H.O. #10

Winter 98-99

Part II

Internet Protocols

Internet RFCs

- Internet standards are published as RFCs (Request For Comments)
- The RFCs are all available on-line at several sites:

http://www.cis.ohio-state.edu/hypertext/information/rfc.html

(HTML format, useful for reading on-line)

ftp://ftp.merit.edu/internet/documents/rfc

(ASCII and PS files, useful for printing)

Internet Protocol (IP)

- The *Internet Protocol* defines an unreliable, connectionless, best-effort delivery mechanism for the Internet.
  - Unreliable: packet delivery is not guaranteed
  - Connectionless: packets are treated independently; multiple packets between two nodes may take different paths and arrive out-of-order
  - Best Effort: packets are discarded when underlying networks fail or resources are exhausted
- The protocol specifies packet formats, processing, and error-handling.
- Reference: RFC 791

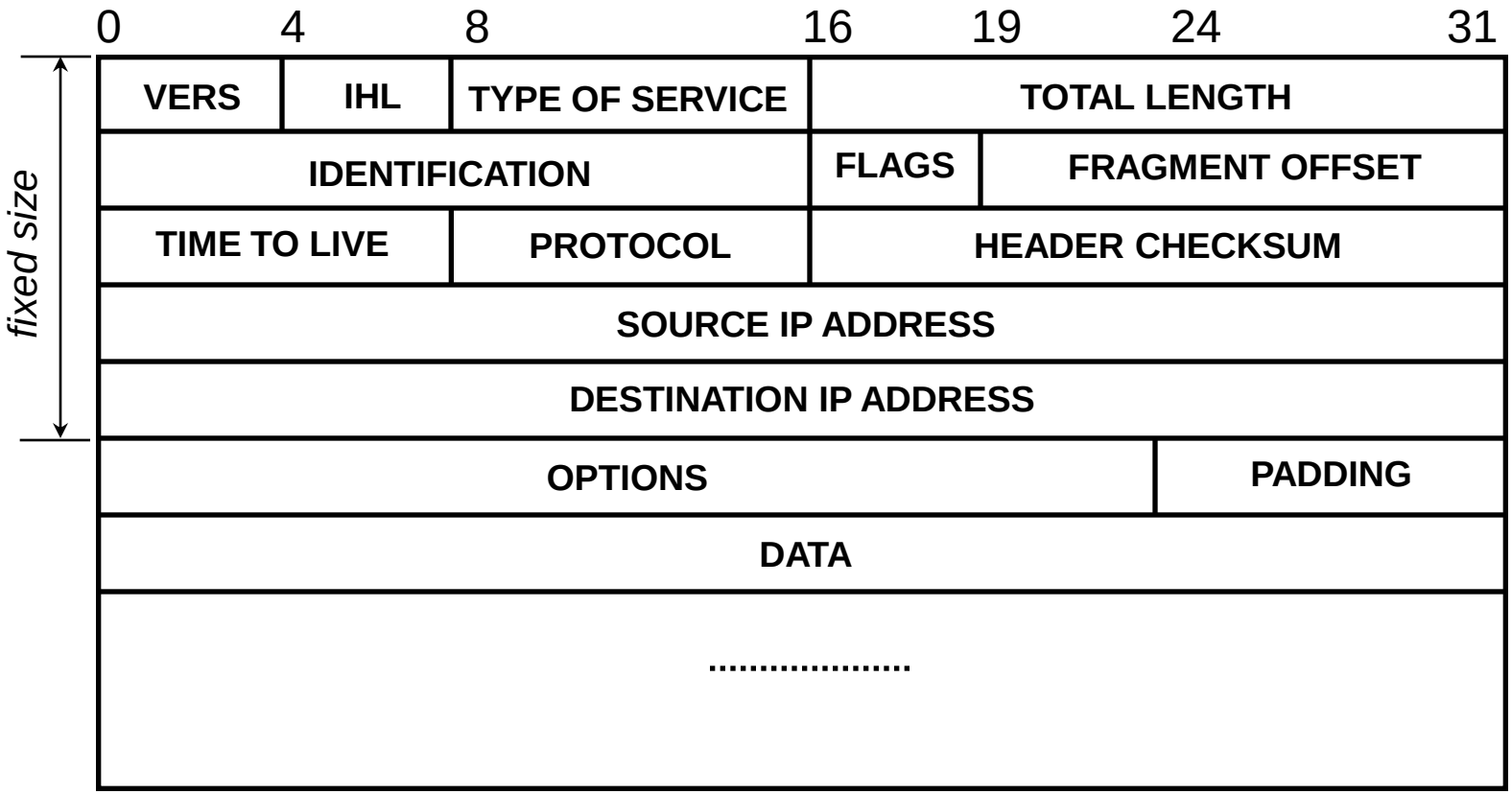
The Internet Datagram

- The *Internet Datagram* is the basic unit of information transfer in the Internet Protocol. It is divided into a datagram header and data area:

- The datagram is transported from source to destination through one or more intermediate networks. Within a *particular network*, it is *encapsulated* into a network frame:

- Sometimes a complete internet datagram will not fit into the frame size of an intermediate physical network. IP allows its datagrams to be *fragmented* . Once a datagram is fragmented, its fragments travel as separate datagrams all the way to the final destination.

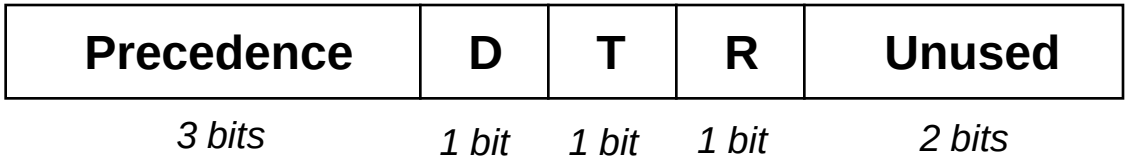
Datagram Format



Fields of the IP Datagram

- ***VERS***: specifies the IP protocol version in use. Machines reject datagrams with versions different from theirs. Currently IP version 4. Assigned version numbers can be found in RFC1700.
- ***IHL***: specifies datagram header length in 32-bit words (needed because field *OPTIONS* is variable length).
  - Minimum value is 5 (when OPTIONS field is 0)
  - Maximum value is 15 (60 bytes header)
- ***TOTAL LENGTH***: specifies total length of IP datagram measured in octets (including header and data).
  - Maximum length is 65,535 bytes

Datagram Type of Service and Precedence

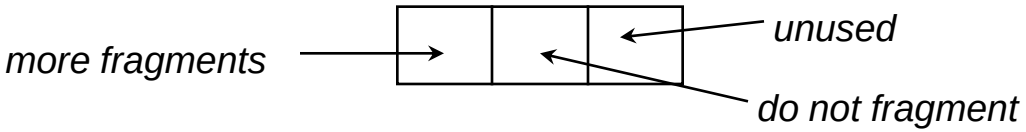


- The 8-bit *Type of Service* field in the IP datagram is subdivided as follows:
- ***Precedence***: (priority) specifies importance of a datagram:

  111 - Network Control
  110 - Internetwork Control
  101 - CRITIC/ECP
  100 - Flash Override
  011 - Flash
  010 - Immediate
  001 - Priority
  000 - Routine
- ***D***: requests low delay service
- ***T***: requests high throughput service
- ***R***: requests high reliability service
- The type of service specification is a "hint" to elements in the Internet which help them to choose among various available paths to a destination. The Internet cannot guarantee the type of service requested. It does not always make sense to choose all three types of service simultaneously, since there are trade-offs.

Fragmentation Control

- The following fields of the datagram header control fragmentation:
  - ***IDENTIFICATION***: contains a unique integer which identifies the datagram. Any gateway that fragments a datagram copies the *IDENTIFICATION* field into every fragment (host chooses a number to uniquely identify each datagram).
  - ***FLAGS***: (3 bits) contains a *do not fragment* bit and a *more fragments* bit, the third bit is unused. The more fragments bit allows a destination to know where the end of the original datagram is.



  - ***FRAGMENT OFFSET***: specifies the offset (in units of ***8 bytes***) of *this* fragment into the original datagram (all fragments except the last one must be multiples of 8 bytes).

Datagram Lifetime

- The *TIME TO LIVE* field specifies how long (in seconds) a datagram is allowed to remain on the Internet system. Packets that exceed their lifetime are discarded. Since it is difficult for routers to know exact transit time in networks, simple rules are used:
  - Each router along the path from source to destination decrements *TIME TO LIVE* by 1 when it processes the datagram header
  - To handle the case of overloaded routers that may introduce long delays, the local arrival time is recorded and the *TIME TO LIVE* counter decrements by the number of seconds the datagram waited for service inside the router.

Other Datagram Header Fields

- ***PROTOCOL***: (protocol ID) specifies which transport layer process is to receive this datagram. Assigned protocol IDs can be found in RFC1700.
- ***HEADER CHECKSUM***: Checksum is computed only on the header (including *OPTIONS*), which reduces processing time at gateways (adds up all the 16 bit halfwords using 1's complement arithmetic then takes the one's complement of the result)
- ***PADDING***: octets containing zeros that are needed to ensure that the Internet header extends to an exact multiple of 32 bits (since the header length is specified in 32-bit words).

Internet Datagram Options

- The *OPTIONS* field is used for testing and debugging in the Internet, and for signaling special options.
- The length varies, depending upon which options are selected. There are two cases for the format of an option:
  - A single option code byte; or
  - An option code byte, an option length byte, and data bytes associated with the option.

Option Codes

- The *option code* octet is divided into three fields, as shown below:

- ***Copy*** specifies how a gateway handles options during fragmentation. Copy=1 means the option is copied onto all fragments; Copy = 0 specifies that the option is only copied onto the first fragment.
- ***Option Class***:

Option Class	*Meaning*
0	Datagram or network control
1	Reserved for future use
2	Debugging and measurement
3	Reserved for future use

IP Option Numbers

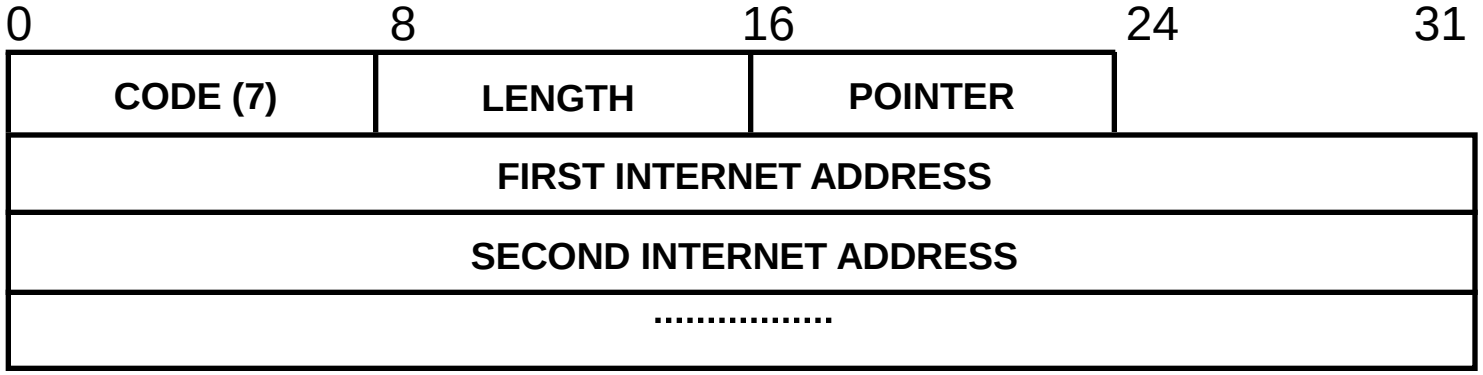
- Some of the defined IP option numbers are listed below:

Option Class	*Option Number*	*Length*	*Description*
0	0	1	End of option list. Used if options do not end at end of datagram
0	1	1	No operation
0	2	11	***Security*** and handling restrictions
0	3	var	***Loose source routing***. Used to route datagram along specified path
0	7	var	***Record route***. Used to trace route
0	9	var	***Strict source routing***. Used to route datagram along a specified path
2	4	var	***Internet timestamp***. Used to record timestamps along the route

Note: var stands for *variable*

Record Route Option

- The *Record Route* option provides a way to monitor how gateways route datagrams. The format is shown below:



- ***CODE***: specifies the option number and class
- ***LENGTH***: gives length of option as it appears in IP datagram.
- ***INTERNET ADDRESS***: denotes the area reserved for internet addresses. This region is initially empty. Each router along the datagram path enters its address on the list.
- ***POINTER***: points to next available internet address slot in the option. When a gateway receives the datagram, it puts its address in the slot given by the pointer.

Source Route Options

- The *Source Route* options allow network designers to dictate the path of a datagram through the network.
- *Strict Source Routing*: specifies a sequence of internet addresses which a datagram must follow. The path between any two addresses can consist of only a single physical network.
- *Loose Source Routing*: specifies a sequence of internet addresses which a datagram must follow. The path between any two addresses may consist of multiple network hops.
- The format of the option is very similar to the Record Route option. There is a code, length, and pointer, along with a list of internet addresses forming the specified route.

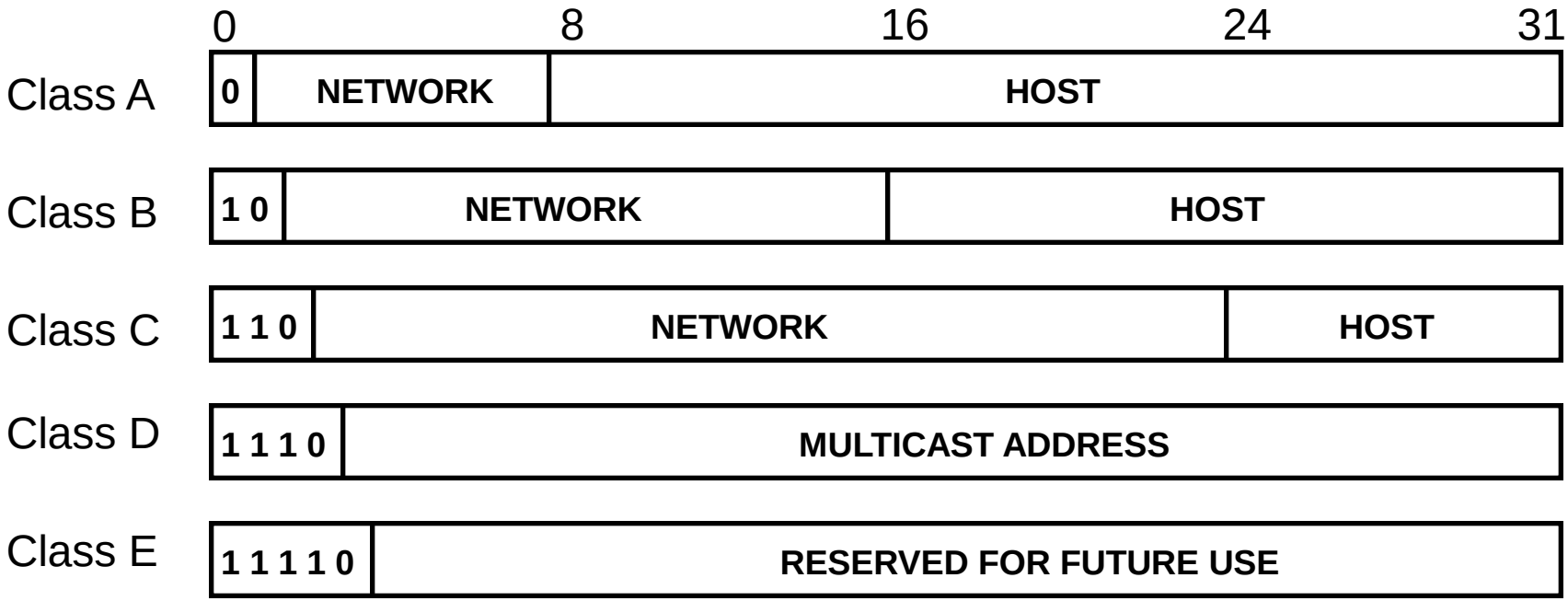
Timestamp Option

- The *timestamp* option, like the record route option, has an initially-empty list, and each router along the path from source to destination fills in one item on the list. Entries here are the time at which the datagram passes through a particular gateway and (possibly) the identity of the gateway.
- The value of the timestamp is the number of milliseconds since midnight, Universal Time.

Internet Addresses (1)

- IP addresses are unique over the whole Internet
- IP addresses are 32-bit long and consist of a *network address* part and a *host address* part. They are represented by the *dotted decimal* notation, where each byte is written in decimal values (from 0 to 255).
  - e.g. 10000000 00001010 00000010 00011110
  - 128. 10. 2. 30.
- IP addresses are grouped intro classes depending on the size of the host part of the address

Internet Addresses (2)



IP address formats

Internet Addresses (3)

- ***class A***
  - handful of networks ($2^7 = 128$)
  - # hosts > 2^{16} (65,536)
- ***class B***
  - intermediate size networks
  - 2^8 (256) < # hosts < 2^{16} (65,536)
- ***class C***
  - smaller networks
  - # hosts < 2^8 (256)

Internet Addresses (4)

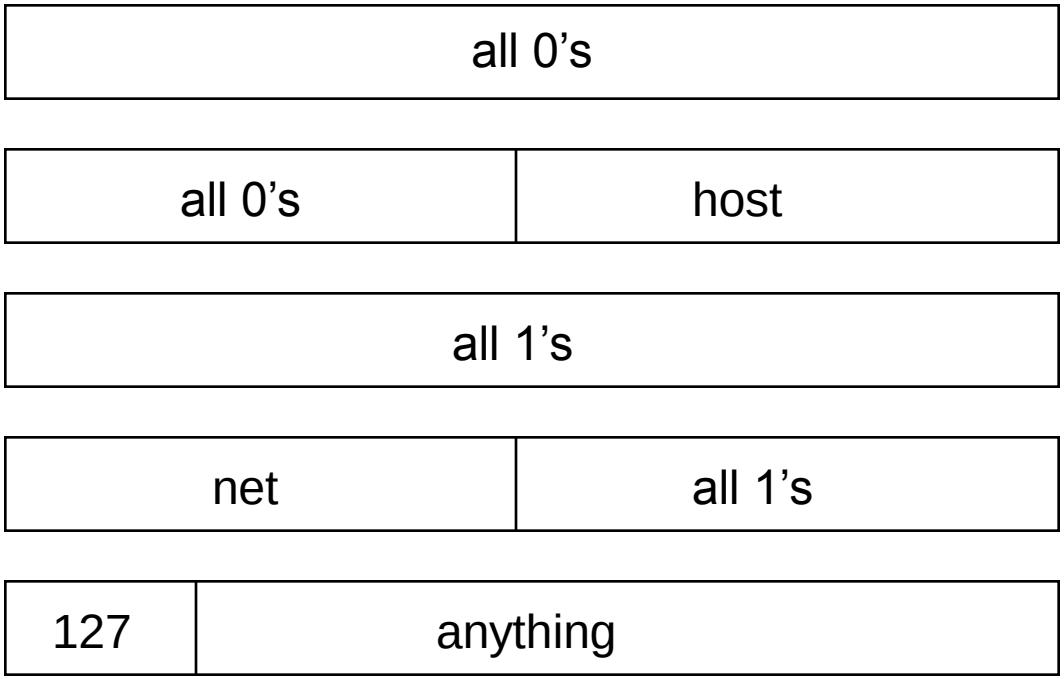
Class	Lowest Address	Highest Address
A	1.0.0.0	126.0.0.0
B	128.0.0.0	191.255.0.0
C	192.0.0.0	223.255.255.0
D	224.0.0.0	239.255.255.255
E	240.0.0.0	247.255.255.255

Note: the advent of CIDR (Classless Inter-Domain Routing) has, by and large, obsoleted the concept of network class.

Internet Addresses (5)

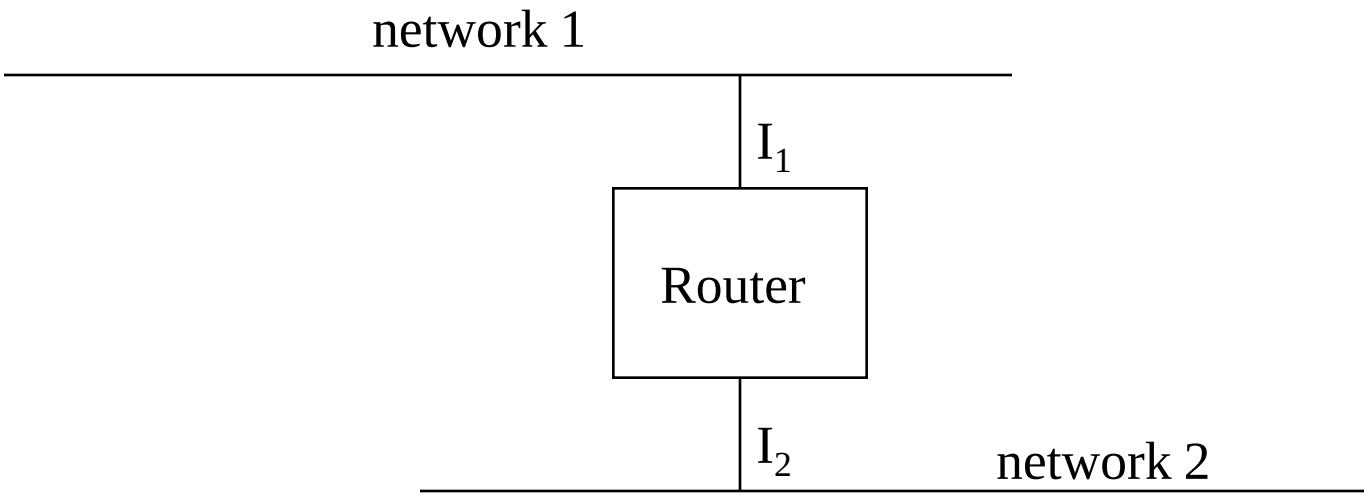
- Special addresses:
  - 0.0.0.0 means ***this*** host, used by machines as source address when they boot up (if they don't know their IP address)
  - 255.255.255.255 broadcast address on local network
  - Network part all zeros: means host on ***this*** network
  - Host part all ones: broadcast address on a specific network; routers typically do not forward these datagrams.
  - 127.x.x.x loopback (datagrams are looped back in software; they are not sent on any physical interface)

Internet Addresses (6)



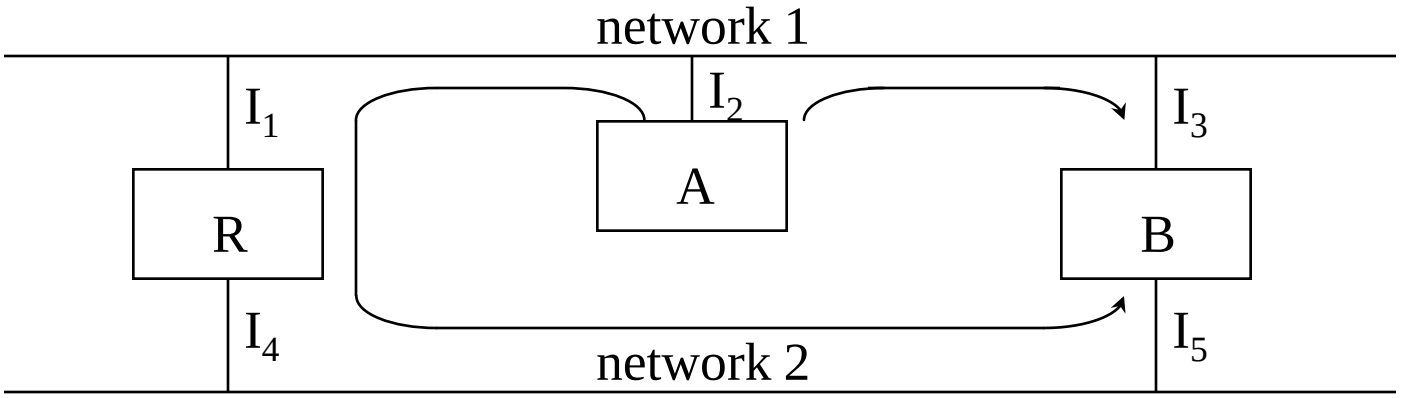
Internet Addresses (7)

- Addresses specify Network Connections
  - multi-homed hosts and routers require multiple IP addresses
  - an IP address identifies an ***interface***



Internet Addresses (8)

- Because routing uses the network portion of the IP address, the path taken by packets traveling to a host with multiple IP addresses depends on the address used.
- It may be impossible to reach the destination knowing only one out of a multiplicity of addresses

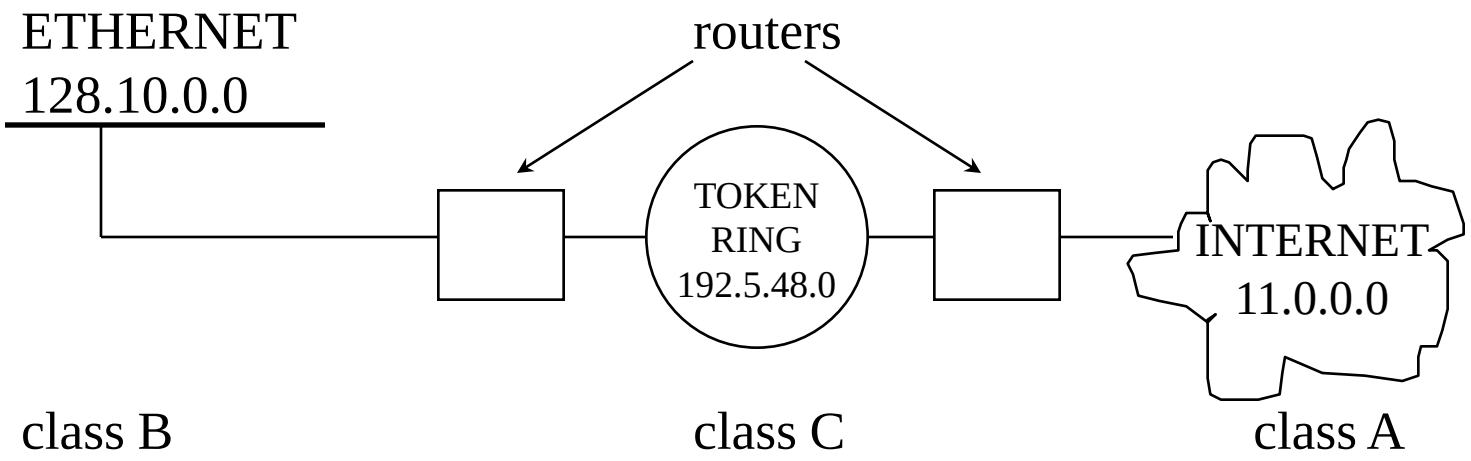


what if the connection of B to network 1 breaks?

Internet Addresses (9)

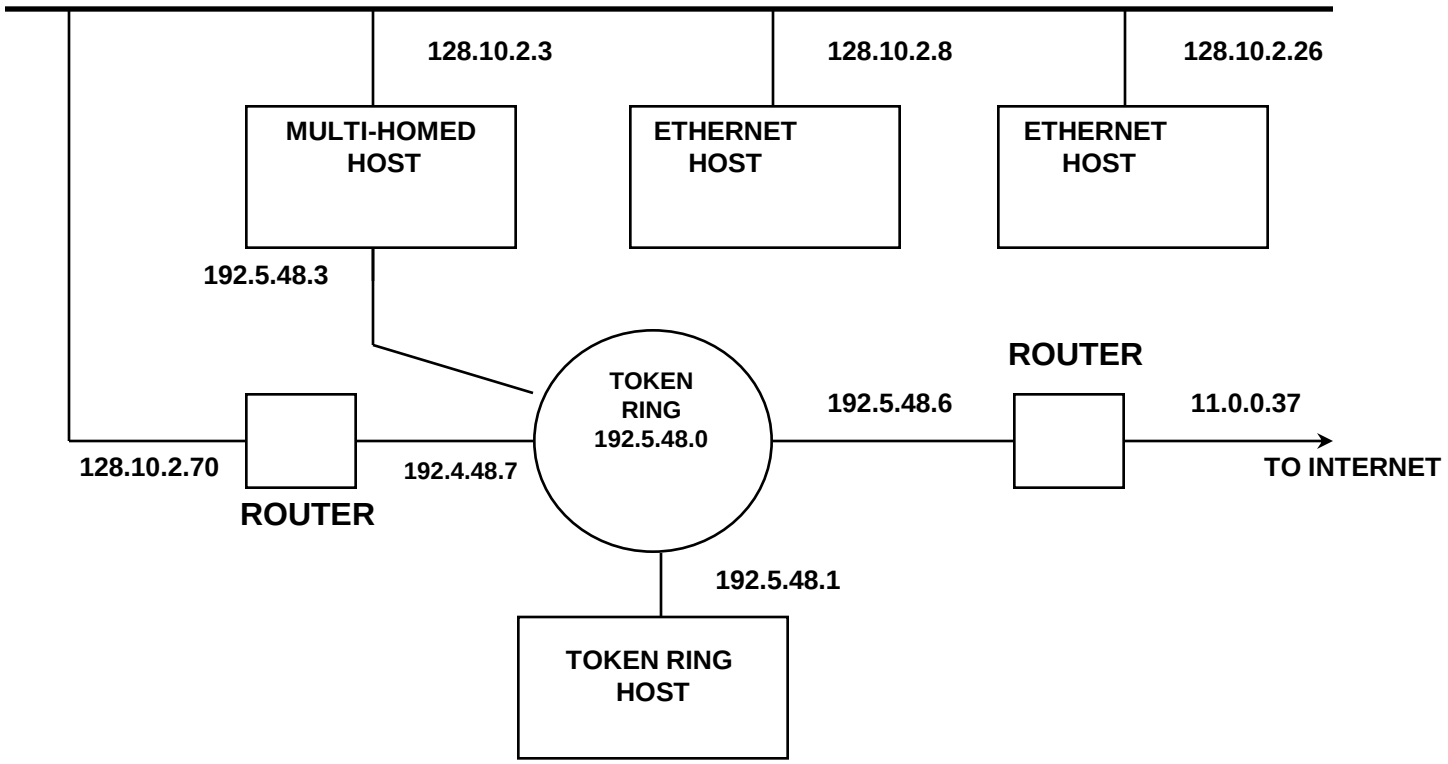
- To ensure that the network portion of an Internet address is unique, all internet addresses are assigned by a The Internet Assigned Number Authority (IANA)
  - policy, and ultimate control over number assigned
- Internet Network Information Center (INTERNIC) issues network addresses. Blocks are administered in a hierarchical fashion.
- The IP address for a host is a function of the network it is connected to; if the host is moved to another network, its address must change.

Internet Addresses (10)



The logical connection of two networks to the Internet backbone. Each network has been assigned an IP address.

Internet Addresses (11)



Private Internets

- Private Internets have no direct connection to the Internet
- Blocks of addresses have been reserved for Private Internets (RFC1918); these addresses will never be used in the Internet.
- Reserved blocks:
  - 10.0.0.0 - 10.255.255.255 (1 class-A network)
  - 172.16.0.0 - 172.31.255.255 (16 class-B networks)
  - 192.168.0.0 - 192.168.255.255 (256 class-C networks)

Translating Between IP Addresses and MAC Addresses

- Each interface has an IP address at Layer 3, and a MAC address at Layer 2.
- Assume that host A wants to send a packet to host B.
- Host A knows the IP address of host B; however, in order to transmit the packet, host A must somehow know or find out what is the MAC (layer 2) address of host B.
- Solution: the Address Resolution Protocol (ARP), RFC826

ARP

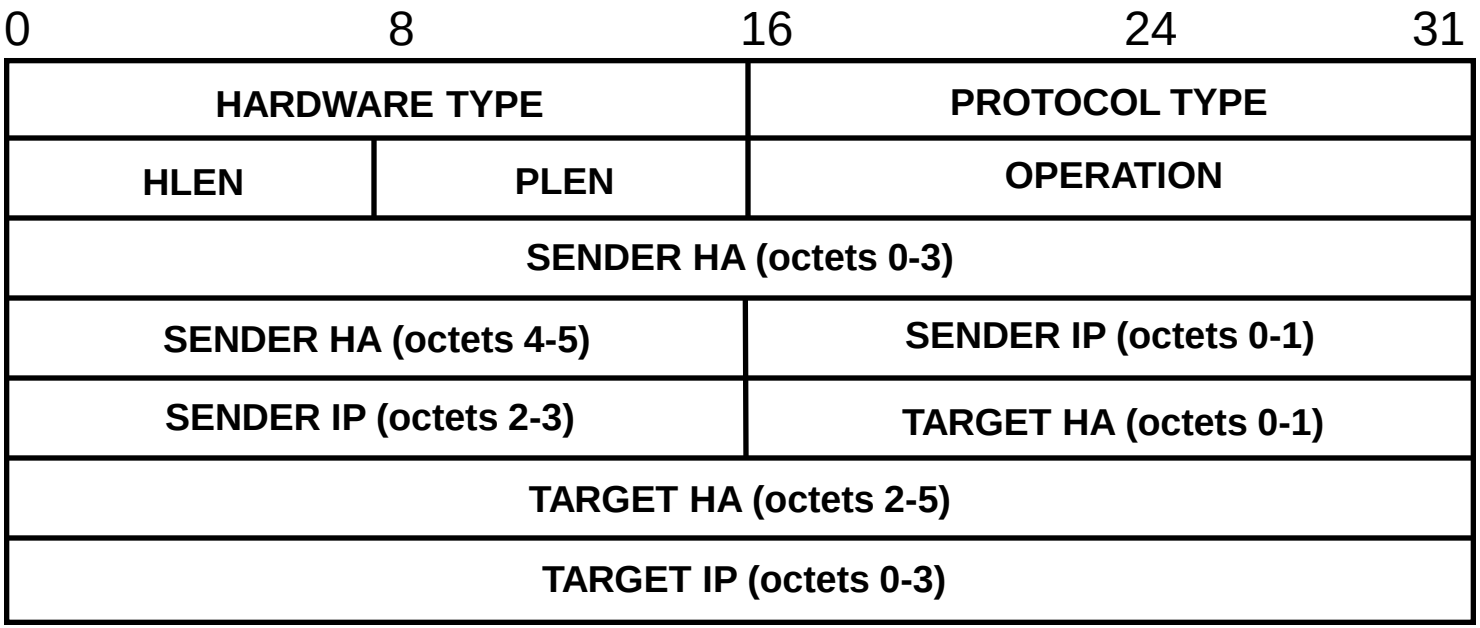
Address Resolution Protocol (1)

- Used to find the physical address of a target host on the local physical network, given only the host's IP address
- Mechanism:
  - The source broadcasts a special packet asking host with the target IP address to respond with a message carrying the (IP address, physical address) mapping
  - All hosts on the local physical network receive the broadcast, but only the target recognizes its IP address and responds to the request
  - When the source receives the reply, it sends the packet to the target using the target's physical address and places the mapping in its cache (a cache is used to prevent repeated broadcasts for the same destination)

ARP (2)

- ARP refinements
  - Source includes its (IP address, physical address) mapping in the ARP request anticipating the target's need for it in the near future. This avoids extra network traffic.
  - When all machines receive the ARP request broadcast, they can store the address mapping in their cache
- ARP is used when a IP to physical address mapping changes to notify hosts on the network of the change
- ARP messages are encapsulated in MAC frames. A special value in the type field of the frame is used to indicate that it is carrying an ARP message
- Entries in the local ARP cache for each host time out after a certain period

ARP (3)

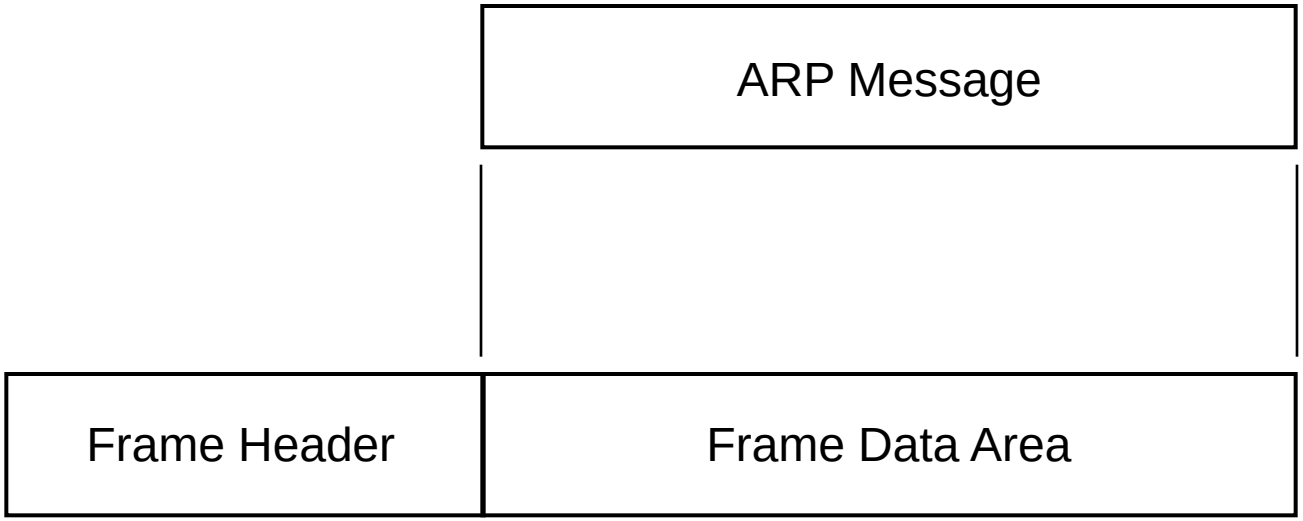


ARP Message Format

ARP (4)

HARDWARE TYPE: specifies type of hardware interface for which the request is made (e.g., 1 for Ethernet)
PROTOCOL TYPE: specifies high level protocol address supplied in message (e.g. 0800_{16} for IP)
HLEN and **PLEN**: specify length of fields for hardware address and protocol address respectively
OPERATION: specifies if this is an ARP request or reply message (1 for ARP request, 2 for ARP response, 3 for RARP request and 4 for RARP response)
HA and **IP**: **ha**rdware and IP addresses respectively

ARP (5)



- Frame type field in Ethernet frames contains 0806_{16}

Determining an Internet Address at Start-up: RARP

- Usually, a machine's IP address is kept on its secondary storage (OS finds it at start up)
- Issue : Diskless Workstations !
  - files are stored on a remote server
  - need IP address to use TCP/IP to obtain initial boot image.
- Solution : Use physical address to identify machine.
  - Given a physical network address, find the corresponding internet address
  - => Reverse Address Resolution Protocol (RARP), RFC903

RARP (2)

- Mechanism
  - Sender broadcasts a RARP request, supplying its physical network address in the Target HA field
  - Only machines authorized to supply the RARP service (RARP servers) process the request and send a reply filling in the target internet address
- Mechanism allows a host to ask about an arbitrary target;
  - thus sender HA is separate from target HA address
  - server replies to sender's HA

RARP (3)

- Issues :
  - Timing RARP Transactions
    - lost or delayed RARP requests
    - Repeated requests (delayed appropriately)
  - Primary and Back-up RARP servers
- Ethernet frame Type for RARP is 8035_{16}

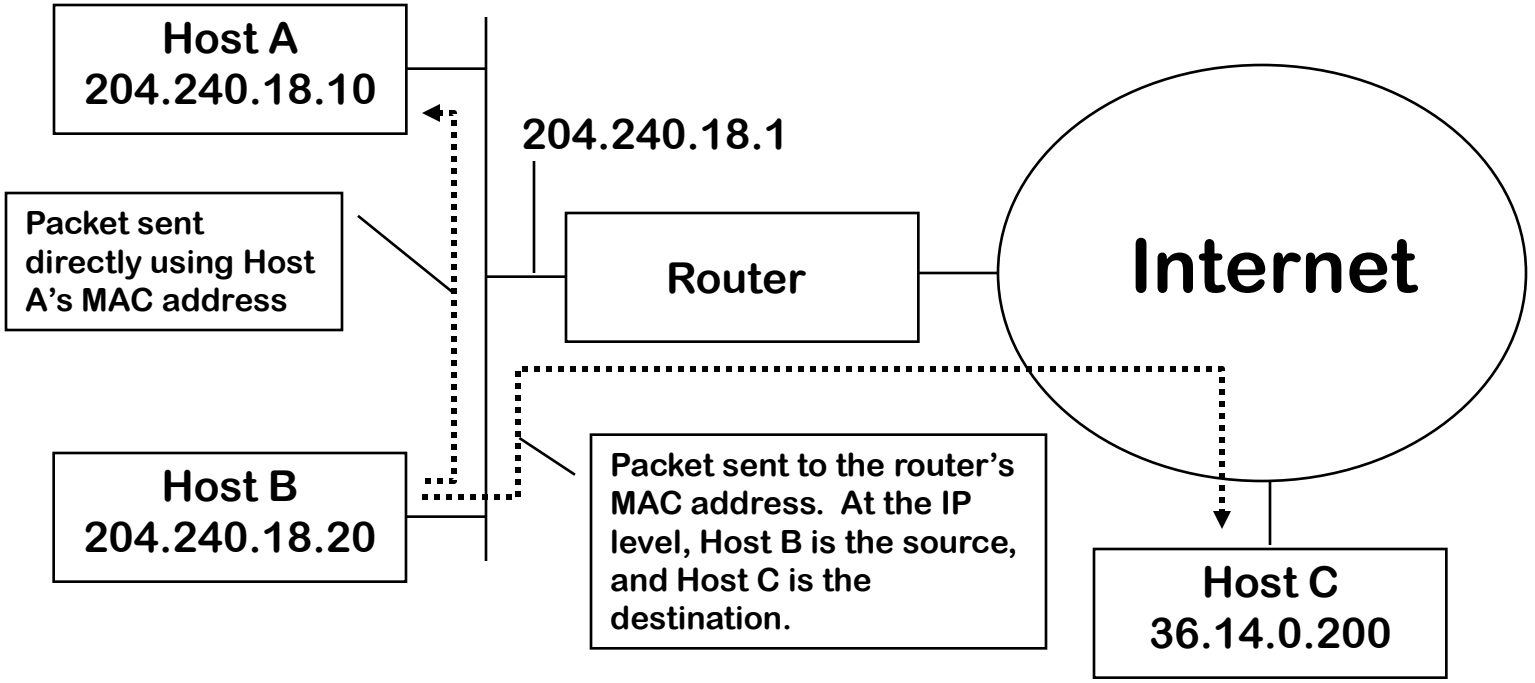
Routing IP Datagrams (1)

- Direct Delivery :
  - Transmission of an IP datagram between two machines on a single physical network does not involve routers
  - The sender encapsulates the datagram in a physical frame, binds the destination IP address to a physical hardware address, and sends the resulting frame directly to the destination.

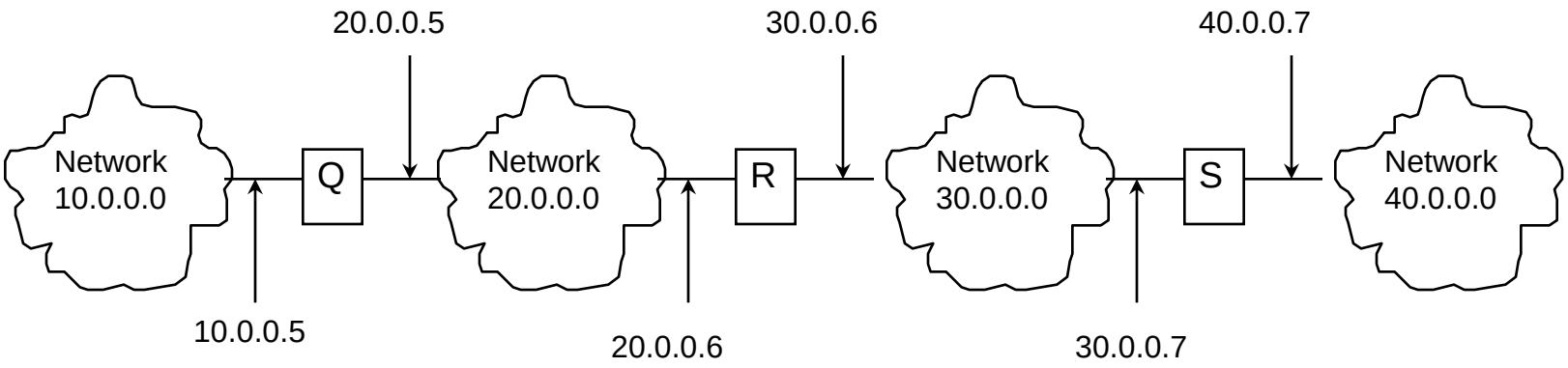
Routing IP Datagrams (2)

- Indirect delivery
  - through intermediate routers
  - At the MAC layer, the packet gets sent to the router's MAC address
  - routing decisions are made based on network prefixes (not full IP address)
  - "Next hop" refers to next router IP address on the route to destination
  - Host also performs routing decisions based on routing tables

Routing IP Datagrams (3)



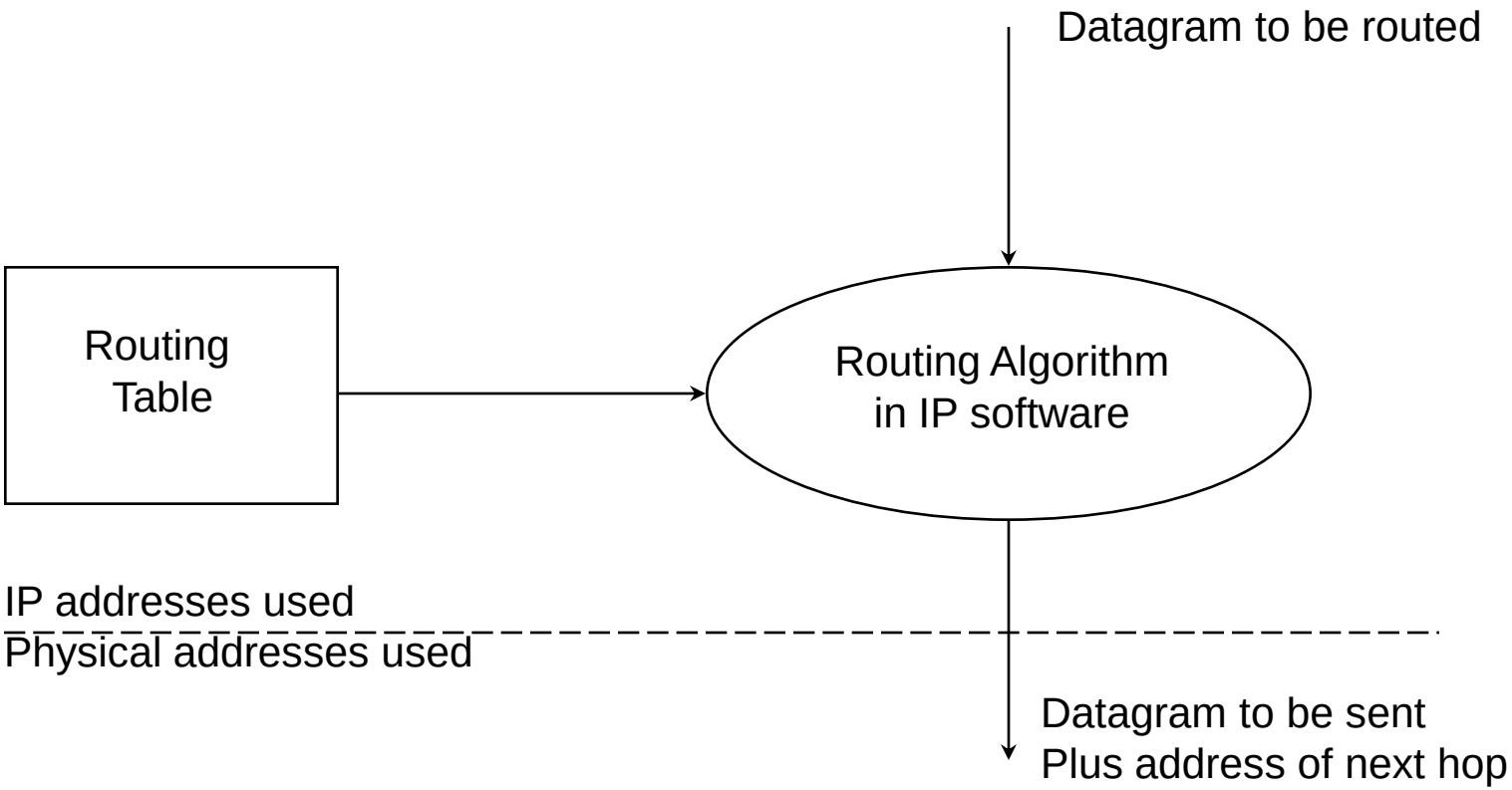
Routing IP Datagrams (4)



To Reach Hosts on Network	Route To This address
20.0.0.0	Deliver Directly
30.0.0.0	Deliver Directly
10.0.0.0	20.0.0.5
40.0.0.0	30.0.0.7

Routing Table in R

Routing IP Datagrams (5)



Subnet and Supernet Address Extensions (1)

- Original IP addressing scheme
  - Each physical network is assigned a unique network address.
  - Each host on a network has the network address as a prefix of the host's individual address.

Subnet and Supernet Address Extensions (2)

- Issues:
  - Large number of small size networks leads to :
    - Immense administrative overhead to manage network addresses.
    - Routing tables in routers extremely large.
    - High load on the internet due to routing information exchanges among routers.
    - Address space will eventually be exhausted.
  - Insufficient class B prefixes to cover all medium-size networks.

Subnet and Supernet Address Extensions (3)

- Observations
  - To minimize network addresses, the same IP network prefix must be shared by multiple physical networks.
  - To minimize class B addresses, class C addresses must be used.

Subnet and Supernet Address Extensions (4)

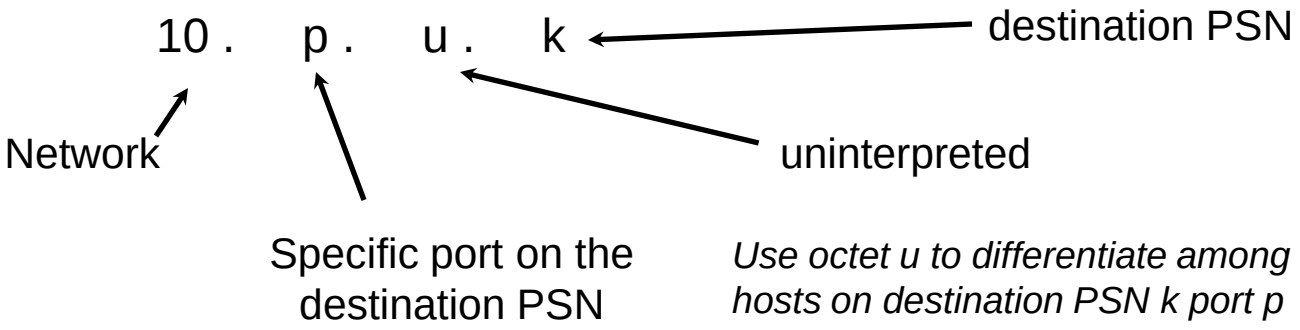
- Another observation
  - Individual sites have the freedom to modify addresses and routes as long as modifications remain invisible to other sites:
    - All hosts & routers at the site agree to honor the site's addressing scheme.
    - Other sites on the internet can treat addresses as in the original scheme.

Subnet Address Extensions

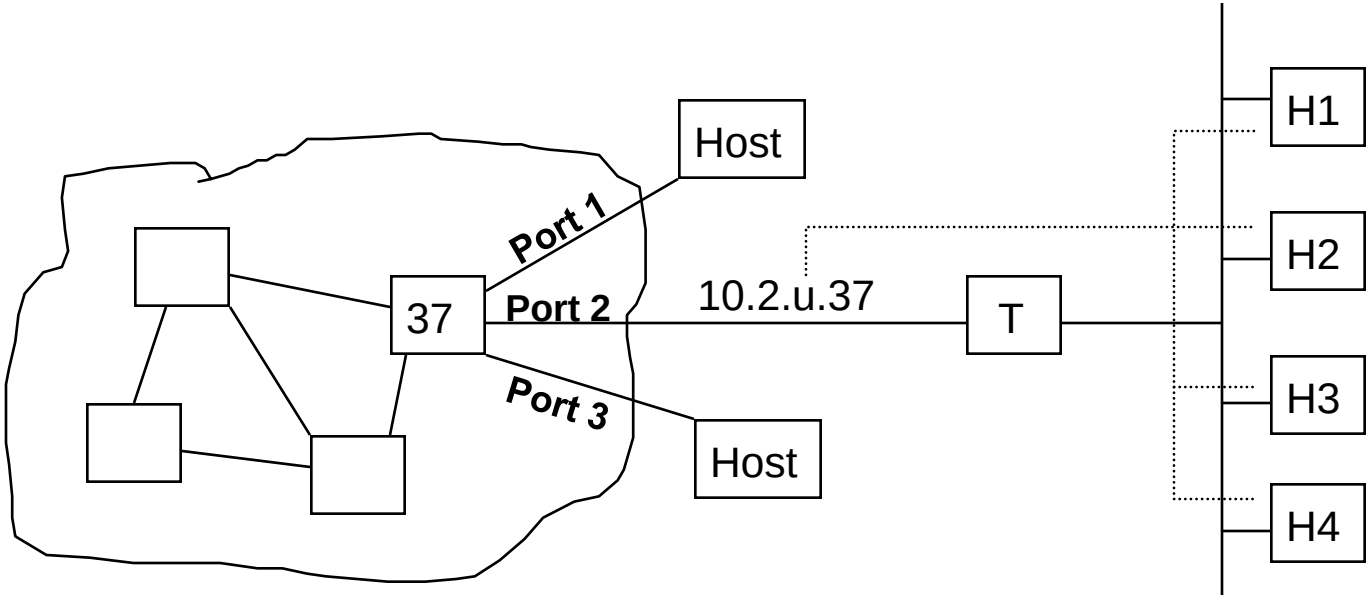
- Techniques used:
  - Transparent routers;
  - Proxy ARP;
  - Subnet Addressing.
- Transparent routers and subnet addressing are based on creating an additional level of hierarchy in the host ID field.
- Proxy ARP is based on an additional (transparent) level of indirection

Subnet Address Extensions
Transparent Routers(1)

- Consider a wide area network assigned a class A IP address; e.g., ARPNET with address 10.0.0.0
- Each packet switch node (PSN) is given a unique address considered to be of the form 10.p.u.k with the following interpretation:

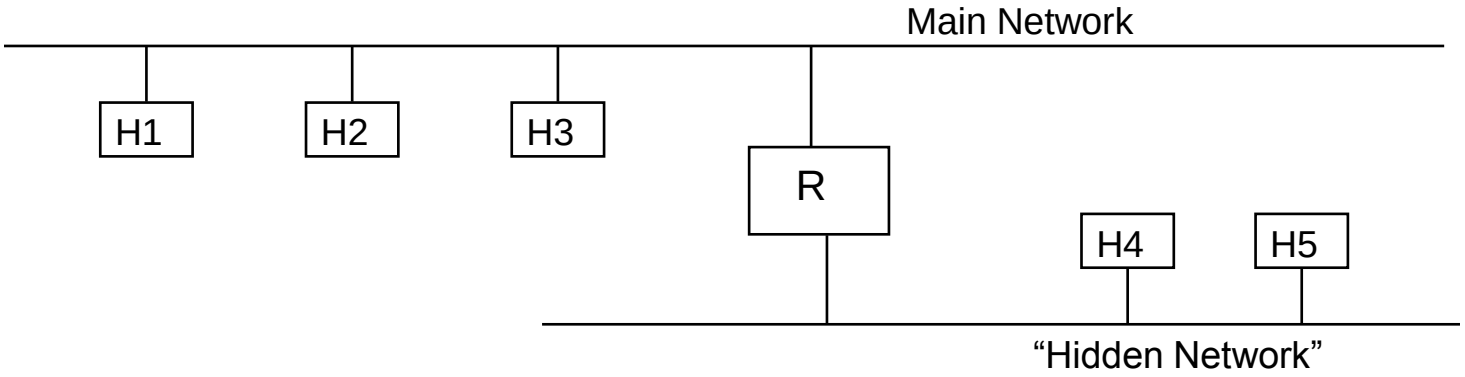


Subnet Address Extensions
Transparent Routers(2)



- Transparent router T extends a wide area network to multiple hosts at a site (no separate IP prefix is needed for the local area network).

Subnet Address Extensions
Proxy ARP (1)



- Used to allow two physical networks to share the same IP network prefix
- Router R answers ARP requests on each network for hosts on the other network, giving its hardware address, and then routing datagram correctly.

Subnet Address Extensions
Proxy ARP (2)

- Issues:
  - Several IP addresses map to the same physical address. How to distinguish between a legitimate Proxy ARP router and spoofing?
- Advantages of Proxy ARP
  - Can be added to a single router without disturbing the routing table in other hosts or routers on that network.
- Disadvantages:
  - Does not work for networks that do not use ARP.
  - Does not generalize to complex network topologies (e.g. multiple routers interconnecting two physical networks).
  - Does not support a reasonable form of routing. (relies on network managers to maintain tables of machines and addresses manually.)

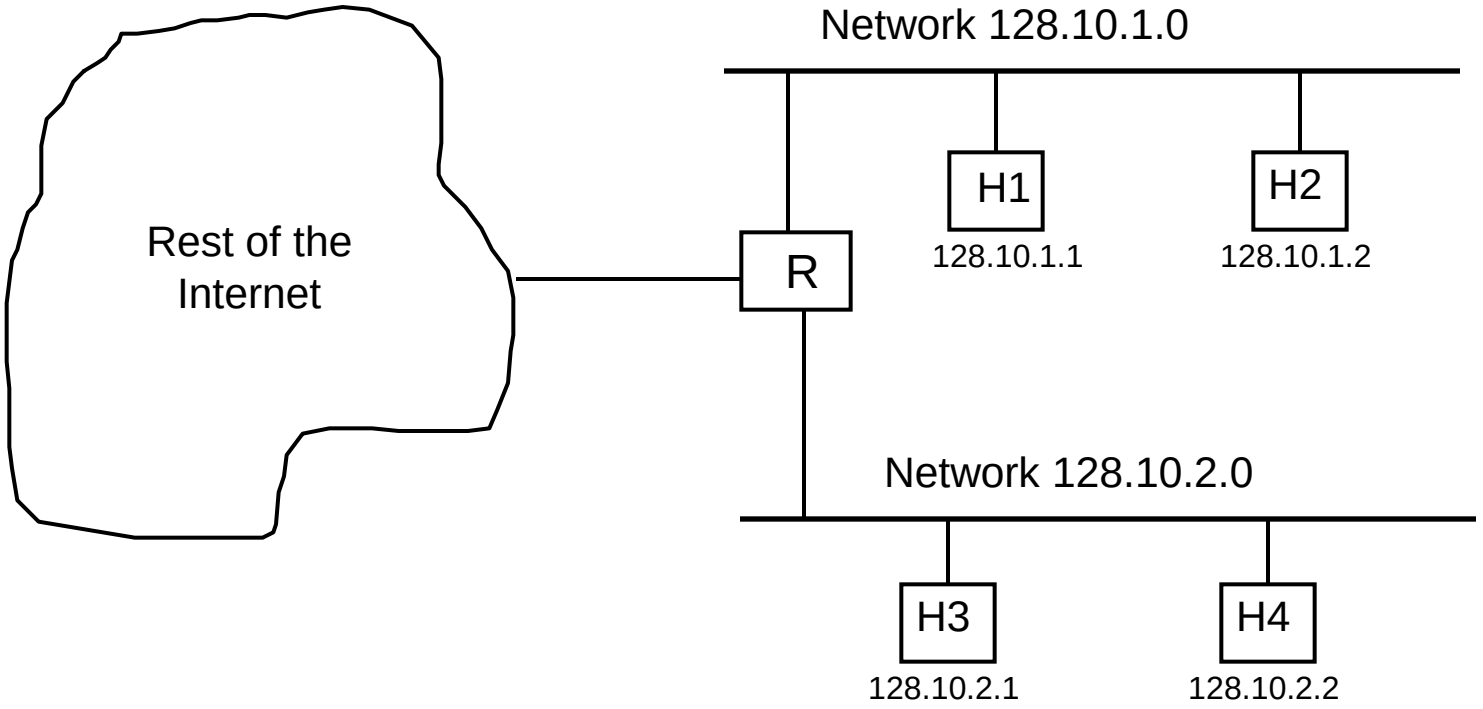
Subnet Address Extensions

Subnet Addressing or Subnetting (1)

- Most widely used. (most general and standardized).
- Considers the Host ID portion of the IP address to be further subdivided into two parts.
  - A physical network address.
  - A specific host on a physical network.

Subnet Address Extensions

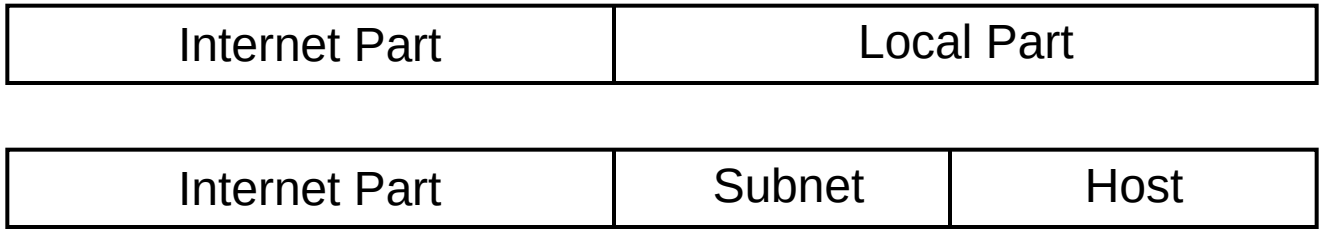
Subnet Addressing or Subnetting (2)



Subnet Address Extensions

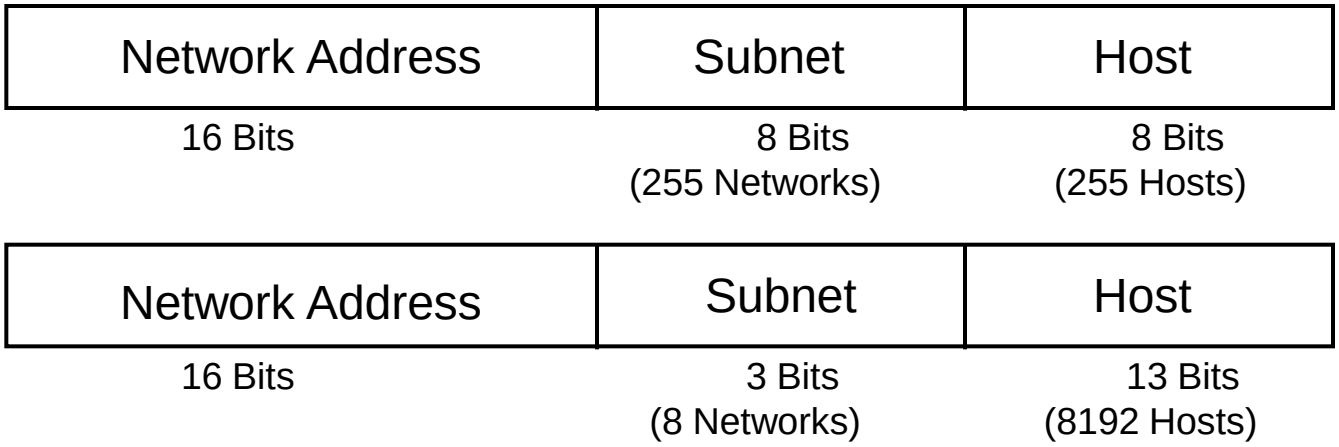
Subnet Addressing or Subnetting (3)

- More generally:
  - Think of a 32-bit IP address as having an Internet portion and a local portion:
  - The internet portion identifies a site (possibly with multiple physical networks)
  - The local portion identifies a physical network and host at that site



Subnet Address Extensions
Subnet Addressing or Subnetting (4)

- Example : class B address
  - Once a subnet partition has been selected, all machines on that network must honor it.



Implementation of Subnets with Masks

	Network Address		Subnet	Host
32 Bit Subnet Mask	111111111	11111111	11111111	00000000
Dotted Decimal Representation	255.	255.	255.	0

- 3-tuple Representation

  { < network number >, < subnet number> , < host number > }

  " all ones" is represented by -1

  Class B example, subnet mask 255.255.255.0 ⇔ {-1,-1,0}

Subnet Routing (1)

- Conventional routing table entry:
  - (network address, next hop address)
  - note : network address format known from address class
- With subnetting, entry becomes:
  - (subnet mask, network address, next hop address)
- Procedure :
  - Extract network address (including subnetwork number) using subnet mask
  - Then compare with network address field of entries to find next hop address.

Subnet Routing (2)

- The use of masks generalizes the subnet routing algorithm to handle all the special cases of the standard algorithm
  - routes to individual hosts
  - default route
  - routes to directly connected networks
  - routes to conventional networks (i.e., not using subnet addressing)
- Merely combine { arbitrary 32-bit mask field, arbitrary 32-bit address }

… in a network address field

- Individual host $\Leftrightarrow$ (masks of all 1's, Host's IP address)
- Default route $\Leftrightarrow$ (masks of all 0's, network address all 0's)
- Non subnet $\Leftrightarrow$ (masks of 2 octets 1's and 2 octets 0s ; 2 octets class B network network address)

Subnet Routing (3)

- Algorithm:
  - Extract destination address, ID, from datagram;
  - Compute IP address of destination network, IN;
  - If IN matches any directly connected network address
    - send datagram to destination over that network (Resolve ID to a physical address, form frame, send frame)
  - Else
    - for each entry in routing table do
      - N = bitwise - AND of ID and subnet mask
      - if N equals the network address field of entry ; then route datagram to the specified next hop address
  - If no matches were found, declare a routing error

Supernet Addressing (1)

- Use many IP network address for a single organization
- Example:
  - To conserve class B address, issue multiple class C addresses to the same organization
- Issue: increase number of entries in routing tables
- Solution :
  - "Classless Inter-Domain Routing" (CIDR - RFC1519) collapse a block of contiguous class C addresses into the pair:

(network address, count)

smallest number in the block

Supernet Addressing (2)

- CIDR requires each block of addresses to be a power of 2 and uses bit mask to identify the size of the block
- Example
  - Dotted Decimal 32-bit Binary Equivalent
  - lowest: 234.170.168.0 11101010 10101010 10101000 00000000
  - highest: 234.170.175.255 11101010 10101010 10101111 11111111
  - A block of 2048 addresses
  - 21 bit mask 11111111 11111111 11111000 00000000

Supernet Addressing (3)

- In the router :
  - the entry consists of
  - (the lowest address and 32-bit mask)
  - a block of addresses can be subdivided, and separate route can be entered for each subdivision
  - when looking up route, the routing software uses a longest-match paradigm to select a route

IPv6

- Motivation
  - limited address space (32 bits)
  - support for new applications (e.g, multimedia streams)
  - security
  - extensibility

Features of IPv6 at a Glance

- Larger addresses (128 bit addresses)
- Flexible Header format (set of optional headers)
- Support for flow identification (needed in resource allocation for multimedia streams)
- Provision for protocol extension