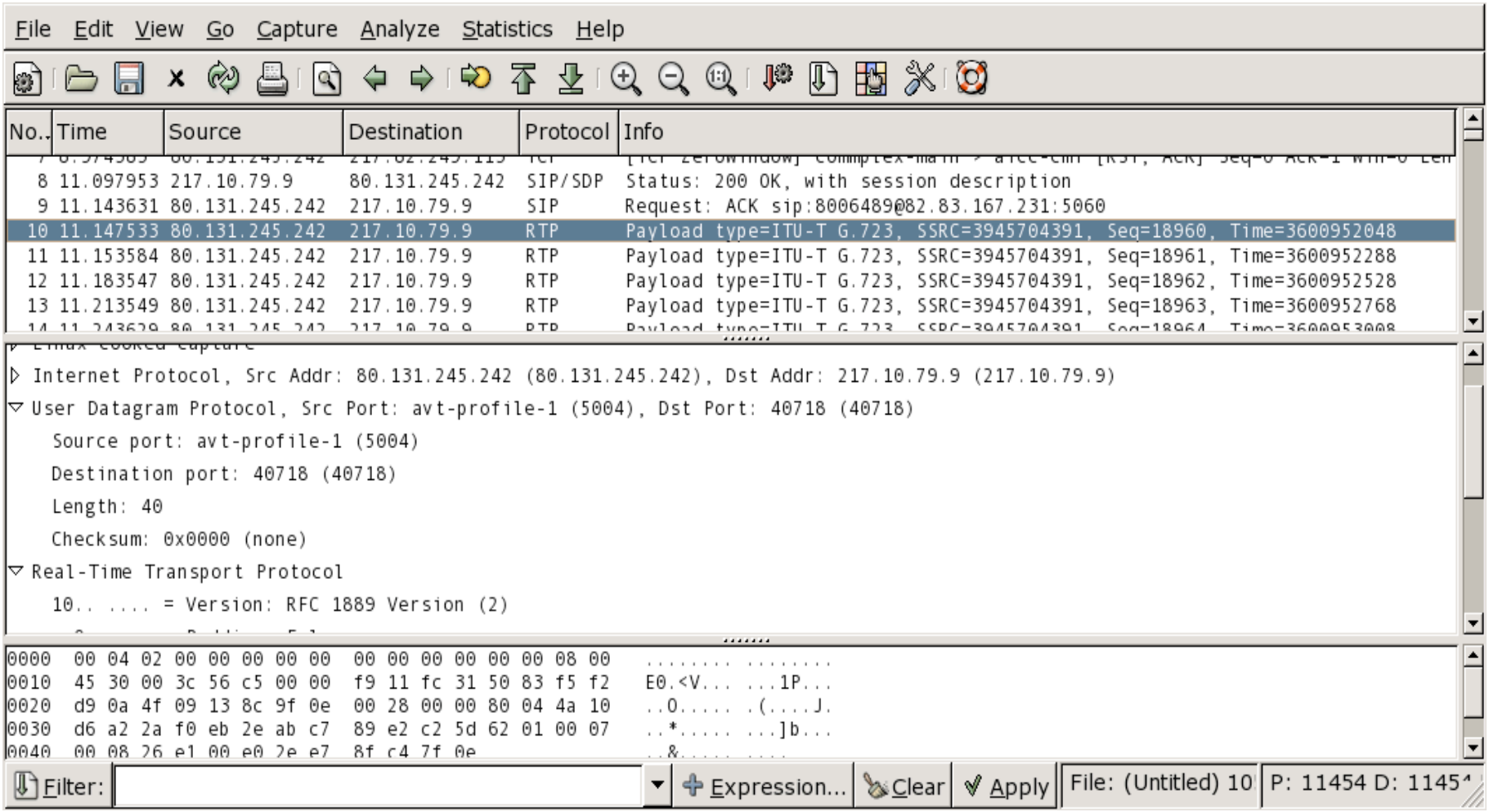
- Introduction of a special multimedia protocol
- Video and audio streaming
- Defined in RFC 1889, RFC 3550.
- Used for transporting common formats such as PCM and GSM for sound, and MPEG1 and MPEG2 for video
- RTP can be viewed as a sublayer of the transport layer
- Usually on top of UDP
  - 8byte header (faster transfer)
  - No setup overhead like with TCP session
  - no explicit connection handling (left to protocols like SIP) – faster

RTP – Packet Header

- RTP packet header
  - Payload type (7 bits): the type of audio/video encoding
  - Sequence number (16 bits)
  - Time stamp (32 bits): used for jitter removal - derived from a sampling clock at the sender
  - Synchronization Source Identifier (SSRC) (32 bits): identify the source of the RTP stream
  - It is not the IP address of the sender (would violate the layering) but a number that the source assigns randomly when the new stream is started

RTP Header

RTP – Header in Wireshark



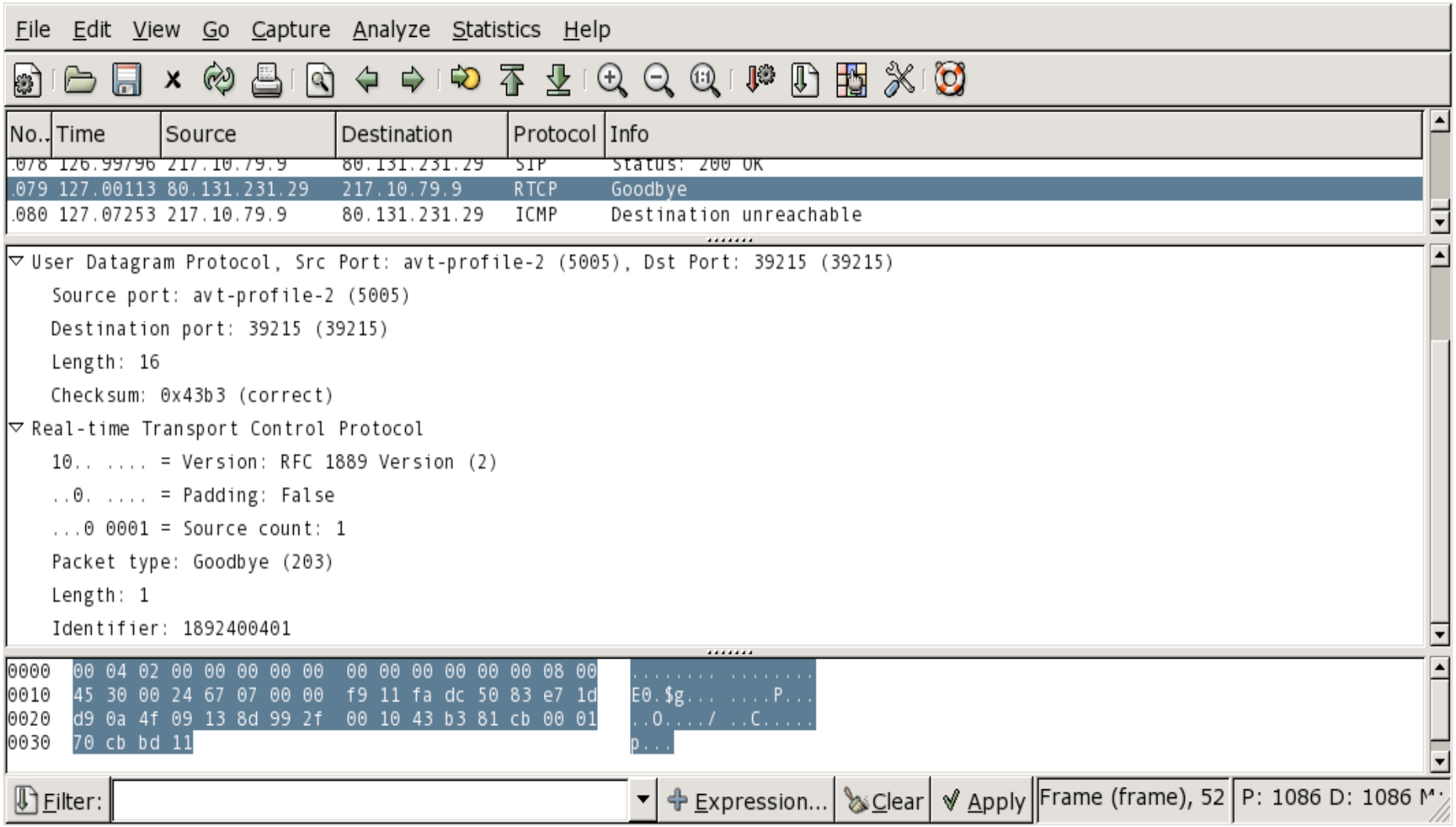
RTP

- At the sender, the application puts its audio/video data with an RTP header and sends into the UDP socket
- The application in the receiver extracts the audio/video data from the RTP packet
  - Uses the header fields of the RTP packet to properly decode and playback the audio/video data
- Helper protocol: RTCP (RTP Control Protocol)
- RTCP packets do not encapsulate audio/video data

RTCP

- RTCP packets sent periodically between sender and receivers to gather useful statistics
  - number of packets sent
  - number of packets lost
  - inter arrival jitter
- RTP and RTCP packets are distinguished from each other through the use of distinct port numbers

RTCP – header in wireshark



Resource Reservation Protocol (RSVP)

- RTP needs a bandwidth at least of the rate as packets are sent in each direction
  - Otherwise packet loss or delays will occur and decrease the quality of data stream
- A special protocol was developed to add service quality parameters to the packet orientated internet
  - RSVP - part of a larger effort to enhance the current Internet architecture with support for Quality of Service flows
  - RFC 2205
- RSVP requests will generally result in resources being reserved in each node along the data path
  - Resource we speak of is bandwidth (delay is much more complicated to “reserve” within IP networks)

RSVP

- Signaling protocol introduced to reserve bandwidth between a source and its corresponding destination
- Main features of RSVP are
  - Use of “soft state" in the routers
  - receiver-controlled reservation requests
  - flexible control over sharing of reservations
  - forwarding of subflows
  - the use of IP multicast for data distribution
- Source → Destination: RSVP path message
- Destination → Source: RSVP reserve message
- Nice try – but ...

RSVP – Problems

- Routers cannot not store state information about packets – often too slow
- Simpler technique: mark each packet with a simple flag indicating how to treat it
- Individual flows are classified into different traffic classes
- Each router sorts packets into queues via differentiated services (DS) flag
- Queues get different treatment (e.g. priority, share of bandwidth, probability of discard)

RSVP – Problems

- Result is coarsely predictable class of service for each "differentiated services" field value
- Cost of transmission varies by type of service
- Each traffic class is reserved a defined level of resources, e.g. buffer and bandwidth
- Different QoS guarantee policies can be applied in different traffic classes
  - When congestion occurs, packets in low priority traffic classes will be dropped first
  - The buffer and the bandwidth in a router for high priority traffic classes are more than low priority traffic classes
- More scalable than RSVP but cannot allocate resources precisely

Multimedia Challenges and Packet Classification

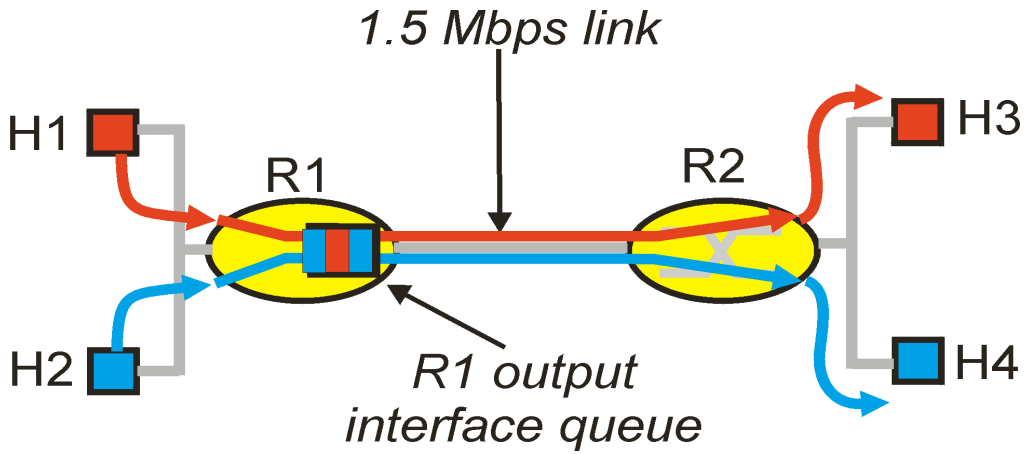
- Remember the packet filtering lectures two weeks ago – IP is a service not offering much QoS features out of itself
- Reconsidering packet filtering from traffic shaping point of view
- Most router implementations:
  - Use only First-Come-First-Serve (FCFS), which might generate suboptimal results
  - Imagine running several VoIP connections on a shared DSL line with P2P file sharing
  - Limited packet processing and transmission scheduling
- To mitigate impact of "best-effort" protocols, we can:
  - Use UDP to avoid TCP and its slow-start phase…
  - Buffer content at client and control playback to remedy jitter
  - Adapt compression level to available bandwidth

Multimedia Challenges – Solutions

- Just add more bandwidth and enhance caching capabilities (over-provisioning)!
- Need major change of the protocols:
  - Incorporate resource reservation (bandwidth, processing, buffering), and new scheduling policies
  - Set up service level agreements with applications, monitor and enforce the agreements, charge accordingly
- Need moderate changes (“Differentiated Services”):
  - Use two traffic classes for all packets and differentiate service accordingly
  - Charge based on class of packets
  - Network capacity is provided to ensure first class packets incur no significant delay at routers

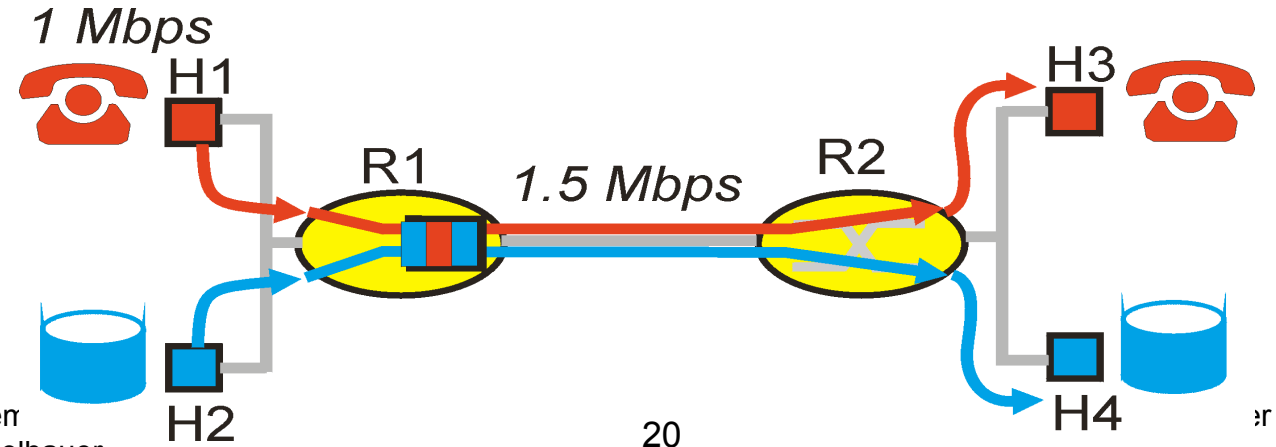
Quality of Service (QoS) – intro

- Talked earlier on new protocols like RTP, RTCP and RSVP – concentrate now on bandwidth management
- IETF groups are working on proposals to provide QOS control in IP networks, e.g., going beyond best effort to provide some assurance for QOS
- Work in Progress includes RSVP, Differentiated Services, and Integrated Services
- Simple model for sharing and congestion studies:



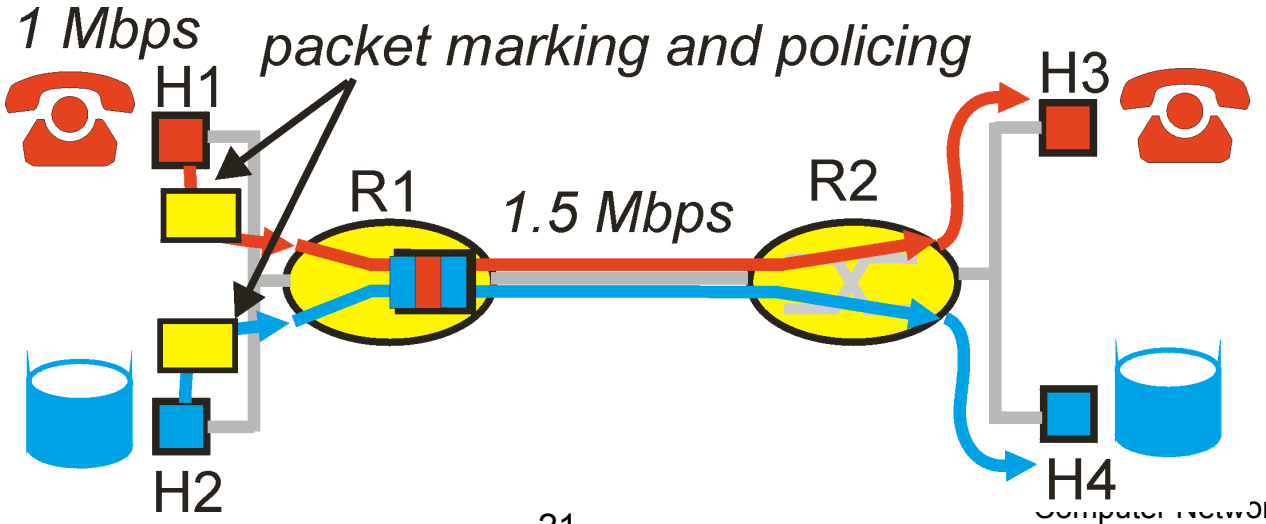
Quality of Service (QoS) – Intro

- Consider a phone/video application at 1Mbit/s and an FTP application sharing a 1.5 Mbit/s link.
  - bursts of FTP can congest the router and cause multimedia packets to be dropped.
  - want to give priority to audio/video streams over FTP
- PRINCIPLE 1: Marking of packets is needed for router to distinguish between different classes; and new router policy to treat packets accordingly



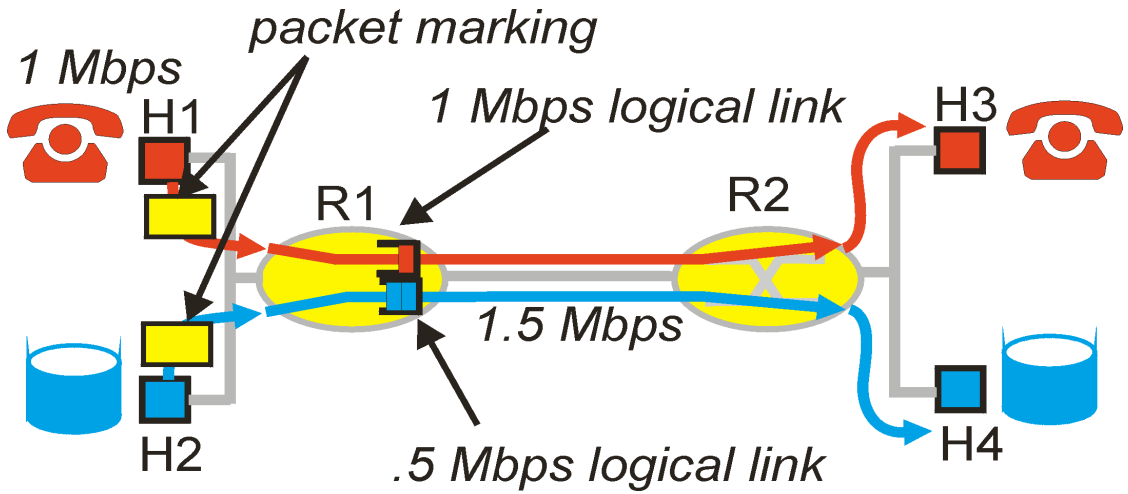
Quality of Service (QoS) – Intro

- Applications misbehave (audio/video sends packets at a rate higher than 1Mbit/s assumed above)
- PRINCIPLE 2: provide protection (isolation) for one class from other classes
- Require Policing Mechanisms to ensure sources adhere to bandwidth requirements; Marking and Policing need to be done at the edges:



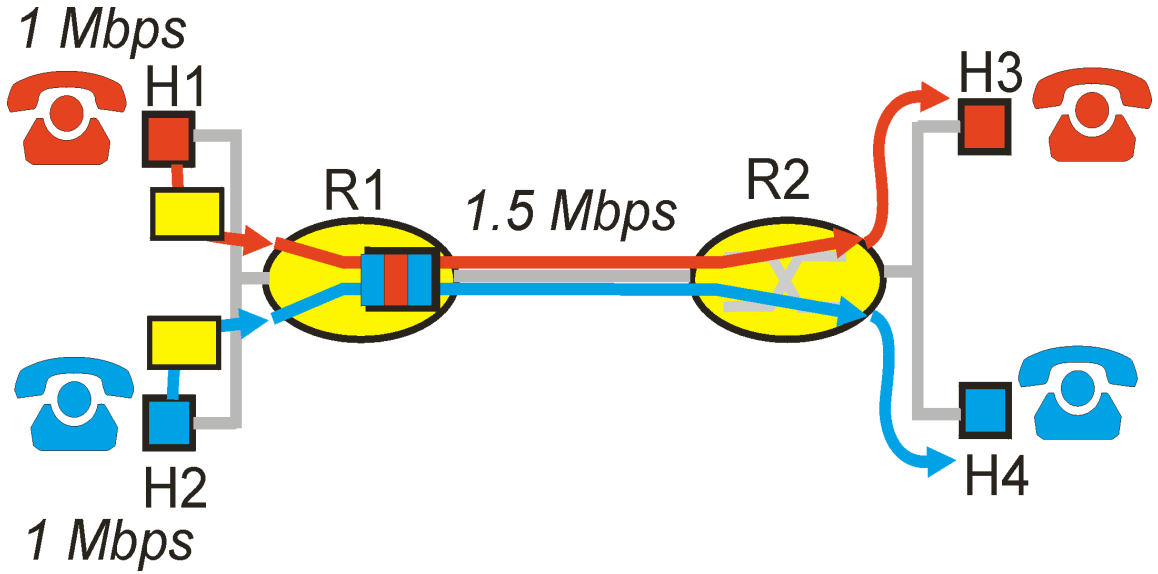
Quality of Service (QoS) – Intro

- Alternative to Marking and Policing: allocate a set portion of bandwidth to each application flow; can lead to inefficient use of bandwidth if one of the flows does not use its allocation
- PRINCIPLE 3: While providing isolation, it is desirable to use resources as efficiently as possible



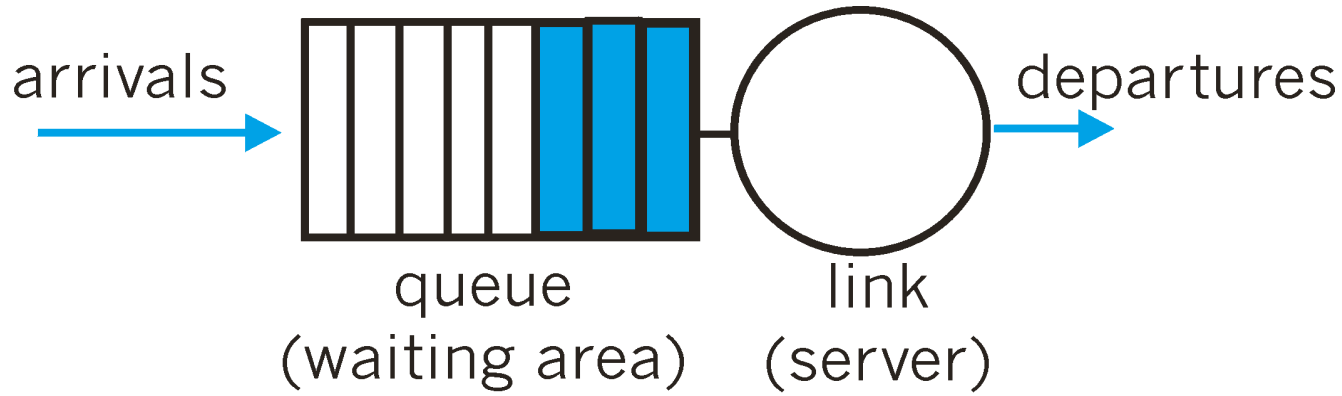
Quality of Service (QoS) – Intro

- Cannot support traffic beyond link capacity
  - Two phone calls each requests 1 Mbit/s
- PRINCIPLE 4: Need a Call Admission Process; application flow declares its needs, network may block call if it cannot satisfy the needs



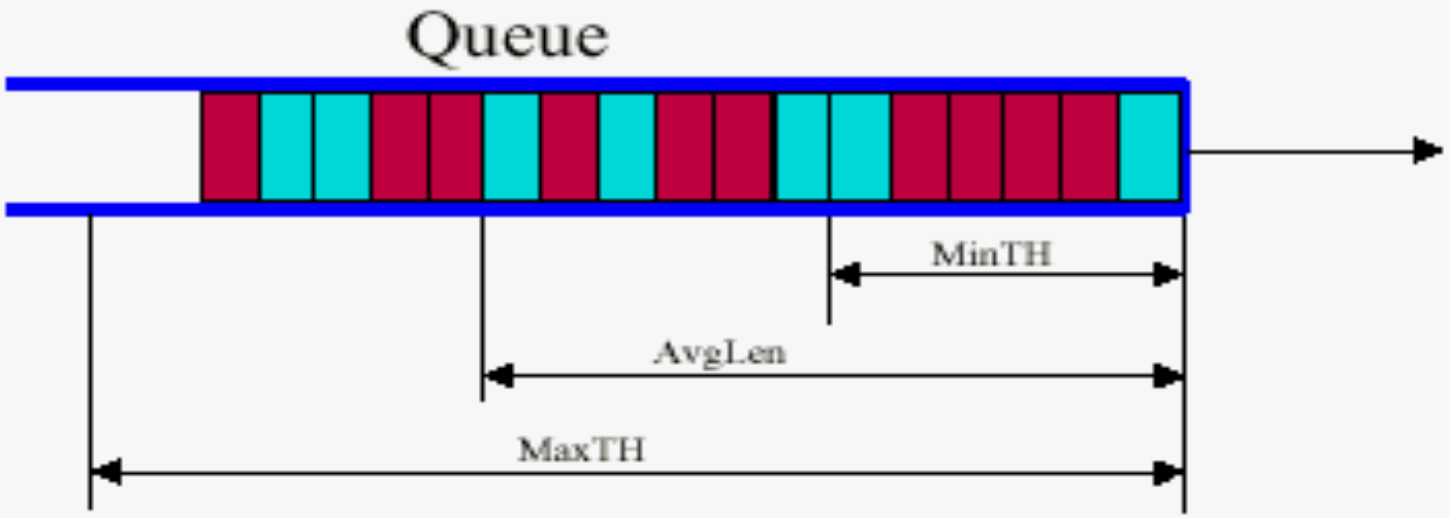
Quality of Service (QoS) – Packet Scheduling

- Scheduling: choosing the next packet for transmission
  - FIFO
  - Priority Queue
  - Round Robin
  - Weighted Fair Queuing



Quality of Service (QoS) – Packet Scheduling

Queue Management Using RED



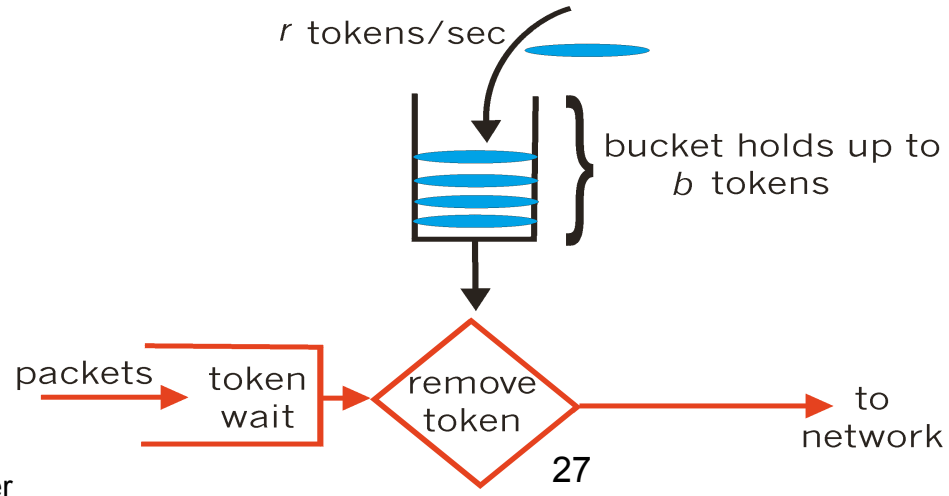
✓ Random Early Detection
- monitor time-based average queue length (AvgLen) and drops arriving packets with increasing probability as AvgLen increases
- no action if AvgLen < MinTH and all packets dropped if AvgLen > MaxTH
- Variants (e.g. WRED, FRED, RIO) increase fairness
- http://www-nrg.ee.lbl.gov/floyd/red.html

Quality of Service (QoS) – Packet Scheduling

- Policing mechanisms
  - (Long term) Average Rate
    - 100 packets per sec or 6000 packets per min??
      - crucial aspect is the interval length
  - Peak Rate:
    - e.g., 6000 p p minute Avg and 1500 p p sec Peak
  - (Max.) Burst Size:
    - Max. number of packets sent consecutively, e.g. over a short period of time
  - Units of measurement
    - Packets versus bits

Quality of Service (QoS) – Packet Scheduling

- Token Bucket mechanism, provides a means for limiting input to specified Burst Size and Average Rate.
- Bucket can hold b tokens
- tokens are generated at a rate of r token/sec
  - unless bucket is full of tokens
- Over an interval of length t, the number of packets that are admitted is less than or equal to (r t + b)



Quality of Service (QoS) – Routing

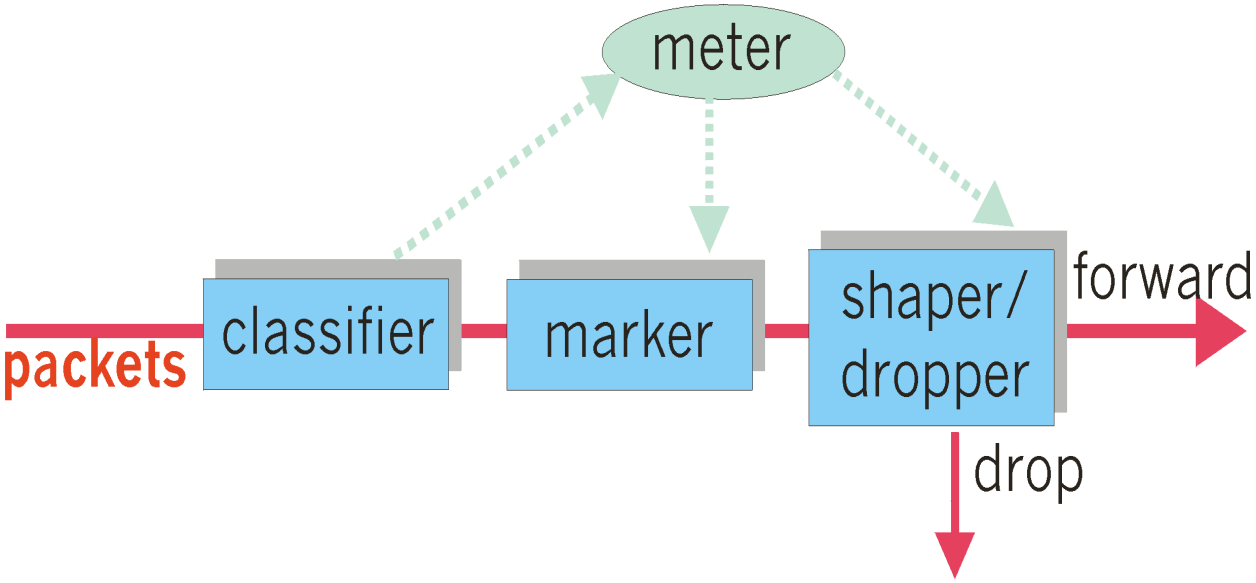
- QoS routing – multiple restraints
- A request specifies the desired QoS requirements
  - e.g., BW, Delay, Jitter, packet loss, path reliability etc
- Two type of constraints:
  - Additive: e.g., delay
  - Maximum (or Minimum): e.g., Bandwidth
- Task
  - Find a (min cost) path which satisfies the constraints
  - if no feasible path found, reject the connection

Quality of Service (QoS) – Classification of Packets

- But often to complicated/impossible to define a path first, so use mechanism on “per-hop-behaviour” (PHB) - simply let routers decide on each hop what to do
  - Big advantage over protocols like RSVP – no state to be kept
- Give routers hints how to handle different packets
- 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

Quality of Service (QoS) – Classification of Packets

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



Quality of Service (QoS) – Classification of Packets

- 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

Quality of Service (QoS) – Classification of Packets

- PHBs under consideration:
  - Expedited Forwarding: departure rate of packets from a class equals or exceeds a specified rate (logical link with a minimum guaranteed rate)
  - Assured Forwarding: 4 classes, each guaranteed a minimum amount of bandwidth and buffering; each with three drop preference partitions
- But: AF and EF are not even in a standard track yet… research ongoing
- “Virtual Leased lines” and “Olympic” services are being discussed
- Impact of crossing multiple ASs and routers that are not DS-capable

Quality of Service (QoS) – Linux Implementation

- Practical implementations – deployed Linux QoS in an earlier practical session already, so tools should be familiar already
- Linux kernel includes several types of QoS features
  - Hierarchy token bucket (HTB)
  - Statistical fair queuing (SFQ)
  - Hierarchical Fair Service Curve Packet Scheduler
  - ...
- The iproute2 package is used to handle traffic classes (tc command)
- Linux packet filter is able to mark packets – so they could be handled later in QoS queues

Queueing Disciplines (qdisc) in Linux

- Queueing Discipline (qdisc) is an algorithm that manages the queue of a device, either incoming (ingress) or outgoing (egress).
- “tc” command in Linux
- Classless qdisc
  - shape traffic for an entire interface, without any subdivisions.
  - fifo_fast, Token Bucket Filter (TBF), Stochastic Fairness Queueing (SFQ)
- Classful qdisc
  - contains multiple classes having different priorities, different kinds of traffic can have different treatment.
  - PRIO , Class Based Queueing (CBQ), Hierarchical Token Bucket (HTB)

Classless Queueing

- pfifo_fast
  - First In, First Out. No packet receives special treatment.
  - The queue has 3 bands. Within each band, FIFO rules apply
- Token Bucket Filter (TBF)
  - Only passes packets arriving at a rate which is not exceeding the administratively set rate
  - But allow short bursts in excess of this rate
  - Have a buffer (bucket), constantly filled by tokens, at a specific rate (token rate)
  - Each arriving token collects one incoming data packet from the data queue and is then deleted from the bucket
  - The first choice if you just want to slow down an interface

Classless Queueing

- Stochastic Fairness Queueing (SFQ)
  - Traffic is divided into a pretty large number (limited number) of FIFO queues using hashing algorithm (hence stochastic)
  - One queue for one session
- Traffic is then sent in a round robin fashion, giving each session the chance to send data in turn

Classful queueing

- Contains multiple classes with different priorities, so different kinds of traffic can have different treatment.
- When the traffic enters a classful qdisc, it needs to be classified according to the 'filters'.
- PRIO (Priority queueing)
  - No shaping, only subdivides traffic based on filters
  - When a packet is enqueued to the PRIO qdisc, a class is chosen based on the filters
  - Very useful in case you want to prioritize certain kinds of traffic, without using only TOS-flags but using all the power of the tc filters

Classful Queuing

- Class Based Queuing (CBQ)
  - The most complex qdisc available
  - Implement shaping by measuring effective idletime, to make sure that the link is idle just long enough to bring down the real bandwidth to the configured rate
  - Subdivides traffic based on filters
  - When sending out a packet, uses a weighted round robin process ('WRR'), beginning with the lower-numbered priority classes

Classful Queuing

- Hierarchical Token Bucket (HTB)
- CBQ is complex and does not seem optimized for many typical situations
- HTB is well suited for setups where
  - you have a fixed amount of bandwidth
  - you want to divide the bandwidth for different traffics and give each traffic a guaranteed bandwidth
  - and specify how much bandwidth can be borrowed
- HTB works just like CBQ but does not resort to idle time calculations to shape
  - Instead, it is a classful Token Bucket Filter (hence the name :-))

Quality of Service (QoS) – Conclusion

- In most cases bandwidth (and IP first-come-first served) suffices
  - But you may have to connect a flatsharing community of students over a single DSL line
  - Provide Internet services for a student dormitory over a WLAN link with limited capacity
- Congested lines may render the whole service unusable
  - SSH gets unbearable delays, Mail download via POP or IMAP takes hours
  - Even filesharing does not work – ACK to downloaded packets have to wait to long ...
- That way you might solve a range of bandwidth related problems without the need to upgrade the connection
- Nevertheless at corporate level it is often cheaper just to add bandwidth than starting a sophisticated QoS management on switch and IP level

ALBERT-LUDWIGS-
UNIVERSITÄT FREIBURG

Communication Systems

University of Freiburg
Computer Science
Computer Networks and Telematics
Prof. Christian Schindelhauer

CoNe
Freiburg

IIF
INSTITUT FÜR
INFORMATIK
FREIBURG