ANC4
(2018)

Lewis Mackenzie

ANC4 Topics

- Internet architecture and routing;
- Ethernet and VLANs;
- Congestion control and quality of service;
- Overlay networks;
- Multimedia transmission;
- Communications system implementation.

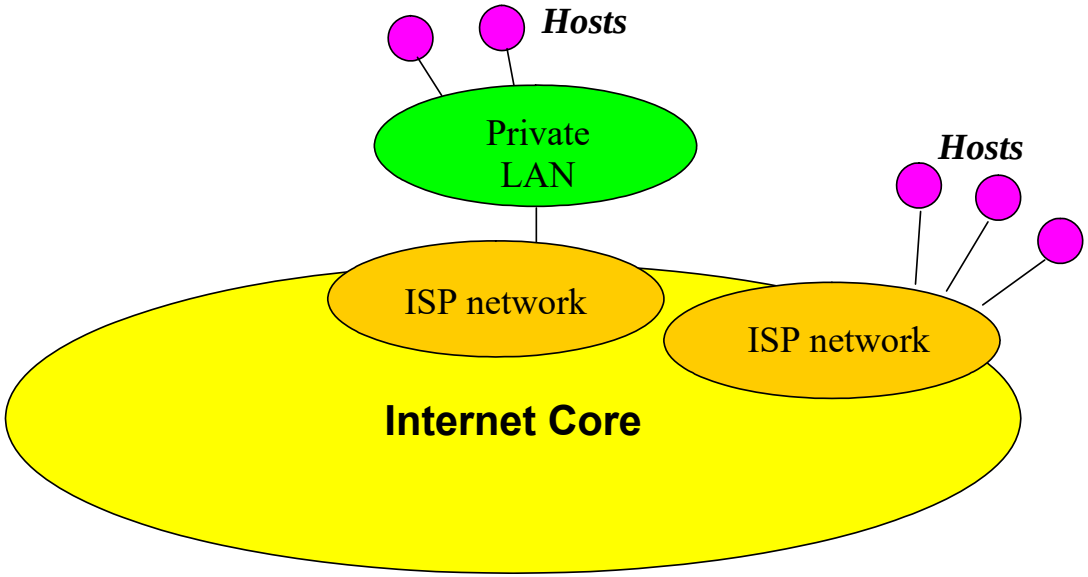
There are no compulsory texts for this course. All essential material is provided.

For background reading: see ANC4 Moodle page.

1. Internet Architecture and Routing

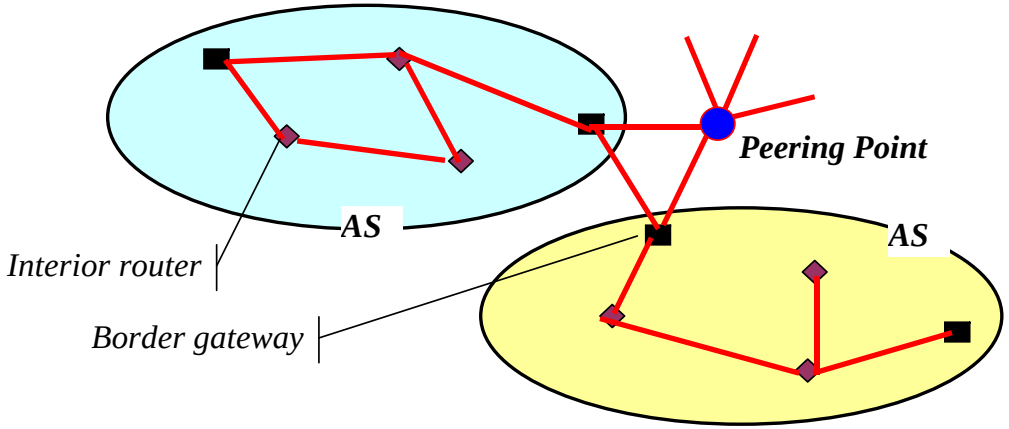
Internet Structure I

- Hosts and private LANs (mostly wired Ethernets or wireless 802.11 networks) are attached to the Internet via edge networks run by ISPs
- Customer networks may be attached to ISPs via ADSL/VDSL, cable, wireless (e.g. 3G, 4G etc), dedicated fibre etc.
- Edge (consumer) ISPs link to higher level ISPs forming what is loosely called “Internet core” (networks which provide transit to other ISPs).
- In addition to ISPs there are dedicated private networks which do not offer transit.



Internet Structure II

- Internet hosts use IP to transmit/receive data. IP packets are routed geographically. Internet is divided into regions called autonomous systems (AS) each controlled by single routing authority. E.g JANET is AS786 (for an accurate definition see RFC 1930)



- AS connected to others via border gateways (BGs). AS-AS interface may involve peering, voluntary settlement-free (sender keeps all) interconnection for traffic exchange or transit, where one AS is a customer of the other. Peering may be private (point-point) or multiple parties may link at peering point such as LINX (London Internet Exchange) under publicly agreed rules: see www.linx.net. Private peering agreements are political and usually negotiated between ostensible equals.
- Interior routers handle local traffic. Each router has routing table indexed on network portion of IP address. BG may send incoming packet to another BG or an interior router.

Internet Structure III

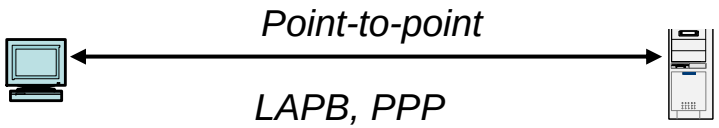
- AS may be a stub (only one external connection), multihomed (>1 connection but no through traffic) or transit (allows through traffic).
- Categorize AS's as follows:
  - Tier 1 :can access the whole Internet settlement free (everybody else is a customer or a peer). Only a few (~15) such networks exist. Global Internet backbones (e.g. AT&T, NTT, Sprint).
  - Tier 2 :peers with some networks but buys transit (is a customer) to reach some parts of Internet (typically national carriers)
  - Tier 3 : no peering, buys transit from (usually) a Tier 2 AS. Many consumer ISPs.
- BG tables are big. Border Gateway Protocol (BGP) used to update. Optimisation not feasible; aim is reachability (given settlement and peering relationships).
- Within an AS interior routers use an interior gateway protocol (IGP) to update routing tables. Examples are Routing Information Protocol (RIP) and Open Shortest Path First Protocol (OSPF). These try to optimise routes.

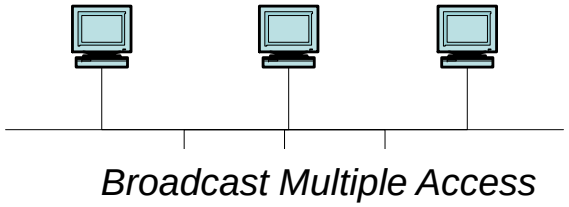
Communication Architectures

- Series of layers composing communications system form a communication architecture.
  - specifies functionality of each layer and rules for interaction between layers. Each layer has one or more protocols.
- Revision: Compare OSI & Internet architectures!
- In an architecture a layer may support multiple protocols (e.g. Internet transport layer has TCP and UDP).
- Implementation of architecture with specified protocols for each layer is a protocol stack.
- In Internet architecture at network layer all networks use IP. Currently primary version is still IPv4. We focus here on this unless otherwise stated.
- Subnet access layer in Internet sometimes called “Layer 2”. Why?
- Following this logic the IP Layer is Layer 3 and the Transport Layer is Layer 4.

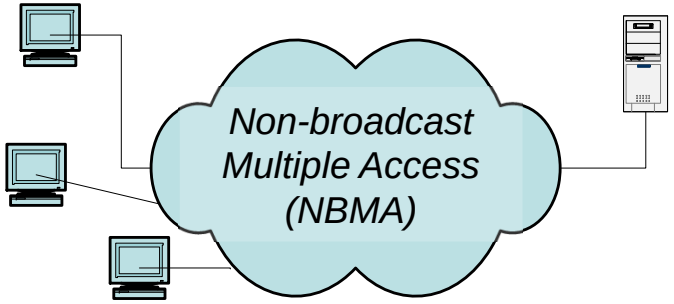
Internet “Layer 2”

- Can actually consist of complex sub-architecture with multiple sublayers.
- Provides data pipes between routers or routers and hosts.
- Different possibilities:





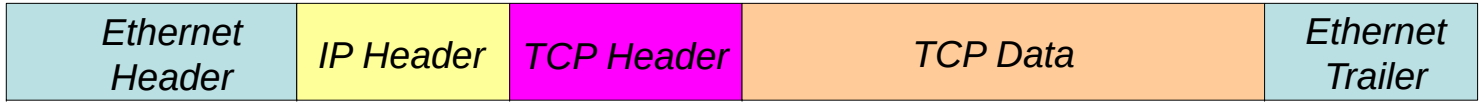
E.g. Ethernet



E.g. ATM

Encapsulation

- Consider TCP connection to host attached to an Ethernet.
  - On local net data is transmitted as Ethernet frames (max payload 1500bytes).
  - Within each Ethernet frame is an IP packet
  - Within each IP packet is a TCP segment



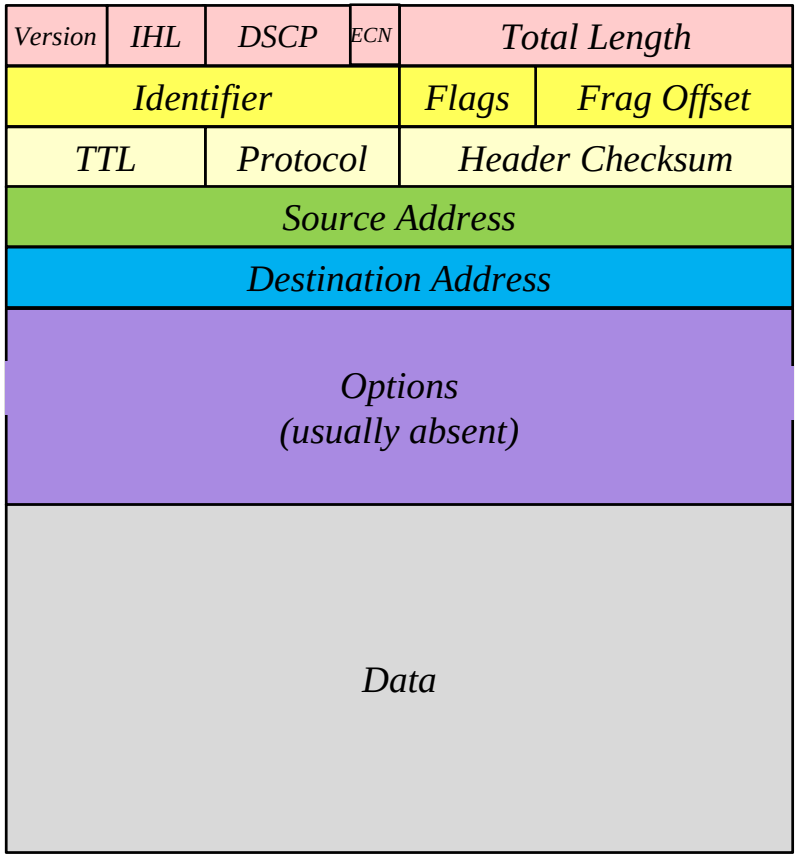
On any TCP/IP network,

  - max IP packet size is called maximum transmission unit (MTU);
  - max data in a TCP segment is called maximum segment size (MSS).

- In example above MTU=1500, MSS=1460 (if no options are used).
- MSS can be set for a given direction when a TCP connection is set up (uses the TCP header MSS option)
- Most systems will attempt to avoid necessity for fragmentation by keeping packets smaller than the minimum MTU they are likely to encounter (see later).
- Absolute minimum MTU for IPv4 is 68 bytes; for IPv6 1280 bytes
- All IPv4 hosts must be willing to receive packets of at least 576 bytes (for IPv6 1280 bytes).

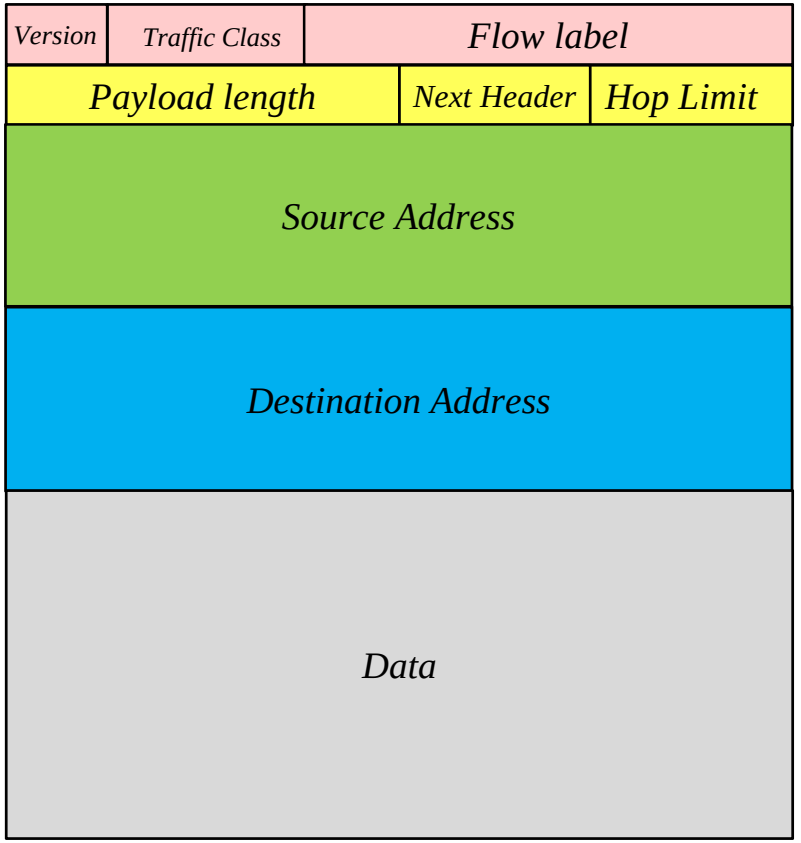
IPv4 Datagram Format

- IHL (Internet Header Length) 4-bit field giving number of words in header (5 common). Total Length is in bytes (max 2^{16} bytes).
- 6 bits DSCP allows for priorities but not widely used; 2 bits ECN is for congestion control. This byte was formerly known as Type of Service (also not much used).
- Fragments from same datagram have same Identifier. 3 flags: one unused, MF (More Fragments) indicates more fragments to come, DF means Don't Fragment.
- 13-bit Fragment Offset says where current fragment comes in its parent datagram (units are 8 bytes).
- Time-to-live used to limit packet lifetimes: decremented each hop, kills packet when 0. Often set to 30.
- Protocol field tells which type of payload datagram is carrying: e.g. protocol 6 is TCP, 17 is UDP.
- Checksum is taken over the header only.



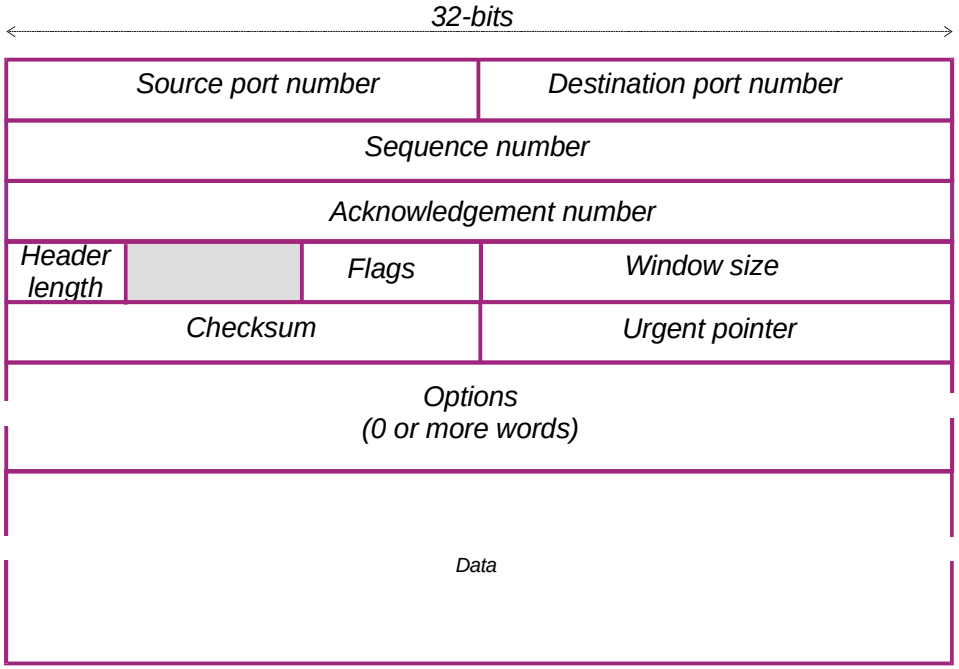
IPv6 Datagram Format (Fixed Header)

- Traffic Class 8-bit field equivalent to DSCP+ECN in IPv4
- Flow Label (20 bits) used to associate packets in flows for real-time applications.
- Payload Length is in bytes and includes any extension headers. Set to zero when jumbogram option (RFC 2675) is in use.
- Next Header (8-bits) carries type of next header. Usually set to cargo protocol number (same as Ipv4) but may indicate extension header.
- Hop Limit is like TTL field of IPv4
- Source and destination addresses are 128 bit IPv6 format.
- Several extension header types are defined: e.g Hop-by-Hop Options, Destination Options, Authentication (IPSec), Encapsulating Security Payload (also IPSec), Fragment, Routing etc.



TCP Segment Format

- Exchange of TPDUs called segments. Header of five 32-bit words. Each segment carries a sequence number. TCP sees transmission as stream of (data) bytes. Every byte in stream has a number. Sequence number of segment is number of its first data byte. Acknowledgement number, if present, is number of next expected byte.



Header length (4 bits) gives number of 32 bit words (including options).

Next 3-bit field is reserved

There are 9 1-bit flags: NS, CWR, ECE, URG, ACK, PSH, RST, SYN, FIN used for e.g. setting up connections (via 3 way handshake).

Window size is number of bytes receiver will accept (sliding window flow control)

Checksum is over TCP segment plus pseudo-header.

If URG =1, urgent pointer gives number of urgent bytes that follow sequence number.

Options have several uses: e.g. MSS negotiation (Option 2, length 4 bytes) in SYN segments.

Path MTUs

- Minimum MTU encountered along a route is called the path MTU or PMTU.
- Most modern systems will try to establish the PMTU for a TCP connection by initially sending IP packets (of a size in accordance with the initial MSS negotiation) with DF set and then seeing what happens. The technique is called PMTU discovery (RFC 1191) and it relies on the Internet Control Message Protocol (ICMP).
- If a router cannot forward a packet because DF is set it will discard the packet and inform the sender via an ICMP packet (Type 3, Code 4). This "Packet Too Big" (PTB) message contains a 16-bit next-hop MTU (for the inaccessible network), allowing the sender to adjust the next data packet accordingly.
- Some firewalls block ICMP in which case a failed PMTUD attempt receives no PTB reply. This is called a PMTU black hole (see RFC 2923).This can lead to occasional or sustained packet loss and failure of the transport connection.
- Can use ping tool to determine PMTU.

  `ping -f -n <number of pings> -l <size> <dest IP addr>`

- Ping uses an ICMP Echo Request packet which has a 8 byte header and is contained in an IP packet so the PMTU is 28 bytes bigger than the largest successful **`<size>`.**

More on PMTUD

- MSS negotiation uses MTU of local network of host. This leads to assumed PMTU based on minimum MTU of end networks. If no MSS is specified the Internet default of 536 is assumed. Commonly the initial PMTU is 1500. Why?
- After MSS negotiation, PMTU discovery is conducted at the IP layer. If a reduction in the PMTU is detected (due to an intermediate hop) it must be reported to the transport layer which is responsible for packetization.
- If ICMP is blocked, some routers attached to links with MTUs lower than 1500, reduce the MSS of TCP connections passing through them. This is called TCP clamping but is not an ideal solution. Why?
- A robust mechanism for PMTUD in the absence of ICMP is described in RFC 4821. This requires active participation by the transport layer and is called Packetization Layer PMTUD (PLPMTUD).
- A PMTU is associated with a path to a particular destination. Re-computation for each active destination is desirable at regular but not too frequent intervals. This leads to the concept of cached PMTU values held by the IP Layer with each entry having a finite lifetime (10 minutes is typical).
- PMTU is also defined for IPv6 (see RFC 1981).

ICMP

- ICMP (RFC 792) carries control messages between hosts and routers. Generated in various circumstances including diagnostics, routing or IP errors (see also RFC 1122: Requirements for Internet Hosts).
- Carried in IP datagrams with Protocol Number 1.
- First byte of header is 8-bit type field, second is 8-bit code (additional info). This is followed by 16-bit checksum and a number of 32-bit words (format is determined by type + code) e.g. Type 8 (Echo Request) can carry arbitrary data (must be included in the Echo Reply).
- Some example message types.
  - Type 3: destination unreachable (can't forward datagram). Code gives reason (4 means datagram is too big (see PMTUD).
  - Types 9/10: router advertisement/router solicitation (identify routers on local network: see RFC 1256).
  - Type 5: routing redirect (tell source there is a better router to use)
  - Type 4: source quench (sender slow down) deprecated by RFC 6633 (2012)
  - Types 8/0 echo request/reply (used in **`ping`** and **`tracert`** tools)
  - Types 13/14 timestamp request/reply
  - Type 11: time exceeded (sent to source when a packet has reached ttl limit)

More on ICMP

- ICMP is used for reporting non-transient error conditions or for querying network with request reply.
  - No port numbers
  - No client-server concept
  - No need for services and ports to be listening.
- Special conditions (RFC 792)
  - No ICMP error messages in response to ICMP error messages
  - No ICMP error messages in response to multicast or broadcast IP packet.
  - No ICMP error messages in response to packet not from a unique host
  - For fragmented datagram errors, ICMP replies only to first fragment
- ICMP is widely used but not designed for security and is vulnerable to exploitation. An attacker can use the protocol to gather information (reconnaissance), launch denial of service attacks or act to implement covert channels.
- IPv6 has very similar ICMPv6 defined in RFC 4443

ICMP Security Issues

- ICMP sweep attacker sends ICMP requests to target network range; sees who replies (automated pings).
- Traceroute (Windows tracert) relies on a response with ICMP Type 11 “Time exceeded” packets and can be used to map out a network. Traceroute can probe with ICMP or UDP packets. UDP probes often sent to Port 53 (DNS port) to try to penetrate firewalls. Attacker wants to learn firewall filtering policies/open ports.
- Firewalk is a tool that traceroutes a network firewall, then pings hosts one hop behind and waits for an ICMP error message: if it gets one, the ICMP request got past the firewall and the host(s) exist(s); if not, the attacker may learn something about the firewall filter rules. Learning such rules greatly helps the planning of an attack
- Inverse Mapping: attacker sends ICMP replies. Often firewall allows these even if blocking incoming requests. If router replies with Type 3 Code 1 packets (Host unreachable) this shows hosts exist at given addresses.
- OS Fingerprinting different OSs respond slightly differently in the data they include in a Destination Unreachable response. This can give away info about which OS a host is running.
- Router impersonation: Attacker can hijack the router solicitation protocol
- Ping of death; Attacker sends oversized packet (via fragmentation)
- Denial of service (Smurf attack): spoof IP address of host and send ICMP Echo Request to directed broadcast address. Replies overwhelm target.
- ICMP tunnelling exploits arbitrary data length in ICMP echo traffic to transfer information between an infected machine and a client (e.g. Loki, Phrack Magazine, 49/6, 1996).
- Requires compromised target machine so tunneling software has admin/root privileges.
  - Can be used to tunnel IP over ICMP (ICMPTX).
  - Very hard to detect. Defence is blocking ICMP, limiting echo packet size, or Intrusion Detection System (IDS) analysis.

ICMPv6

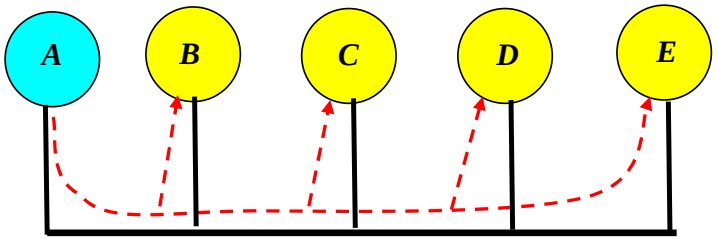
- ICMPv6 (RFC 4443) has many similarities to ICMPv4 but here also supports other protocols such as Neighbour Discovery Protocol (NDP), Secure Neighbour Discovery (SEND), Multicast Router Discovery (MRD)
  - NDP subsumes role of ARP and Router Solicitation/Advertisement
- Fixed header is just: 8-bit Type, 8-bit Code, 16-bit Checksum.
- Types 0-127 are used to signal error conditions
  - Type 1: Destination Unreachable
  - Type 4: Packet Too Big
  - Type 3: Time Exceeded
- Types 128-255 are information messages.
  - Some like ICMPv4. e.g. Echo Request (128), Echo Reply (129),
  - Some are extensions supporting NDP, SEND, MRD, etc.
- ICMPv6 has many of the same vulnerabilities as ICMPv4. E.g.
  - Covert channels are possible via Echo Request/Response
  - Router Impersonation via NDP Router Advertisement (Type 134)
- Fragmentation by router is not allowed in IPv6, so any packet that exceeds its PMTU will elicit a PTB response.

Layer Addressing

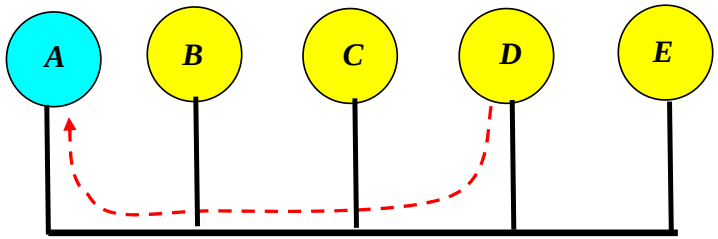
- Layer 2 address is network point of attachment (NPA) or MAC address:
  - format varies but most common is IEEE 802 format 48-bits;
  - flat namespace;
  - each interface has independent MAC address;
  - hardwired.
- Layer 3 addresses are 32-bit IP numbers:
  - one per interface for any system using Layer 3 routing (one-one correspondence with MAC addresses)
  - in bridged (layer 2 switched) system can have multiple MAC addresses to one IP number
  - IP addresses are hierarchical (network number concatenated with host number);
  - soft address.
- 16-bit Layer 4 subaddresses (port numbers) identifying TCP/UDP ports:
  - Uses system of well-known ports associated with particular services;
- In Windows use **`ipconfig /all`** to determine MAC and IP addresses for each interface on an Internet host. In Unix ifconfig is similar.

Address Resolution

- Typically host or router wants to forward IP packet to destination locally connected (broadcast LAN or NBMA network).
  - Use Layer 2
  - Must resolve IP address to Layer 2 address.
- In broadcast LAN usually use address resolution protocol (ARP): local broadcast on LAN.



ARP request with IP address D

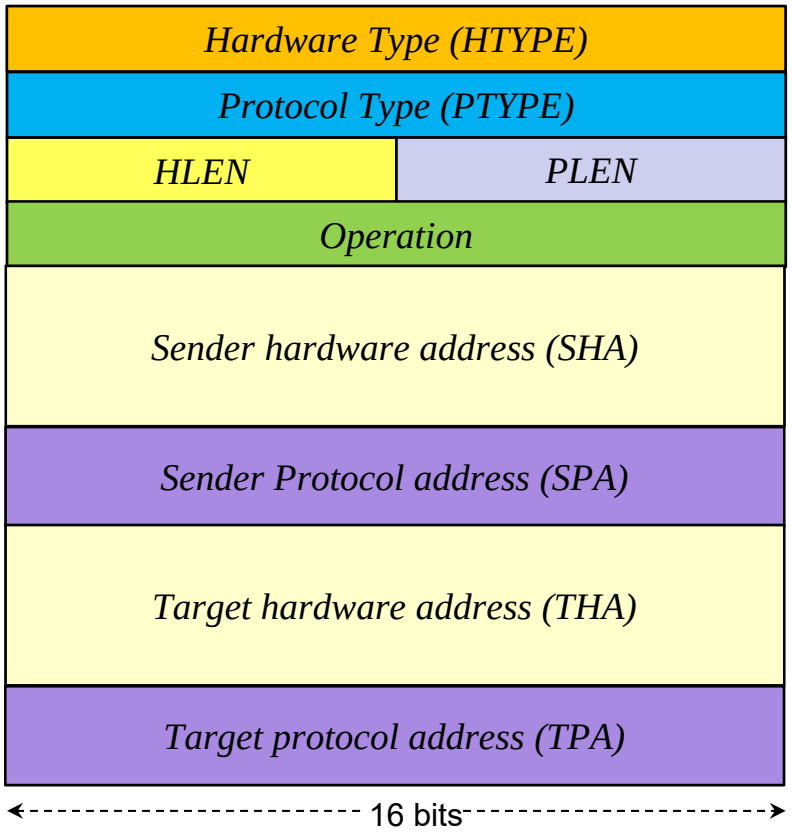


ARP response with MAC address for D

- Stations maintain ARP cache (examine using **`arp -a`**).
- Conventional ARP relies on LAN broadcast so in NBMA network, need different approach: e.g. ARP Server.

ARP

- ARP (RFC 826) is general address resolution mechanism. Packet format is shown:
  - HTYPE specifies LAN network type (Ethernet =1)
  - PTYPE specifies internetwork protocol type (IPv4 is 0x0800)
  - HLEN is hardware address length
  - PLEN is protocol address length
  - Operation is 1 for request, 2 for reply.
- ARP probe is request sent with SPA = 0.0.0.0
  - Used to check that a newly assigned IP address is not already in use (assignment error)-RFC 5227.
- ARP can make announcements (aka gratuitous ARP message) usually broadcast as ARP request containing senders address in TPA field (SPA=TPA) and THA = 0.
  - Used to update caches with new ARP information so as to cut down on future traffic
- Like ICMP ARP is vulnerable to attack. ARP spoofing involves the generation of fake ARP messages to poison ARP caches opening way to e.g. man-in-the middle and denial of service attacks.
- One solution is static cache entries for key systems.



Exercise: Explain how these attacks might be conducted and suggest ways to defend against them.

Neighbour Discovery Protocol (IPv6)

- NDP (RFC 4861) uses five different ICMPv6 types:
  - Router Solicitation
  - Router Advertisement
  - Redirect
  - Neighbour Solicitation
  - Neighbour Advertisement
- NS messages are multicast when resolving an address and unicast when checking reachability.
  - Uses solicited-node multicast address formed from bottom 24 bits of target's IPv6 address (prefixed by ff02:0:0:0:0:1:ff00::/104
  - This is more efficient than broadcast and only goes to nodes with same bottom 24 address bits.
  - Solicited NA messages are targeted back at originator of NS; unsolicited NA messages go to all-nodes multicast (same approach as ARP).
- NDP has same security issues as ARP.
- Incorporation of Router Discovery allows a new node to auto-configure its address

Sending data via TCP (the story top-down)

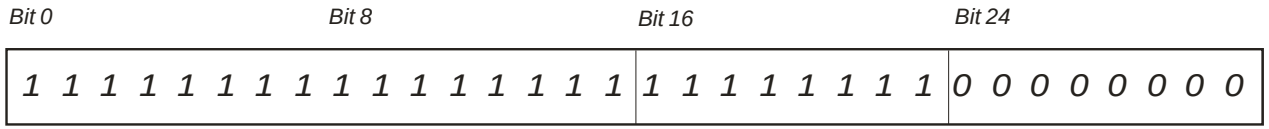
- Client app submits data to TCP layer giving destination IP address and port number (or equivalently, socket ID).
- TCP layer prepares TCP segment consistent with MSS (established at outset) and attaches 3 word IP pseudo-header
  - source and destination IP addresses, protocol number, overall segment length
  - TCP (and UDP) header checksum computed over segment + pseudo-header
  - pseudo-header is not transmitted but is a means of passing info between TCP and IP layers. If packet gets misdirected (changed destination IP address), checksum will fail.
- TCP layer submits segment + PH to IP layer
- IP layer prepares IP packet containing segment
- IP layer invokes routing function and consults forwarding table to determine next hop (to destination host or next router).
- IP layer determines forwarding MAC address (via e.g. ARP).
- IP layer submits IP packet to network access layer.

IP Addressing

- Recall that IP addresses are divided into 5 classes A to E.
- A-C addresses are divided into 2 fields: network number and host number.
  - **A from 0.0.0.0 - 127.0.0.0**
  - **B from 128.0.0.0 - 191.255.0.0**
  - **C from 192.0.0.0 - 223.255.255.0**
- Class D addresses (**224.0.0.0-239.255.255.255**) are for multicasting.
- Class E addresses are reserved.
- Three ranges are reserved for private networks:
  - **10.0.0.0** **1 Class A network**
  - **172.16.0.0 – 172.31.0.0** **16 Class B networks**
  - **192.168.0.0 – 192.168.255.0** **256 Class C networks**
- Network with private addresses must not be directly connected to Internet (must use address translation at gateway).
- A few other regions of the space are reserved: e.g 127.0.0.0 is loopback.
- Network number space is under control of IANA but allocations are delegated to Regional Internet Registries.

Address Masks

- Rigid use of class A, B, C networks (classful addressing) has proved inefficient.
- Motivated introduction of classless addressing where network number can be any number of bits long.
- Can indicate split using address mask (also quoted in dotted decimal notation).
  - bitwise AND of mask with address gives network number



Mask

IP Number

- Classless Inter-Domain Routing (CIDR) notation uses /n suffix after address to indicate number of bits in network number.
  - Thus 130.209.240.50/20 means a network number of 130.209.240.0, host number of 50

Special Addresses

- Network address: address with host number of 0.
- Network number zero means “this network”.
- 0.0.0.0 means “this host” (used when host doesn’t know its IP number).
- 255.255.255.255 (limited broadcast address): all hosts on current physical network.
- Directed broadcast address (host number all 1s) means all hosts on all subnets of current network.
- Loopback address (any address beginning with 127) goes straight from host’s output to its input.

Forwarding

- Internet routing nodes use destination IP address to direct IP packets along next hop.
- Use network number of destination IP address to consult routing table (a.k.a. forwarding table).
- Each table entry refers to some network *A* and has the structure:

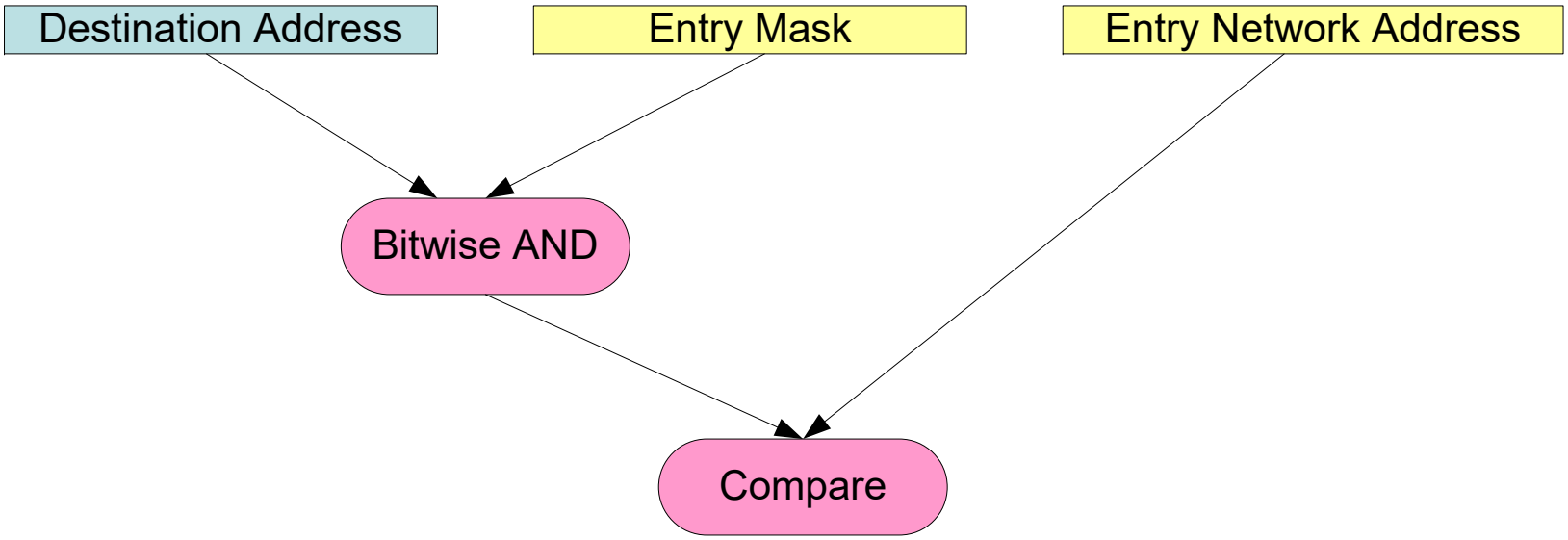
  `network address  network mask  gateway  interface  metric`
- Where
  - **`network address`** & **`network mask`** are the address and mask of A respectively;
  - **`gateway`** is IP address where any packet trying to reach A should be sent next;
  - **`interface`** is the IP number of the routing node's interface that should be used;
  - **`metric`** is the number of hops still to go to *A*.
- If *A* is directly attached to the node, gateway is the same as interface.
- If *A* is a remote network (no interface to node) gateway is IP address of a router on the local network.

Matching Routing Table Entries

- For each entry, destination address in IP packet is bitwise ANDed with network mask and result compared with network address. If there is a match, entry is used.
- If there are multiple matches, the longest network mask is preferred
  - E.g packet with address 130.209.240.50 and table with entries

    130.209.240.0 255.255.240.0 …

    130.209.0.0 255.255.0.0 …
  - Will be routed by the first entry



Host routing tables

- Each network host has (assume one network interface) an IP number, *N*, a mask called a subnet mask, *S*, which identifies the network it belongs and an address for a local default gateway, *G*.
  - Host uses these items to construct routing table.
- Every routing table will contain a default route with network number 0.0.0.0 and mask 0.0.0.0.
  - Every address matches this, so it will be used if no other match is found.
- Every routing table includes a loopback route which matches any address of the form 127.x.y.z and routes to the special loopback address 127.0.0.1.
- Other routes always present include: directly attached network; local host, network broadcast; limited broadcast and multicast.
- To examine the forwarding table on a Windows host use:

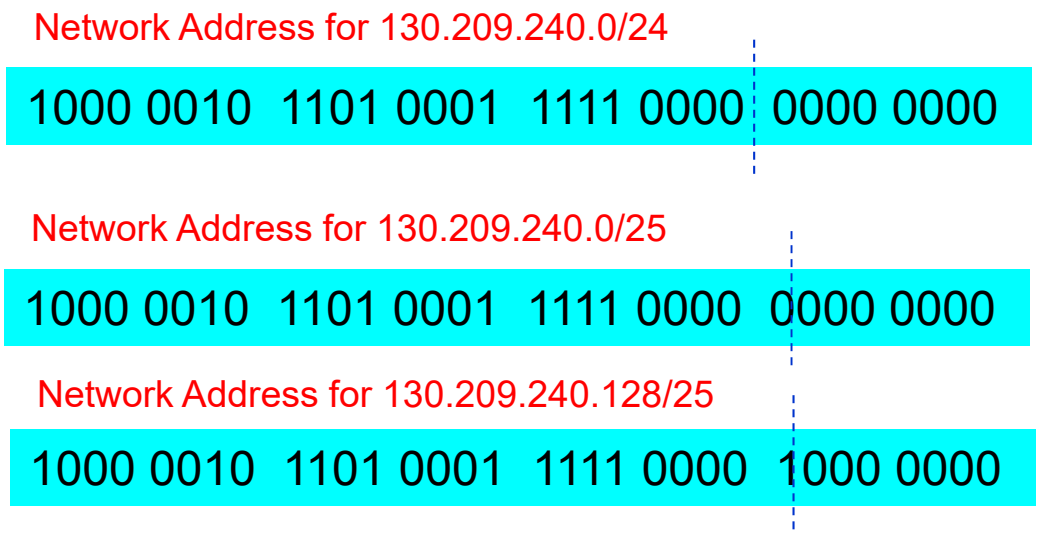
```
route print
```

Routers

- Routers have two or more NICs.
  - Multihomed conventional computer can act as a router (e.g. Windows Server includes software to act as an IP router);
  - Modern high performance routers use ASICs to minimise delay
- IP layer on router uses routing table to decide how to forward any packet arriving on any interface.
- Routing tables are indexed on network number.
  - For some routers (e.g. in backbone networks), routing tables can be very large.
  - Routing table sizes can be reduced by grouping geographically close networks with adjacent network numbers into one entry.
- Routers generally use adaptive routing with a routing protocol (e.g. RIP, OSPF, BGP etc) operating between routers to allow new information to update routes.
- Routers use a routing algorithm to compute new optimal routes on the basis of information conveyed by routing protocol.

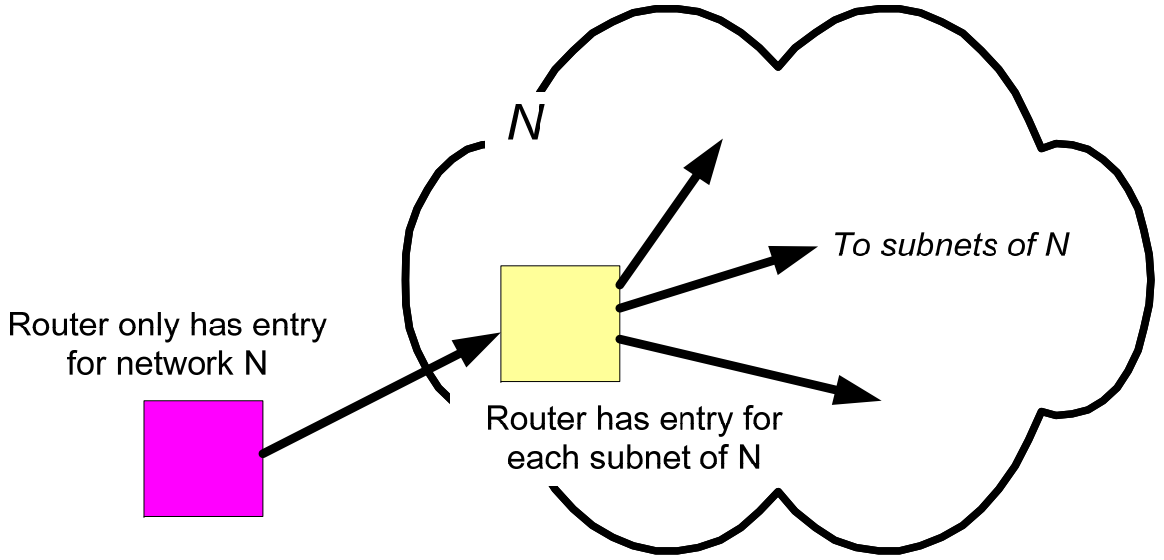
Subnetting

- Subnetting allows a network to be split into smaller sections, or subnets, by splitting original host number field into subnet field and new host number field.
- This is done by using an extended subnet mask.
- Example: split network 130.209.240.0/24 into 2 equal subnets 130.209.240.0/25 and 130.209.240.128/25.



Subnet Routing

- Hosts in different subnets have different subnet numbers. Routing table will send datagrams targeted to other subnets to a router even if the targets are on same physical network
- External routers are not aware of subnets in a network. Only when a datagram enters a network does it get routed according to subnet.



Supernetting

- Allocate multiple Class C networks with adjacent addresses to one organisation in same geographical vicinity.
  - E.g use 22-bit mask to group 4 Class C networks together
- Routers far from such supernetted networks need only one entry
  - `xxxxxxxx.xxxxxxxx.xxxxxx00.00000000  255.255.252.0  …`
- Routers nearby have four distinct entries
  - `xxxxxxxx.xxxxxxxx.xxxxxx00.00000000  255.255.255.0  …`
  - `xxxxxxxx.xxxxxxxx.xxxxxx01.00000000  255.255.255.0  …`
  - `xxxxxxxx.xxxxxxxx.xxxxxx10.00000000  255.255.255.0  …`
  - `xxxxxxxx.xxxxxxxx.xxxxxx11.00000000  255.255.255.0  …`
- This reduces size of routing tables in backbone routers.
- Group of supernetted networks is a CIDR (Classless Inter-domain Routing) block.
- Note that since masks are used to describe extent of block, the number of component networks must be a power of two.
- In CIDR environment, routing protocols must be able to exchange mask information. RIPv1 is not CIDR compliant; RIPv2 and OSPF are.
- A network address/mask combination may refer to one or multiple networks. For the purposes of routing such a combination is called a routing prefix.

Routing Algorithms

- View abstractly first.
- Aim is to establish best route between every source-destination pair in current circumstances.
- Shortest path routing:
  - Assign cost (e.g. hops, latency, financial) to every link
  - Find paths that minimise cumulative cost
- Two broad approaches:
  - Source routing (path is computed at source and sent with packet)
  - Per-hop routing (each node makes routing decision)
- Per hop requires routing tables at each node
- Best approach is to allow each node to compute its own table.
- Internet uses per-hop routing.

Routing: general

- Routing can be static (one table computation) or adaptive.
  - If adaptive routing is used, how often should updates be undertaken?
- Per-hop routing tables can be computed centrally and issued, but this is not popular. Why?
- Internet uses decentralised adaptive routing. Routers exchange routing information to allow adaptation of tables: RIP, OSPF, BGP.
- General adaptive routing problems include:
  - Different picture of network at different nodes (may lead to looping)
  - Path oscillation
- Two common approaches to computing shortest paths:
  - Link state routing (Dijkstra algorithm);
  - Distance vector routing (Bellman Ford algorithm).

Link state routing

- Each node checks which nodes are attached to it by sending HELLO packets.
  - Functioning connection with a neighbour is an adjacency.
  - Subsequent periodic use of HELLOs allows routers to detect failures of links or neighbours.
  - Once a node knows its neighbours it can establish link costs using any chosen metric (e.g. hops, bandwidth, time delay).
  - Newly discovered link costs are flooded as messages called link state advertisements (LSAs) to all other network routers.
- Every node maintains database of link costs
  - Used to compute a shortest path tree for each node.
  - Can be run in centralised or distributed form.

Distance-vector routing

- In this approach, each router exchanges with its neighbours the distance column (distance vector), suitably indexed, from its routing table.
- A router compares its own current routes against the new ones it has been offered and adjusts its table accordingly if it is offered a better one.
- Dangers:
  - Looping
  - Bad news travels slowly (count-to-infinity problem)
- Can be executed centrally but is primarily designed for distributed use.
- No need for link database at every node so more economical in terms of storage.
- In practice, not as stable as link state routing, but easier to implement.
- Enhancements to the algorithm such as spilt horizon (do not advertise routes back in the direction from which they were learned) can help speed convergence in *some* situations.

RIP

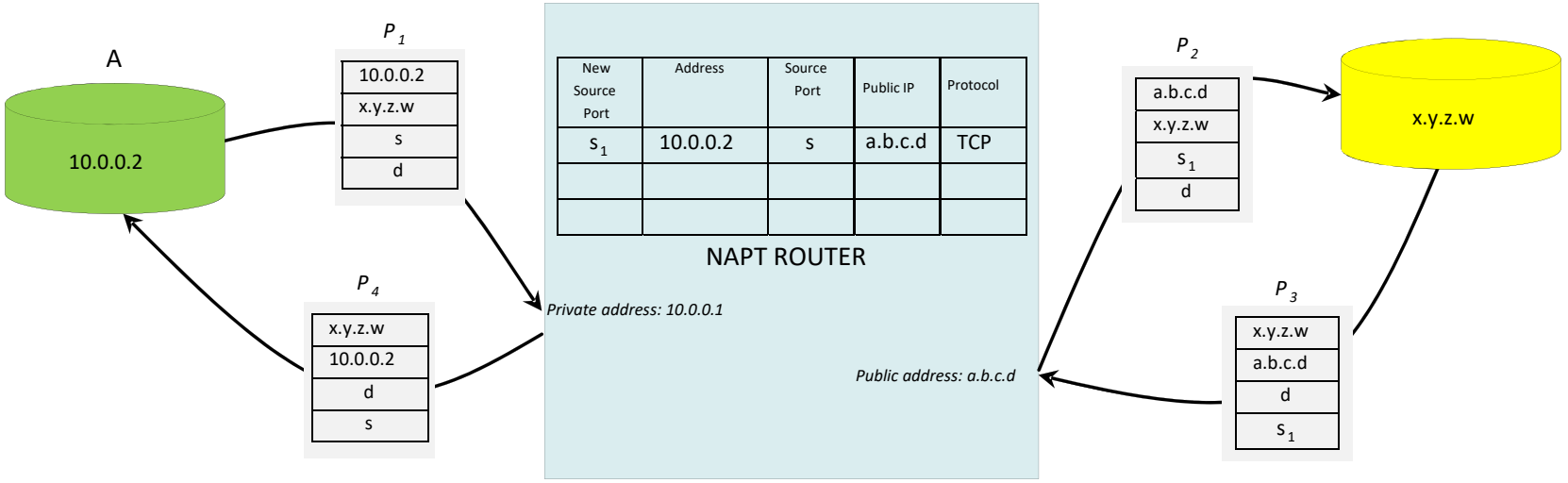
- RIP (v2 is RFC 2453) uses distance-vector routing.
- Router advertises distances in hops every 30 secs
  - RIP message is list of routes (address, mask, next hop, distance)
  - Messages are carried in UDP PDUs
- RIP has maximum hop count of 15
  - In RIP 16 is infinity (unreachable)
  - Only usable on small to medium networks
- RIP v2 has several enhancements:
  - supports classless addressing;
  - supports authentication (why?)
  - uses multicasts instead of broadcasts on multiaccess networks
  - supports route tags.

OSPF

- OSPF (RFC 2328) uses link state routing with any metric.
- Every router builds a Link State Database (LSDB).
  - Composed of a Link State Advertisements (LSAs)
- Routers find neighbours with HELLO packets
  - Build an adjacency
  - Routers synchronise LSDBs with neighbours when adjacency formed.
- Routers must keep LSDBs up to date
  - If local change is detected, router sends a Link State Update Packet to send new LSAs to rest of network (flooding via multicast)
  - Routers continue to check status of neighbours in case of fault
- To control size of task, OSPF can divide an AS into several routing areas.
  - Connected by area border routers (ABRs)
  - Area identified by 32-bit Area ID .
  - Always have one backbone area (Area 0).

Network Address Translation

- NAT router translates between private address and one or more public ones.
- In a common situation, there are fewer public IP addresses than private IP/port pairs wanting Internet access at the same time. Use process called NAPT in RFC 2663 (summarised below).
- NAT router table maps private address/port pair (IP_{Priv}, s) onto an external address/port pair (IP_{Pub}, s_1) called a server reflexive address). It is this that is the source address in any public packets that are sent to the target of the communication.
- Router records (IP_{Priv}, s) and (IP_{Pub}, s_1) together with protocol in use (TCP, UDP) in an internal table. These 5 items form a so-called 5-tuple. This table entry remains valid until inactivity timeout deletes it.



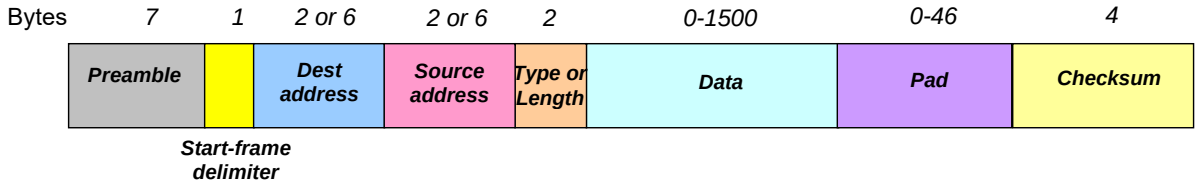
NAT Traversal.

- NAT connections are initiated by private hosts
- But once a 5-tuple exists, questions arise:
  1. If same (IP_{Priv}, s) initiates a connection to a new target, can same SRA be used?
  2. Can remote device initiate a connection through the existing 5-tuple?
- In most forms answer to Q1 is Y. Exception is symmetric NAT.
- Answer to Q2 varies:
  - Full cone NAT, any remote device can initiate.
  - Address restricted cone NAT, any port on same remote IP can initiate
  - Port restricted cone NAT, only same remote IP and port can initiate
- Some protocols fail over NAT because private host is required to communicate its contact address in segment payload. Unfortunately this address is private, so peer device cannot make contact. Examples include VoIP, gaming and peer-peer protocols.
- To let such protocols work a private host needs to be able to discover its SRA.
- One means of doing this is via Session Traversal Utilities for NAT (STUN) which uses a STUN server on the public Internet to reveal a private client's SRA (RFC 5389).
  - Unfortunately STUN does not work with symmetric NAT. Why?
- For symmetric NAT solution is TURN (Traversal Using Relays around NAT): see RFC 5766. Uses TURN relay server to avoid symmetric NAT problem BUT expensive in resource and delay.
- Interactive Connectivity Establishment (ICE) seeks best approach in given scenario (RFC 5245)

2. Layer 2

Ethernet Operation

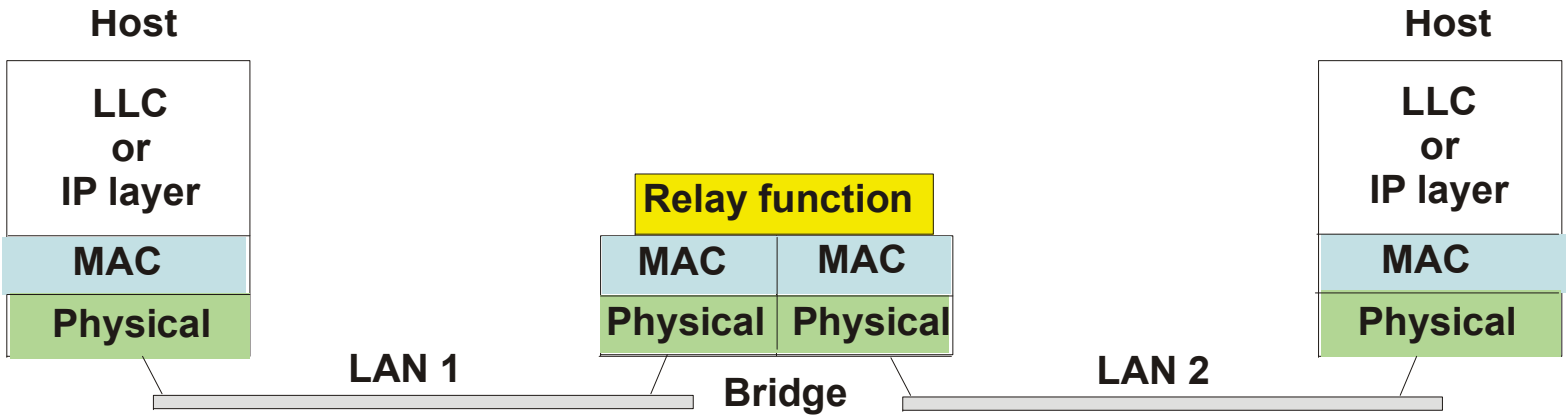
- Ethernet uses frames with format:



- Checksum is CRC-32.
- Ethernet addresses are 48-bits long and globally unique.
- Data field size is can be 0-1500bytes
- Minimum length 64 bytes. If data field too small, pad field must compensate. 64 bytes corresponds to time to cross max network size twice (51.2μs at 10Mbps and 5.12μs at 100Mbps). If a packet collides, its transmitting station must know (and abort) before it finishes sending it. Malformed short packet formed by collision is a runt.
- Type field identifies payload (e.g. 0x0800 is IP).
- In IEEE 802.3 “Type” replaced by “Length” (for compatibility Type > 0x05DC). 802.3 uses IEEE simplified variant of HDLC, Logic Link Control (LLC) for LANs. 0x8870 is reserved to indicate a jumbo frame (up to 9000 bytes) option used in Gigabit Ethernet.
- LLC and IP can be thought of as operating at layers higher than MAC. IEEE views MAC and LLC as sublayers of OSI Data-Link Layer.
- IP packets carried over Ethernet may have to be fragmented (max IP size, 64Kbytes).
- There is a required inter-frame gap of 12 bytes.

Bridges

- Bridges are intended to link LANs together



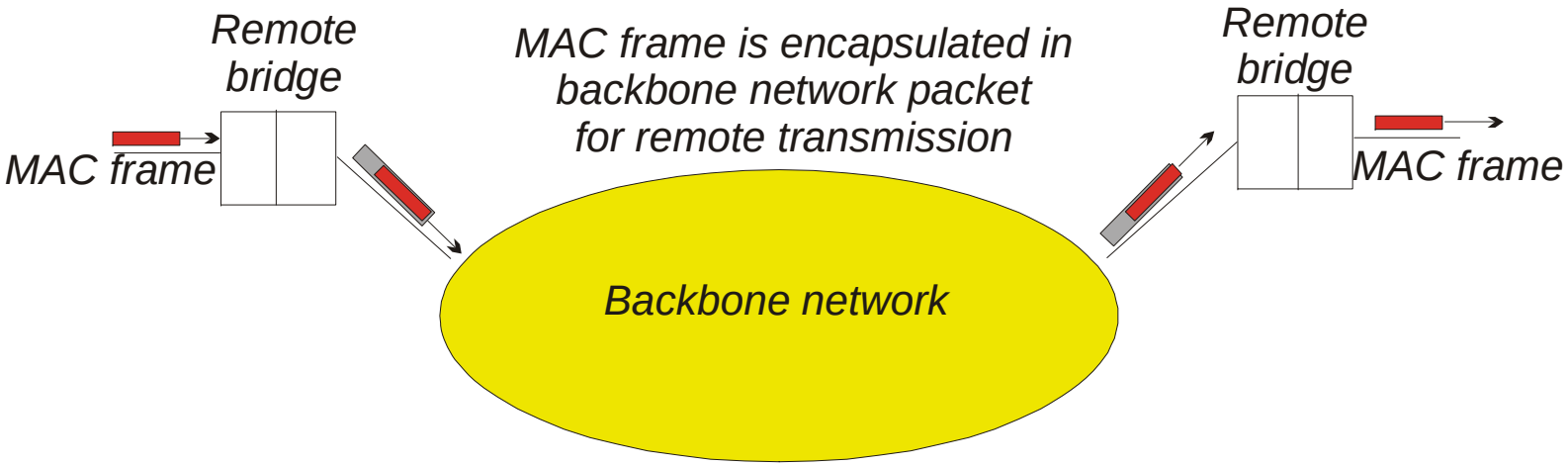
- Transforms one MAC frame format to another
  - No packet (layer 3) level functionality
  - Relies on connected LANs having similar formats
- IEEE 802 defined transparent bridges (IEEE 802.1D)
  - Transparently translates frames from one 802 format to another
  - Routes based on intelligent interpretation of destination MAC address

Routing without routers

- Bridges may be used to construct internetworks.
  - Route without Layer 3 involvement
- Layer 2 addresses have no hierarchical structure
  - Point of attachment of hosts may be altered at any time
- Transparent bridges try to learn where specific addresses currently located
  - Observe source addresses (backward learning)
  - Build filter table (output port indexed on destination MAC address)
  - Floods frames whose destination is unknown to all ports
  - Can lead to looping behaviour
- 802.1D tries to build spanning tree which all bridges share
  - Use spanning tree protocol (STP) with BPDUs directed to a reserved multicast address
  - Establish one root bridge
  - Each LAN gets one designated bridge forwarding frames from root direction.
  - Original SPT can be quite slow to converge (10s of seconds)
  - Rapid STP (RSTP) specified in 802.1w is order of magnitude faster (see 802.1D-2004)
  - Shortest Path Bridging (SPB) in 802.1aq allows all links to be active (802.1aq-2012)

Remote bridging

- Two bridge LANs not directly connected can tunnel through an intermediate network: remote bridging.
- MAC frames encapsulated in backbone network PDUs.
- Internet may be used as vehicle to connect multiple private LANs and individuals: forms a virtual private network (VPN).

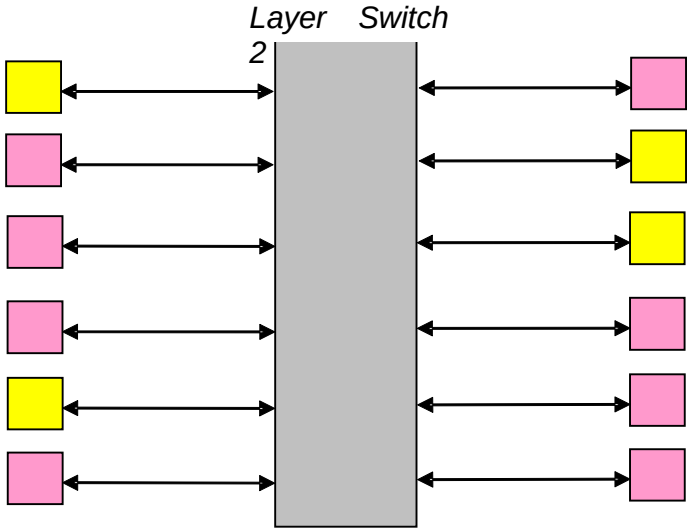


Layer 2 Switches

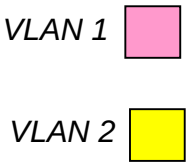
- Layer 2 switches are similar in function to bridges but:
  - Faster (based on more recent technology);
  - More restricted (usually connect only LANs of same type)
- Ethernet switches developed from 10BaseT technology
  - Very widely used for 100Mbps, almost exclusively for gigabit Ethernet
  - Can be used to connect segments together
- Allow full duplex Ethernet (802.3x-1997)
  - No collisions, frames discarded if switch buffers full
  - Effectively no concept of collision domain
  - Increases maximum size of Ethernet
  - PAUSE frame defined to implement flow control

VLANs

- Usually each LAN is a broadcast domain.
- Using switches, an (Ethernet) LAN may be divided into several VLANs (virtual LANs), each its own broadcast domain.
- VLANs may be port based (isolated), overlapping, MAC address based, protocol based; or IP subnet based.
  - In every case, intelligence required is integrated into the switch



Each switch port belongs to just one VLAN



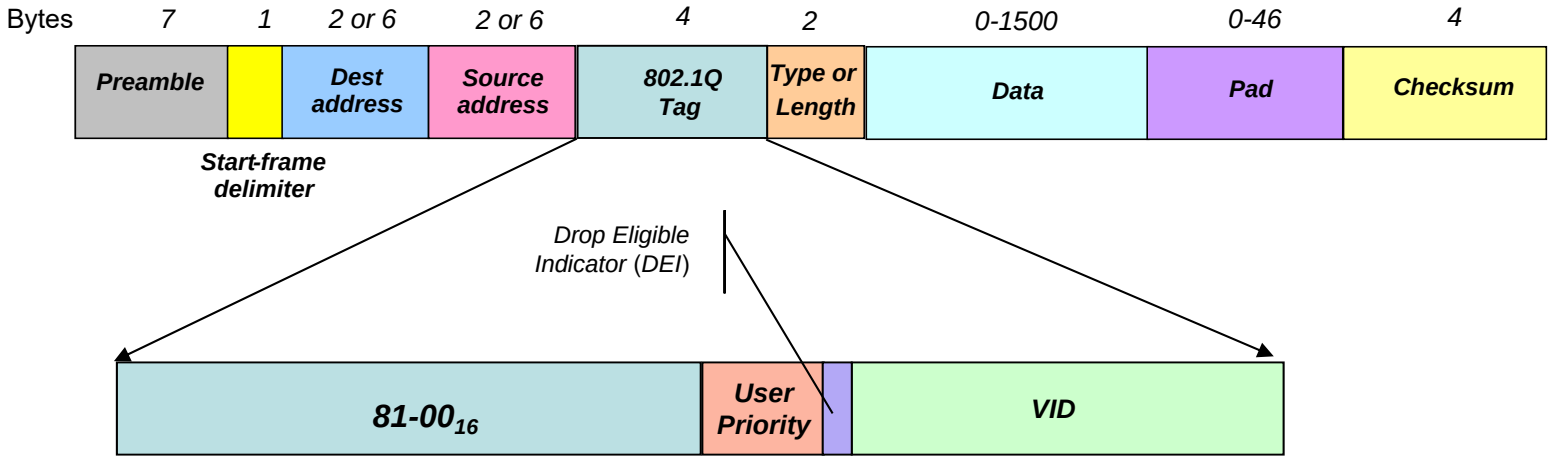
- Challenge is VLANs distributed over several switches.
- Problem is equivalent to ensuring multicast frames are directed to hosts in a multicast group

VLANs and Multicasts

- First note similarities between multicasting and multi-switch VLANs.
- 802.1D (1990) handled multicasts by forwarding on all ports. In 1998 two key extensions: 802.1p addendum to 802.1D-1998 enhanced support for groups across interconnected switches; 802.1Q-1998 (separate) is VLAN standard.
- Ideas introduced in 802.1p:
  - filter table are updated when a station joins a group; frames multicast to each group can then be forwarded intelligently.
  - *Generic* protocol called GARP (generic attribute registration protocol) which lets stations request receipt of traffic with specific attributes. Filter tables in GARP-compliant switches then updated accordingly.
  - Frame priority scheme for differentiating types of traffic (used in 802.1Q: next slide).
- GARP framework used for 2 derived protocols: GMRP for multicast group membership defined in 802.1p; GVRP for VLANS defined in 802.1Q-1998.
- GARP later superseded by a successor protocol, MRP (Multiple Registration Protocol) defined in 802.1ak addendum to 802.1Q-2005.
- 802.1Q-2014: incorporates relevant 802.1D into 802.1Q standard (unification).

802.1Q

- 802.1Q is aimed at supporting VLANs. From outset (1998) defined new frame format.
- Each VLAN is allocated a 12-bit VLAN ID.
- 802.1Q adds a 4-byte tag to Ethernet frame containing VID and a 3-bit priority field.
- Most stations are not 802.1Q aware. Untagged frames are tagged at port where they enter a switch.
  - Tag includes port VID (PVID) and priority.
  - Frame then transferred to every port willing to accept that VID
- Links joining switches (trunks) carry tagged frames

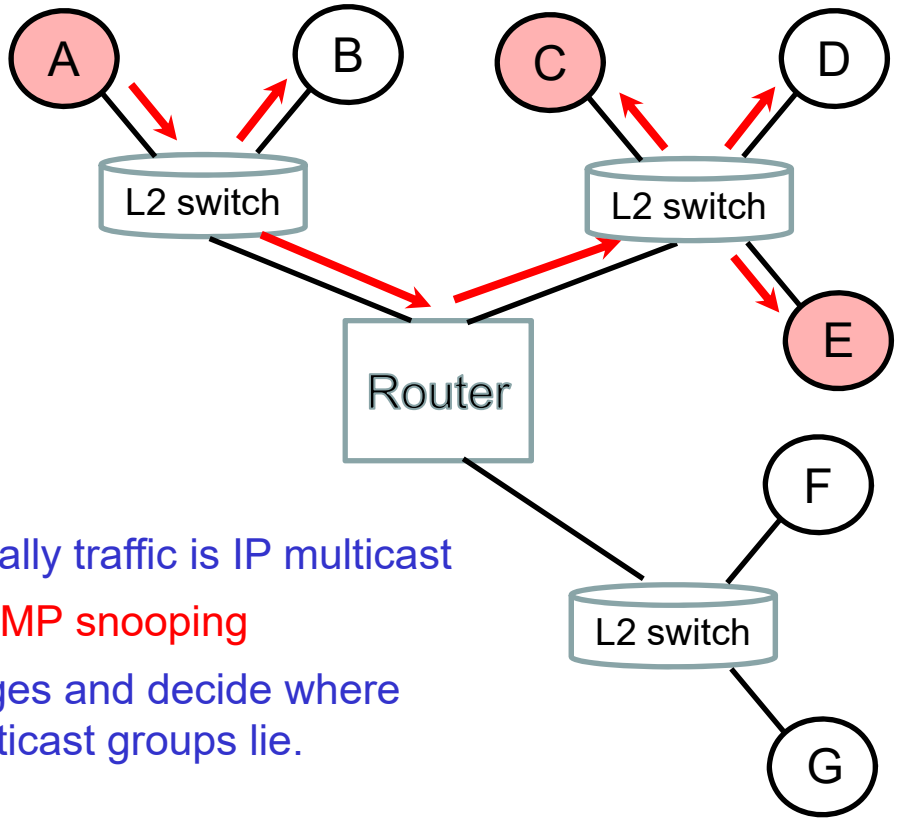


Routers vs Switches

- Switches were initially much faster than routers
  - No need to process IP headers
  - Layer 2 switch consults forwarding table in hardware (c.f. ATM switching)
- One solution is label switching
  - Identify flows (collections of packets with same endpoints)
  - Attach a label to each packet
  - Construct forwarding table of labels against next hop (Ethernet address).
  - Switch packets on flow without looking at their headers.
- IETF label switching standard is multiprotocol label switching (MPLS)
  - Operates within an MPLS domain.
  - Effectively creates virtual circuits to carry IP packets through domain.
  - Allows use of quality of service differentiators.
  - Can be used with protocols other than IP.
- Another option is the routing switch which is a router with routing functions implemented via high-speed ASICs: sometimes called wire-speed router.

Multicast Support at Layer 2

- Layer 2 supports multicast.
- 802.1D switches can flood multicast frames
- Better solution MMRP, (MRP Multicast Registration Protocol): 802.1ak-2007, MRP applied to multicast.



- MMRP can carry any protocol but usually traffic is IP multicast
- Alternative in this case to MMRP is IGMP snooping
- Layer 2 switches watch IGMP messages and decide where stations that have joined or left IP multicast groups lie.
- Filter tables are adapted accordingly.
- IGMP snooping is covered by RFC 4541 (informational).

3. Congestion Control and QoS

Congestion in IP networks

- Congestion occurs when router receives more packets than it can transmit:
  - Buffers fill up.
  - Packets are discarded.
- Congestion control may attempt avoidance or response or both.
  - Can be responsibility of hosts (slow down send rate) or routers (selective discard).
- TCP has end-end congestion response.
  - Reduces rate in response to packet drop.
  - What about UDP-type communications?
  - Not suitable for isochronous traffic
- Concept of flow as sequence of packets passing along same route is useful for router-based control.
  - Routers can act on flows to control congestion and prioritise traffic
  - Can do this even if flow is not TCP-based.
- Routers handle congestion via a queuing algorithm

Quality of Service

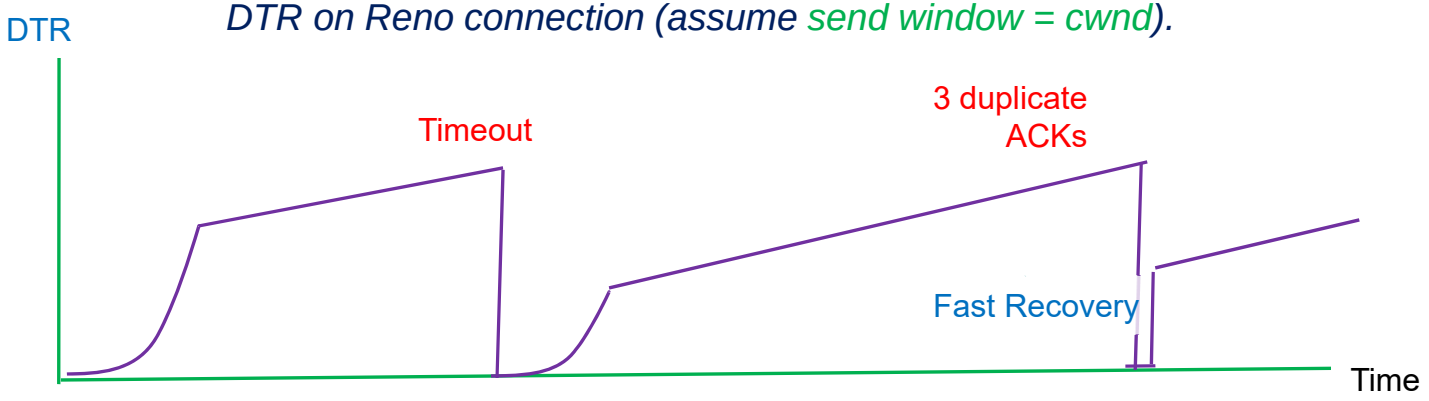
- Basic Internet operates a best effort service
  - All packets treated the same.
- Some types of traffic need guarantees
  - Implies a need for a service model with different levels of quality of service (QoS).
- Main issue is allocation of network resources (bandwidth, router buffer space etc) so as to satisfy competing requests fairly
- Closely connected with congestion control (how network reacts when excessive demands are made on resources)
- Resource allocation can be responsibility of hosts or routers or both.
- Best effort service network typically uses hosts with feedback to reduce demand (e.g. TCP uses implicit feedback) .
- Support for multiple QoS levels is usually via resource reservation (e.g. to a flow) and relies on router behaviour.

TCP Tahoe Congestion Control

- TCP Tahoe refers to TCP with original Jacobsen congestion control algorithm.
- TCP sender maintains congestion window (cwnd).
  - Incoming ACKs clock outgoing packets.
  - When connection starts, congestion window set to 1 segment.
  - ~doubles every round trip time (RTT) until it reaches preset threshold (sstresh): this is called slow start.
  - Then increases by 1 every RTT (additive increase) unless timeout occurs: this is the congestion avoidance phase.
- On timeout, sender assumes congestion
  - Set sstresh = cwnd/2 (multiplicative decrease)
  - Re-initiates slow start from cwnd of 1

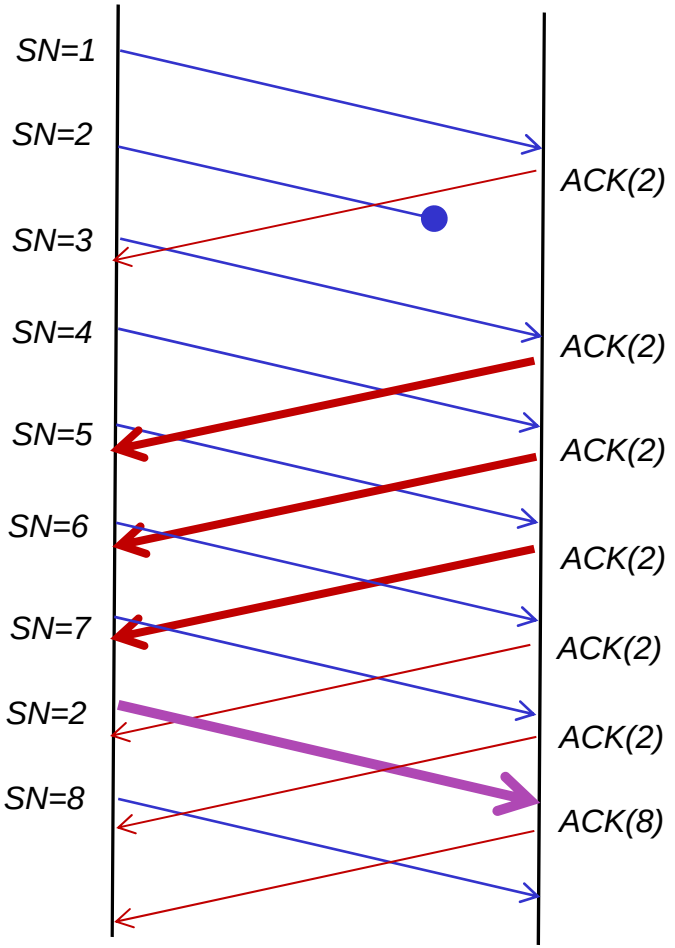
Fast Retransmit and Fast Recovery

- In TCP Tahoe when 3 duplicate ACKs are received:
  - Packet is probably lost so…
  - fast retransmit (retransmit segment without waiting for timeout)
- TCP Reno (RFC 2001 then RFC 2581) adds fast recovery
  - If ACKs are getting through, congestion is less likely.
  - set sstresh=cwnd/2 and set cwnd =sstresh+3
  - wait until new data (retransmitted packet) is ACKed
  - then go straight to additive increase from sstresh



Limitations of TCP Reno

- Time sequence diagram (right) shows fast retransmit in Reno
- Note that after fast retransmit, sender may still get ACKs requesting missing frame up to one RTT from retransmit.
- Reno can get into trouble if more than one packet goes missing in a single window of data. Why?
- Improved version called TCP NewReno (RFC 3782) attempts to fix this.
- Also note more complex TCP SACK (RFC 2018) which uses selective acknowledgements.



TCP Vegas

- TCP Vegas (Brakmo, O'Malley & Peterson, 1994)
  - proactive approach to congestion detection
  - try to predict congestion by measuring current throughput on the connection against expected throughput in an uncongested network (delay rather than loss-based)
    - this can be estimated by comparing current throughput against the best achieved so far;
  - if difference is small cwnd is increased; if large, cwnd is decreased
  - attempt to correct for congestion before it takes hold while maintaining throughput
  - TCP Vegas not widely deployed
  - Other delay based approaches FAST TCP (commercial); Compound TCP (Microsoft)

$$ExpectedT = \frac{cwnd}{BaseRTT}$$

$$CurrentT = \frac{cwnd}{RTT}$$

$$Diff = ExpectedT - CurrentT$$

$$If\ Diff < \alpha\ then\ cwnd = cwnd + 1$$
$$If\ Diff > \beta\ then\ cwnd = cwnd - 1$$

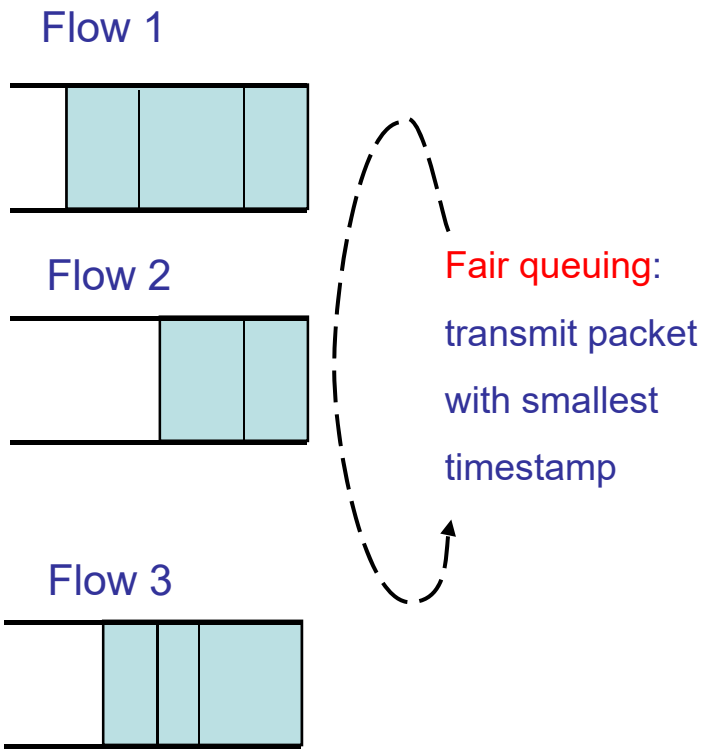
The Vegas algorithm

BaseRTT is lowest RTT observed so far.
α, β constants.

Queuing

- Queuing algorithm dictates how to handle packets arriving for given output. Includes:
  - scheduling discipline (which packet should be transmitted next) and
  - drop discipline (which packet to throw away).
- Simplest algorithm is FIFO with tail drop
  - commonest in Internet routers.
  - but single FIFO queue creates restrictions
- Can use multiple queue scheduling to separate flows out for fair treatment or to allow priorities.
- Could use round-robin scheduling (one packet from each queue in turn) but some applications may send big packets thus obtaining unfair advantage.
  - fair queuing counters this using a timestamp system (see right and next slide)
  - can be adapted to include carefully controlled priorities (weighted fair queuing)

When packet, *P*, arrives allocate a numeric timestamp, TS(*P*) using a fair queuing algorithm



Fair Queuing

- Aim is to compensate for varying packet size in different flows (queues).
- Suppose that:
  - When packet, *P*, arrives at an output, X, it is allocated to a queue, Q_i.
  - X is currently transmitting a packet P_T, from one of its queues (not necessarily Q_i).
  - The estimated transmission time for *P* is t(*P*)
- If there is a packet, P_Q , in front of *P* in Q_i , *P* is time-stamped with value:

$$TS(P) = TS(P_Q) + t(P).$$

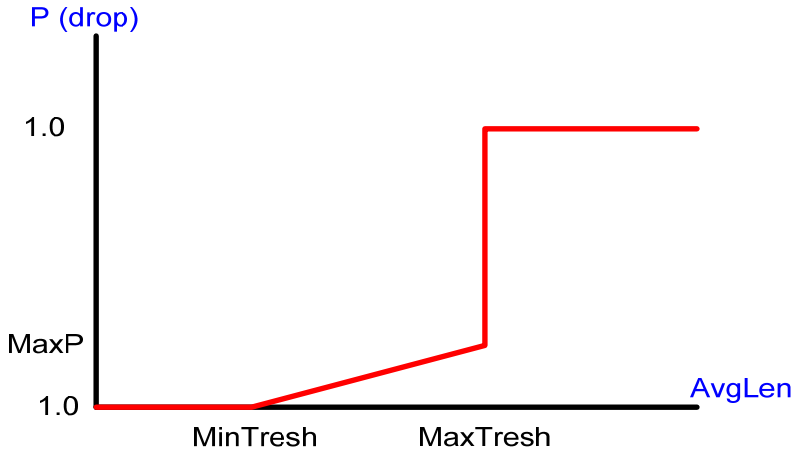
- If Q_i is empty, *P* is time-stamped with value:

$$TS(P) = TS(P_T) + t(P).$$

- The fair queuing algorithm always transmits the packet with the smallest timestamp.
- Easy to adapt this to implement weighted fair queuing by scaling transmit times.

RED

- Random Early Detection (RED) proposed for use with TCP (Floyd et al 1993; see RFC 2309)
  - Drop discipline to avoid congestion
  - drop packets before buffers full.
  - TCP sources slow as soon as packets get lost.
  - Every packet arriving at a given queue is dropped with a probability determined by the current average queue length computed as shown.



$$AvgLen = (1 - Weight) \times AvgLen + Weight \times SampleLen$$

0<*Weight*<*1* *SampleLen* is length of queue when sample taken

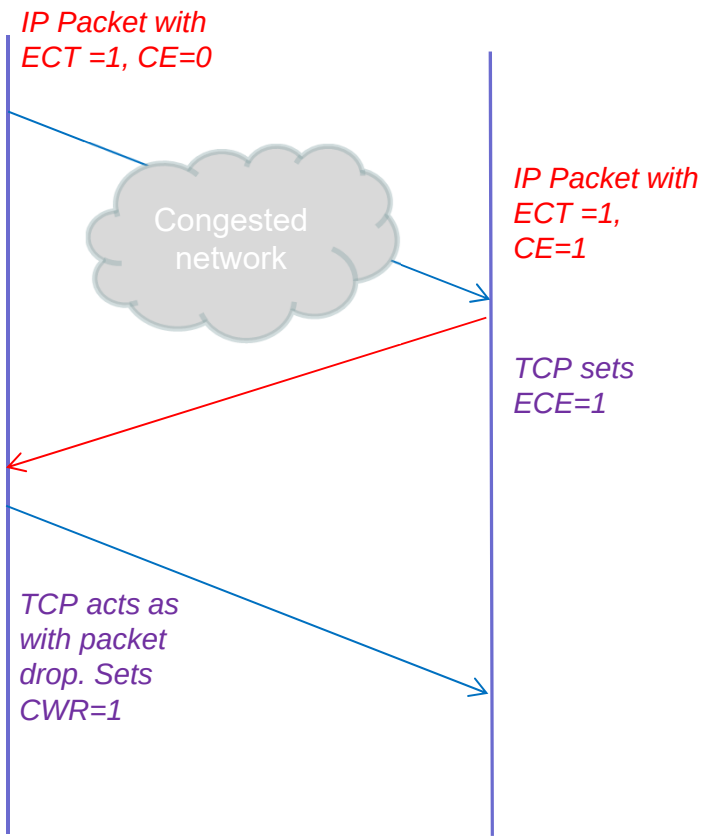
If AvgLen is between MinTresh and MaxTresh, each packet dropped with probability *P*

Below *MinTresh P* = 0; above *MaxTresh* *P*=1.

Choice of actual parameter values has proved hard; limited deployment in practice.

Explicit Congestion Notification (ECN)

- The ECN scheme (RFC 3168) needs compliant transport protocol (E.g. TCP)
- Uses two extra bits in the IP Header and two in the TCP header.
- IP bits (in old TOS byte) are: ECT (ECN Capable Transport) and CE (Congestion Experienced)
- TCP bits are ECE (ECN Echo) and CWR (Congestion Window Reduced)
- When ECN is in effect, IP packets start out with ECT set, CE cleared.
- If an ECN compliant router encounters congestion and sees a packet with ECT set, it can choose to forward with CE set instead of discarding.
- At receiver, transport layer sets ECE bit in response. Sender then reacts as with packet drop but also sends next segment with CWR set.



UDP Issues

- UDP was originally used for short request-response transfers
- More recently used for streaming multimedia, VoIP, gaming etc.
  - TCP is too constrained for these applications
  - retransmission, in-order delivery, get in the way;
  - congestion control is not smooth
- Recently, significant increase in proportion of UDP on Internet
- But UDP has no congestion control which is bad for Internet as a whole.
- Suggestion 1: Add congestion control mechanism to UDP
  - TCP Friendly Rate Control (TFRC) outlined in RFC 3448 can be used with RTP (common cargo for UDP). See also update in RFC 5348.
  - TFRC continually measures RTT and receive rates (c.f. TCP Vegas) and uses equation-based control to then adjust send rates
  - Feedback packets are used to convey measurements back to the sender
  - Congestion response smoother than TCP but slower (only use when smooth response is needed).
- Suggestion 2: Design new protocol, “UDP+” including congestion control mechanisms but omitting unwanted features of TCP
  - DCCP (Datagram Congestion Control Protocol)
- At other pole is UDP-lite (RFC 3828): reduced error checking compared to UDP.

SYN flooding attacks

- A SYN flood is a denial of service attack:
  - Attacker sends many TCP SYN requests to open spurious connections to server
  - Server responds with SYN ACK, but attacker fails to return ACK (third element of 3-way handshake) and waits for half-open (embryonic) connection to time out.
- In TCP, half-open connection allocates resources on server (records details of connection) while it waits for ACK.
  - By constantly initiating spurious connections, attacker aims to exhaust space available on server to handle legitimate connections.
- Attacker may spoof packets to make them seem to have multiple sources.
  - makes defence based on identifying attacking IP numbers very difficult.
- A SYN cookie (TCP: Bernstein 1996) or Init cookie (DCCP) mounts defence:
  - server dispenses with local record of half-open connection
  - instead encodes state of connection in SYN ACK response and returns to client.
  - If client wants to open connection, must return state information with ACK.
  - If the client is spurious (no ACK is returned) no buffer space reserved on server.
  - SYN cookie supports only limited state (no TCP options, 8 allowable MSS values)
  - DCCP Init cookie allows more state.
  - see also TCP Cookie Transactions (RFC 6013).

Traffic types and QoS

- Distinguish between real-time and elastic applications (e.g. file transfer).
- Real-time (aka isochronous) applications vary in type.
  - Typically application data unit (audio sample, video frame, real time command) must be used at destination at some playback time.
  - Need predictable latency but can use startup delay and playback buffer to give some tolerance.
  - Some real-time applications are tolerant of loss but there are also intolerant real-time applications.
  - Some real-time applications can accommodate small variations in playback point (delay adaptive) or rate (rate adaptive) by degrading gracefully.
- QoS must provide suitable service for various traffic types
- Two broad approaches to QoS
  - Fine-grained provides QoS to flows.
  - Course-grained provides QoS to classes of data

Integrated Services

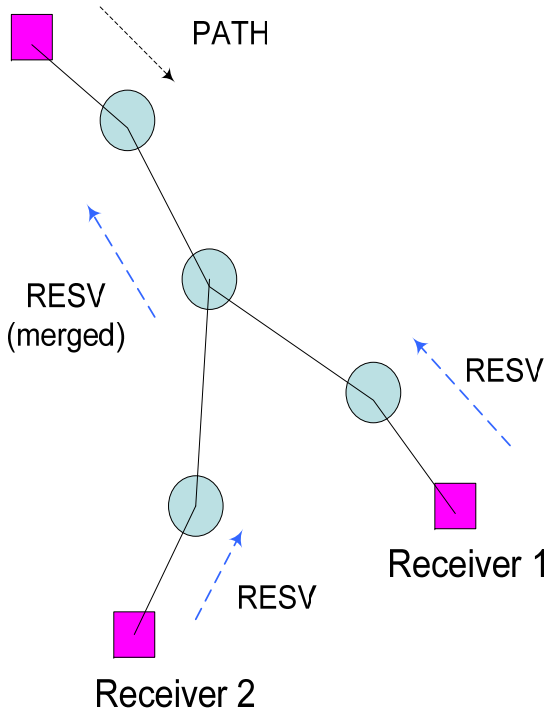
- Fine grained approach to Internet QoS called integrated services (IntServ).
  - Not widely used (does not scale well) but worth studying
- Service classes:
  - guaranteed for intolerant real-time applications (see RFC 2212)
  - controlled load for tolerant adaptive real-time applications (see RFC 2211)
- A traffic flow may be multicast and is initially defined by a Tspec generated by the sender and based on a traffic shaping algorithm.
- Each receiver chooses a Tspec it wants but may add capabilities of its own via an Rspec (only used in guaranteed service): may e.g. allow faster clearing of queues.
- Integrated Services approach requires:
  - Admission control —can flow be supported by network? This can be hard on Ethernet networks.
  - RSVP (Resource Reservation Protocol) —setup protocol for an *admitted* flow, reserves resources at routers (RFC 2205).
  - Policing —does packet adhere to agreed TSpec.
  - Packet Classification —associate packet with reservation (use IP addresses, port numbers, protocol number). May use 802.1p priorities or IP DSCP field (formerly TOS)
  - Packet Scheduling —needs sophisticated queuing.

Traffic Shaping

- Policing involves traffic shaping (rate limiting).
- Traffic shaper delays metered traffic if it tries to exceed terms of its contract (e.g.Tspec in IntSev).
- IP typically metered with token bucket algorithm.
  - "bucket" contains tokens needed to send a unit of data.
  - Tokens arrive at some predetermined rate, *r*, and are placed in bucket unless bucket is full, *b*.
  - Tokens used up every time packet is sent.
  - Allows burstiness up to maximum of bucket size but will impose peak bandwidth limit, *p.*
  - *r, b* and *p* are main elements of Tspec in IntSrv.
  - also used in other QoS approaches.
- Metered packets stored in FIFO buffer until they can be sent
  - If buffer overflows packets discarded.

RSVP

- Creates illusion of virtual circuits in IP network
- Receivers make resource reservations
- Sender sends PATH message with TSpec to receivers.
- Each router on (multicast) path records reverse path. May modify TSpec
- Receivers return RESV message with Flowspec (Tspec + Rspec)
  - Passed back up route
  - Requirements may be combined at some routers to serve multiple receivers
  - RSpec may give routers licence to send a little faster and increase delays a little (slack) where useful.
- Reservations time out (no explicit delete)—soft state.
  - Receivers repeat RESVs regularly (~30s)
  - Senders repeat PATH regularly (~30s).



Differentiated Services

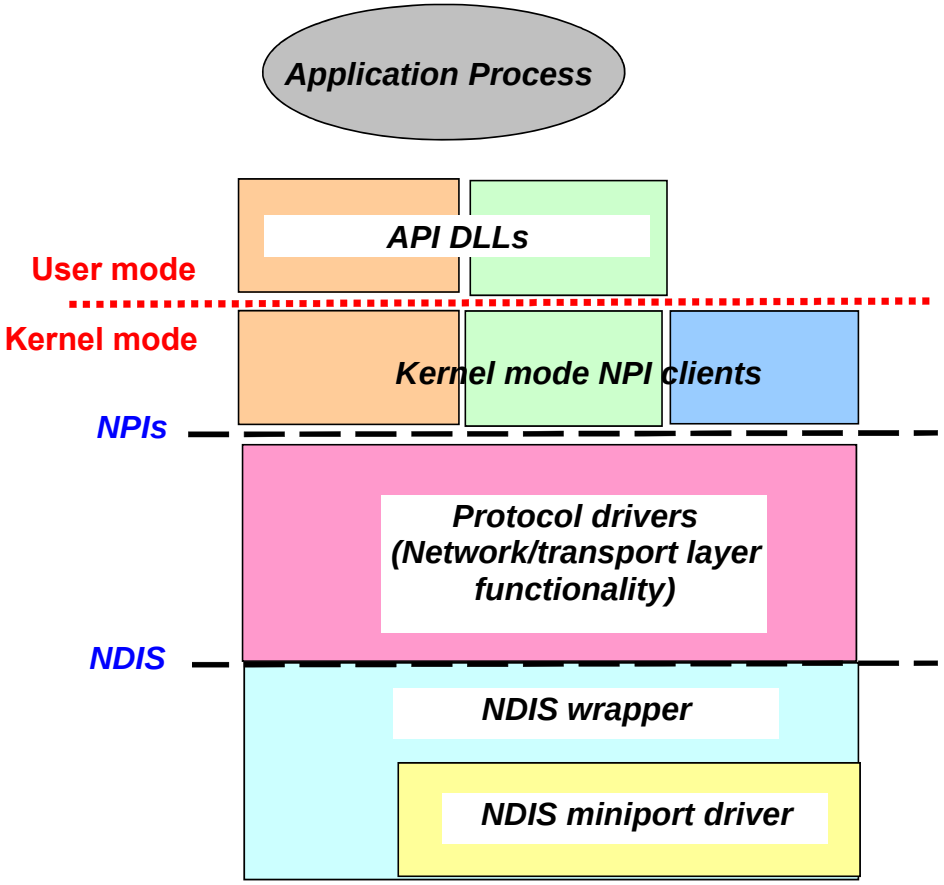
- Alternative to IntServ is differentiated services (DiffServ) (RFCs 2474, 2475)
  - mark packets at edge of an administrative area (DS domain).
  - 6 bits of old TOS field have been redefined as differentiated services code points (DSCP), used to identify classes.
  - Different classes of packets are then treated differently by routers (known as per hop behaviours or PHBs)
- PHB classes have been defined as follows.
  - Expedited forwarding (EF) guarantees immediate forwarding for EF packets (number of EF packets in area must be limited) RFC 3246. Aimed at real-time services like voice and video.
  - Assured forwarding (AF) is a group of 4 classes each with 3 levels (RFC 2597). Can use e.g. RED with differently handled packet types to discriminate between priorities— Weighted RED (WRED). Flexible use to be determined by service provider.
  - Use Weighted Fair Queuing to handle different DSCP packets in different queues.
- Packets marked (classified) at edge of domain by DS boundary nodes. Interior nodes forward marked packets according to pre-agreed rules.
- Actual behaviour is dependent on service provider (may differ between domains).
- Some form of admission control is necessary to allow a boundary node to know how to classify a packet. Traffic may then be conditioned (metered, shaped and policed) to prevent a source from injecting more at a given level than has been agreed.

Internet QoS: a critique

- Work on Internet QoS has been going on for ~ 20 years. Yet there is no wide-scale deployment.
- In practice QoS is restricted to LANs, VPNs and specialist providers.
- QoS is complicated to implement and this makes it unattractive to graft onto a working system unless a very clear business benefit can be demonstrated. See, e.g. (copies on ANC4 web page).
  - *Bell, "Failure to thrive: QoS and the Culture of Operational Networking"*
  - *Davie, "Deployment Experience with Differentiated Services"*
  - both from ACM SIGCOMM 2003 Workshops
- Some oppose deployment of QoS on the Internet as a whole. There are practical and political reasons for such a stance:
  - Practically it is argued that QoS is only needed if network resources are scare so instead we should always ensure significant overprovisioning in the core (may need restrictions at edge).
  - Politically, opposition is tied to arguments about network neutrality. Proponents of neutrality fear that QoS will result in capacity being shared in such a way that the "*rich win at the expense of the poor*" (c.f. artificial scarcity). BUT network neutrality requires government regulation which opponents fear will stifle innovation (e.g. note debate in the US especially since 2015 – see article from IEEE Spectrum on ANC4 web page). In essence this is a classic free market vs government regulation argument.

4. Real World Implementation: The Windows Communications System

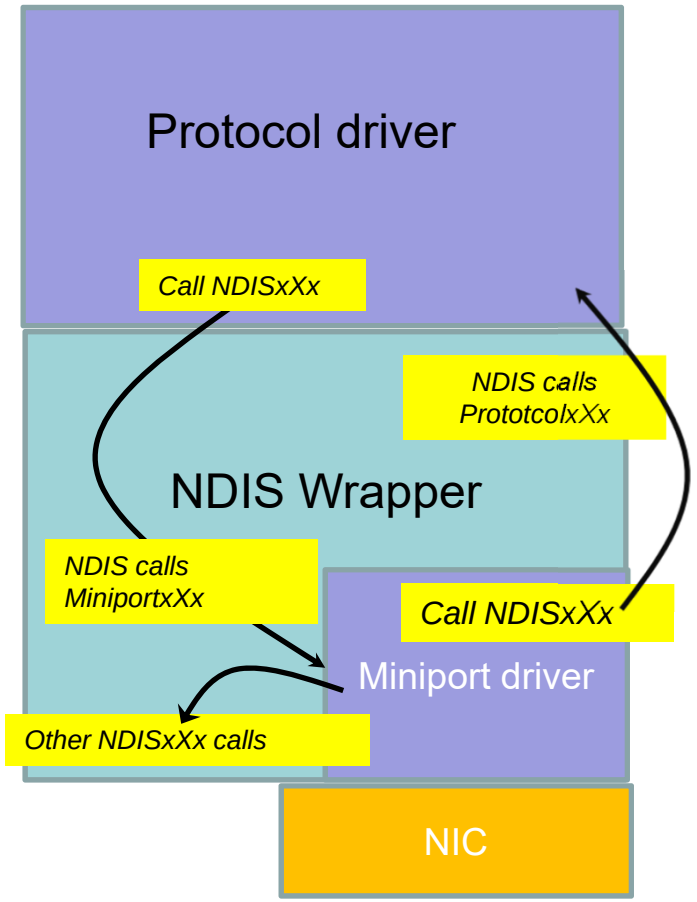
Windows communication system



- Windows communications architecture has always supported protocol stacks other than TCP/IP but latter is primary.
- Layer 2 functions implemented by NIC hardware and drivers which adhere to Network Driver Interface Specification (NDIS) specification (exposes device independent functions which can be used to program against Layer 2)
- Above NDIS are protocol drivers such as Tcpip.sys.
- the TCP/IP Protocol drivers interact with kernel mode clients via Network Programming Interfaces (NPIs).
- A user networking API like Winsock is implemented by a kernel mode NPI client and user mode DLLs.

NDIS

- Network Driver Specification Interface (NDIS): API developed by Microsoft and 3com.
- Allows NDIS compliant communications drivers to communicate with each other and with OS via standard calls.
- Network Interface Card (NIC) is controlled by an NDIS (compatible) miniport driver.
- NDIS consists of a library (sometimes called the wrapper) of functions which can be called by a protocol driver or a miniport driver.
- Protocol driver "lower edge" first binds to a miniport driver for chosen NIC. Then NIC can send and receive.
- Miniport driver calls NDIS library to communicate with protocol driver, with OS and also for setting up NIC.
- NDIS miniport driver is more portable than device driver not using NDIS.
- Current version is 6.6 (Windows 10, Server 2016).

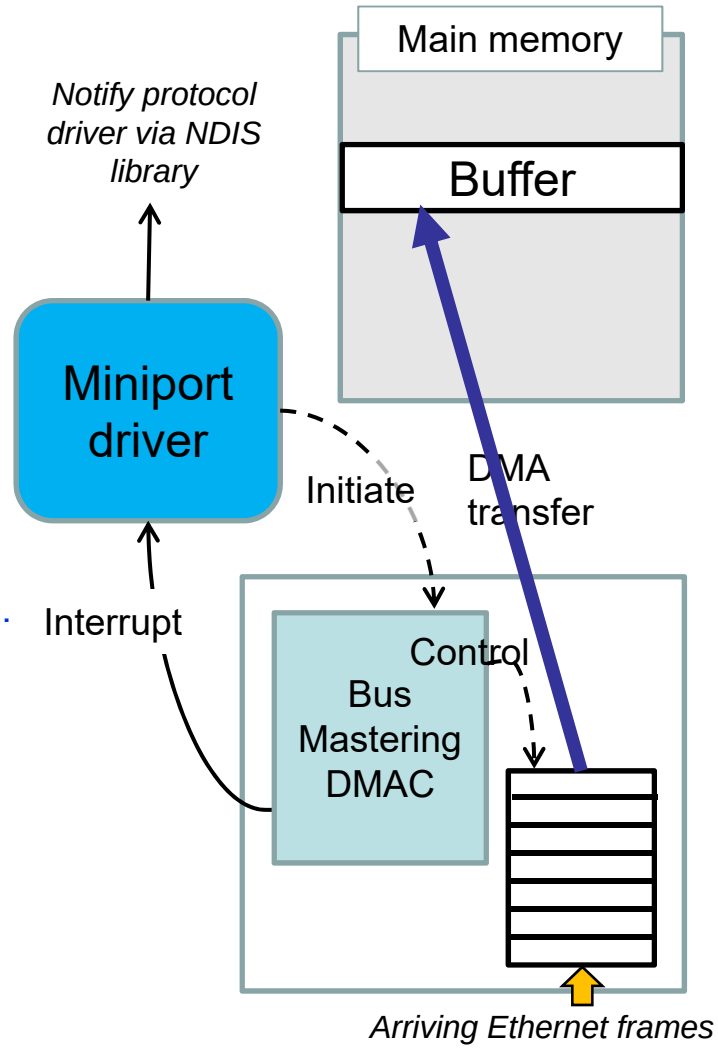


Implementing the Protocol Stack

- Protocol driver implements combined transport and network functionality with data-link framing for selected Layer 2.
  - handles segmentation, reassembly, acknowledgement etc
- Transport functions accessible to kernel mode clients via an NPI. NPI in current versions of Windows is the Transport Layer NPI (TLNPI) which gives access to Tcpip.sys.
- An NPI makes functions available to user-mode applications via NPI clients in kernel mode, accessed through user mode DLLs.
- Winsock uses the driver Afd.sys (Ancillary Function Driver) as its kernel mode client and the the core user-mode library Ws2_32.dll provides the API to user programs .
- Multiple protocol drivers can bind to a single miniport. This is how WinPcap and Win10Pcap (used with Wireshark) capture network traffic.
- Protocol driver and miniport driver communicate via NDIS library, e.g. when protocol driver wants to send data or miniport driver wants to deliver data.
- Intermediate or filter drivers can be inserted between the protocol driver and miniport driver. These all communicate via the NDIS library. An example is the network capture driver of Microsoft Network Monitor tool and its successor Microsoft Message Analyser.

The NIC

- Incoming packets are stored in NIC FIFO according to filter currently in effect (e.g. exact address, promiscuous unicast, promiscuous multicast, VLAN). See diagram.
- Interrupt is generated which schedules a service routine associated with the miniport driver.
- Typically data is then transferred by Direct Memory Access (DMA) to a buffer in main memory controlled by protocol driver. Further memory-memory copy typically needed to transfer data to application.
- DMA device must have access to physical memory addresses. NDIS uses NET_BUFFER structure to store data. Includes Memory Descriptor List (MDL) which describes virtual memory buffer layout in physical memory.
- Outgoing data is sent via NDIS call to miniport driver API. NDIS library passes calls to miniport driver via MiniportXxx calls.
- DMA is then arranged to transfer from memory buffer to NIC FIFO. Protocol driver notified via NDIS call by miniport driver.

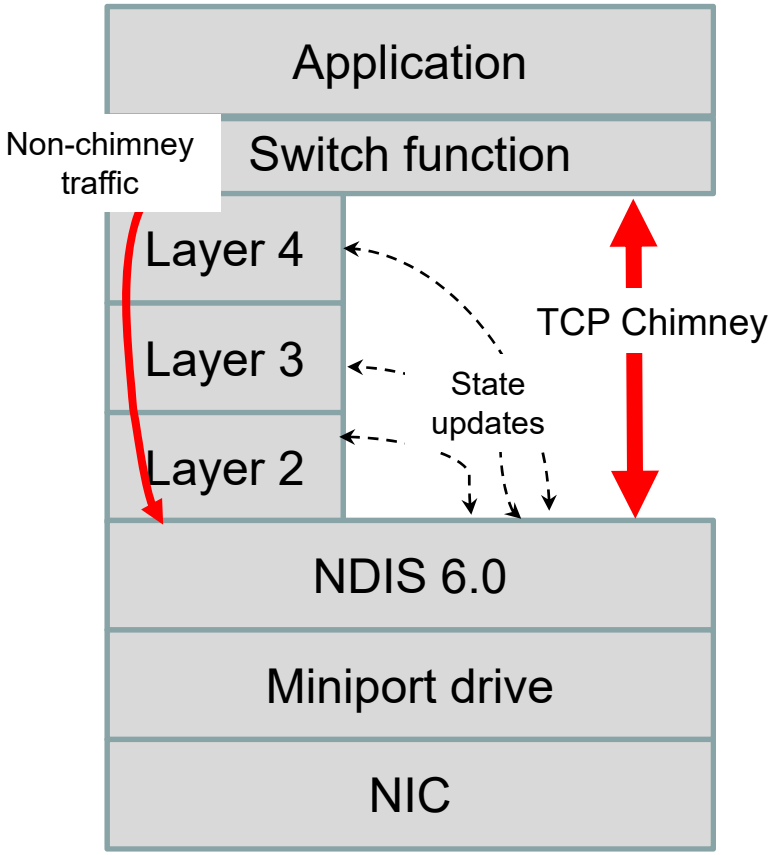


Scalable Networking

- Substantial overhead in processing TCP/IP protocol stack:
  - Comms drivers must segment, reassemble, checksum, incoming and outgoing data as well as handle protocol functions.
  - Many interrupts from NICs (usually constrained to a single core even when, as usual in modern systems, there are several present).
  - Memory-to-memory copying
- These issues become more critical at high networking speeds and especially on servers.
- Number of scalable networking techniques have been developed to reduce this overhead. Usually rely on more intelligent network hardware with suitable OS support. Examples.
  - TCP Chimney Offload
  - Receive Side Scaling
  - IPSec Offload

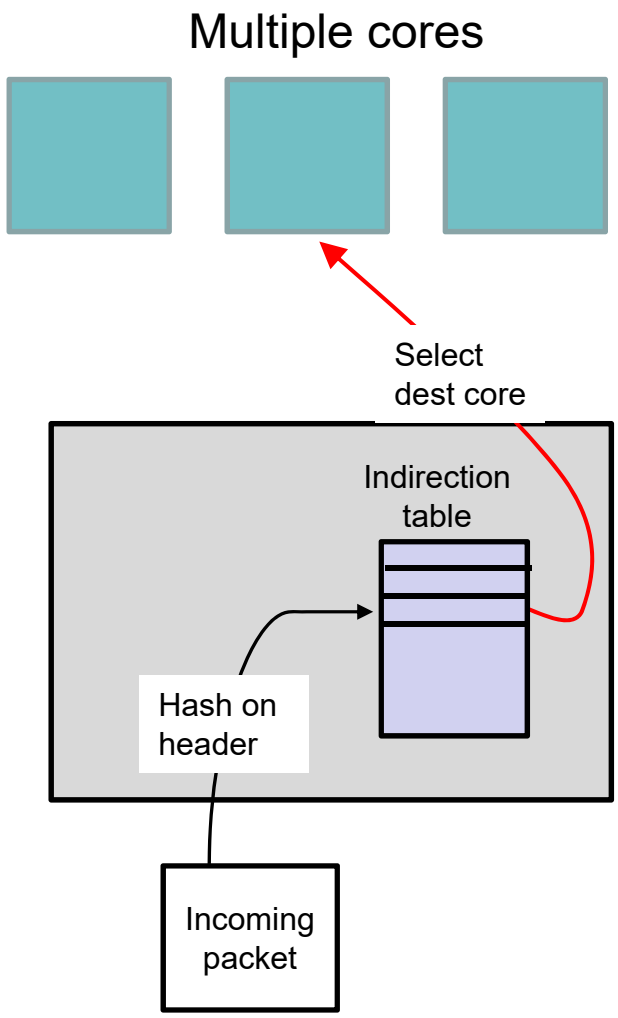
TCP Offload

- A TCP Offload Engine (TOE) is a technology incorporated in some NICs which allows the card to process the entire TCP/IP stack.
- Full offload supports connection management as well as data transfer.
- Windows supports partial (stateless) offload, known as TCP Chimney Offload: connections handled by software as usual but data transfer on a TCP connection can be handled by the NIC.
- Chimney offload does not apply to non-TCP packets.
- TOE has been criticised as being difficult to patch (and thus possibly less secure), may run out of hardware resources, is proprietary and may have a limited lifetime (given rate of technological advance).
- It is argued that some of these criticisms have less applicability to partial offload approaches.



Receive Side Scaling

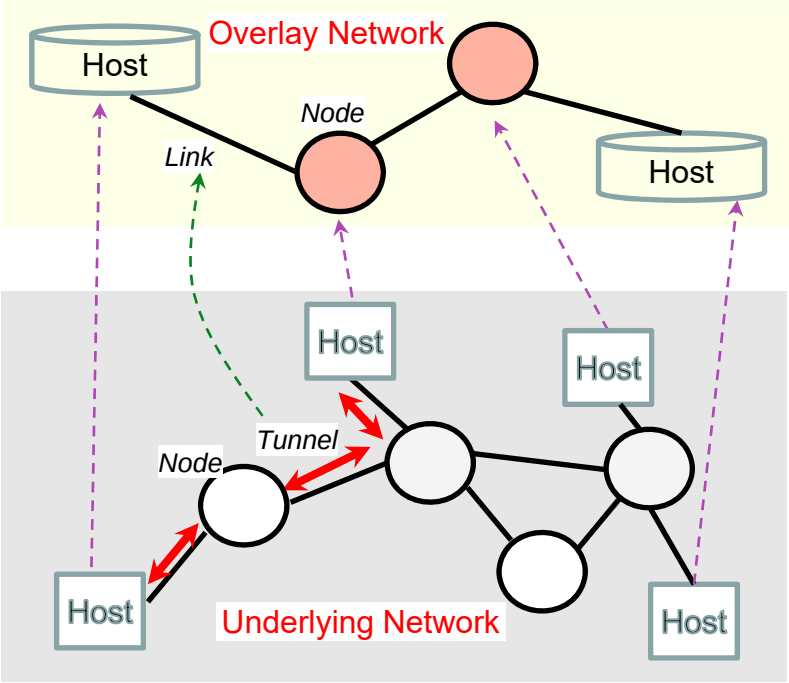
- When data arrives on a connection an interrupt is generated.
- Modern servers have multiple CPUs/cores to help share load but to distribute arriving data from TCP over many cores is problematic since for any given connection, the protocol driver requires in-order delivery. .
- Windows Receive Side Scaling (RSS) tries to solve this by hashing on selected fields of the packet header and using an indirection table to decide which core will handle that packet.
- All packets on a connection are then handled by the same core.
- If system is sufficiently flexible (e.g. Message Signalled Interrupt or MSI support) ISR can be made to run on the chosen core. Otherwise ISR always runs on one core and work is distributed from there.



5. Overlay Networks

Overlay Networks

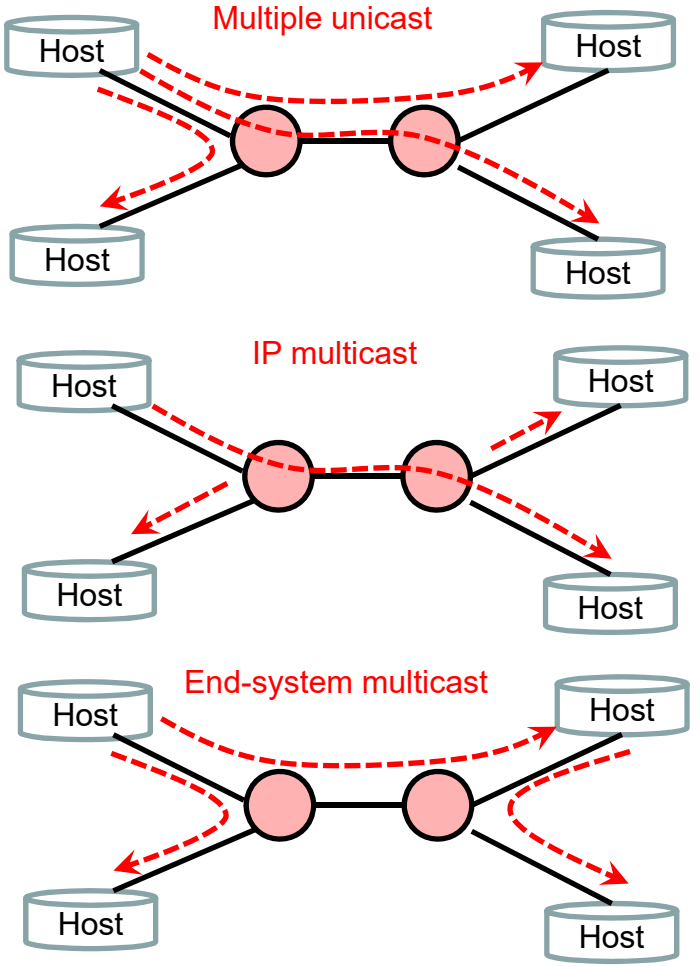
- An overlay network, *A*, is a logical network implemented on top of another underlying network, *B*.
  - Nodes in *A* are typically hosts or nodes of *B*. In many cases only hosts are used.
  - Links in *A* are typically tunnels through *B*.
- For example:
  - The Internet can be seen as an overlay of the telephone network.
  - A VPN is a private overlay of some more public network (e.g. the Internet, a carrier network etc).
  - Mbone and 6-BONE were routing overlays used on the Internet to test and deploy new forms of IP routing.



- Overlay networks based on Internet hosts are of particular interest.
  - Internet has limited capacity for experimentation with new protocols because it provides an essential service (this leads to what is sometimes called ossification)
  - However as an underlying network it provides direct connectivity between any two hosts or nodes.
  - Internet hosts and, in some cases, routers, can then be interconnected by tunnels into any desired overlay topology.

End-system multicast

- IP multicast has not been universally deployed.
- Alternative is end-system multicast.
  - *Host-only* overlay network using underlying Internet.
  - Less efficient than IP multicast but much more so than multiple unicasts (see diagram).
  - Links are UDP tunnels.
  - Important to efficiency how links are chosen (why?).
- Members of multicast network keep measuring round-trip times to estimate best tunnels.
  - Current best tunnels used to form underlying mesh of tunnels (this is the overlay network proper).
  - Multicast algorithm (e.g. DVMRP) now run to form multicast tree using mesh edges only.
  - When host joins network it notifies existing member and forms tunnel to it.
  - When host leaves, neighbours reconfigure tunnels.
  - Mesh must be maintained as join and leave operations will result in sub-optimal tunnels.



Peer-Peer Networks

- Peer-peer (P2P) networks attempt to decentralise resources with limited or no server support.
- Implemented as host-only (application layer) overlay networks
- First achieved prominence with music sharing systems like Napster and more general file-sharing systems like Kazaa (early 2000s).
- Networks are dynamic (peers can join or leave at any time) and resources are distributed across all peers.
- An unstructured P2P network attempts to address issues of peer discovery and indexing (resource location) without *a priori* organisation.
- A structured peer-peer network will attempt to organise connections between peers and sometimes also resources using techniques like distributed hash tables.
- Unstructured networks can be:
  - Centralised (central server used for indexing) e.g. Napster
  - Hybrid (special infrastructure supernodes but no single central server) e.g. Kazaa
  - Pure (all peers are entirely equal) e.g. Gnutella

Gnutella: pure unstructured P2P

- Peers organise into overlay network connected by UDP tunnels.
- A peer, *A*, is connected to all the others it knows about but this is only small subset of entire network.
- If *A* wants an object it queries its neighbours.
  - If one has the required item, it responds and *A* can access it
  - If no neighbour has item, query is forwarded to all its neighbours and so on (effectively query is flooded across network)
- Queries contain Query ID (QID) and record of the upstream neighbour from which it came but no trace of source.
  - QIDs used to damp flooding
  - Responses sent upstream until they return to A.
- Nodes discover new nodes when they respond to a query
  - Such a discovered node may be retained as a new neighbour.
  - Neighbours periodically check each other for life via PING and PONG messages.
- Flooding is an expensive mechanism for broadcasting:
  - Other options exist that try to reduce load (e.g. probabilistic forwarding)
  - Similar problems encountered broadcasting in mobile ad hoc networks (MANETS)

Structured P2P networks

- Here we try to map objects onto nodes in a distributed fashion. Issues are:
  - How do we map?
  - How do we route request for object to correct node?
- Natural approach to distribute objects is to use hash function but:
  - In simple hashing need to decide in advance how many buckets;
  - Not suitable for network where peers can join and leave.
- A distributed hash table (DHT) is a decentralised distributed lookup system that allows any peer in an overlay network to retrieve an object given its key (hash) in some very large keyspace (e.g. 160 bits).
- Most DHT systems use consistent hashing to partition keyspace and key-based routing mechanism.
- In consistent hashing define a distance function that measures "closeness" of any two keys. Then:
  - Generate a key in the space from each object's ID (e.g. SHA-1 hash of the name).
  - Generate a key in the space from each peer's address (e.g. IP number)
  - Allocate a new object to the peer with key closest to the object's own key.
  - A host owns the key-subspace consisting of all keys that are closer to it than any other peer.
  - If peers leave or join, only the key-subspace of neighbours (in key-space) is affected.
- In key-based routing, for any key, *k*, every node, *A*, that does not own *k* has at least one overlay neighbour whose key is closer to *k's* than *A's. A* forwards to the neighbour whose key is closest to *k*'s.
- See *Rowstron & Druschel's* paper on the Pastry system (ANC4 website)

BitTorrent

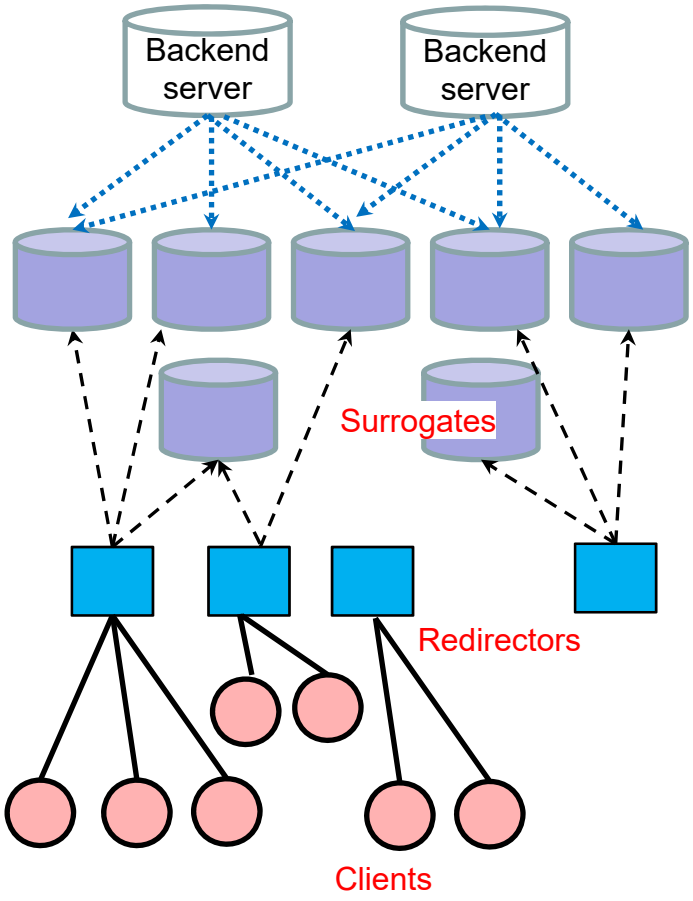
- BitTorrent is a peer-peer file sharing protocol designed by Bram Cohen (2001).
- Files are divided into pieces downloadable separately.
- Each file shared by its own independent overlay network or swarm.
- Swarm starts as one node with complete file.
- Node that wants file, joins swarm as leecher. When it has downloaded a piece it becomes another source for that piece.
- Once a node in the swarm has a complete copy it becomes a seed.
- In conventional BitTorrent, swarm has a server called its tracker that keeps track of current swarm membership.
- To join swarm a new peer, *A*, uses a .torrent file typically downloaded from a web server to locate the tracker which it then contacts.
- Tracker replies with partial list of peers to which *A* connects and with which it can exchange data. Linked peers inform each other of which pieces they have completed.
- Newer versions of BitTorrent support trackerless swarms using a DHT implemented over a peer-finder network which spans swarms. DHT based on Kademlia protocol. Client acquires hashed version of desired item called a magnet link and uses this to search DHT for peer information. Replaces .torrent file, though both systems may run together.

Operation of BitTorrent

- A peer-peer network is much more efficient than a central server at distributing a file to multiple recipients. Why?
- However any peer-peer network will only function well if participants upload as well as download. BitTorrent was designed to try to ensure that free-riders are contained.
- BitTorrent enforces fair behaviour by swarm members. A peer chokes another if the latter is not uploading at required rate.
- When a new peer, *A*, joins a swarm it is allocated a set of swarm neighbours by the torrent, with which it forms TCP connections. Each neighbour sends a list of pieces.
- *A* requests pieces, usually adopting a strategy of rarest first. Why? As *A* accumulates pieces, it receives requests itself.
- BitTorrent uses a fairness approach sometimes called tit-for-tat as follows.
  - *A* prioritises those neighbours who are sending bits to it fastest selecting the 4 most active. Neighbours not in this group are said to be choked.
  - It also chooses one 5^{th} neighbour randomly and optimistically unchokes it.
  - The slowest of the 5 unchoked neighbours is replaced by a new optimistically unchoked candidate every 10 seconds.
  - In this way new peers can participate in the swarm but free-riders are discriminated against.

Content Distribution Networks

- Client-server interaction is subject to several potential bottlenecks: client-side server-side and network-based.
- Server-side and network bottlenecks are especially problematic when a server simultaneously becomes the target of a large number of clients: e.g a flash crowd or a distributed DOS attack.
- Solution commonly adopted is a content distribution network (CDN):
  - use *geographically distributed* surrogate servers to cache pages held by the main backend servers
  - use redirector functions to intercept client requests and forward them to most appropriate surrogate.
- Redirection may be achieved via DNS request routing (return different server addresses to clients) and URL rewriting by surrogate servers.
- Alternatively redirection can be done by physical proxies which direct the client to an appropriate server.
  - Choice of server can be made using a DHT hashing on URL and server ID.
  - Mechanism must also take account of server load and network proximity.
  - DHT means no communication between redirectors needed.
  - Examples include Kankan and Coral CDNs.



Yet another approach uses IP anycast (RFC 1546) with BGP so a single IP address locates nearest server; however this is not responsive to dynamic changes in traffic.

More on CDNs

- CDN requires major infrastructure. Only largest content providers operate CDNs (e.g. YouTube/Google Amazon does); others use 3rd party e.g. Akamai
- Infrastructure must be distributed globally. Two strategies:
  - Enter deep (e.g. Akamai) places server clusters inside ISP access networks.
  - Bring home (e.g. Limelight, YouTube) places smaller number of larger server clusters equidistant from multiple Tier 1 ISP Points of Presence (PoPs)
- DNS request routing (see RFC 3568) used with DNS extensions (RFC 6891). Consider case where content provider (CP) uses third party CDN provider.
  - Client, *A,* visits content provider (CP) site, *B,* and accesses content url
  - *B* returns DNS query, *q*, for content.
  - User host submits *q* to local DNS server L.
  - *L* passes *q* to CP's authoritative DNS server, *C*.
  - *C* returns name, *H* of a host in CDN network to *L*
  - L consults CDN provider's authoritative DNS server, *D,*
  - *D* returns hostname, S_i, of most suitable surrogate server to *L*.
  - *L* returns S_i to *A.*
  - Client now uses second DNS query to resolve IP address of S.
- Surrogate server is chosen by some cluster selection strategy which may be based on simple geography or, in more sophisticated implementations, by measuring RTTs to client networks.