Internet

Communications, Networking & Computer Security

Sanjay Goel

University at Albany

Outline

- What is Internet?
- Internet Protocols
- Protocol hierarchies
- The OSI reference model
- Services in the OSI model

Internet

What is it?

- It is a network of networks
- Any network connected to the internet
  - Conform to certain naming conventions
  - Must run the IP protocol
  - IP protocol is also called Internet dial tone
- Internet has a hierarchical topology
  - End Systems connected to local ISPs through access networks
  - Access Network examples – LAN, telephone line with a modem, high speed cable networks
  - Local ISPs connected to regional ISPs, regional ISPs connected to national & international ISPs
  - Construction analogous with Lego construction

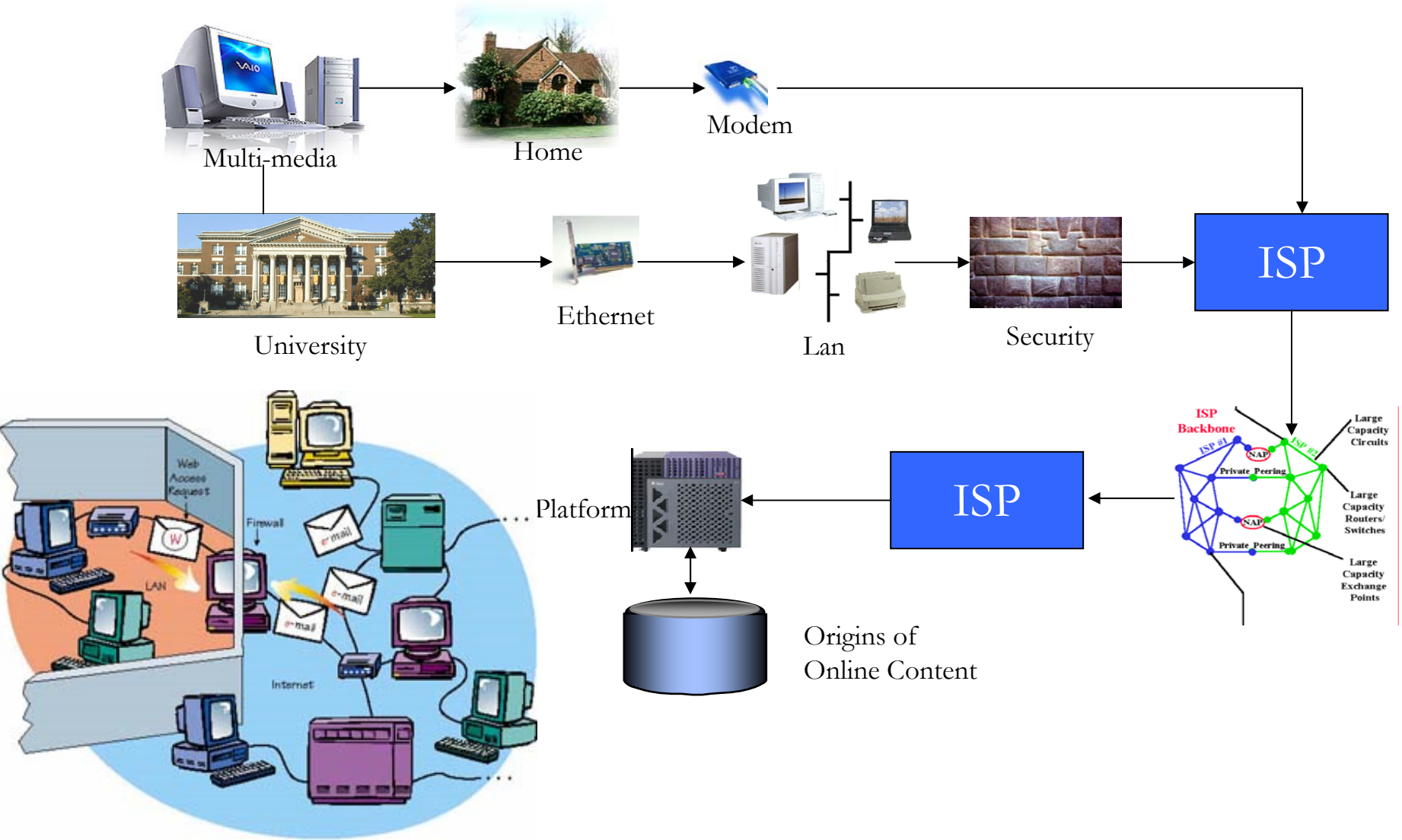
Internet

Role

- Allows distributed applications to exchange data with each other
- Applications include: FTP, Telnet, Mail, WWW, distributed games, video conferencing
- Provides two kinds of services
  - Connection Oriented Service (TCP): Establish connection prior to data exchange, coupled with reliable data transfer, flow control, congestion control etc.
  - Connectionless Service (UDP): No handshake prior to data exchange, No acknowledgement of data received, no flow/congestion control

Internet

Information Flow



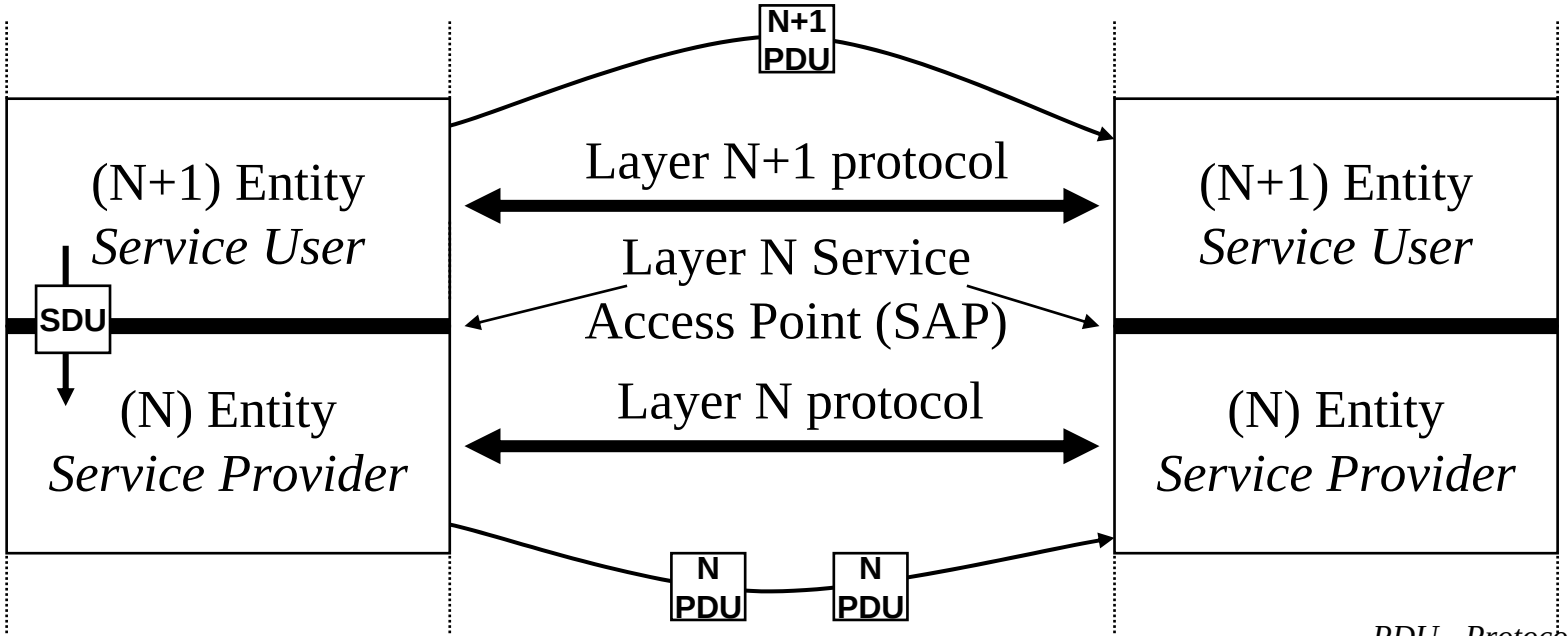
Internet

Protocol Hierarchies

- Internet is a very complex system
- Set of layers and protocols represents the *Network Architecture*.
- Protocols are stacked vertically as series of 'layers'.
- Each layer has a well defined interface.
  - Allows for easy replacement of layer
- Each layer offers *Services* to layer above, shielding implementation details.
- Each layer on one machine communicates with corresponding layer on another machine using Protocol for the Layer.

Internet

Layering Principle



PDU - Protocol Data Unit
SDU - Service Data Unit

- Service = set of primitives provided by one layer to layer above.
- Service defines what layer can do (but not how it does it).
- Protocol = set of rules governing data communication between peer entities, i.e. format and meaning of frames/packets.
- Service/protocol decoupling very important.

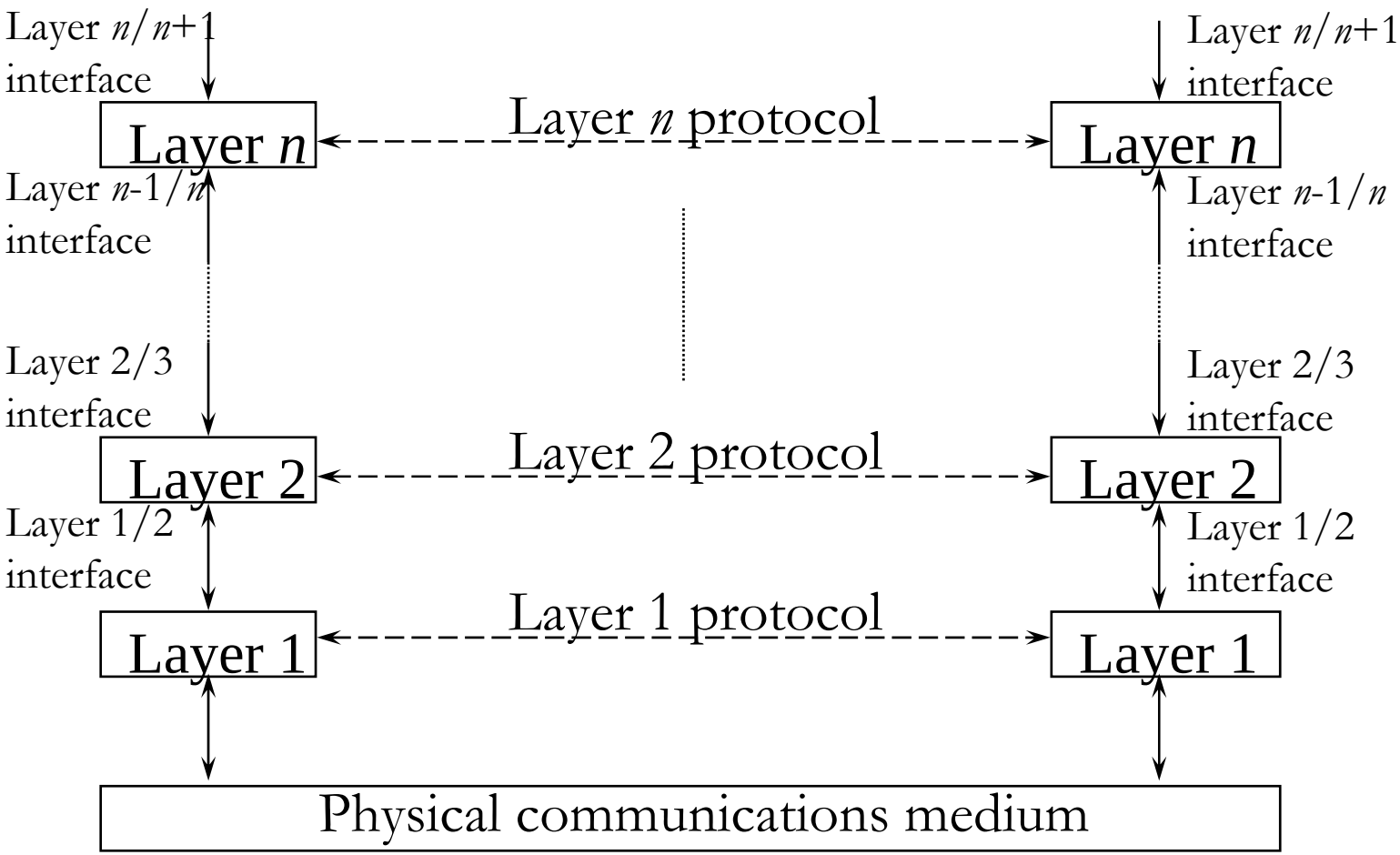
Internet

Connections & Reliability

- Connections
  - Layers can offer *connection-oriented* or *connectionless* services.
  - Connection-oriented like telephone system.
  - Connectionless like postal system.
  - Each service has an associated *Quality-of-service* (e.g. reliable or unreliable).
- Reliability
  - Reliable services never lose/corrupt data.
  - Reliable service costs more.
  - Typical application for reliable service is file transfer.
  - Typical application not needing reliable service is voice.
  - Not all applications need connections.

Internet

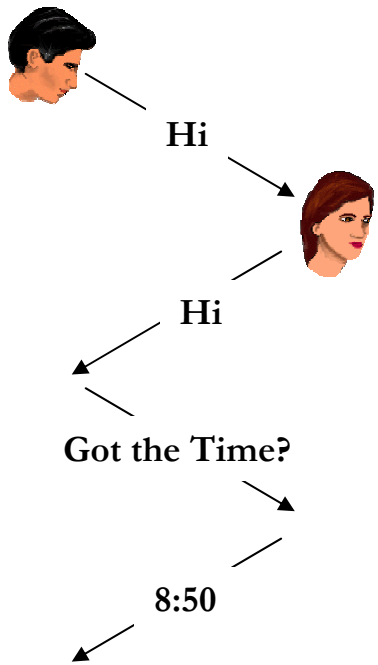
Layers, Protocols & Information Flow

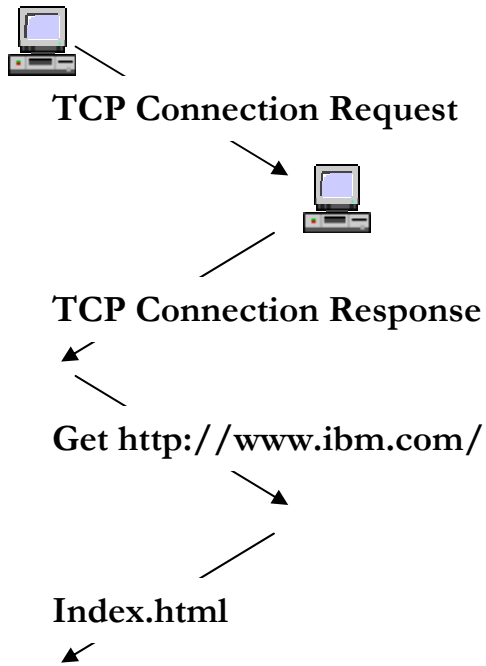


Protocol

Definition

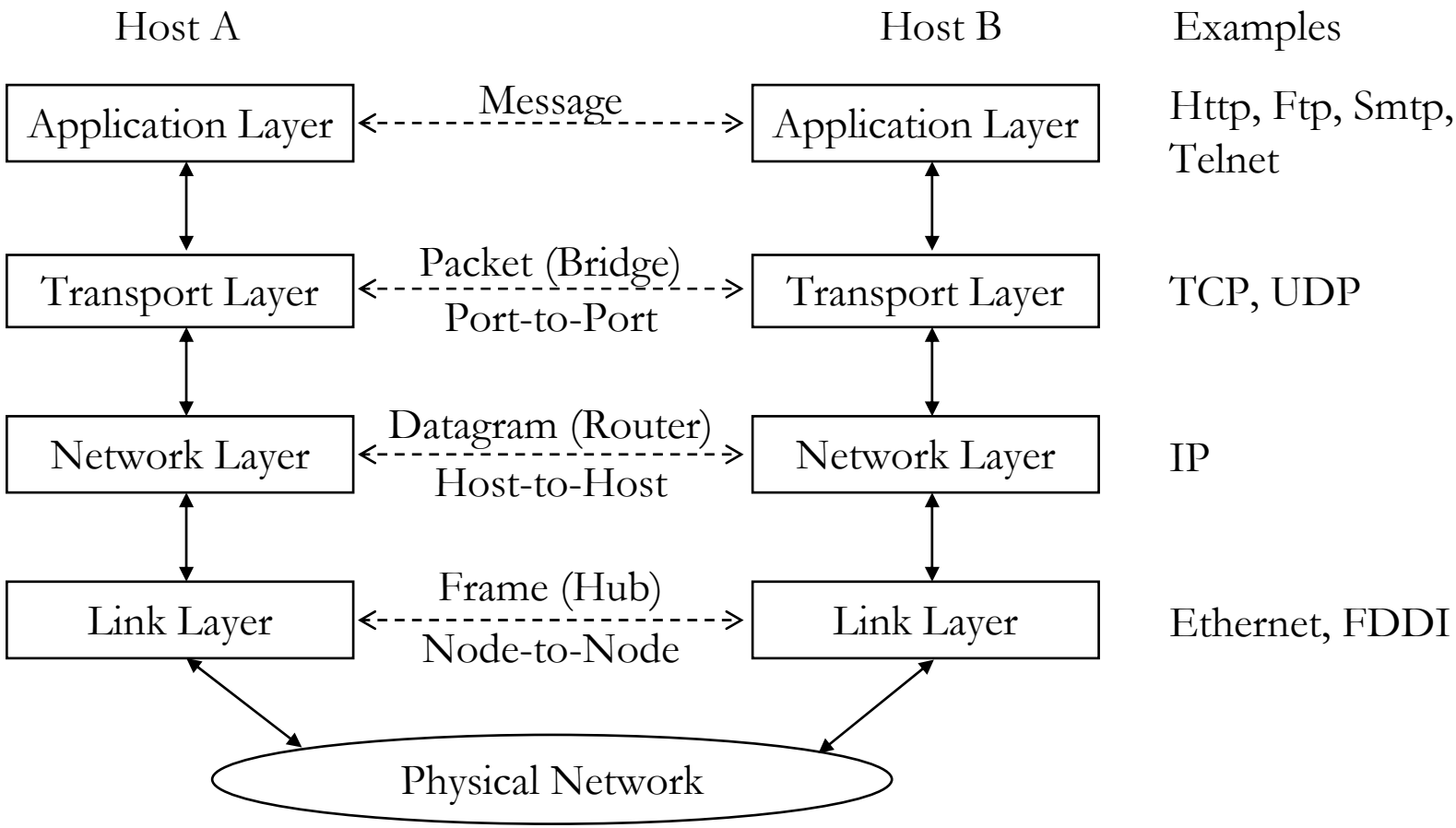
- A protocol defines the format and order of messages exchanged between two of more communicating entities as well as the actions taken on the transmission and/or receipt of a message or event.





Internet

Architecture



- Analogous to the mail system in context of layering & standardized protocols.

Application Layer

Function

- Implements application protocol
  - Users invoke applications using this protocol
- Application Layer Protocol defines
  - Types of messages exchanged e.g. request or response
  - Syntax of the various message types, such as, fields in the messages and how they are delineated
  - Semantics of the fields i.e. meaning of information in each field
  - Rules for determining when and how a process sends messages and responds to messages

Application Layer

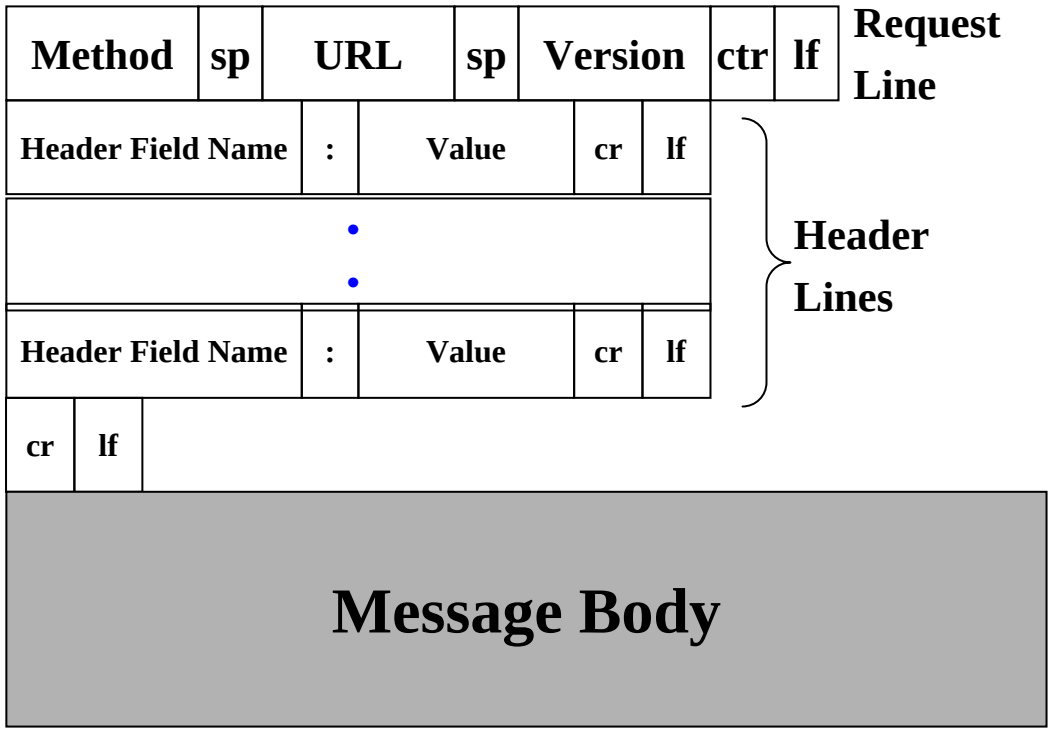
Function

- Different applications use different protocols
  - Web Servers/Browsers use HTTP
  - File Transfer Utilities use FTP
  - Electronic Mail applications use SMTP
  - Naming Servers use DNS
- Interacts with transport layer to send messages
  - Choose the transport layer protocol
  - Fix transport layer parameters, such as, buffer/segment sizes

Application Layer

Format

Http Request Message Format



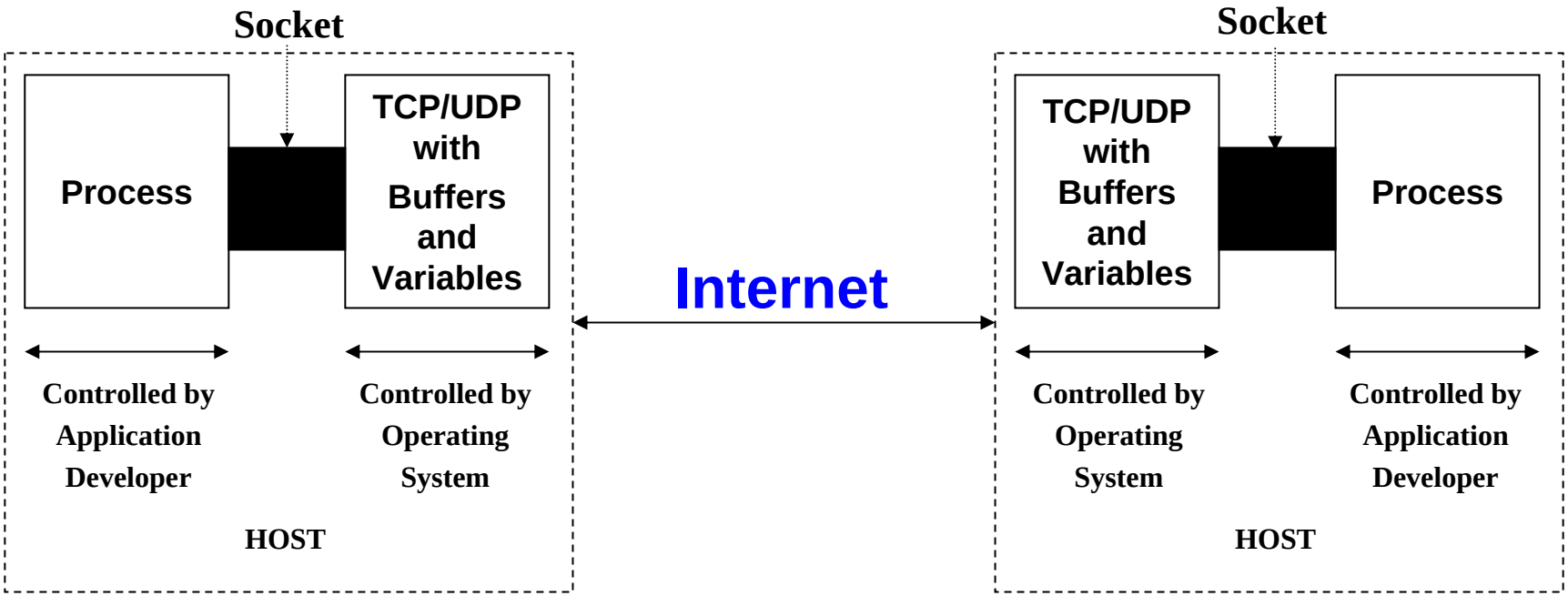
Http Request Message Example

Get /somedir/page.html HTTP/1.1
Connection: close
User-agent: Mozilla
Accept: text/html, image/gif, image/jpeg
Accept-language: fr
(extra carriage return, line feed)

- Types of messages
  - HTTP request, HTTP response, HTTP head

Application Layer

Interface



- Socket is the interface between the application layer and the transport layer
- Two parameter are required for identifying receiving process
  - Host machine identifier - IP Address
  - Host machine process identifier - Port

Transport Layer

Function

- Provides for logical communication between applications running on different hosts
  - Application multiplexing and de-multiplexing
- Implemented in the end systems but not in network routers
- On sending side
  - Divides stream of application message into smaller units (packets),
  - Adds the transport header to each chunk
  - Sends message to network layer
- On receiving side
  - Takes the header off the message packets
  - Reassembles the packets in order
  - Sends message to the application layer
- Two internet transport protocols available
  - TCP, UDP

Transport Layer

Protocol: TCP

- TCP (Transmission Control Protocol)
- Connection Oriented Service (requires handshake)
  - Duplex
  - Simplex
- Reliable Data Transfer
  - Guaranteed delivery of packets
- Congestion Control
  - Throttles process when network is congested
- No guarantee of a minimum transmission rate
- Suitable for reliability critical/ non time critical applications
  - FTP, SMTP, Telnet, HTTP

Transport Layer

Protocol: UDP

- Stands for User Datagram Protocol
- Lightweight transport protocol
- Connectionless (no handshake)
- Unreliable data transport service
  - No acknowledgements (lost packets not resent)
- Messages may arrive out of order
- No congestion control
  - Application can pump as many packets over the socket as it chooses
- Suitable for loss-tolerant time critical applications
  - Audio/Video streaming
  - Internet Telephony

Transport Layer

TCP Example

Source Port Number									Destination Port Number
Sequence Number									
Acknowledgement Number									
Header Length	Unused	URG	ACK	PSH	RST	SYN	FIN		Receiver Window Size
Source Port									Pointer to Urgent Data
Options									
Data									

32-bits

TCP header	HTTP Message
Src: 1081 Dst: 80 Chksum: 0xa858	GET /directory/dirsearch.html HTTP/1.1 Host: www.phoenix.co.uk

- Source / Destination Port Numbers
  - Multiplexing / Demultiplexing
- Sequence Number & Acknowledgement Number
  - Congestion Control
- Window size
  - Flow control
- Length Field
  - Length of TCP header in 32-bit words
- Unused field is currently unused
- Flag Field contains 6 bits
  - ACK: shows value in acknowledgement field is valid
  - RST, SYN, FIN bits used for connection setup and teardown
  - PSH bit indicates data should be passed to upper layer immediately
  - URG – indicates that there is data in the segment which is marked as urgent
- Ptr to urgent data

  Points to last byte of the urgent data
- Options field is used when sender and receiver negotiate the maximum segment size.

Network Layer

Function

- Provides communication service between two hosts
  - Transports packets from sending host to receiving host
- Encapsulates packets in IP datagram with IP header
- Three primary tasks
  - Path Determination: Determine the route taken by a packet as it flows from sender to receiver
  - Switching: Arriving packet is moved to the appropriate output link
  - Call Setup: Handshake prior to routing packets (required by some network architectures)
- If addressed to local machine, remove the IP datagram header and pass up to transport layer.

Network Layer

Protocols

- Network Layer contains several protocols including
  - Internet Protocol
  - Address Resolution Protocol (ARP)
  - Internet Control Message Protocol (ICMP)
  - Internet Group Message Protocol (IGMP)

Network Layer

Internet Protocol

- Internet Protocol
  - Determines the source and destination IP address of all packets
  - IP address is a unique address on a network assigned to a device
  - If the packet is meant for a device on the local host IP gets the MAC address for the device and sends it directly to the host
  - For a remote packet it first looks up the routing table for an explicit route to the network.
  - If an explicit route is not available it sends it to a default gateway

Network Layer

Internet Protocol: Example

Version	Header Length	Type of Service	Packet Length (bytes)	
16-bit identifier			Flags	13-bit Fragm-entation Offset
Time to Live	Upper Layer Protocol	Headerchecksum		
32-bit source IP Address				
32-bit destination IP Address				
Options (if any)				
Data				

32-bits

- Version: IP protocol version
- Header Length
- TOS
  - Allows different types of IP datagrams to be differentiated
- Datagram Length
  - Length of data + header
- Identifiers, Flags Fragmentation offset
  - Deal with fragmentation
- Time-to-live (TTL)
  - Decremented each time a router processes a datagram Datagram dropped when field is zero
- Protocol
  - Indicates transport level protocol
    e.g. 6 indicates TCP, 17 indicates UDP
- Checksum: Used for error checking
- Data – Contains the transport layer segment

IP datagram header	TCP header	HTTP Message
Src: 192.168.0.40 Dst: 192.168.0.50 TTL: 128	Src: 1081 Dst: 80 Chksum: 0xa858	GET /directory/dirsearch.html HTTP/1.1 Host: www.phoenix.co.uk

Network Layer

MAC vs. IP

- Physical (MAC) addresses identify the hardware and are configured by the manufacturer
- Logical (IP) addresses identify the node and are configured by the customer
  - IP addresses may be reused if a machine is replaced
  - IP addresses depend on the customers location
    - Network number
- Configuring client nodes is tedious and error-prone

Network Layer

Address Resolution Protocol

- Translates MAC address to IP addresses and vice-versa
- 2 types of ARP packets: replies and requests
- Using ARP for each packet causes a 2 packet overhead for each packet
- ARP thus caches the packets
  - Cache flushed at startup
  - Cache periodically cleaned up
  - Cache searched prior to sending the ARP request

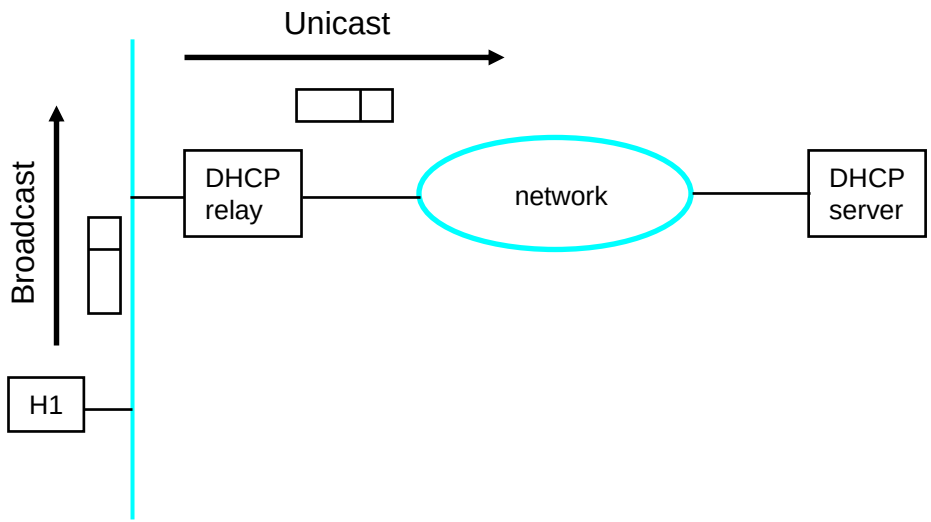
Hard Type	Prot Type	Hard Size	Prot Size	Opcode	Sender Hardware Address	Sender IP Address	Target Hardware Address	Target IP Address
2	2	1	1	2	6	4	6	4

bytes

Network Layer

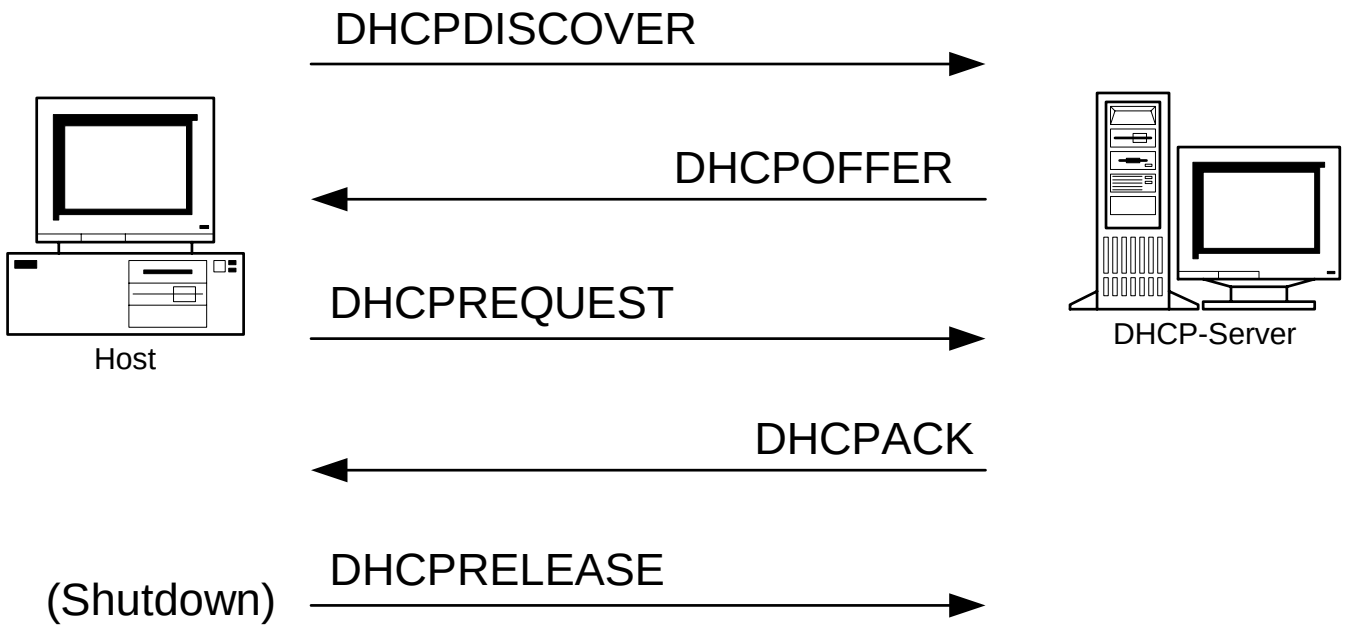
Dynamic Host Configuration Protocol

- DHCP server maintains configuration information about clients
  - IP addresses
  - default router
- Clients broadcast to locate server
- Server replies with configuration information
- Client IP addresses may be temporary (leased)



Network Layer

Address Assignment



Link Layer

Function

- Purpose is to defines the interface between device & network
  - It contains a driver that is compatible with the network that the device is connected to
- Transfers network layer datagrams over a link from node to node
  - A node is a router or a host
  - A link is the communication path along two nodes
- Link Layer protocol defines the format of the packets exchanged between the nodes
- The packets exchanged by link layer are called frames
  - Each frame typically encapsulates one datagram
  - e.g. Ethernet, token ring, FDDI, PPP

Link layer

Hardware Address

- Each frame contains the physical address of the hardware of the packet destination.
- This physical address is called the Media Access Control (MAC) address and is burned into the network interface card
  - This is a 12 character hexadecimal number (analogy social security #)
  - Example: 00:A0:C9:0F:92:A5
  - Here the first six digits are the manufacturers ID and the last six digits are the device ID
  - As each packet arrives at the network interface card the MAC address on the packet is compared to the MAC address of the device
  - If the addresses match the packet is accepted

Link Layer

Services

- Framing and Link Access
- Reliable delivery
- Flow Control
- Error Detection
- Error Correction
- Two types of services are provided
  - Half-Duplex and Full-Duplex
- Implemented in adapters commonly called Network Interface Cards (NIC)

Link Layer

Example

- Ethernet is the dominant protocol in the LAN market
  - Primary factors are cost and complexity
- Many topologies of Ethernet
  - Bus Topology
  - Star Topology
- Supports multiple medium
  - Coaxial Cable
  - Copper Wire
  - Fiber Optic
- Can transmit data at different rates
  - 10Mbps, 100Mbps, 1Gbps

Physical Layer

Function

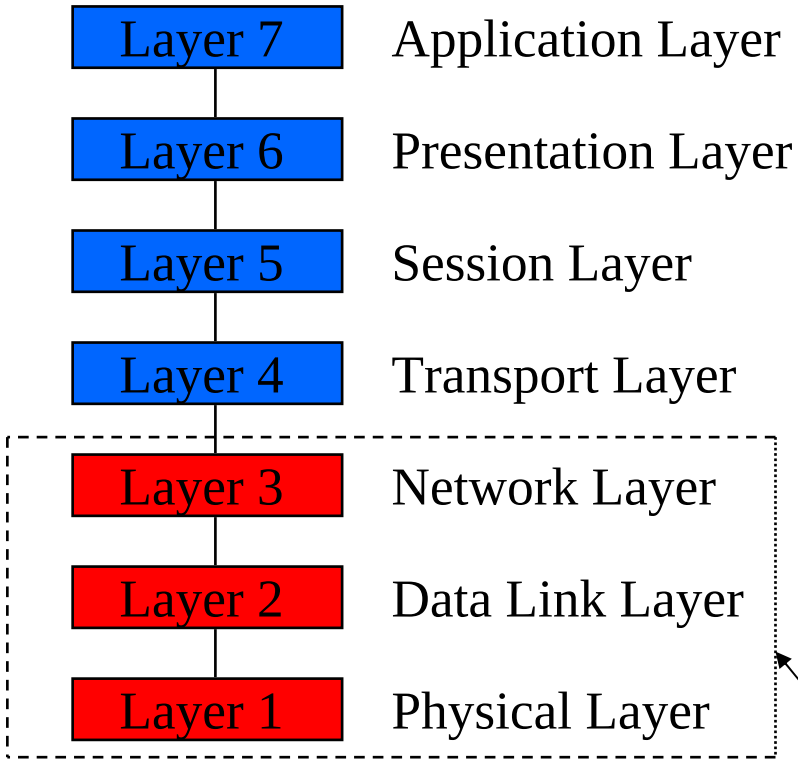
- Accepts IP datagrams and transmits over specific networks.
- Maybe a simple device driver (e.g. an Ethernet driver) or a complex subsystem with further data link protocols.

OSI Model

OSI Reference Model

- OSI Reference Model - internationally standardised network architecture.
- OSI = *Open Systems Interconnection*: deals with *open systems*, i.e. systems open for communications with other systems.
- Specified in ISO 7498.
- Model has 7 layers.

7-Layer OSI Model

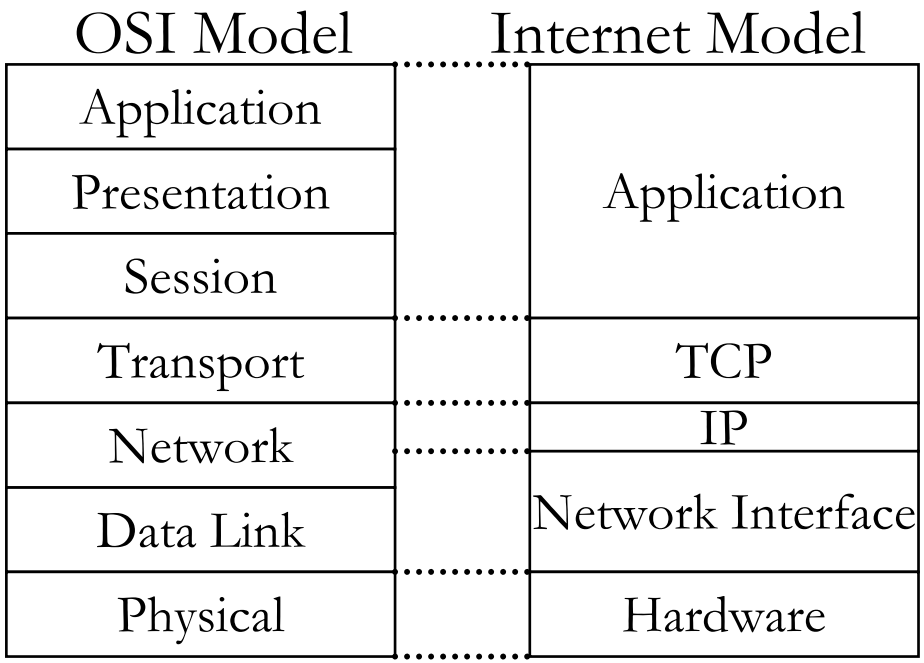


- Layers 1-4 relate to communications technology.
- Layers 5-7 relate to user applications.

Communications subnet boundary

Protocols

Comparison



- No Explicit Presentation and session layers in Internet Protocol
- Data Link and Network Layers redesigned

- In OSI model, each layer provide services to layer above, and 'consumes' services provided by layer below.
- Active elements in a layer called entities.
- Entities in same layer in different machines called peer entities.

Application

Function

- Level at which applications access network services.
  - Represents services that directly support software applications for file transfers, database access, and electronic mail etc.

Presentation

Function

- Related to representation of transmitted data
  - Translates different data representations from the Application layer into uniform standard format
- Providing services for secure efficient data transmission
  - e.g. data encryption, and data compression.

Session

Function

- Allows two applications on different computers to establish, use, and end a session.
  - e.g. file transfer, remote login
- Establishes dialog control
  - Regulates which side transmits, plus when and how long it transmits.
- Performs *token management* and *synchronization*.

Internet Addressing

Internet Address

Introduction

- Host identifiers are classified in three categories
  - Names: Identify what an object is
  - Addresses: Identify where object is
  - Routes: Identify how to get to the object
- Each host on a TCP/IP internet is assigned a unique 32-bit internet address that is used in all communications with that host.
  - Bits of IP addresses on the same host Provide unique address to each host
  - Written as four decimal integers separated by decimal points
  - Each integer gives the value of one octet of the IP address
- The 32-bit internet address
  - 10000000 00001010 00000010 00011110
  - 128.10.2.30

Internet Address

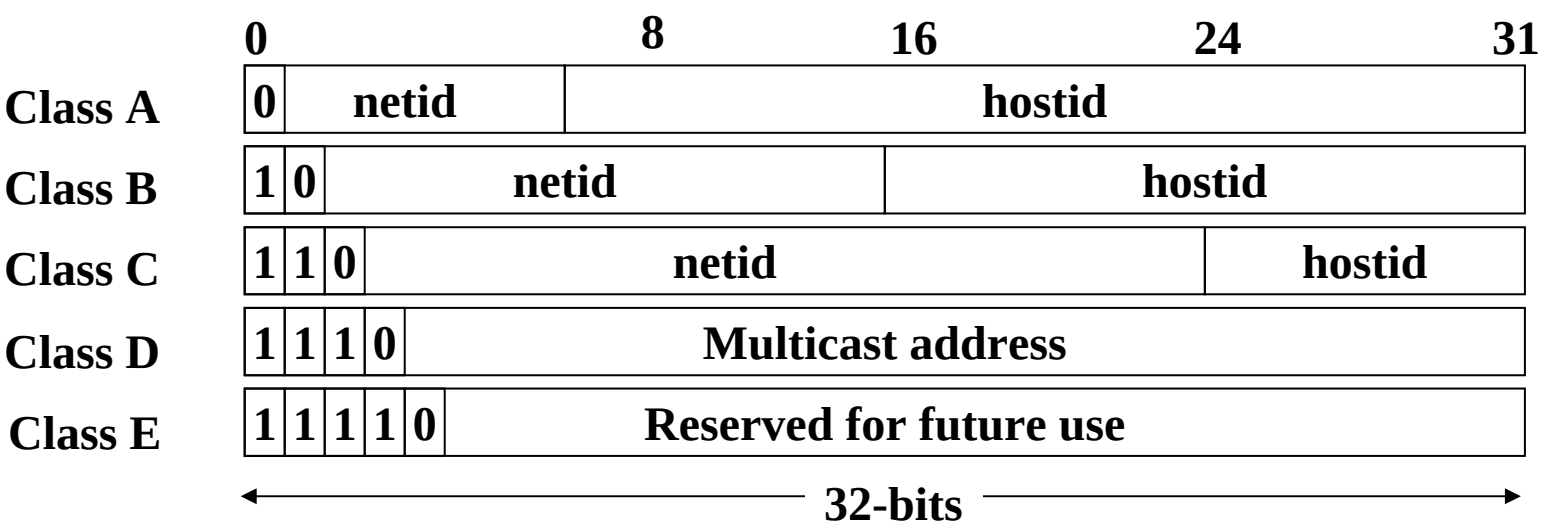
Introduction

- Conceptually each address is a pair (hostid, netid)
  - Netid identifies a network
  - Hostid identifies a host on that network
- Since IP addresses encode both a network and a host on that network, they do not specify individual computers, but a connection to a network
  - A router which connects to n networks will have n addresses
  - A multi-homed host will have multiple addresses

IP Addresses

Classes

- Class of address assigned depends network size
- Each IP address should be class A, B, or C
  - Class A used for more than 2^{16} hosts on network
  - Class B used when more than 2^{8} but less than 2^{16} hosts on network
  - Class C used for less than 2^{8} hosts on network



IP Addresses

Network & Broadcast Address

- Internet addresses can be used to refer to networks as well as individual hosts
  - An address with all bits of the hostid equal to 0 is reserved to refer to the network
- IP addresses can be used to specify a broadcast
  - Directed broadcasts are used to broadcast messages to target networks
  - A directed broadcast address has a valid netid hostid with all bits set to 1
  - Local network broadcast address is used for broadcast to local network independent of any host address
  - Local broadcast address consists of 32 1's

IP Addresses

Network & Broadcast Address

- A field consisting of zeros means this
  - IP address with all hostid fields 0 means this host
  - A netid of 0 means the current network
- 127.0.0.0 is the loop back address and used primarily for testing TCP/IP as well as for inter-process comm.
- Dotted Decimal Notation is used to represent IP addresses
  - IP addresses are written as four decimal integers separated by decimal points
  - Each integer gives the value of one octet of the IP address
  - 10000000 00001010 00000010 00011110 → 128.10.2.30

IP Addresses

Summary of Exceptions

all 0s		This host[1]
all 0s	host	Host on this net[1]
all l s		Limited broadcast (local net)[2]
net	all 1s	Directed broadcast for net[2]
127	Anything (often 1)	Loopback[3]

Footnotes:

[1] Allowed only at system startup and is never a valid destination address.

[2] Never a valid source address.

[3] Should never appear on a network.

IP Addresses

Issues

- Inadequate to respond to the fast growth of networks
  - Immense administrative overhead to manage network addresses
  - Routing tables in routers extremely large causing large overheads when routers exchange routing table information
  - Address space of networks will be eventually exhausted (Already short of class B addresses)
- Original scheme modified to allow sharing of network addresses
  - Transparent Routers
  - ARP
  - Standard IP Subnets

IP Addresses

Subnets

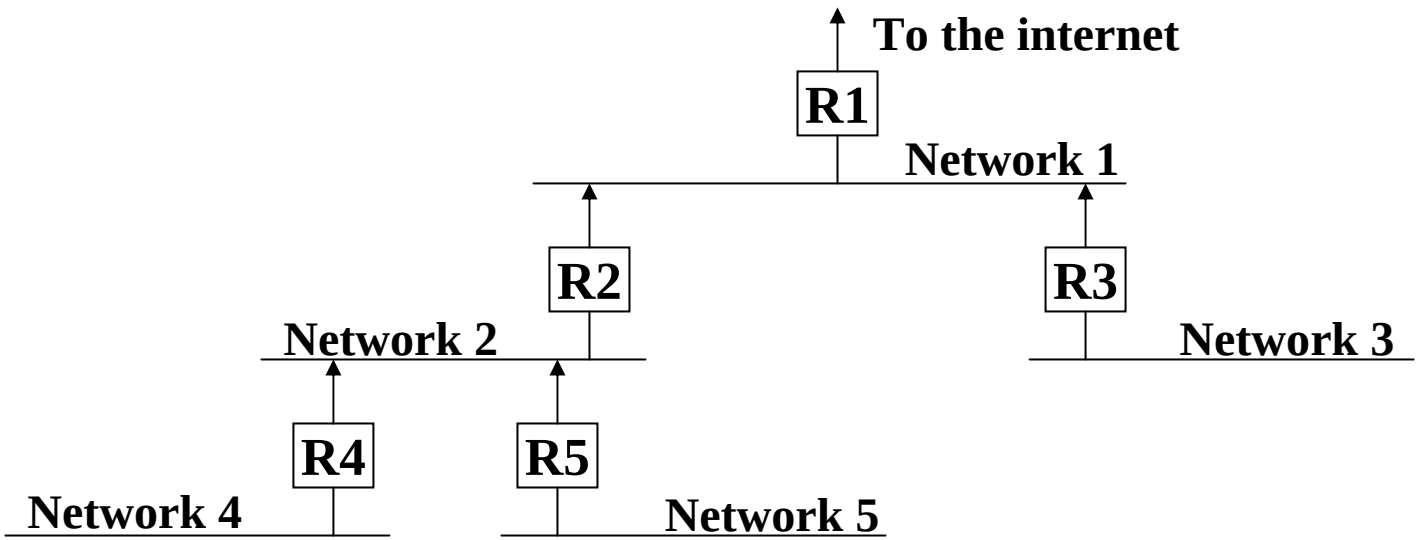
- Allows multiple networks to share the same network address
- The IP address is redefined such that
  - The network id is left intact
  - The host id portion is split into subnetwork id and hostid
- TCP/IP subnet standard permits subnet interpretation to be chosen independently of each physical network.
- Once a subnet partition is selected all the machines on the network must honour it.

Original Scheme	**1**	**0**	**netid**	**host id**	
Subnet Scheme	**1**	**0**	**Netid (Internet)**	**Netid (Local)**	**host id**

IP Addresses

Hierarchical Addressing

- Allows multiple networks to share the same network address



IP Addresses

Subnet Masks

- Sites that use subnet addressing must also choose a 32-bit subnet mask for each network.
  - Bits for network identifier are set to 1
  - Bits for the host identifier are set to 0
- For a class B address if the third octet is used for local netid
  - Subnet Mask → 11111111 11111111 11111111 00000000
- Dotted Decimal Notation is also popular for subnet masks
  - {<network number>, <subnet number>, <host number>}
  - e.g. 128.10.6.62

Mail

Mail

RFC 822

- Messages consist of a primitive envelope (described in RFC 821), some number of header fields, a blank line, and then the message body.
- Each header field (logically) consists of a single line of ASCII text containing the field name, a colon, and, for most fields a value.
- RFC822 was designed decades ago and does not clearly distinguish the envelope fields from the header fields.

Mail

RFC 822 Cont'd.

Header	Meaning
To:	E-mail address(es) of primary recipient(s)
Cc:	E-mail address(es) of secondary recipient(s)
Bcc:	E-mail address(es) for blind carbon copies
From:	Person or people who created the message
Sender:	E-mail address of the actual sender
Received:	Line added by each transfer agent along the route
Return-path:	Can be used to identify a path back to the sender

RFC 822 header fields related to message transport.

Mail

RFC 822 Cont'd.

Header	Meaning
Date:	The date and time the message was sent
Reply-To:	E-mail address to which replies should be sent
Message-Id:	Unique number used for referencing this message later
In-Reply To:	Message-Id of to which this is a reply
References:	Other relevant message-Ids
Keywords:	User-chosen keywords
Subject:	Short summary of the message for the one-line display

Some fields used in the RFC 822 message header.

Mail

Multipurpose Internet Mail Extensions (MIME)

- Allows multilingual ability for mail
- Messages in languages with accents
  - (e.g., French and German)
- Messages in non-Latin alphabets
  - (e.g., Hebrew and Russian)
- Messages in languages without alphabets
  - (e.g., Chinese and Japanese)
- Messages not containing text at all
  - (e.g., audio or images)

Mail

MIME Header

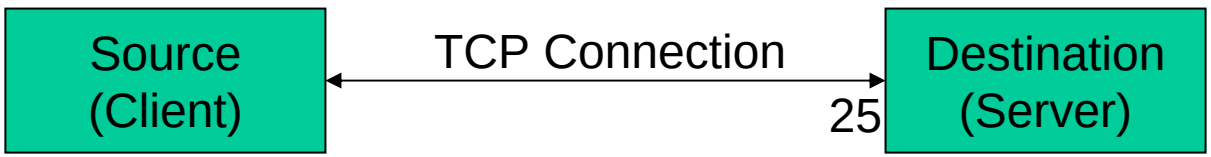
Header	Meaning
MIME-Version:	Identifies the MIME version
Content-Description:	Human- readable string telling what is in the message
Content-Id:	Unique identifier
Content-Transfer-Encoding	How the body is wrapped for transmission
Content-Type:	Type and format of the content

RFC 822 headers added by MIME.

Mail

SMTP

- **Simple Mail Transfer Protocol**
  - SMTP is a simple ASCII protocol for transfer of email from source to destination
  - To deliver mail the source machine establishes a TCP connection to port 25 of the destination machine.
  - Listening to this port (25) is an e-mail daemon that speaks SMTP.
  - After establishing the TCP connection to port 25, the sending machine, operating as the client, waits for the receiving machine, operating as the server, to talk first.



Mail

SMTP

- The server starts by sending a line of text giving its identity and telling whether it is prepared to receive mail.
- If it is not, the client releases the connection and tries again later.
- Otherwise the client starts sending the messages

Mail

Post Office Protocol (Version 3) : POP3

- Allows messages to be delivered even when the receiver is not online
- This protocol allows mail stored on the server to be downloaded to the client
- POP3 is invoked when the user starts the mail reader
  - The mail reader calls up the ISP and establishes a TCP connection with the message transfer agent at port 110.
  - Once the connection has been established, the POP3 protocol goes through three states in sequence:
    - Authorization
    - Transaction
    - Update

Mail

Post Office Protocol (Version 3) : POP3

- POP3 protocol supports the ability to download a specific message or set of messages and leave them on the server
  - most e-mail programs however just download everything and empty the mailbox.
- This behavior means that in practice, the only copy is on the user's hard disk.
  - If that crashes, all e-mail may lost permanently.

Mail

Internet Message Access Protocol (IMAP)

- IMAP assumes that all the e-mail will remain on the server indefinitely in multiple mailboxes.
  - Unlike POP3, which assumes that user will empty the mailbox on every contact and work off-line after that
- IMAP provides extensive mechanisms for reading messages or even parts if messages
  - This feature is useful when a slow modem is used to read the text part of multipart message with large audio and video attachments.

Mail

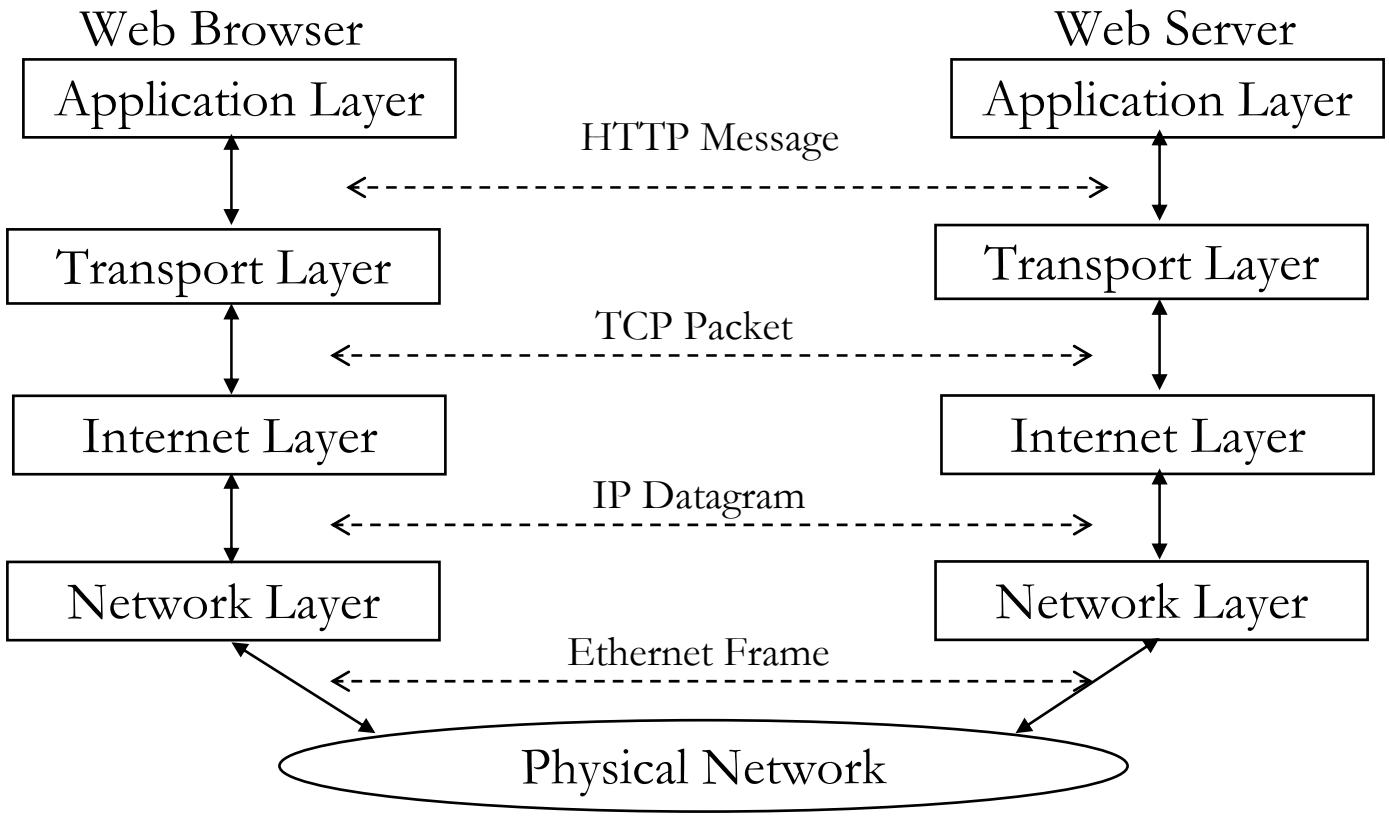
Internet Message Access Protocol (IMAP)

- IMAP provides mechanisms for creating, destroying, and manipulating multiple mailboxes on the server.
- Unlike POP3, IMAP can also accept outgoing e-mail for shipment to the destination as well as deliver incoming e-mail.

Routing

World Wide Web

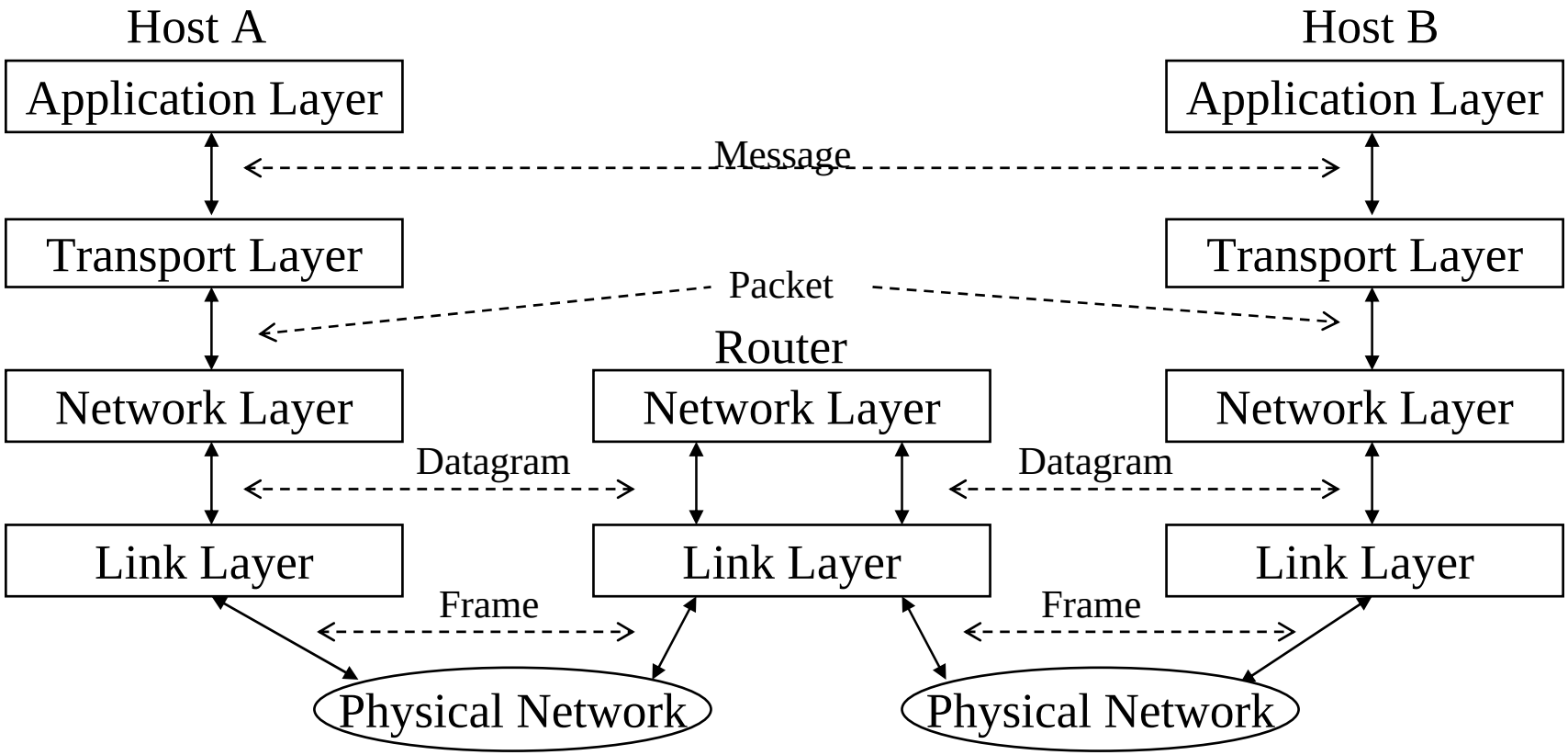
Protocols



- The data may be routed via numerous nodes called routers

Routing

Protocols

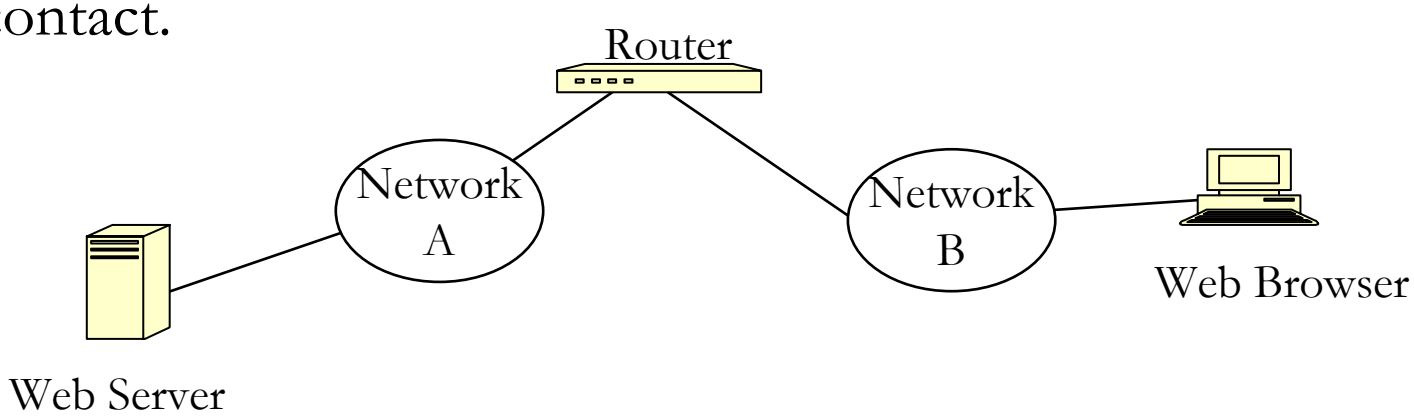


- The data may be routed via numerous nodes called routers

Routing

Protocols

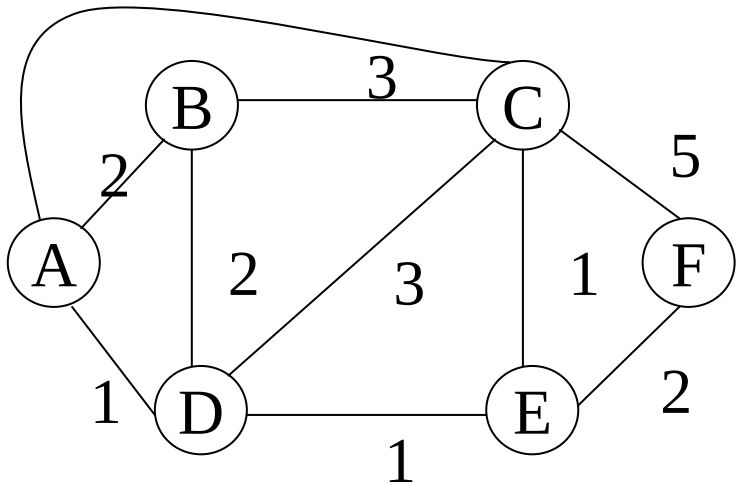
- In TCP/IP any machine on the same network can be contacted directly, but machines on another network must be contacted through a router or gateway.
- Router is a specific device (software or hardware) that forwards a transmission from a local network to other networks.
- Since the router is another device on the network, it needs to have its own internal IP address that the computers can contact.



Routing

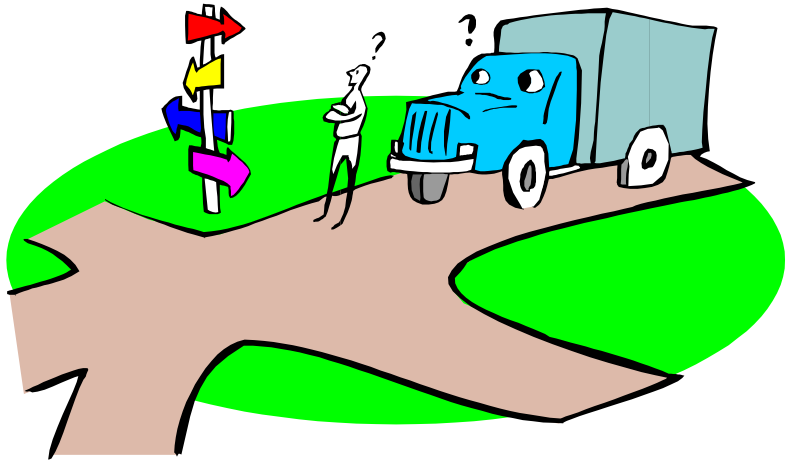
Protocols

- Objective: Determining optimum path through a sequence of routers that packets should take in going from one host to destination
- Graph abstraction for routing algorithms:
  - Graph nodes are routers
  - Graph edges are physical links
  - link cost: delay, $ cost, or congestion level



Routing

Algorithms



- Logic for deciding the path
- Two kinds of routing algorithms
  - Link State Algorithm: Global routing algorithm that uses knowledge of the entire network while making selection
  - Distance Vector Algorithm: Decentralized algorithm compues least cost path in iterative distributed manner
- The routing algorithms can also be classified as
  - Static: Routes change slowly over time (usually via manual intervention)
  - Dynamic: Routing paths change as network traffic loads or network topology changes.

Routing

Dijkstra's Algorithm

- Net topology, link costs known to all nodes
  - accomplished via "link state broadcast"
  - all nodes have same info
- Computes least cost paths from one node ('source") to all other nodes
  - gives routing table for that node
- iterative: after k iterations, know least cost path to k dest.'s
- Notation:
  - c(i,j): link cost from node i to j. cost infinite if not direct neighbors
  - D(v): current value of cost of path from source to dest. V
  - p(v): predecessor node along path from source to v, that is next v
  - N: set of nodes whose least cost path definitively known

Routing

Dijkstra's Algorithm - Steps

Initialization:

```
N = {A}
for all nodes v {
  if v adjacent to A
  then D(v) = c(A,v)
  else D(v) = infty
}
```

Loop (until all nodes in N) {

```
find w not in N such that D(w) is a minimum
 add w to N
 update D(v) for all v adjacent to w and not in N:
 D(v) = min( D(v), D(w) + c(w,v) )
/* new cost to v is either old cost to v or known shortest path cost to w
    plus cost from w to v */
```

}

- Algorithm complexity (n nodes)
  - each iteration: need to check all nodes, w, not in N
  - n*(n+1)/2 comparisons: O(n**2)
  - Efficient implementations O(nlogn) possible

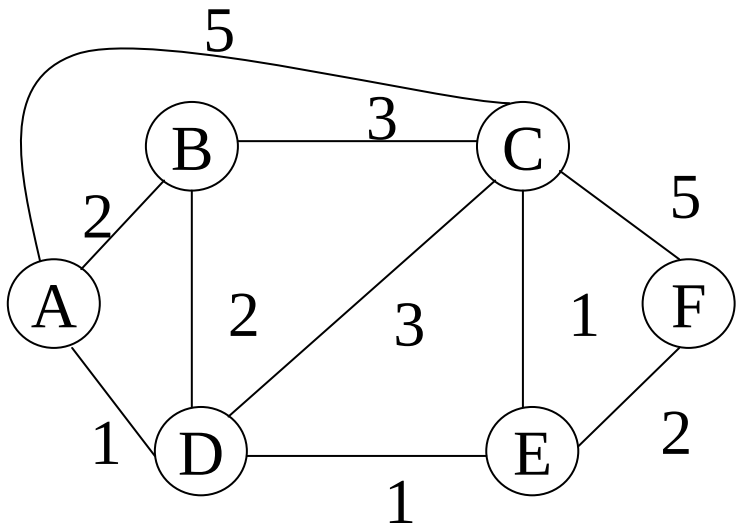
Routing

Dijkstra's Algorithm - Example

Step	start N	D(B),p(B)	D(C),p(C)	D(D),p(D)	D(E),p(E)	D(F),p(F)
0	A	2,A	5,A	1,A	infinity	infinity
1	AD	2,A	4,D		2,D	infinity
2	ADE	2,A	3,E			4,E
3	ADEB		3,E			4,E
4	ADEBC					4,E
5	ADEBCF					

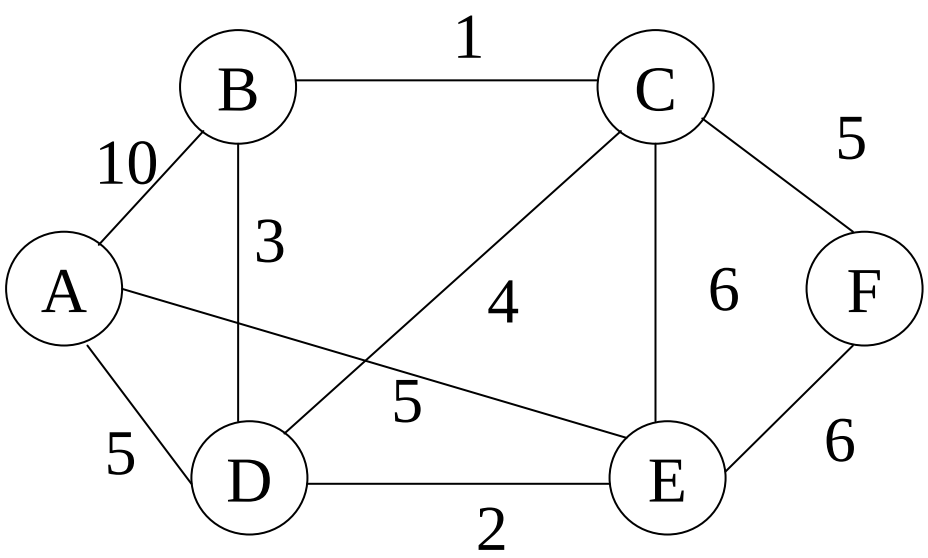
Compute Distance Vector for node A

Node	Distance
A	-
B	2
C	3
D	1
E	2
F	4



Routing

Dijkstra's Algorithm - Problem



Routing

Distance Vector Algorithm

- Each router starts with a distance table consisting of the value "0" for itself and the value "infinity" for every other destination
- Each router will transmit its distance vector to each of its neighbors whenever the information changes (as well as when a link to a neighbor first comes up)
- Each router saves the most recently received distance vector from each of its neighbors, and calculate its own distance vector, based on minimizing the cost to each destination

Routing

Distance Vector Algorithm (Kurose)

iterative:

- continues until no nodes exchange info.
- *self-terminating*: no "signal" to stop

asynchronous:

- nodes need *not* exchange info/iterate in lock step!

distributed:

- each node communicates *only* with directly-attached neighbors

Distance Table data structure

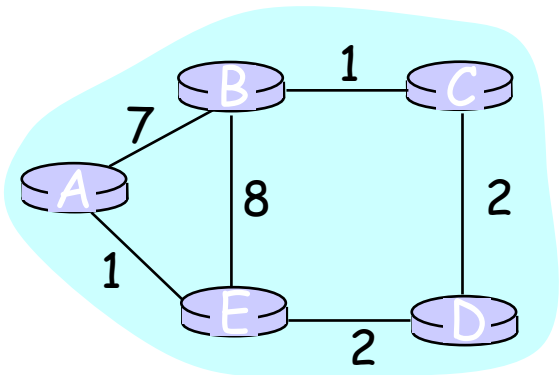
- each node has its own
- row for each possible destination
- column for each directly-attached neighbor to node
- example: in node X, for dest. Y via neighbor Z:

$$D^X(Y,Z) = \text{distance } \textit{from } X \textit{ to } Y, \textit{ via } Z \text{ as next hop}$$

$$= c(X,Z) + \min_w \{D^Z(Y,w)\}$$

Routing

Distance Table (Example)



$$D^E(C,D) = c(E,D) + \min_w\{D^D(C,w)\}$$
$$= 2+2 = 4$$

$$D^E(A,D) = c(E,D) + \min_w\{D^D(A,w)\}$$
$$= 2+3 = 5$$

loop!

$$D^E(A,B) = c(E,B) + \min_w\{D^B(A,w)\}$$
$$= 8+6 = 14$$

loop!

cost to destination via

$D^E()$	A	B	D
A	1	14	5
B	7	8	5
C	6	9	4
D	4	11	2

destination

Routing

Distance Table Gives Routing Table

cost to destination via

$D^E()$ (destination)	A	B	D
A	1	14	5
B	7	8	5
C	6	9	4
D	4	11	2

Outgoing link to use, cost

destination	
A	A,1
B	D,5
C	D,4
D	D,4

Distance table → Routing table

Distance Routing

Overview

Iterative, asynchronous:

each local iteration caused by:

- local link cost change
- message from neighbor: its least cost path change from neighbor

Distributed:

- each node notifies neighbors *only* when its least cost path to any destination changes
  - neighbors then notify their neighbors if necessary

Each node:

wait for (change in local link cost of msg from neighbor)

recompute distance table

if least cost path to any dest has changed, *notify* neighbors

Distance Routing

Distance Vector Algorithm

At all nodes, X:

```
Initialization:
 for all adjacent nodes v:
    D^X(*,v) = infty        /* the * operator means "for all rows" */
    D^X(v,v) = c(X,v)
 for all destinations, y
    send min_w D^X(y,w) to each neighbor  /* w over all X's neighbors */
```

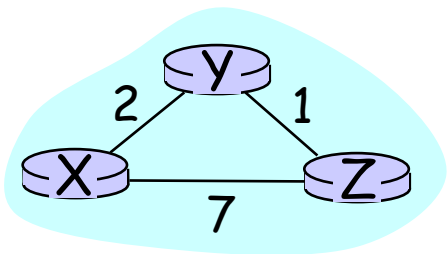
Distance Routing

Distance Vector Algorithm Cont'

```
loop
  wait (until I see a link cost change to neighbor V
        or until I receive update from neighbor V)

  if (c(X,V) changes by d)
    /* change cost to all dest's via neighbor v by d */
    /* note: d could be positive or negative */
    for all destinations y:  D^X(y,V) =  D^X(y,V) + d

  else if (update received from V wrt destination Y)
    /* shortest path from V to some Y has changed  */
    /* V has sent a new value for its  min_w DV(Y,w) */
    /* call this received new value is "newval"     */
    for the single destination y: D^X(Y,V) = c(X,V) + newval

  if we have a new min_w D^X(Y,w)for any destination Y
     send new value of min_w D^X(Y,w) to all neighbors
forever
```

Distance Routing

Distance Vector Algorithm Example:



X	cost via Y	cost via Z
dest Y	2	∞
dest Z	∞	7

Y	cost via X	cost via Z
dest X	2	∞
dest Z	∞	1

Z	cost via X	cost via Y
dest X	7	∞
dest Y	∞	1

X	cost via Y	cost via Z
dest Y	2	8
dest Z	3	7

Y	cost via X	cost via Z
dest X	2	8
dest Z	9	1

Z	cost via X	cost via Y
dest X	7	3
dest Y	9	1

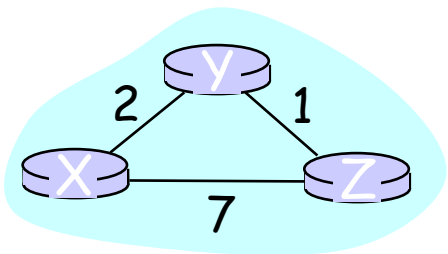
X	cost via Y	cost via Z
dest Y		
dest Z		

Y	cost via X	cost via Z
dest X		
dest Z		

Z	cost via X	cost via Y
dest X		
dest Y		

Distance Routing

Distance Vector Algorithm Example:



D^X

dest	cost via Y	cost via Z
Y	2	∞
Z	∞	7

D^Y

dest	cost via X	cost via Z
X	2	∞
Z	∞	1

D^Z

dest	cost via X	cost via Y
X	7	∞
Y	∞	1

D^X

dest	cost via Y	cost via Z
Y	2	8
Z	3	7

$$D^X(Y,Z) = c(X,Z) + \min_w\{D^Z(Y,w)\}$$
$$= 7+1 = 8$$

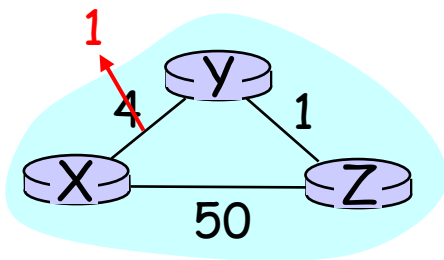
$$D^X(Z,Y) = c(X,Y) + \min_w\{D^Y(Z,w)\}$$
$$= 2+1 = 3$$

Distance Routing

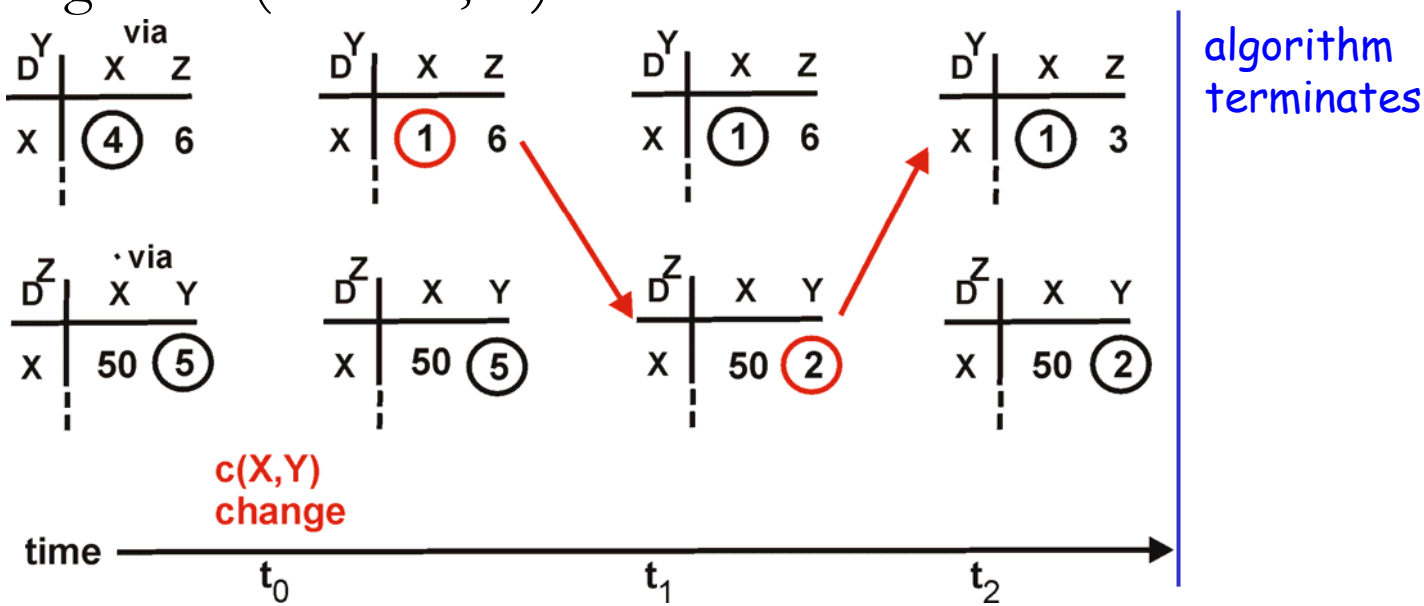
Distance Vector Algorithm: Link Cost Changes

Link cost changes:

- node detects local link cost change
- updates distance table (line 15)
- if cost change in least cost path, notify neighbors (lines 23,24)



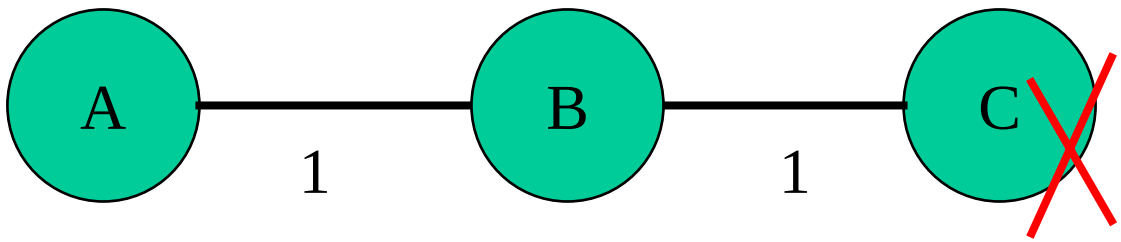
"good news travels fast"



Distance Routing

Distance Vector Algorithm: Example

Count to infinity

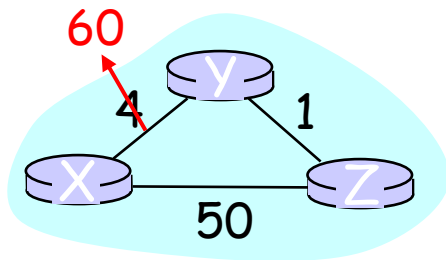


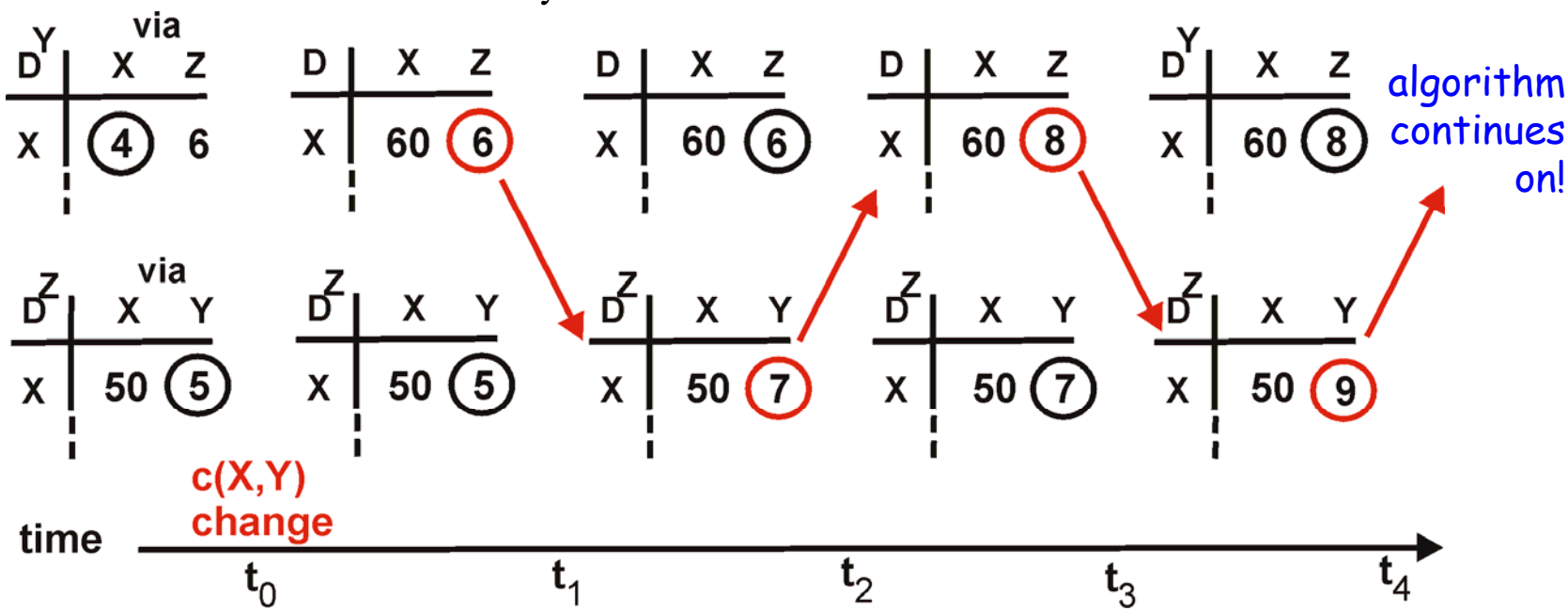
Distance Routing

Distance Vector Algorithm: Link Cost Changes

Link cost changes:

- good news travels fast
- bad news travels slow - "count to infinity"



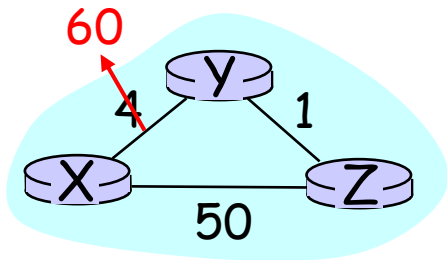


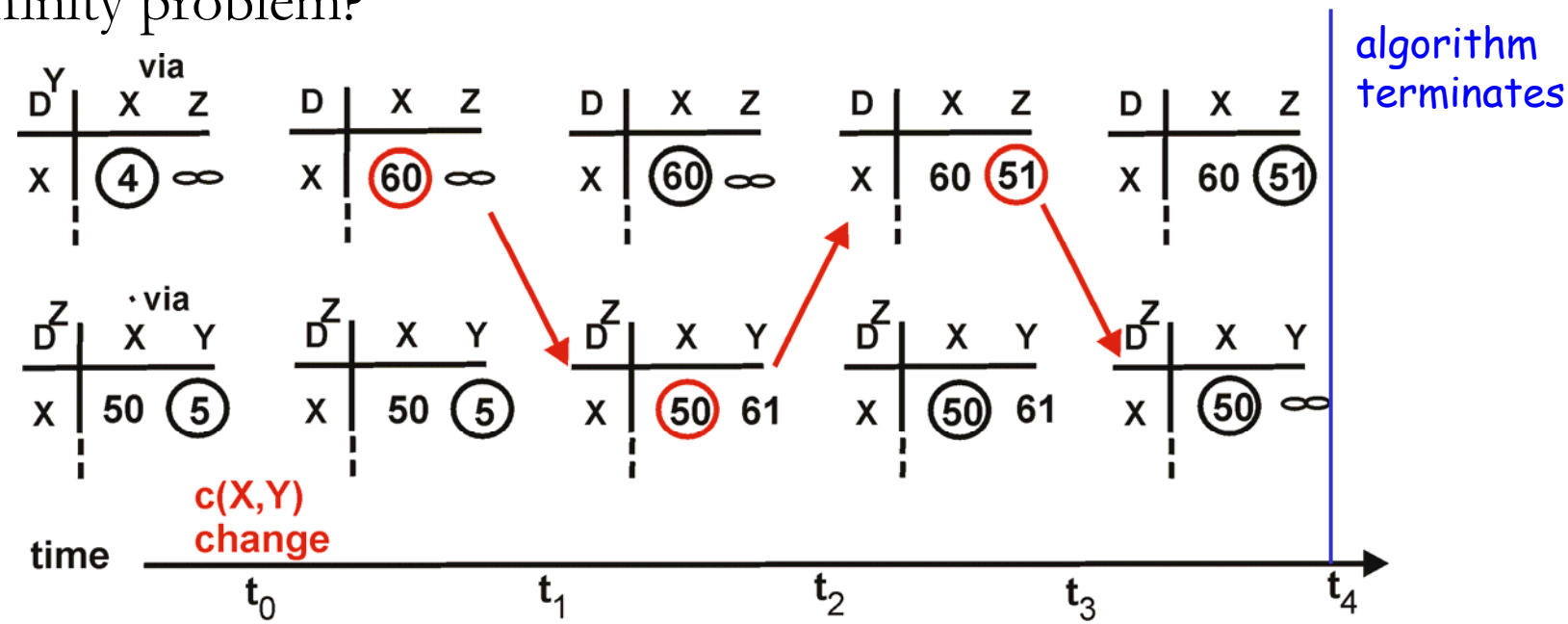
Distance Routing

Distance Vector Algorithm: Position Reverse

If Z routes through Y to get to X :

- Z tells Y its (Z's) distance to X is infinite (so Y won't route to X via Z)
- will this completely solve count to infinity problem?





Distance Routing

Comparison of LS & DV

Message complexity

- LS: with n nodes, E links, O(nE) msgs sent each
- DV: exchange between neighbors only
  - convergence time varies

Speed of Convergence

- LS: O(n**2) algorithm requires O(nE) msgs
  - may have oscillations
- DV: convergence time varies
  - may be routing loops
  - count-to-infinity problem

Robustness: what happens if router malfunctions?

LS:

- node can advertise incorrect *link* cost
- each node computes only its *own* table

DV:

- DV node can advertise incorrect *path* cost
- each node's table used by others
  - error propagate thru network

Routing Algorithms

Summary

- LS and DV are representative
- There are other type of routing algorithms, especially in circuit switching world, e.g., hot potato algorithm
- Most of the internet routing protocols (think OSPF, BGP etc.) are based on these fundamental algorithms we introduced just now

Physical Hardware - Routers

Routers

Devices

- In TCP/IP any machine on the same network can be contacted directly, but machines on another network must be contacted through a router or gateway.
- Router is a specific device (software or hardware) that forwards a transmission from a local network to other networks.
- Since the router is another device on the network, it needs to have its own internal IP address that the computers can contact.

Routers

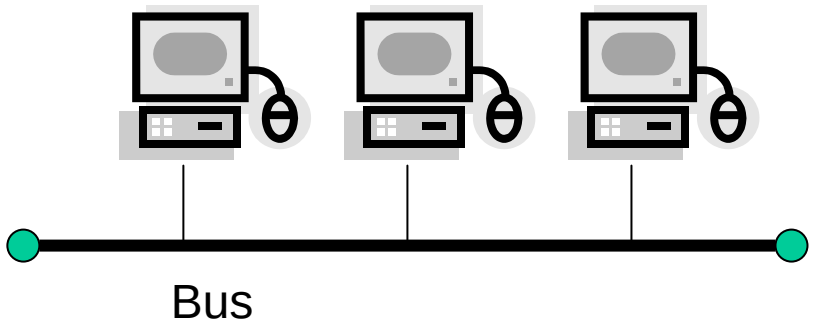
Ethernet

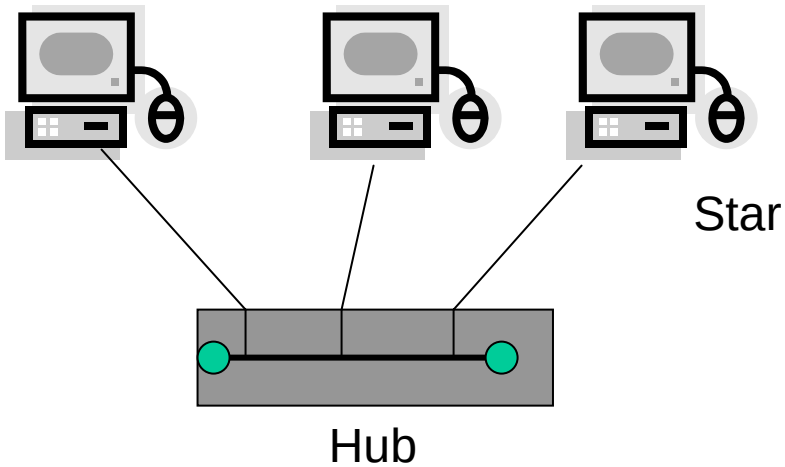
- There can only be one device transmitting on a segment at any given time. If two or more devices attempt to transmit at the same time, a collision occurs.
- After a collision, all devices must retransmit. As you can imagine, as the number of devices on an Ethernet segment increases, the probability for collisions increase.
- Because devices must spend more time re-transmitting data, the network is perceived to be slow.

Routers

LAN Topology

- **Logical topology** is how the network works conceptually
- **Physical topology** is how the network is physically installed
- Logical topology is not necessarily the same as the physical topology





Routing

Devices

- Switch is a generic term for a device that switches data (packets or frames)
- Hub is link layer switch (node to node)
  - Operates on ethernet frames
  - L2 switch
  - Uses physical addresses (MAC addresses)
- Bridge connects two LANS or two segments of the same LAN
  - Protocol Independent
  - Operates on ethernet frames
  - L2 switch
  - Uses physical addresses (mac addresses)

Routing

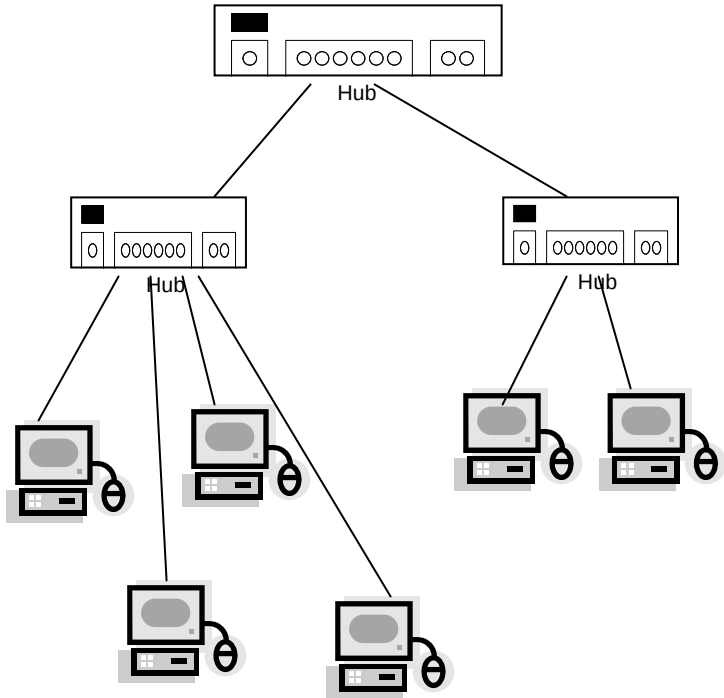
Devices

- Router is a network layer switch (host to host)
  - Also called L3 switch
  - Uses network addresses
  - Operates on packets
- Gateway is a generic term for an internetworking system
  - Can be implemented in software and/or hardware
  - Can operate at any level of the OSI model from application protocols to low-level signaling.

Routers

HUB

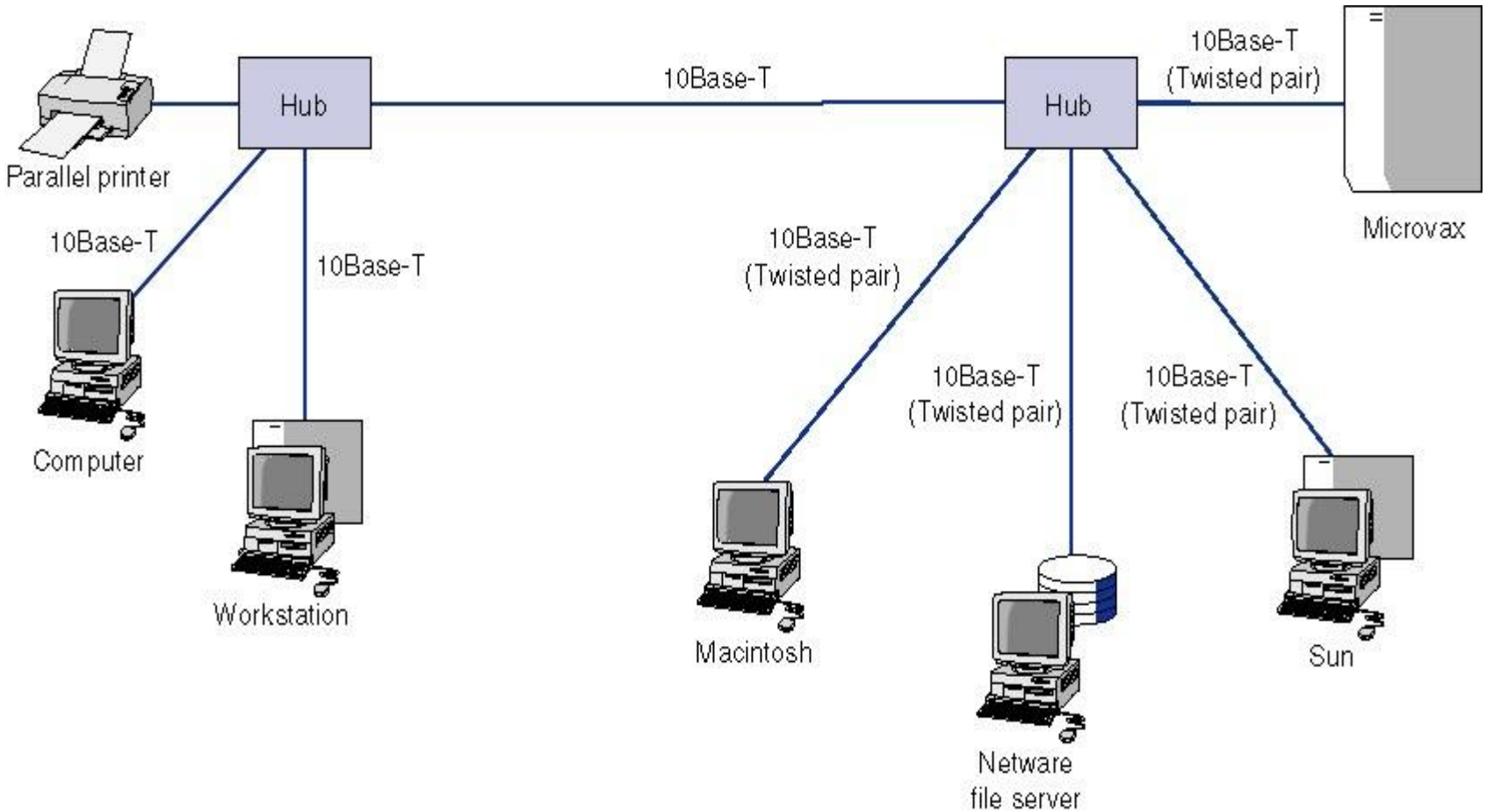
- A device that connects several computer on Ethernet
- A hub has 4/8/16/24 RJ-45 ports.
  - Signals are *broadcast* all the cables connected to all other ports.
- Hubs do no processing on network traffic
  - they simply repeat the incoming signal to all available ports.
- A hub is an alternative to the bus topology
  - make network connection easier
  - Hubs can act as repeaters or amplifiers



- Hubs can be used hierarchically

Routers

Multi-HUB Ethernet



Routers

Repeater

- Copies bits from one network to another
- Does not look at any bits
- Allows the extension of a network beyond physical length limitations

Routers

Bridges

- Network can be divided into segments with a bridge
- Have two Ethernet ports
- Bridge learns which devices are on each side by MAC address
- It makes decision to forward or not to forward each packet to the other side
  - Check the destination MAC address in frame
- Extends the network beyond physical length limitations.
- Improve network efficiency

Routers

Switch

- Connects several computers in a network by a number of RJ-45 ports
  - Same as Hubs
- Every port works as a Bridge
  - A switch has table of (MAC, port) pairs
- Each device can act independently from other devices

Routers

Switch Cont'd.

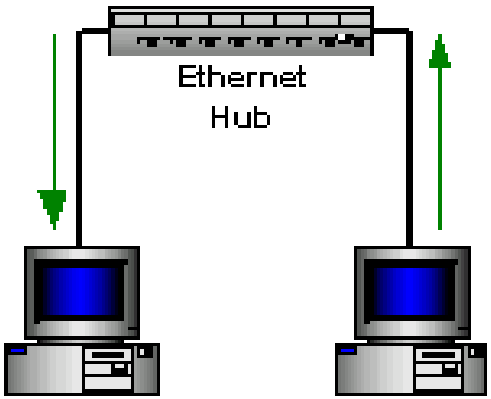
- Consider a switch with the following devices attached:
  - computer 1
  - computer 2
  - computer 3
  - printer
  - file server
  - uplink to the Internet
  - In this case:
    - computer 1 could be printing a document
    - computer 2 connects to a files server
    - computer 3 accesses the Internet.
    - Because the switch intelligently forwards traffic only to the devices involved, there can be multiple independent simultaneous conversations.

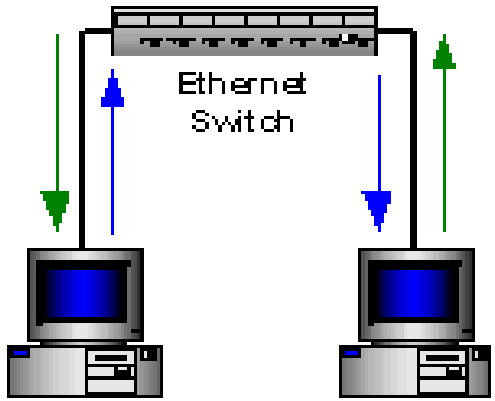
Routers

Hub vs. Switch

Bandwidth Limitations

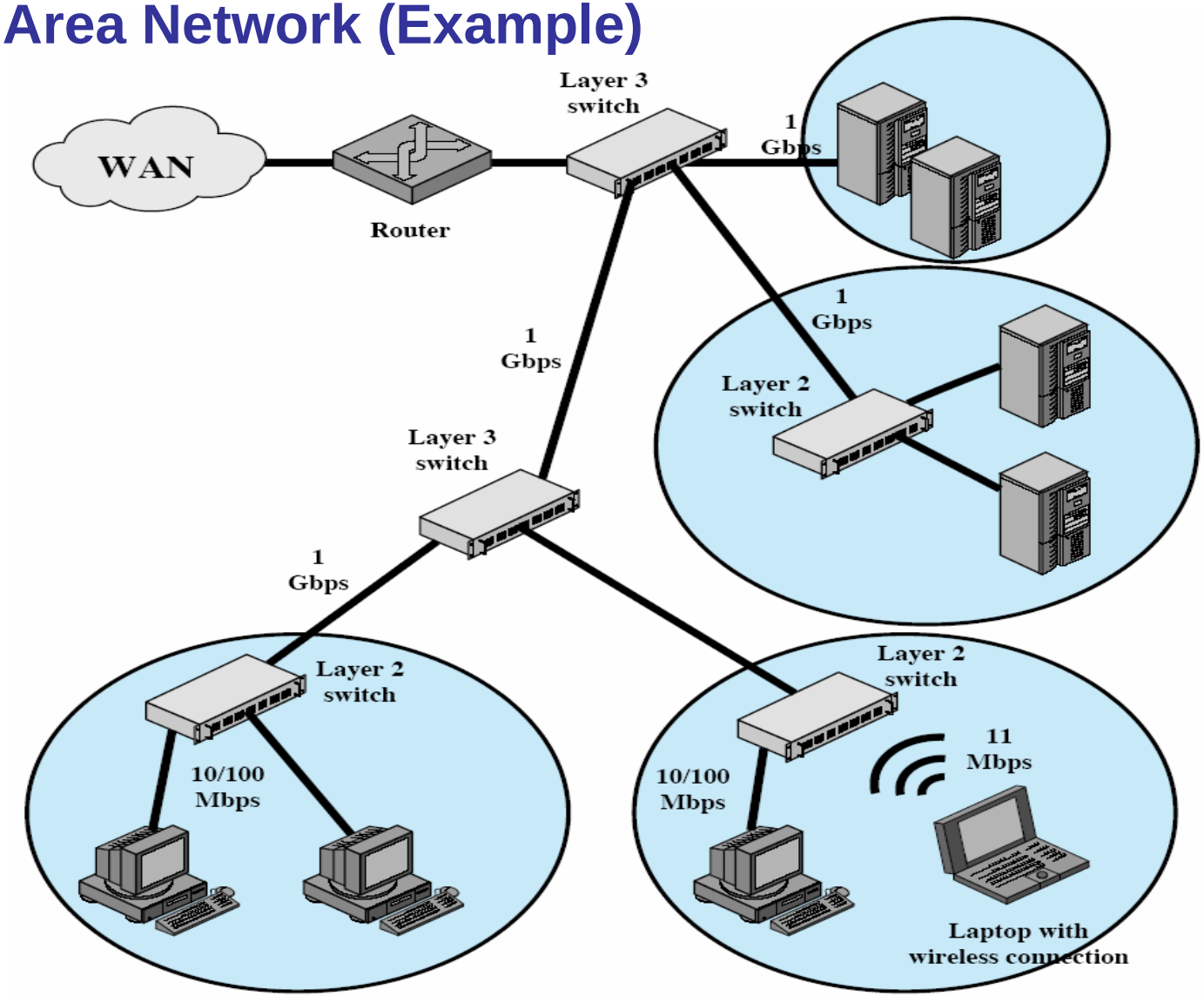
- Total network bandwidth is limited to the speed of the hub, i.e. a 10Base-T hub provides 10Mb bandwidth max, no matter how many ports it has.
- Total network bandwidth is determined by the number of ports on the switch. i.e. an 8 port 100Mb switch can support up to 800Mb/s bandwidth.





Routers

Local Area Network (Example)



Routers

Layer 3 Switch

- Layer 3 switching refers to a class of high-performance routers optimized for the campus LAN or intranet.
- Difference with router:
  - software (router) vs. hardware (switch)
- Layer 3 switch: basically an optimization of Internet class routers for the campus intranet

Sniffing

Sniffing

Definition

- Network sniffing is used to eavesdrop the network to capture the packets transmitted over the network.
- Components of a Sniffer
  - The hardware: adapter with promiscuous mode capability
  - Driver: capture the packets and store them in the buffer.
  - Packet filter: filter the packets according to user rules.
  - Packet analyzer: analyses the packets, and generate human readable reports.
  - Examples:
    - TcpDump, WinDump, Ethereal

Sniffing

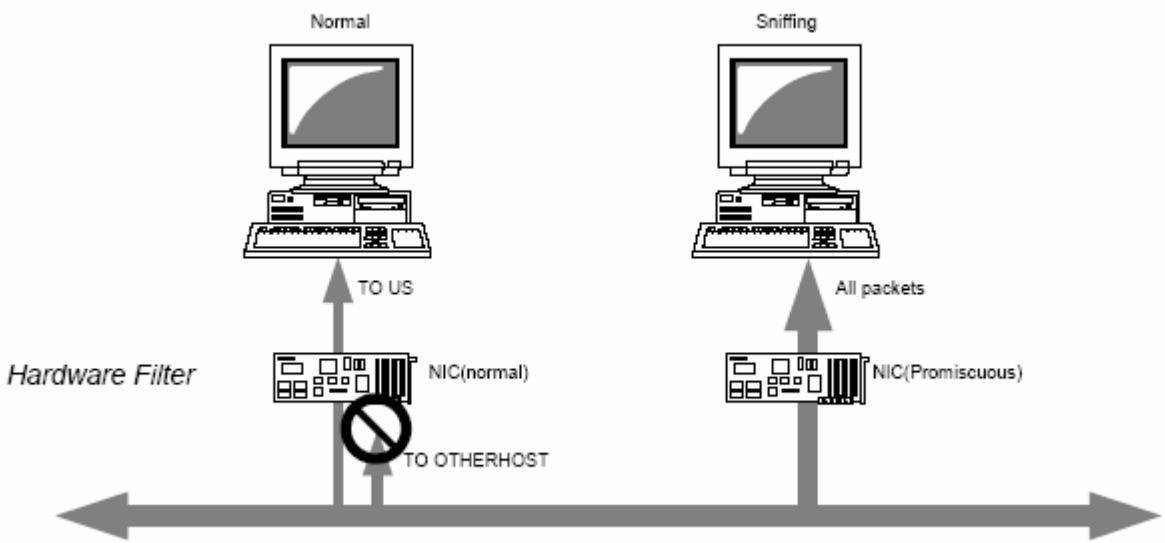
How

- Frames are transmitted on Ethernet
  - Broadcast Frames
    - Examples?
    - All computers read the frame
  - Non-broadcast frames
    - Examples?
    - Only the target computer reads the frame
    - Can the frame be read by other computers?
      - Hub?
      - Switch?

Sniffing

Promiscuous Mode

- Machines can be set in promiscuous mode
  - This allows them to intercept all the packets coming into it
  - Introduced in the specification to allow debugging & testing of networks



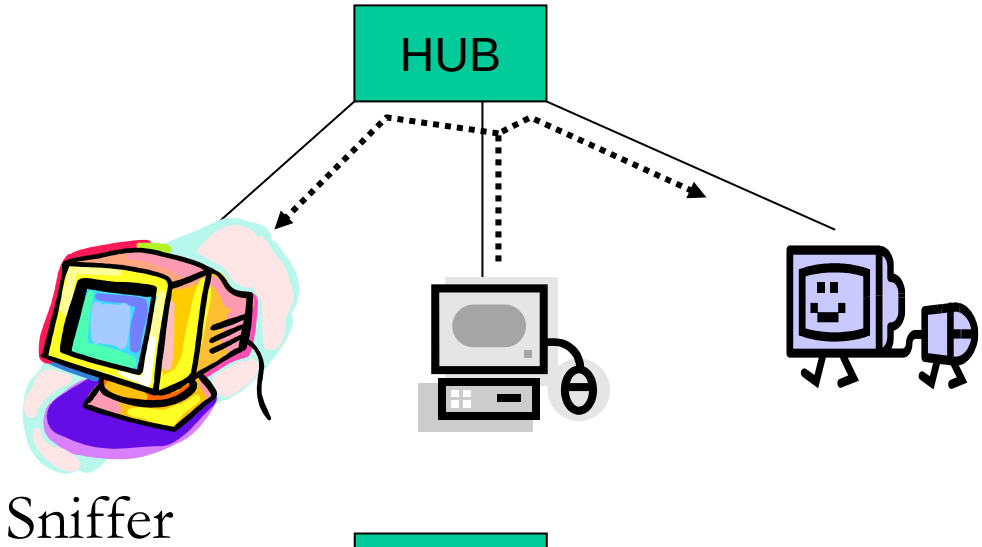
Sniffing

Promiscuous Mode

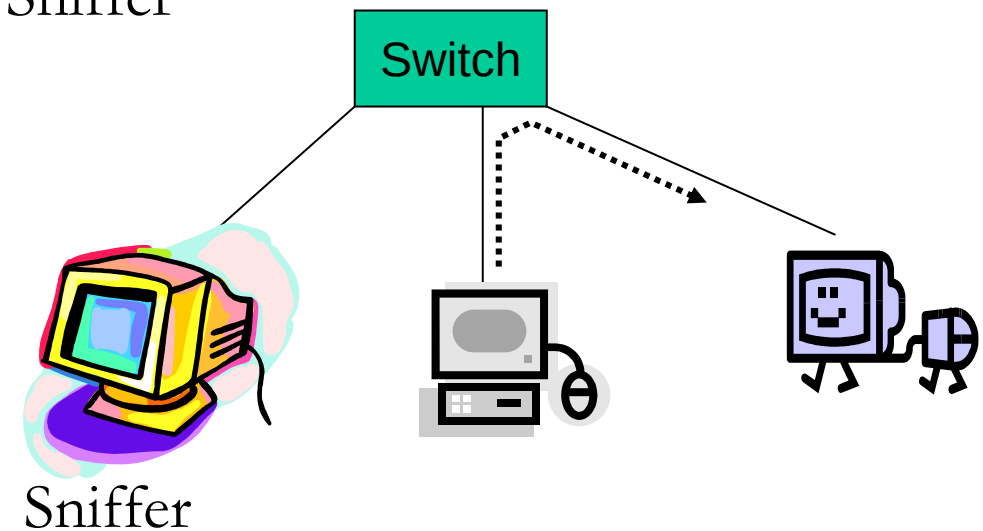
- For most network card, you only need to set a register bit
  - Then, you can get every frame on the wire
  - More overhead to computer
- At higher level use Windows NDIS to set promiscuous mode
  - NDIS: Network Driver Interface Specification
    - Standard API to Network Interface Card
  - Promiscuous mode is required part of NDIS specification

Sniffing

HUB vs. Switch



- Able to sniff the packets
- Able to receive broadcast
- Why?



- Unable to sniff the packets
- Able to receive broadcast
- Why?

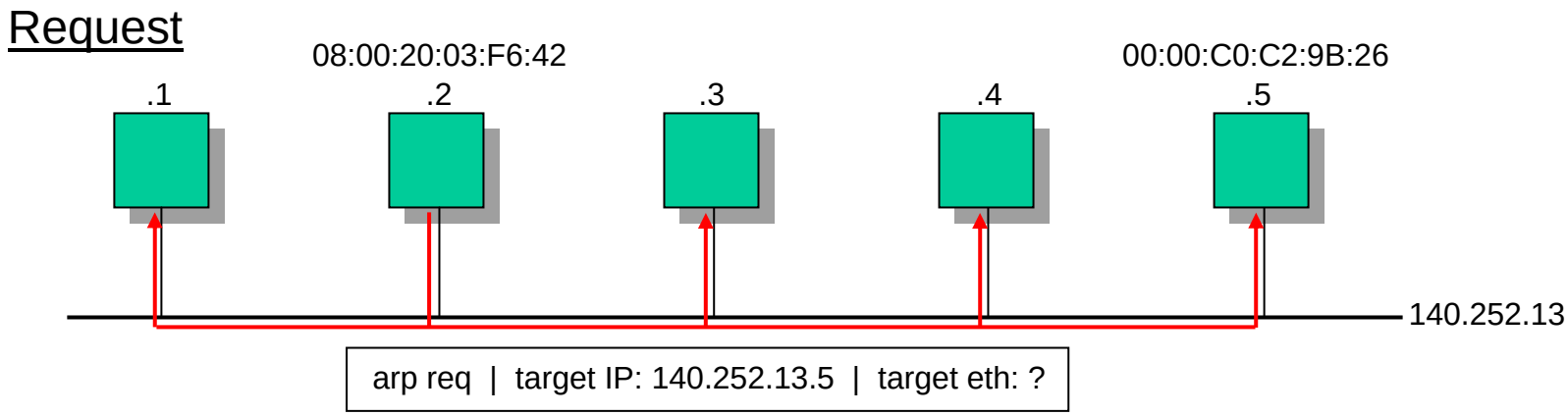
Sniffing

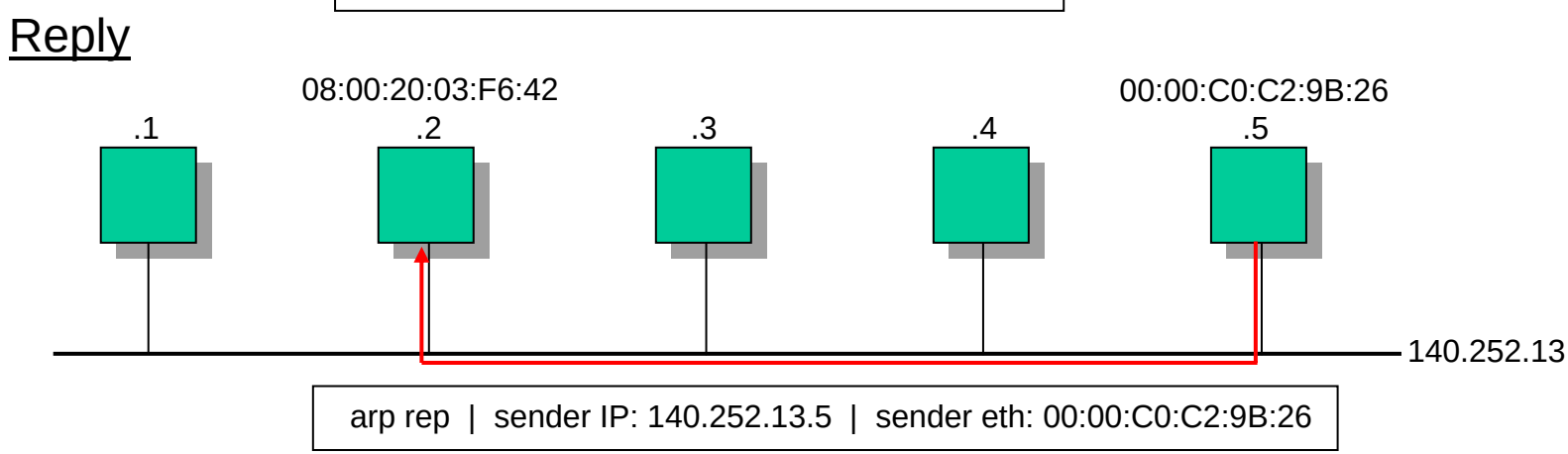
ARP Spoofing (Redirecting)

- Broadcast a response to ARP request by containing the victim's IP address and this hacker's MAC address as the source.
- Others will believe that the hacker has the victim's IP address, and send packets for the victim to this host.
- The hacker would then forward the packets to the victim.

Sniffing

ARP (Mapping IP Addresses to MAC Addresses)

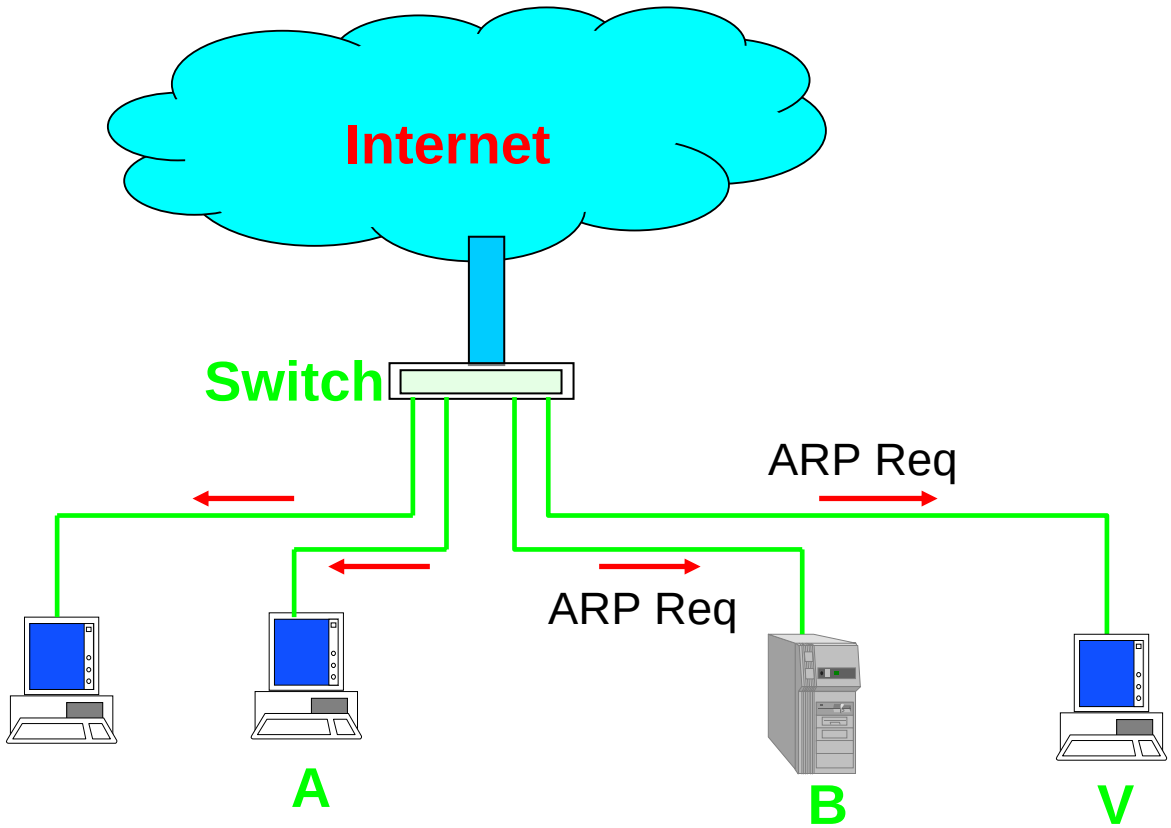




Sniffing

ARP Redirect

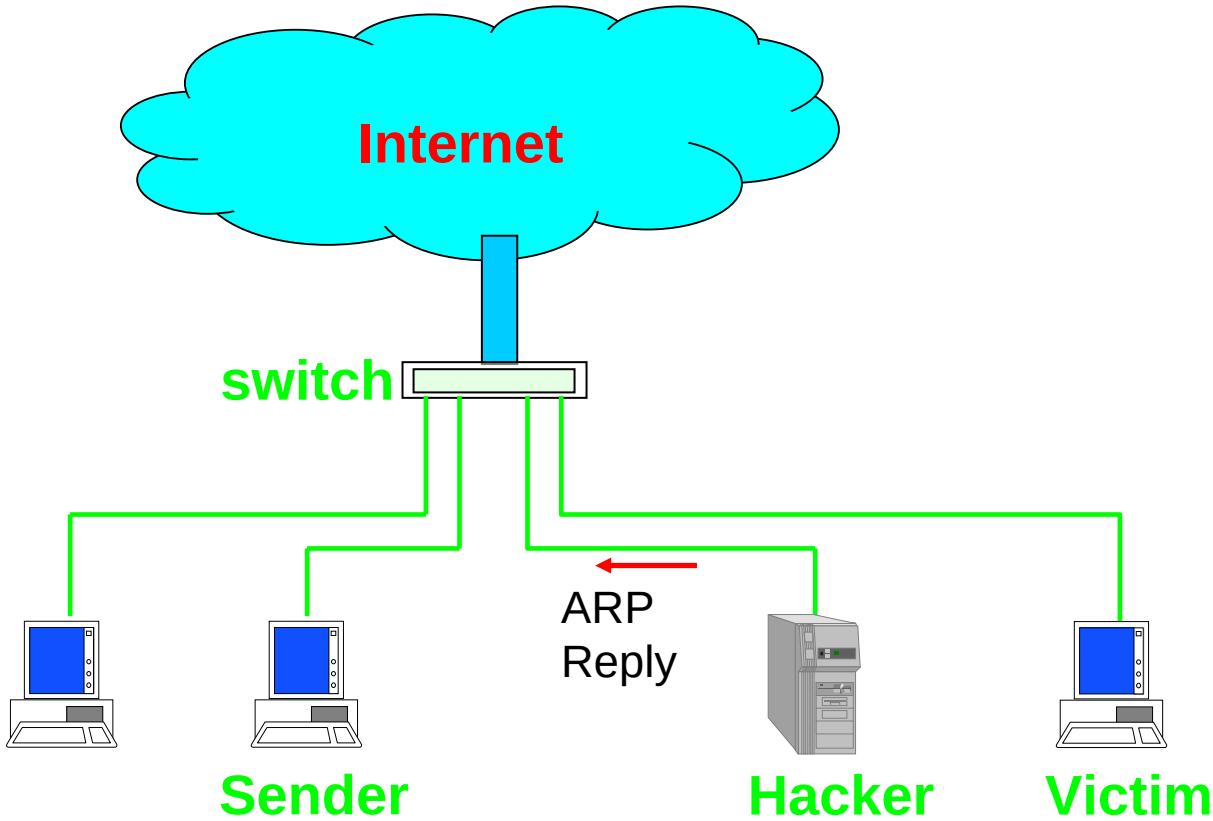
- An ARP request is made by the Sender



Sniffing

ARP Redirect

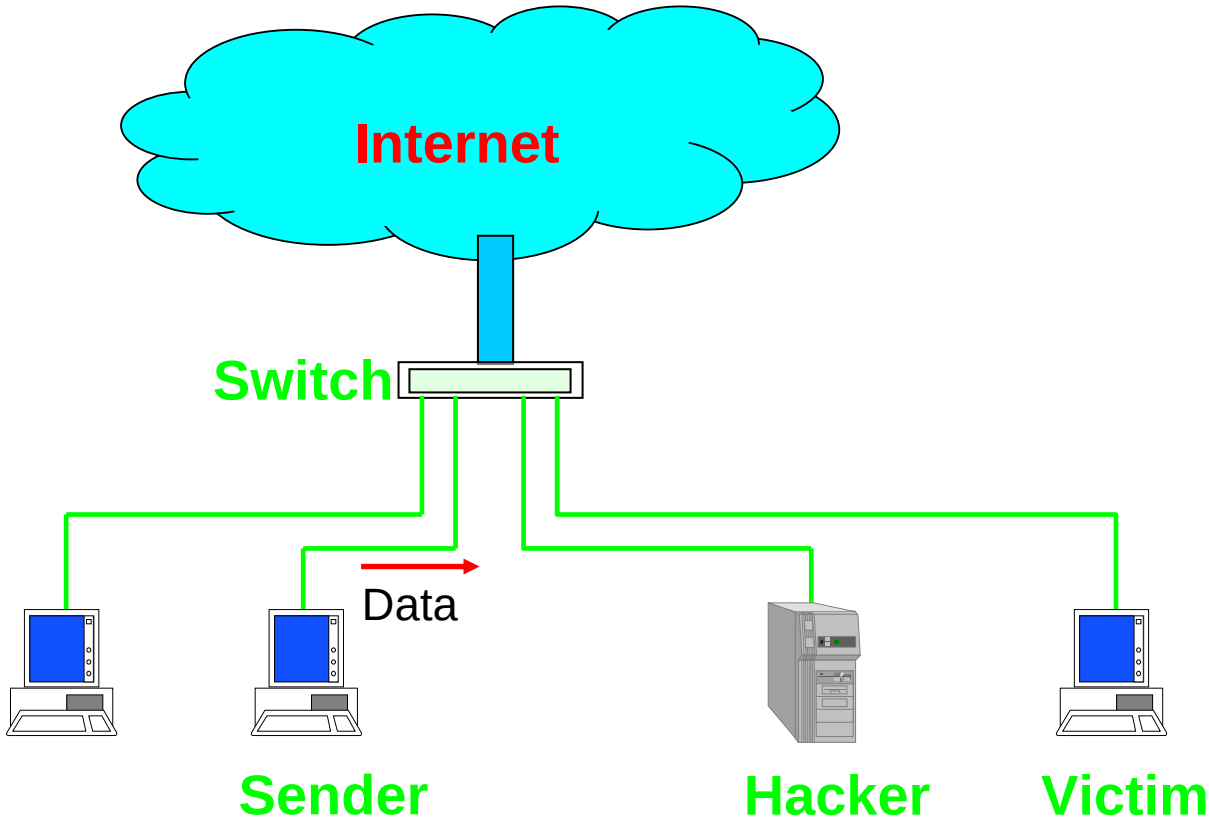
- Hacker Replies to the Request



Sniffing

ARP Redirect

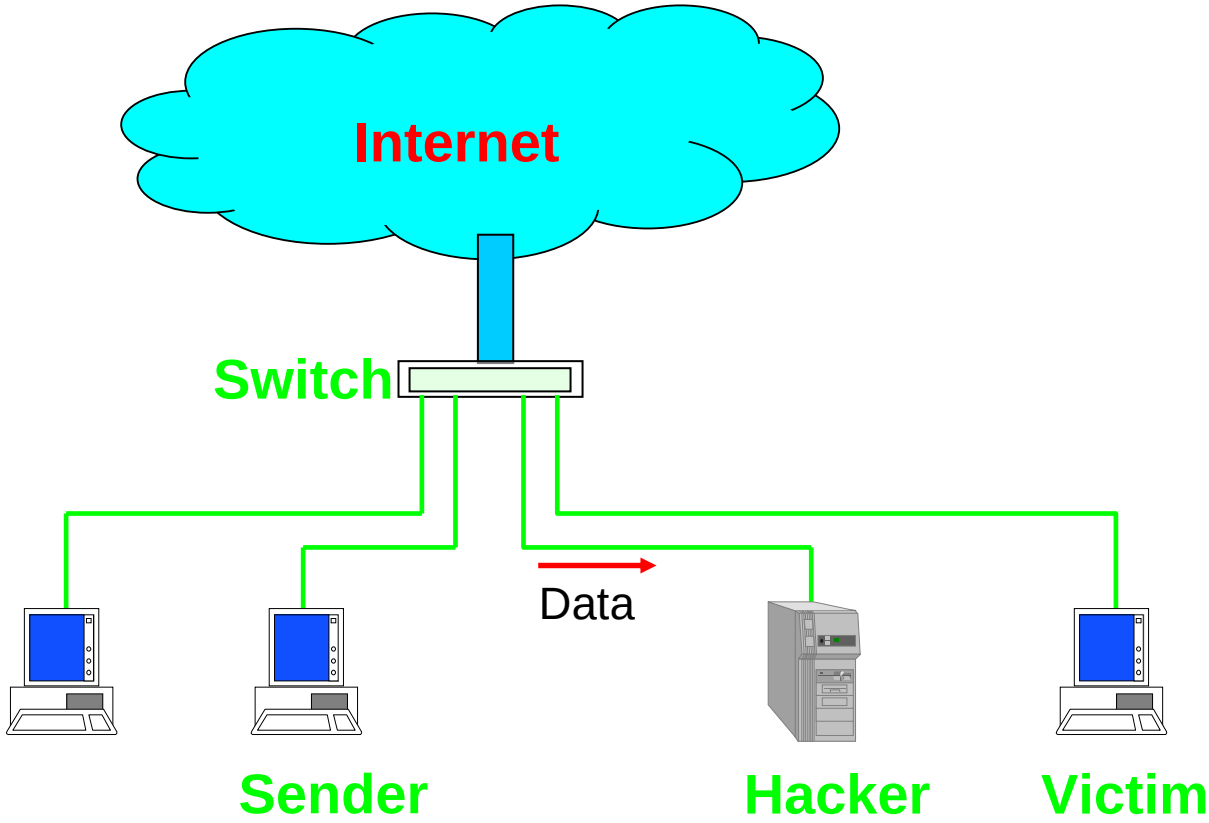
- Sender sends the data on the network



Sniffing

ARP Redirect

- Sender sends the data on the network



Sniffing

ARP Command

- Displays and modifies the IP-to-Physical address translation tables used by address resolution protocol
- Command

  ARP -s inet_addr eth_addr [if_addr]
  ARP -d inet_addr [if_addr]
  ARP -a [inet_addr] [-N if_addr]

inet_addr:	internet address.
eth_addr:	physical address.
if_addr:	Interface address

- Options

  -s: Adds the host and maps Internet address inet_addr to Physical address eth_addr. The entry is permanent.

  -d: Deletes the host specified by inet_addr. inet_addr may be wildcarded with * to delete all hosts.

  -a: Displays current ARP entries by interrogating the current protocol data.

  (Note: If inet_addr is specified, the IP and Physical addresses for only the specified computer are displayed. If more than one network interface uses ARP, entries for each ARP table are displayed.)

  -N if_addr: Displays the ARP entries for the network interface specified by if_addr.

  (Note: The Physical address is given as 6 hexadecimal bytes separated by hyphens.

  Example:

  > arp -s 157.55.85.212 00-aa-00-62-c6-09 .... Adds a static entry.

  > arp -a .... Displays the arp table.

Sniffing

ARP Detection

- Detection
  - A LAN with many computers, we want to detect which one of them is sniffing
  - We know all IP addresses of those computers
  - What happens if we send a ARP request with an IP address and a non-broadcasting MAC address?
    - E.g. fake broadcast FF:FF:FF:FF:FF:FE

Sniffing

ARP Protection

- Set the ARP table static
- Replace Hub with Switch
  - Makes sniffing harder
- Sniff the network for sniffing
  - Monitor changes of mapping of IP and MAC address
- Encryption
  - Ultimate solution: never transmit plain-text information

Ethereal

Ethereal

Functionality

- ?

Ethereal

Download and Install

- ?

Ethereal

Lab 1

- ?

Ethereal

Lab 2

- ?

Junk

Transport

Function

- Manages transmission packets
  - Repackages long messages when necessary into small packets for transmission
  - Reassembles packets in correct order to get the original message.
- Handles error recognition and recovery.
  - Transport layer at receiving acknowledges packet delivery.
  - Resends missing packets

Network

Function

- Manages addressing/routing of data within the subnet
  - Addresses messages and translates logical addresses and names into physical addresses.
  - Determines the route from the source to the destination computer
  - Manages traffic problems, such as switching, routing, and controlling the congestion of data packets.
- Routing can be:
  - Based on static tables
  - determined at start of each session
  - Individually determined for each packet, reflecting the current network load.

Datalink

Function

- Packages raw bits from the Physical layer into frames (logical, structured packets for data).
- Provides reliable transmission of frames
  - It waits for an acknowledgment from the receiving computer.
  - Retransmits frames for which acknowledgement not received

Physical

Function

- Transmits bits from one computer to another
- Regulates the transmission of a stream of bits over a physical medium.
- Defines how the cable is attached to the network adapter and what transmission technique is used to send data over the cable. Deals with issues like
  - The definition of 0 and 1, e.g. how many volts represents a 1, and how long a bit lasts?
  - Whether the channel is simplex or duplex?
  - How many pins a connector has, and what the function of each pin is?