Internetworking: addressing, forwarding, resolution, fragmentation

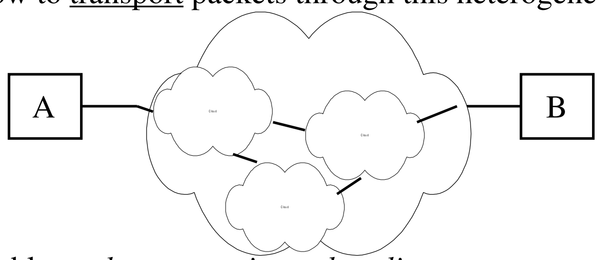
Shivkumar Kalyanaraman
Rensselaer Polytechnic Institute
shivkuma@ecse.rpi.edu
http://www.ecse.rpi.edu/Homepages/shivkuma

Based in part upon the slides of Prof. Raj Jain (OSU), S. Keshav (Cornell), L. Peterson (Arizona)

- Internetworking: heterogeneity & scale
- IP solution:
  - Provide new packet format and overlay it on subnets.
  - Implications: Hierarchical address, address resolution, fragmentation/re-assembly, packet format design, forwarding algorithm etc

The Internetworking Problem

- Two nodes communicating across a "***network of networks***"… How to <u>transport</u> packets through this heterogeneous mass ?



- Problems: *<u>heterogeneity and scaling</u>*
- *<u>Solution:</u>* *<u>Overlay model:</u>* New IP protocol, best-effort forwarding, address hierarchy, address resolution, fragmentation
  - Alternative: *translation* (eg: bridges) or *hybrid* protocol (eg: MPLS used instead IP/ATM overlays)

How does IP forwarding work ?

- A) *Source & Destination in same network*
  - Recognize that destination IP address is on same network. [1]
  - Find the destination LAN address. [2]
  - Send IP packet encapsulated in LAN frame directly to the destination LAN address.
    - Encapsulation => source/destination IP addresses don't change

IP forwarding (contd)

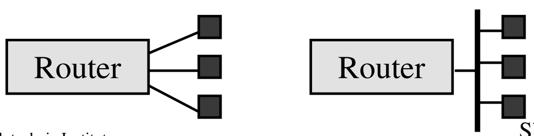
- B) *Source & Destination in different networks*
  - Recognize that destination IP address is not on same network. [1]
  - Look up destination IP address in a (routing) table to find a match, called the next hop router IP address.
  - Send packet encapsulated in a LAN frame to the LAN address corresponding to the IP address of the next-hop router. [2]

Addressing & Resolution

- *[1] How to find if destination is in the same network ?*
  - IP address = network ID + host ID. *Source and destination network IDs match => same network*
  - Splitting address into multiple parts is called hierarchical addressing
- *[2]: How to find the LAN address corresponding to an IP address ?*
  - Address Resolution Problem.
  - Solution: ARP, RARP

IP Address Formats

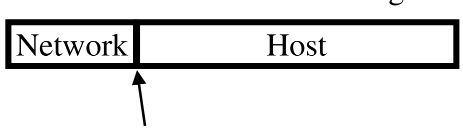
- **Class A:** | 0 | Network | Host | — 1, 7, 24 bits
- Class B: | 10 | Network | Host | — 2, 14, 16 bits
- Class C: | 110 | Network | Host | — 3, 21, 8 bits
- Class D: | 1110 | Multicast Group addresses | — 4, 28 bits
- Class E: Reserved.



Rensselaer Polytechnic Institute
Shivkumar Kalyanaraman

Subnet Addressing

- Classful addressing inefficient: Everyone wants class B addresses
- Can we split class A, B addresses spaces and accommodate more networks ?
  - Need another level of hierarchy. Defined by "***subnet mask***", which is general specifies the sets of bits belonging to the network address and host address respectively
  - External routers send to "network" specified by the "network ID" and have smaller routing tables



Boundary is flexible, and defined by subnet mask

Rensselaer Polytechnic Institute
Shivkumar Kalyanaraman

Subnet Addressing (Contd)

- Internal routers & hosts use subnet mask to identify "subnet ID" and route packets between "subnets" within the "network".
- Eg: Mask: 255.255.255.0 => subnet ID = 8 bits with upto 62 hosts/subnet
- Route table lookup:
  - IF ((Mask[i] & Destination Addr) = =

    Destination[i])

    Forward to NextHop[i]

Addressing and Forwarding Summary

- Addressing:
  - Unique IP address per interface
  - Classful (A,B,C) => address allocation not efficient
  - Hierarchical => smaller routing tables
  - Provision for broadcast, multicast, loopback addresses
  - Subnet masks allow "subnets" within a "network" => improved address allocation efficiency
  - Problem: Host moves between networks => IP address changes.

Addressing/Forwarding Summary(contd)

- Forwarding:
  - Simple *"next-hop"* forwarding.
  - Last hop forwards directly to destination
  - *Best-effort delivery :* No error reporting. Delay, out-of-order, corruption, and loss possible => problem of higher layers!
  - Forwarding vs routing: Routing tables setup by separate algorithm (s)

IP Features

- Connectionless service
- Addressing
- Data forwarding
- Fragmentation and reassembly
- Supports variable size datagrams
- Best-effort delivery: Delay, out-of-order, corruption, and loss possible. Higher layers should handle these.
- Provides only "Send" and "Delivery" services
  Error and control messages generated by
  Internet Control Message Protocol (ICMP)

What IP does NOT provide

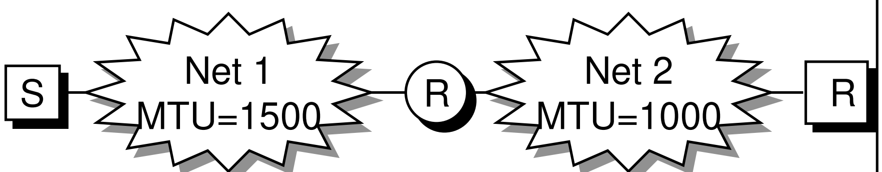
- End-to-end data reliability & flow control (done by TCP or application layer protocols)
- Sequencing of packets (like TCP)
- Error detection in payload (TCP, UDP or other transport layers)
- Error reporting (ICMP)
- Setting up route tables (RIP, OSPF, BGP etc)
- Connection setup (it is connectionless)
- Address/Name resolution (ARP, RARP, DNS)
- Configuration (BOOTP, DHCP)
- Multicast (IGMP, MBONE)

IP Datagram Format

0	4	8	16		32
Vers	H Len	TOS	Total Length		
Identification			Flags	Fragment Offset	
Time to live		Protocol	Header Checksum		
Source IP Address					
Destination IP Address					
IP Options (if any)				Padding	
Data					

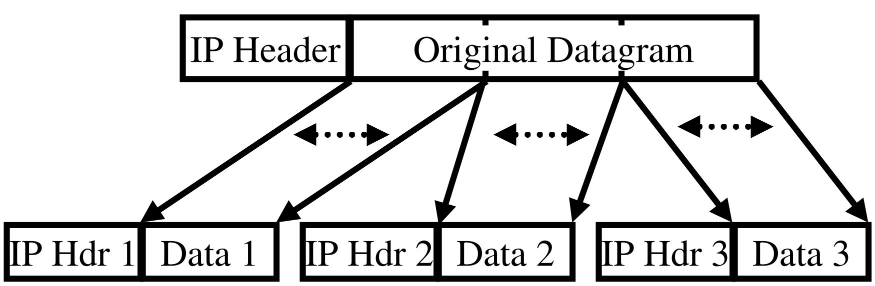
Maximum Transmission Unit

- Each subnet has a maximum frame size
  Ethernet: 1518 bytes
  FDDI: 4500 bytes
  Token Ring: 2 to 4 kB
- Transmission Unit = IP datagram (data + header)
- Each subnet has a maximum IP datagram length (header + payload) = MTU



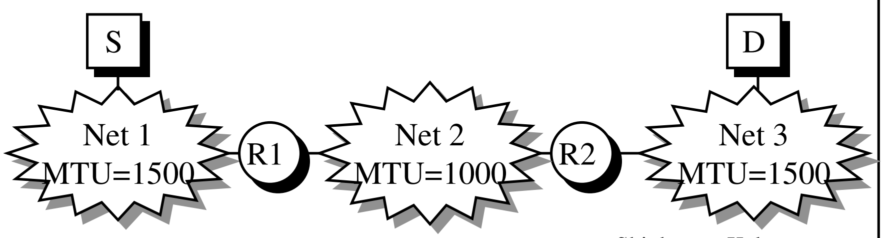
Fragmentation

- Datagrams larger than MTU are fragmented
- Original header is copied to each fragment and then modified (fragment flag, fragment offset, length,...)
- Some option fields are copied (see RFC 791)



Reassembly

- ❑ Reassembly only at the final destination
- ❑ Partial datagrams are discarded after a timeout
- ❑ Fragments can be further fragmented along the path. Subfragments have a format similar to fragments.
- ❑ Minimum MTU along a path ⇒ Path MTU



Further notes on Fragmentation

- ❑ Performance: single fragment lost => entire packet useless. Waste of resources all along the way. Ref: Kent & Mogul, 1987
- ❑ Don't Fragment (DF) bit set => datagram discarded if need to fragment. ICMP message generated: may specify MTU (default = 0)
- ❑ Used to determine Path MTU (in TCP & UDP)
- ❑ The transport and application layer headers do not appear in all fragments. Problem if you need to peep into those headers.

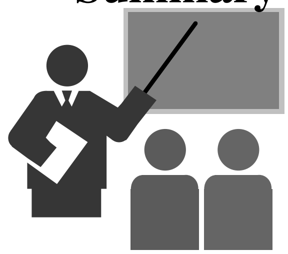
Address Resolution

- Indirection through addressing/naming => requires resolution
- Problem usually is to map destination layer N address to its layer N-1 address to allow packet transmission in layer N-1.
- **1. Direct mapping:** Make the physical addresses equal to the host ID part.
  - Mapping is easy.
  - Only possible if admin has power to choose both IP and physical address.
  - Ethernet addresses come pre-assigned (so do part of IP addresses!).
  - Ethernet addresses are 48 bits vs IP addresses which are 32-bits.

ARP techniques (contd)

- **2: Table Lookup:**
  Searching or indexing to get MAC addresses
  - Similar to lookup in /etc/hosts for names
  - Problem: change Ethernet card => change table
- **3. Dynamic Binding: ARP**
  - The host *broadcasts* a request:
    "What is the MAC address of 127.123.115.08?"
  - The host whose IP address is 127.123.115.08 replies back:
    "The MAC address for 127.123.115.08 is 8A-5F-3C-23-45-56_{16}"
- All three methods are allowed in TCP/IP networks.

Summary



- IP header: supports connectionless delivery, variable length pkts/headers/options, fragmentation, reassembly, path MTU discovery
- New forwarding algorithm, ARP for address resolution