Client – Server Architecture

Machine address

versus

Transport address

ARASH HABIBI LASHKARI
(April - 2011)

Transport provider

Layer of software that accepts a network message and sends it to a remote machine

Two categories:

connection-oriented protocols

connectionless protocols

Connection-oriented Protocols

1. establish connection
2. [negotiate protocol]
3. exchange data
4. terminate connection

Connection-oriented Protocols

	analogous to phone call
1. establish connection	*dial phone number*
2. [negotiate protocol]	*[decide on a language]*
3. exchange data	*speak*
4. terminate connection	*hang up*

virtual circuit service

- provides illusion of having a dedicated circuit
- messages guaranteed to arrive in-order
- application does not have to address each message

vs. circuit-switched service

Connectionless Protocols

- no call setup
- send/receive data
  (each packet addressed)
- no termination

Connectionless Protocols

analogous to mailbox

- no call setup
- send/receive data — *drop letter in mailbox*
  (each packet addressed) — *(each letter addressed)*
- no termination

datagram service

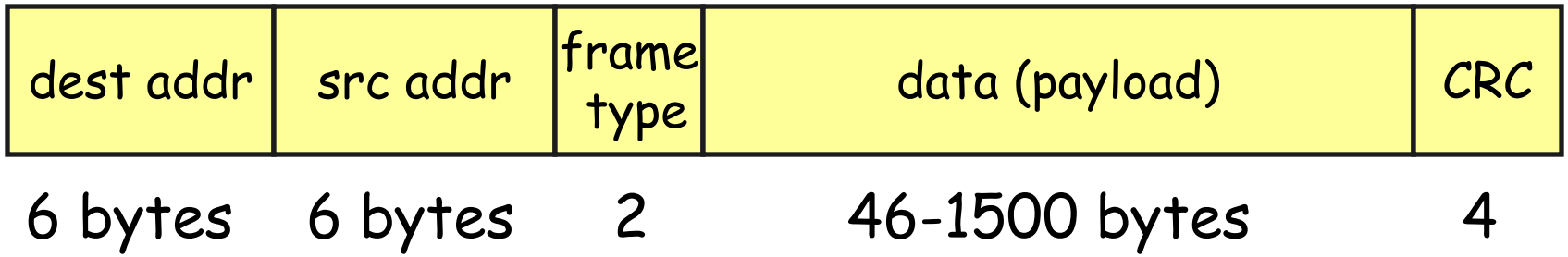
- client is not positive whether message arrived at destination
- no state has to be maintained at client or server
- cheaper but less reliable than virtual circuit service

Ethernet

- Layers 1 & 2 of OSI model
  - Physical (1)
    - Cables: 10Base-T, 100Base-T, 1000Base-T, etc.
  - Data Link (2)
    - Ethernet bridging
    - Data frame parsing
    - Data frame transmission
    - Error detection
- Unreliable, connectionless communication

Ethernet

- 48-byte ethernet address
- Variable-length packet
  - 1518-byte MTU
    - 18-byte header, 1500 bytes data
- Jumbo packets for Gigabit ethernet
  - 9000-byte MTU



18 bytes + data

IP – Internet Protocol

Born in 1969 as a research network of 4 machines
Funded by DoD's ARPA

Goal:

build an efficient fault-tolerant network that could connect heterogeneous machines and link separately connected networks.

Internet Protocol

Connectionless protocol designed to handle the interconnection of a large number of local and wide-area networks that comprise the internet

IP can route from one physical network to another

IP Addressing

Each machine on an IP network is assigned a unique 32-bit number for each network interface:

- **IP address**, *not* machine address

A machine connected to several physical networks will have several IP addresses

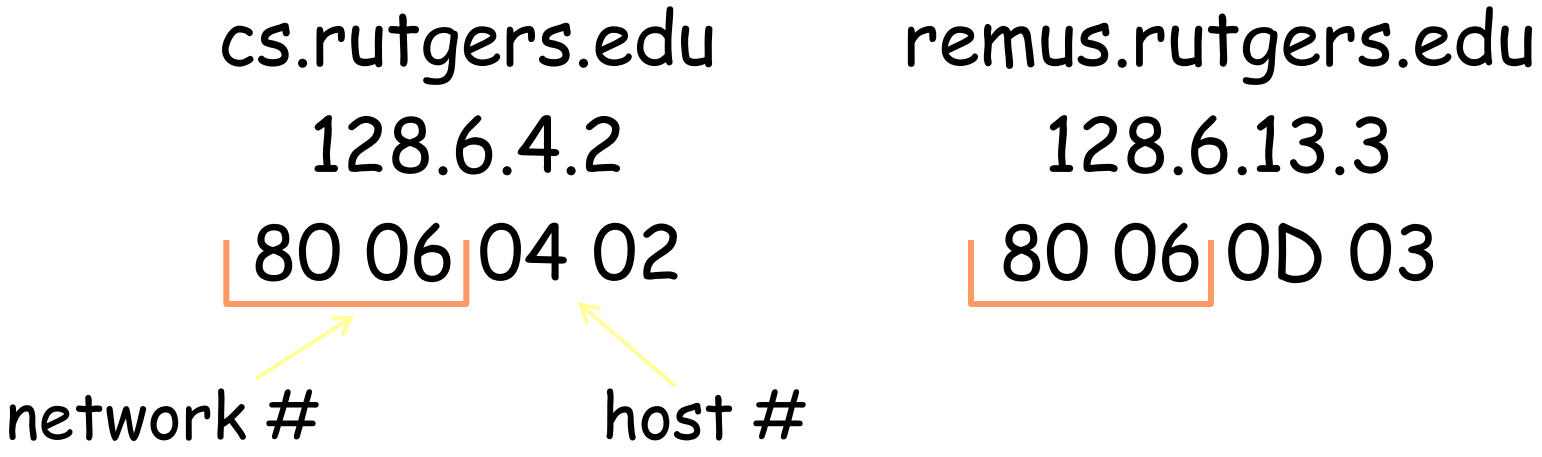
- One for each network

IP Address space

32-bit addresses $\rightarrow$ >4 billion addresses!

- Routers would need a table of 4 billion entries
- Design routing tables so one entry can match multiple addresses
  - hierarchy: addresses physically close will share a common prefix

IP Addressing: networks & hosts



- first 16 bits identify Rutgers
- external routers need only one entry
  - route 128.6.*.* to Rutgers

IP Addressing: networks & hosts

- IP address
  - **network #:** identifies network machine belongs to
  - **host #:** identifies host on the network

- use network number to route packet to correct network
- use host number to identify specific machine

IP Addressing

Expectation:

- a few big networks and many small ones
- create different **classes** of networks
- use leading bits to identify network

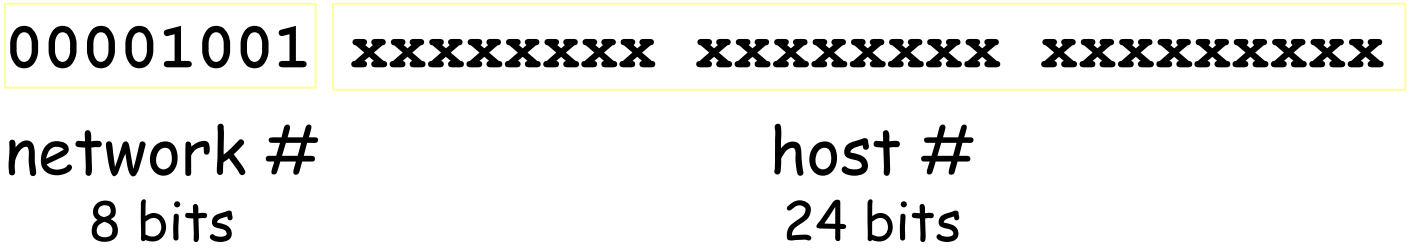
class	leading bits	bits for net #	bits for host
A	0	7 (128)	24 (16M)
B	10	14 (16K)	16 (64K)
C	110	21 (2M)	8 (256)

To allow additional networks within an organization:

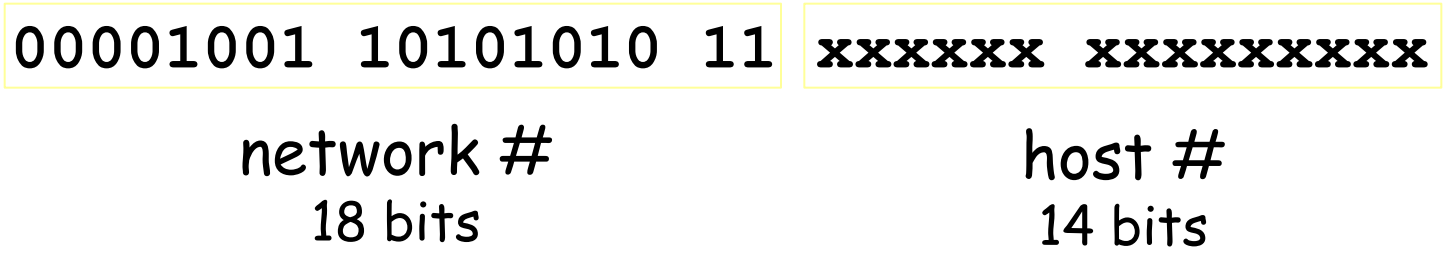
use high bits of host number for a
"network within a network" – **subnet**

IP Addressing

IBM: 9.0.0.0 – 9.255.255.255



Subnet within IBM (internal routers only)



Running out of addresses

- Huge growth
- Wasteful allocation of networks
  - Lots of unused addresses
- Every machine connected to the internet needed a worldwide-unique IP address
- Solutions: CIDR, NAT, IPv6

Classless Inter-Domain Routing (CIDR)

Replace class A, B, C addresses:

- Explicitly specify # of bits for network number
- rather than 8 (A), 16 (B), 24 (C) bits

Better match for organizational needs

machine that needs 500 addresses:

- get a 23-bit network number (512 hosts) instead of a class B address (64K hosts)

Classless Inter-Domain Routing

How does a router determine # bits?

CIDR address specifies it:
32-bit-address/bits-for-network-prefix

- 128.6.13.3/16
- /27 : 1/8 of a class C (32 hosts)
- /24 : class C
- /16 : class B

managing CIDR addresses & prefixes can be a pain

IP Special Addresses

- All bits 0
  - Valid only as *source address*
  - "all addresses for this machine"
  - Not valid over network
- All host bits 1
  - Valid only as destination
  - Broadcast to network
- All bits 1
  - Broadcast to all directly connected networks
- Leading bits 1110
  - Class D network
- 127.0.0.0: reserved for local traffic
  - 127.0.0.1 usually assigned to *loopback device*

IPv6 vs. IPv4

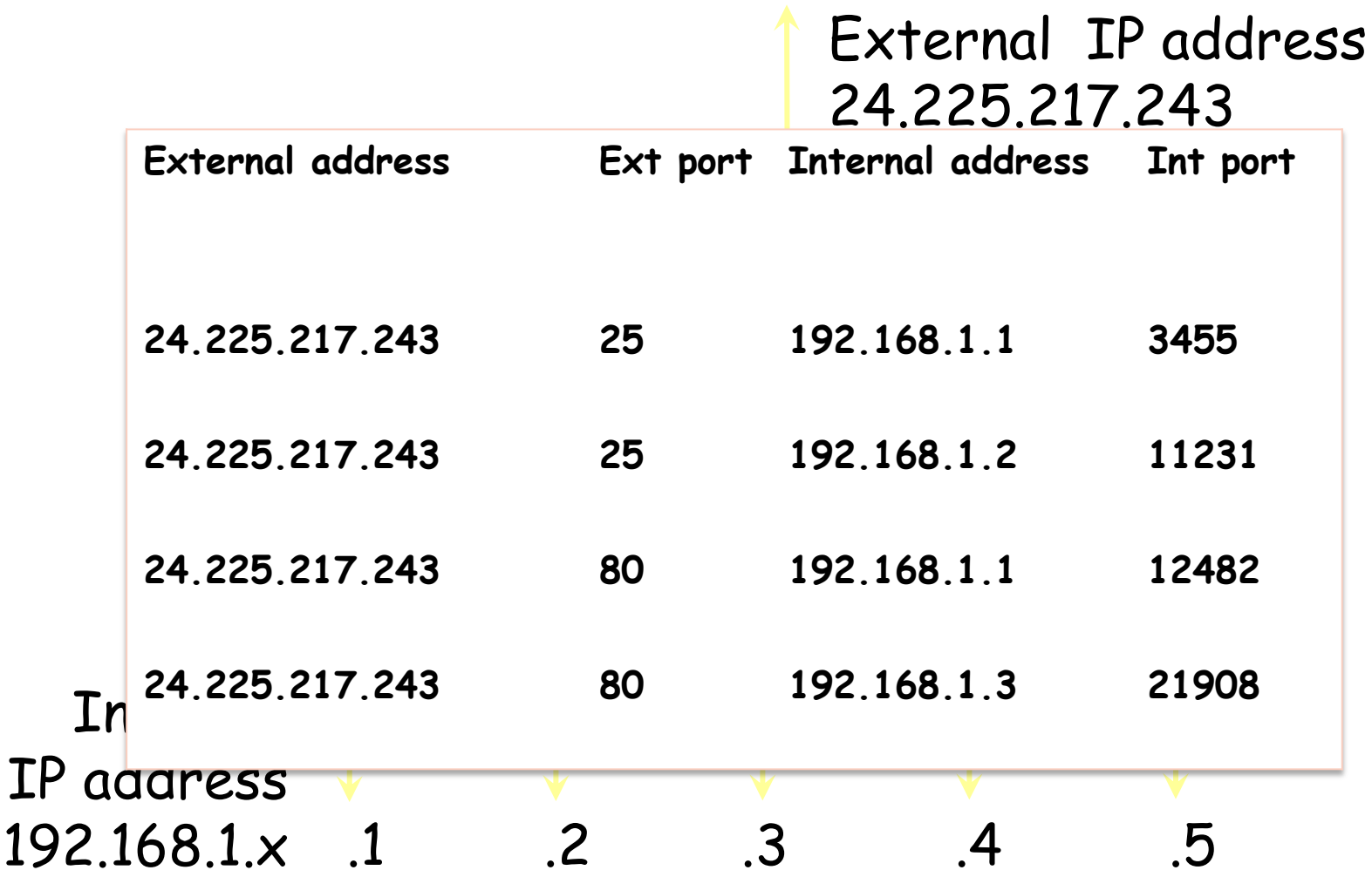
IPv4

- 4 byte (32 bit) addresses

IPv6:

- 16-byte (128 bit) addresses
  3.6×10^{38} possible addresses
  8×10^{28} times more addresses than IPv4
- 4-bit priority field
- Flow label (24-bits)

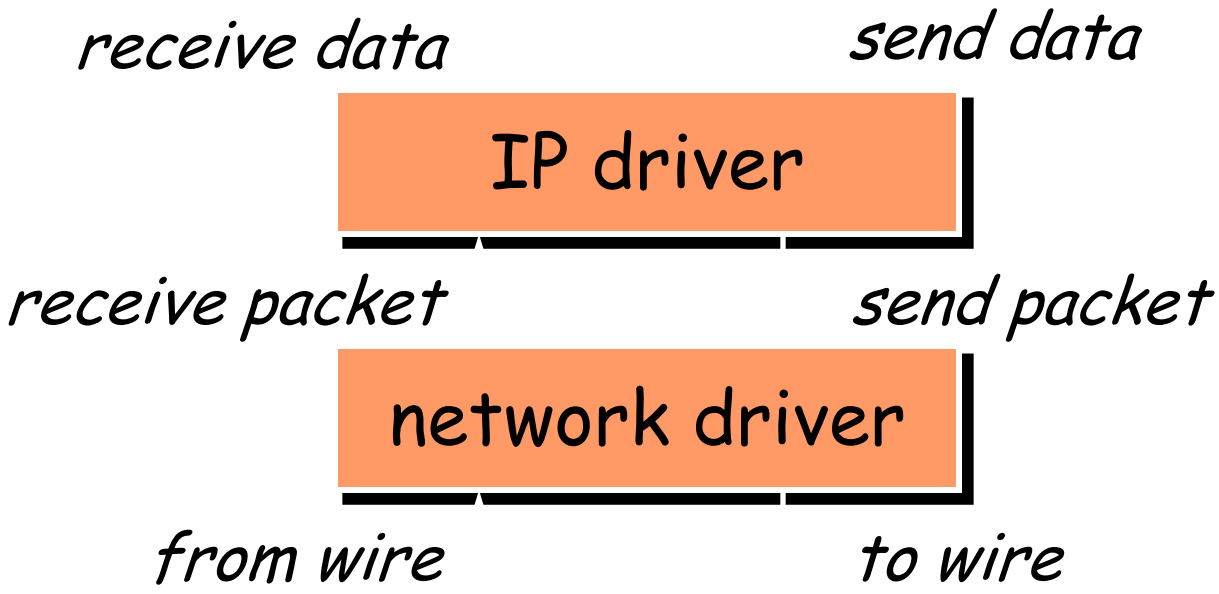
Network Address Translation (NAT)



Getting to the machine

IP is a logical network on top of multiple physical networks

OS support for IP: **IP driver**



IP driver responsibilities

- Get operating parameters from device driver
  - Maximum packet size (MTU)
  - Functions to initialize HW headers
  - Length of HW header
- Routing packets
  - From one physical network to another
- Fragmenting packets
- Send operations from higher-layers
- Receiving data from device driver
- Dropping bad/expired data

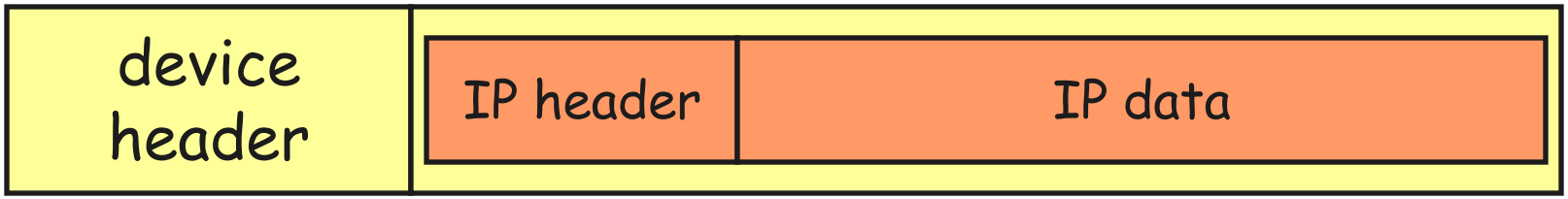
Device driver responsibilities

- Controls network interface card
  - Comparable to character driver
- Processes interrupts from network interface
  - Receive packets
  - Send them to IP driver
- Get packets from IP driver
  - Send them to hardware
  - Ensure packet goes out without collision

Network device

- Network card examines packets on wire
  - Compares destination addresses

- Before packet is sent, it must be **enveloped** for the physical network



Device addressing

IP address $\rightarrow$ ethernet address

Address Resolution Protocol (ARP)

1. Check local ARP cache
2. Send broadcast message requesting ethernet address of machine with certain IP address
3. Wait for response (with timeout)

Routing

Router

- Switching element that connects two or more transmission lines (e.g. Ethernet)
- Routes packets from one network to another (OSI layer 2)
- Special-purpose hardware or a general-purpose computer with two or more network interfaces

Routing

- Packets take a series of **hops** to get to their destination
  - Figure out the path
- Generate/receive packet at machine
  - check destination
    - If destination = local address, deliver locally
  - else
    - Increment hop count (discard if hop # = TTL)
    - Use destination address to search **routing table**
    - Each entry has address and netmask. Match returns interface
    - Transmit to destination interface
- **Static routing**

Dynamic Routing

- Class of protocols by which machines can **adjust routing tables** to benefit from load changes and failures

- Route cost:
  - Hop count (# routers in the path)
  - Time: Tic count – time in 1/18 second intervals

Dynamic Routing Examples

- RIP (Routing Information Protocol)
  - Exchange routing tables with neighboring routers on internal networks
  - Choose best route if multiple routes exist
- OSPF (Open Shortest Path First)
  - Tests status of link to each neighbor. Sends status info on link availability to neighbors.
  - Cost can be assigned on reliability & time
- BGP (Border Gateway Protocol)
  - TCP connection between pair of machines
  - Route selection based on distance vector
  - Exchanges information about reachable networks
  - Periodic keep-alive messages

Transport-layer protocols over IP

- IP sends packets to machine
  - No mechanism for identifying sending or receiving application
- Transport layer uses a **port number** to identify the application
- TCP – Transmission Control Protocol
- UDP – User Datagram Protocol

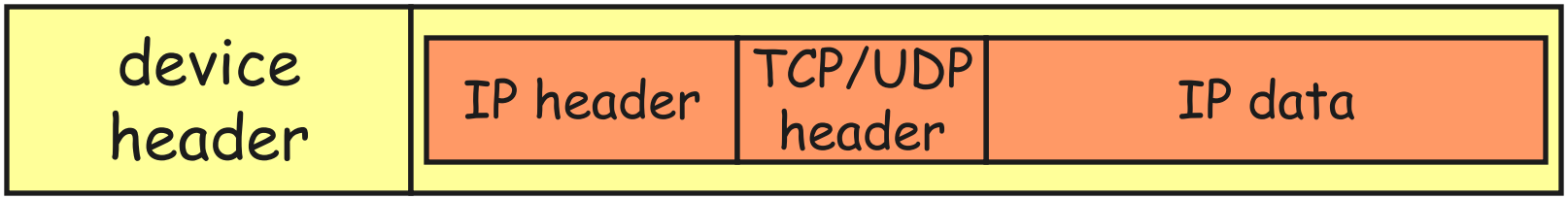
TCP – Transmission Control Protocol

- Virtual circuit service (connection-oriented)
- Send acknowledgement for each received packet
- Checksum to validate data
- Data may be transmitted simultaneously in both directions

UDP – User Datagram Protocol

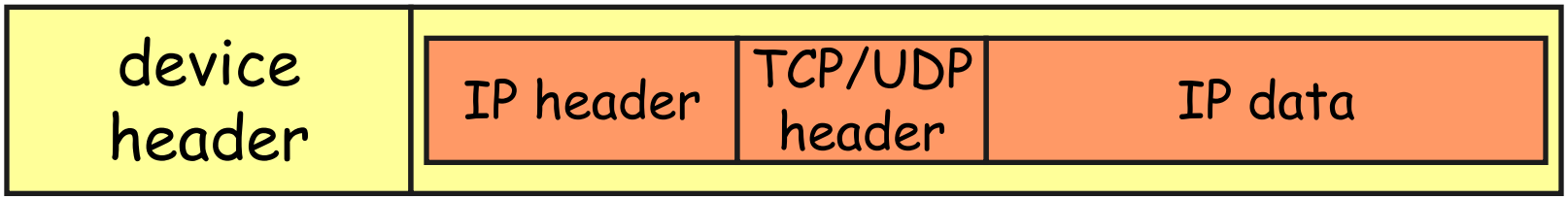
- Datagram service (connectionless)
- Data may be lost
- Data may arrive out of sequence
- Checksum for data but no retransmission
  - Bad packets dropped

IP header



vers	hlen	svc type (TOS)	total length
fragment identification		flags	fragment offset
TTL	protocol	header checksum	
source IP address			
destination IP address			
options and pad			

Headers: TCP & UDP



payload

TCP header

src port	dest port
seq number	
ack number	
hdr len / - / flags	window
checksum	urgent ptr
options and pad	

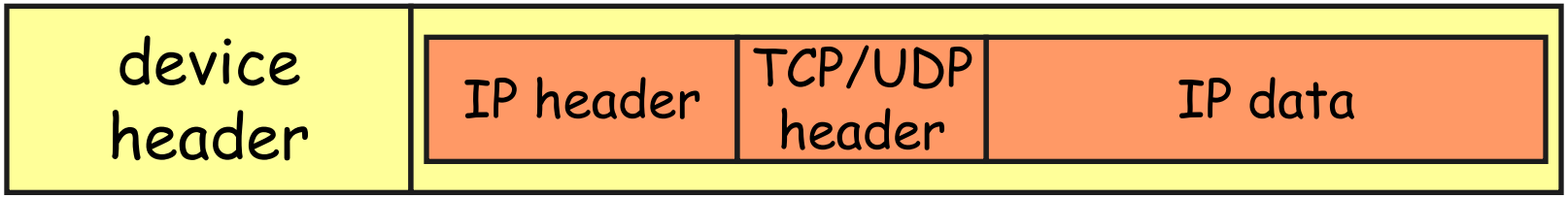
20 bytes

UDP header

src port	dest port
seg length	checksum

8 bytes

Device header (Ethernet II)



payload

6 bytes 6 bytes 2 46-1500 bytes 4

18 bytes + data

Quality of Service Problems in IP

- Too much traffic
  - Congestion
- Inefficient packet transmission
  - 59 bytes to send 1 byte in TCP/IP!
  - 20 bytes TCP + 20 bytes IP + 18 bytes ethernet
- Unreliable delivery
  - Software to the rescue – TCP/IP
- Unpredictable packet delivery

IP Flow Detection

Flow detection in routers:

- Flow: set of packets from one *address:port* to another *address:port* with same protocol
- Network controls flow rate by dropping or delaying packets
- With flow detection:
  - drop TCP packets over UDP
  - Discard UDP flow to ensure QoS for other flows

With flow detection:

- Traffic Shaping
  - Identify traffic flows
  - Queue packets during surges and release later
  - High-bandwidth link to low-bandwidth link
- Traffic Policing
  - Discard traffic that exceeds allotted bandwidth

Dealing with congestion

- FIFO queuing
- Priority queues
- Flow-based weighted fair queuing
  - Group all packets from a flow together
- Class-based weighted fair queuing
  - Based on protocols, access control lists, interfaces, etc.
- Custom queues

Inefficient Packets

- Lots of tiny packets
  - Head-of-line blocking
  - Nagle's algorithm:
    - buffer new data if unacknowledged data outstanding
- Header/packet compression
  - Link-to-link
  - Header compression (RFC 3843)
  - Payload compression (RFC 2393)
  - $ delivery vs. $ compression

Differentiated Services (soft QoS)

Some traffic is treated better than others

- Statistical - no guarantees
- TOS bits & Diff-Serv

- Use on Internet is limited due to peering agreement complexities

TOS bits

- Advisory tag in IP header for use by routers
- TOS: *Type Of Service*, 4 bits
  - Minimum Delay [0x10]
    - FTP, telnet, ssh
  - Maximum Throughput [0x08]
    - ftp-data, www
  - Maximum reliability [0x04]
    - SNMP, DNS
  - Minimum cost [0x02]
    - NNTP, SMTP

RFC 1349, July, 1992

Differentiated Services (Diff-Serv)

- Revision of interpretation of ToS bits
- ToS field in IP header
  - *Differentiated Sevices Control Point (DSCP)*

p p p d t r - -

Priority: 0-7

Reliability: normal/high

Throughput: normal/high

Delay: normal/low

RFC 2475, December 1998

Guaranteed QoS (hard QoS)

- Guarantee via end-to-end reservation

Reservation & Delivery Protocol

- RSVP: ReSerVation Protocol
  - Hosts request specific quality of service
  - Routers reserve resources
  - RFC 2205

Media Delivery Protocols

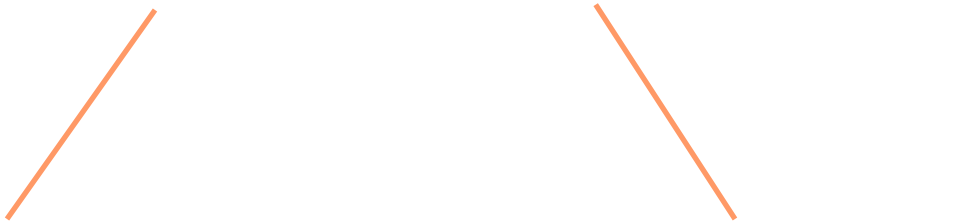
- Real-Time Control Protocol (RTCP)
  - Provides feedback on QoS (jitter, loss, delay)
  - RFC 3550

- RTP: Real-Time Transport Protocol
  - *Not* a routing protocol
  - No service guarantees
  - Provides:
    - Payload identification
    - sequence #
    - time stamp
- RTP/RTCP do not provide QoS controls

ATM: Asynchronous Transfer Mode

Late 1980's

Goal: Merge voice & data networking



low but constant bandwidth

high but bursty bandwidth

ATM

Traditional voice networking

- Circuit switching
  - Too costly
  - Poor use of resource
  - Does not lend to multicasting

ATM

- Based on
  fixed-size packets over **virtual circuits**
- Fixed-size cells provide for
  predictive scheduling
- Large cells will not hold up smaller ones
- Rapid switching

ATM

Current standard:

- 53-byte cell: 48-byte data, 5-byte header

Sender specifies traffic type upon connecting:

CBR	Constant bit-rate	*bandwidth*	Uncompressed video, voice
VBR	Variable bit-rate	*Avg, peak bandwidth*	Compressed video, voice
ABR	Available bit-rate	*-none-*	ftp, web access

ATM

Small cells → lots of interrupts

- >100,000/second

ATM hardware supports an
ATM Adaptation Layer (AAL)

- Converts cells to variable-sized (larger) packets:
  - AAL 1: for CBR
  - AAL 2: for VBR
  - AAL 3/4: ABR data
  - AAL 5: ABR data, simplified
  - AAL 6: MPEG-2 video

Questions