SKIP Until pp.15

Resolving Addresses

- Hardware only recognizes MAC addresses
- IP only uses IP addresses
- Consequence: software needed to perform translation
  - Part of network interface
  - Known as address resolution

Address Resolution

- Layer 2 protocol
- Given
  - A locally-connected network, N
  - IP address C of computer on N
- Find
  - Hardware address for C
- Technique
  - Address Resolution Protocol

Address Resolution Protocol (ARP)

- Keep bindings in table
- Table entry contains pair of addresses for one computer
  - IP address
  - Hardware address
- Build table automatically as needed

ARP Table

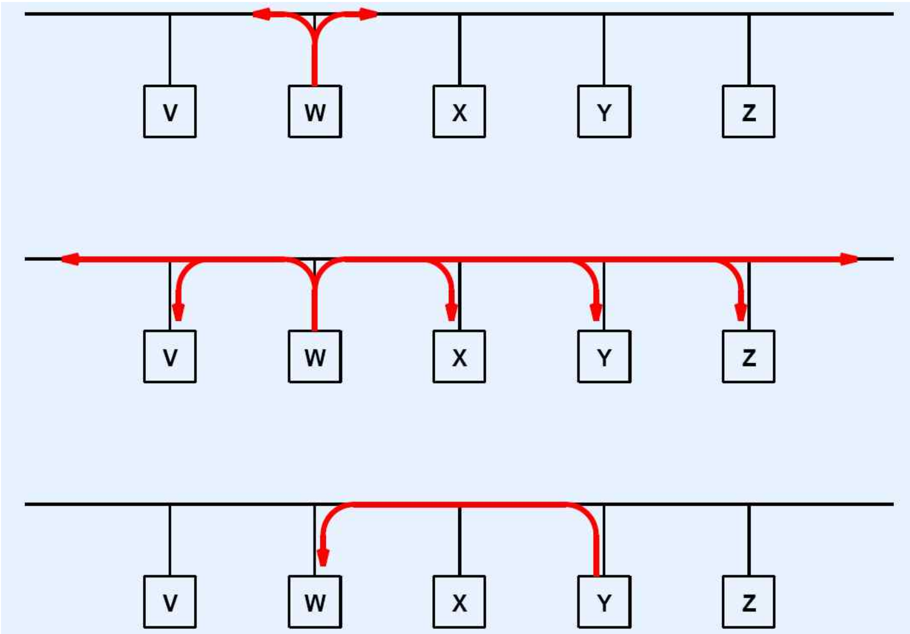
IP Address	Hardware Address
197.15.3.2	0A:07:4B:12:82:36
197.15.3.3	0A:9C:28:71:32:8D
197.15.3.4	0A:11:C3:68:01:99
197.15.3.5	0A:74:59:32:CC:1F
197.15.3.6	0A:04:BC:00:03:28
197.15.3.7	0A:77:81:0E:52:FA

- Only contains entries for computers on local network
- IP network prefix in all entries identical

ARP Lookup Algorithm

- Look for target IP address, T, in ARP table
- If not found
  - Send ARP request message to T
  - Receive reply with T's hardware address
  - Add entry to table
- Return hardware address from table

Illustration Of ARP Exchange



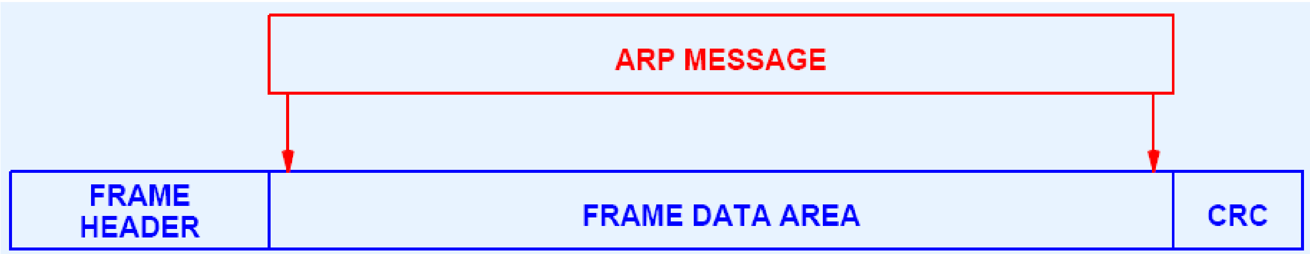
- W needs Y's hardware address
- Request sent via broadcast
- Reply sent via unicast

ARP Message Format (For Ethernet)

0	8	16	24 31
HARDWARE ADDRESS TYPE		PROTOCOL ADDRESS TYPE	
HADDR LEN	PADDR LEN	OPERATION	
SENDER HADDR (first 4 octets)			
SENDER HADDR (last 2 octets)		SENDER PADDR (first 2 octets)	
SENDER PADDR (last 2 octets)		TARGET HADDR (first 2 octets)	
TARGET HADDR (last 4 octets)			
TARGET PADDR (all 4 octets)			

- Length of hardware address fields depend on network type
- Ethernet uses 48-bit addresses

Transmission Of ARP Message In A Frame



- ARP message sent in payload area of frame
- Called encapsulation

Frame Type

Dest. Address	Source Address	Frame Type	Data In Frame
		806	complete ARP message

- Frame type identifies message as ARP
- Receiver examines frame type

Important Note

Because ARP software is part of the network interface software, all higher-layer protocols and applications can use IP addresses exclusively, and remain completely unaware of hardware addresses.

Summary

- Internetworking
  - Solves problem of heterogeneity
  - Includes LANs and WANs
- Internet concept
  - Virtual network
  - Seamless
  - Universal

Summary (continued)

- Internet architecture
  - Multiple networks
  - Interconnected by routers
- Router
  - Special-purpose computer system
  - Interconnects two or more networks
  - Uses table to forward datagrams

Summary (continued)

- Internet Protocol (IP)
  - Fundamental piece of TCP/IP
  - Defines
    - Internet addressing
    - Delivery semantics
    - Internet packet format (IP datagram)

Summary (continued)

- Address resolution
  - Needed to map IP address to equivalent hardware address
  - Part of network interface
  - Uses table
  - Automatically updates table entries
  - Broadcasts requests

PART XI

Internetworking

Part 2

(Datagram Encapsulation, Transmission, Fragmentation, Reassembly)

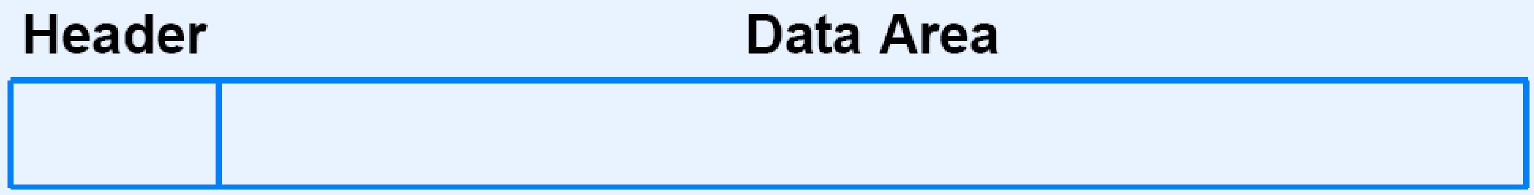
Motivation For IP Packets

Because it can connect heterogeneous networks, a router cannot transmit a copy of a frame that arrives on one network across another. To accommodate heterogeneity, an internet must define a hardware-independent packet format.

Internet Packets

- Abstraction
- Created and understood only by software
- Contains sender and destination addresses
- Size depends on data being carried
- Called IP datagram

The Two Parts Of An IP Datagram



- Header
  - Contains destination address
  - Fixed-size fields
- Payload
  - Variable size up to 64K
  - No minimum size

Datagram Header

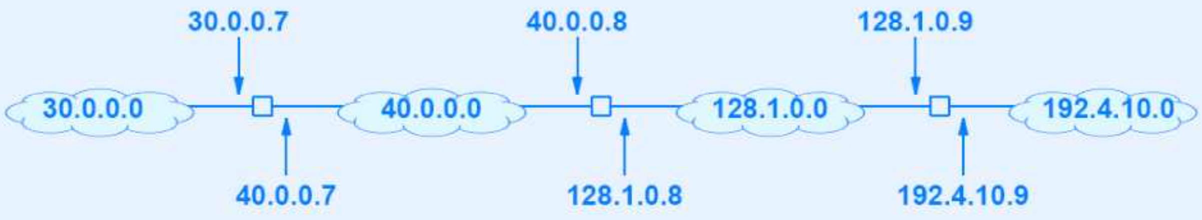
<table>
<tr><th>0</th><th>4</th><th>8</th><th>16</th><th>19</th><th>24   31</th></tr>
<tr><td>VERS</td><td>H. LEN</td><td>SERVICE TYPE</td><td colspan="3">TOTAL LENGTH</td></tr>
<tr><td colspan="3">IDENTIFICATION</td><td>FLAGS</td><td colspan="2">FRAGMENT OFFSET</td></tr>
<tr><td colspan="2">TIME TO LIVE</td><td>TYPE</td><td colspan="3">HEADER CHECKSUM</td></tr>
<tr><td colspan="6">SOURCE IP ADDRESS</td></tr>
<tr><td colspan="6">DESTINATION IP ADDRESS</td></tr>
<tr><td colspan="5">IP OPTIONS (MAY BE OMITTED)</td><td>PADDING</td></tr>
<tr><td colspan="6">BEGINNING OF DATA
⋮</td></tr>
</table>

- Three key fields
  - Source IP address
  - Destination IP address
  - Type (contents)

IP Datagram Forwarding

- Performed by routers
- Similar to WAN forwarding
  - Table-driven
  - Entry specifies next hop
- Unlike WAN forwarding
  - Uses IP addresses
  - Next-hop is router or destination

Example Of An IP Routing Table



(a)

Destination	Mask	Next Hop
30.0.0.0	255.0.0.0	40.0.0.7
40.0.0.0	255.0.0.0	deliver direct
128.1.0.0	255.255.0.0	deliver direct
192.4.10.0	255.255.255.0	128.1.0.9

(b)

Table (b) is for center router in part (a)

Routing Table Size

Because each destination in a routing table corresponds to a network, the number of entries in a routing table is proportional to the number of networks in an internet.

Datagram Forwarding

- Given a datagram
- Extract destination address field, D
- Look up D in routing table
- Find next-hop address, N
- Send datagram to N

Key Concept

The destination address in a datagram header always refers to the ultimate destination. When a router forwards the datagram to another router, the address of the next hop does not appear in the datagram header.

IP Semantics

- IP is connectionless
  - Datagram contains identity of destination
  - Each datagram sent/handled independently
- Routes can change at any time

IP Semantics (continued)

- IP allows datagrams to be
  - Delayed
  - Duplicated
  - Delivered out of order
  - Lost
- Called best effort delivery
- Motivation: accommodate all possible networks

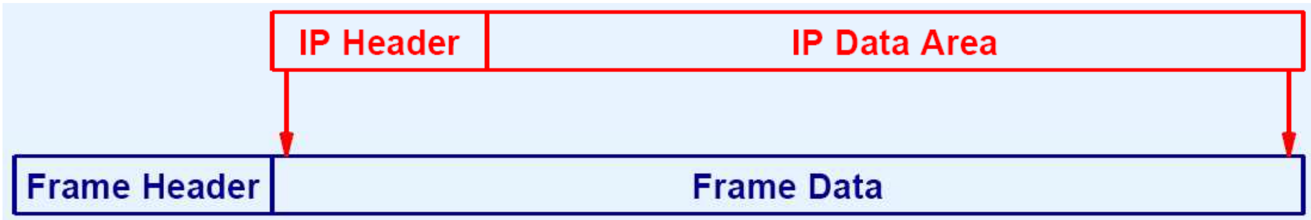
Internet Transmission Paradigm (General Case)

- Source host
  - Forms datagram
  - Includes destination address
  - Sends to nearest router
- Intermediate routers
  - Forward datagram to next router
- Final router
  - Delivers to destination host

Datagram Transmission

- Datagram sent across conventional network
  - From source host and router
  - Between intermediate routers
  - From final router to destination host
- Network hardware does not recognize
  - Datagram format
  - IP addresses
- Encapsulation needed

Illustration Of IP Encapsulation



- Entire datagram treated like data
- Frame type identifies contents as IP datagram
- Frame destination address gives next hop

Frame And Datagram Destination Addresses

- Frame address
  - Hardware (MAC) address
  - Next hop
- Datagram address
  - IP address
  - Ultimate destination

Frame Address For Encapsulated Datagram

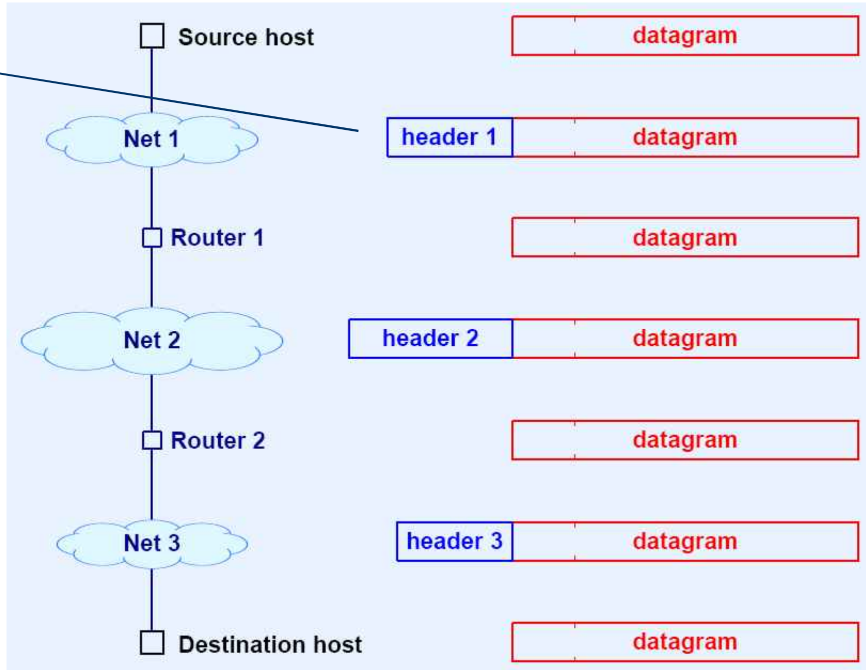
A datagram is encapsulated in a frame for transmission across a physical network. The destination address in the frame is the address of the next hop to which the datagram should be sent; the address is obtained by translating the IP address of the next hop to an equivalent hardware address.

Frames And Datagrams

- Datagram survives entire trip across Internet
- Frame only survives one hop

Illustration Of Frame Headers Used For Datagram Transmission

Depends on each Network



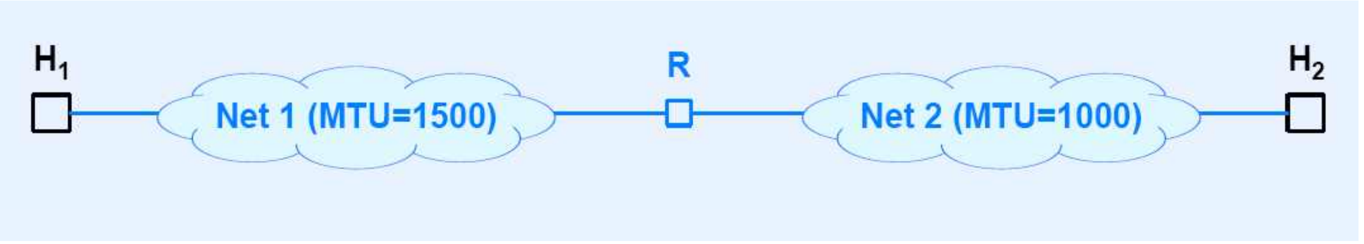
- Each hop extracts datagram and discards frame

Skip from here

Maximum Frame Size

- Each network technology imposes maximum frame size
  - Called Maximum Transmission Unit (MTU)
  - MTUs differ
- Internet
  - Can contain heterogeneous technologies
  - Must accommodate multiple MTUs

Illustration Of How Two MTUs Cause A Problem For IP

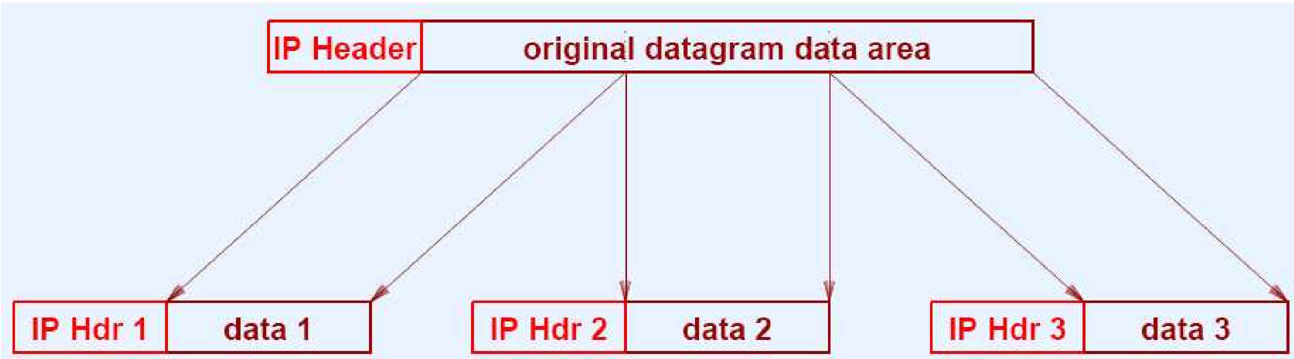


- Host 1
  - Creates datagram for Host 2
  - Chooses datagram size of 1500 octets
  - Transmits datagram across network 1
- Router R
  - Receives datagram over network 1
  - Must send datagram over network 2
  - Employs fragmentation

Datagram Fragmentation

- Performed by routers
- Needed when datagram larger than MTU of network
- Divides datagram into pieces called fragments
- Each fragment has datagram header
- Fragments sent separately
- Ultimate destination reassembles fragments

Illustration Of Datagram Fragmentation



- Each fragment has IP datagram header
- Header fields
  - Identify original datagram
  - Indicate where fragment fits

Example Of Reassembly

- Host H_1 generates 1500-octet datagram
- Router R_1 fragments
- Router R_2 transmits fragments
- Host H_2 reassembles

Multiple Fragmenting Points

- Let MTUs along internet path be
  - 1500
  - 1500
  - 1000
  - 1500
  - 576
  - 1500
- Result: fragmentation can occur twice

Fragmenting A Fragment

- Needed when fragment too large for network MTU
- Arbitrary subfragmentation possible
- Router divides fragments into smaller pieces
- All fragments at same "level"
  - Offset given with respect to original datagram
  - Destination cannot distinguish subfragments

Fragment Loss

- Receiver
  - Collects incoming fragments
  - Reassembles when all fragments arrive
  - Does not know identity of router that did fragmentation
  - Cannot request missing pieces
- Consequence: Loss of one fragment means entire datagram lost

Summary

- Internet transmission paradigm
  - Source host
  - Zero or more routers
  - Destination host
- Datagram encapsulated in network frame for transmission

Summary (continued)

- Network hardware has maximum payload size
  - Called MTU
  - Datagram must be smaller than hardware MTU
- Internet can have multiple MTUs

Summary (continued)

- Datagram fragmentation
  - Accommodates multiple MTUs
  - Performed by router
  - Divides datagram into pieces
  - Ultimate destination reassembles

Summary (continued)

- Fragments can be fragmented
  - Multiple levels possible
  - All offsets at one level
  - Loss of any fragment means loss of entire datagram