



# Computer Networks I

3<sup>rd</sup> stage

## Lecture 3

### Protocol Layers and their Services

DR. Lecturer . Taqwa.F.Hassan

**Computer Department - College of Engineering  
University of Diyala**

**2021 - 2022**

**2nd semester**

# Protocol “layers”

*Networks are complex,  
with many “pieces” :*

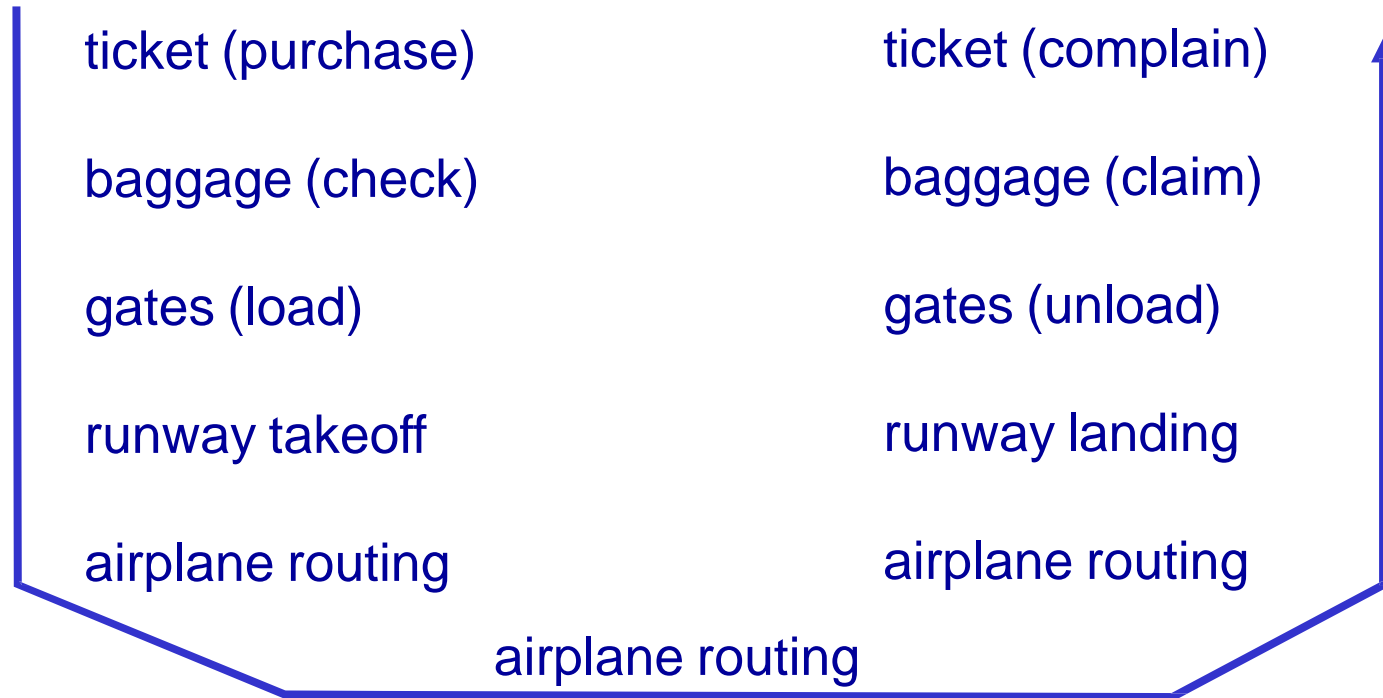
- hosts
- routers
- links of various media
- applications
- protocols
- hardware, software

*Question:*

is there any hope of *organizing*  
structure of network?

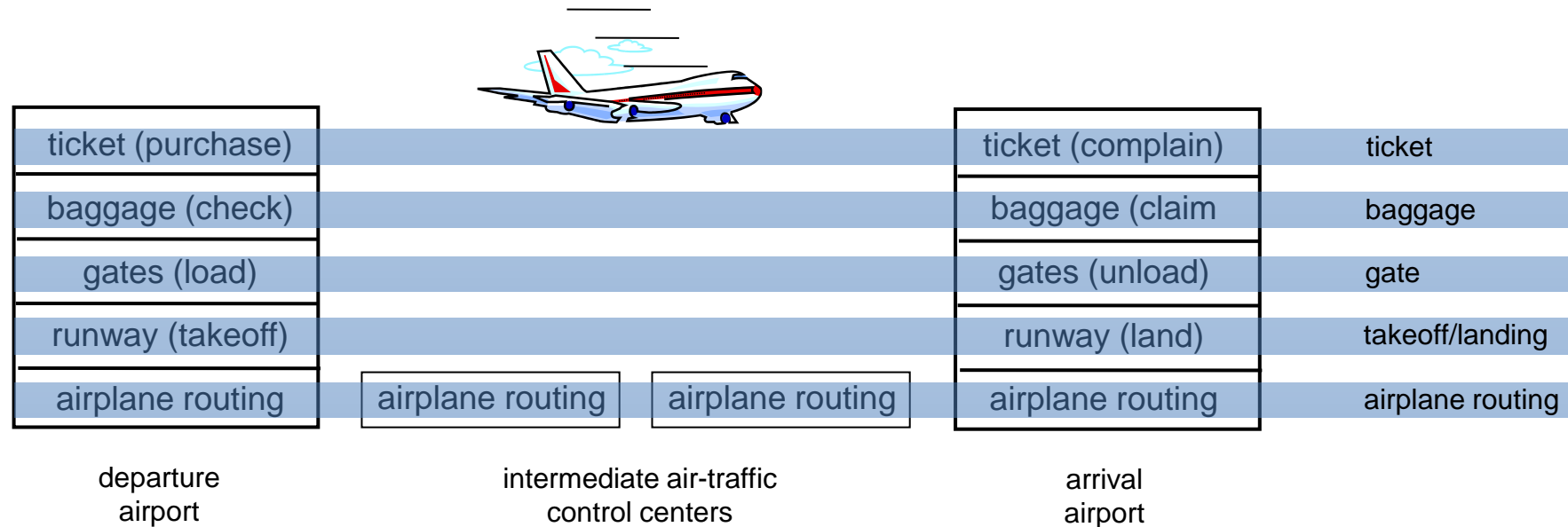
... . or at least our discussion of  
networks?

# Organization of air travel



❖ a series of steps

# Layering of airline functionality



**layers:** each layer implements a service via its own internal-layer actions relying on services provided by layer below

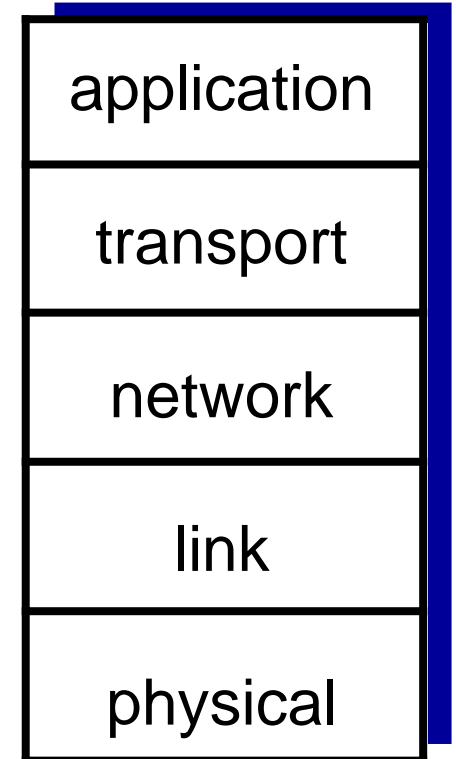
# Why layering?

dealing with complex systems:

- ❖ explicit structure allows identification, relationship of complex system's pieces
  - layered *reference model* for discussion
- ❖ modularization eases maintenance, updating of system
  - change of implementation of layer's service transparent to rest of system
  - e.g., change in gate procedure doesn't affect rest of system
- ❖ layering considered harmful?

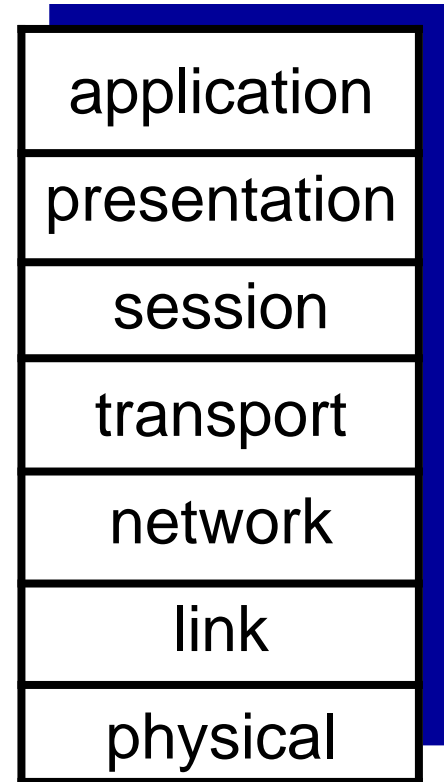
# Internet protocol stack (TCP/IP)

- ❖ *application*: supporting network applications
  - FTP, SMTP, HTTP
- ❖ *transport*: process-process data transfer
  - TCP, UDP
- ❖ *network*: routing of datagrams from source to destination
  - IP, routing protocols
- ❖ *link*: data transfer between neighboring network elements
  - Ethernet, 802.111 (WiFi), PPP
- ❖ *physical*: bits “on the wire”



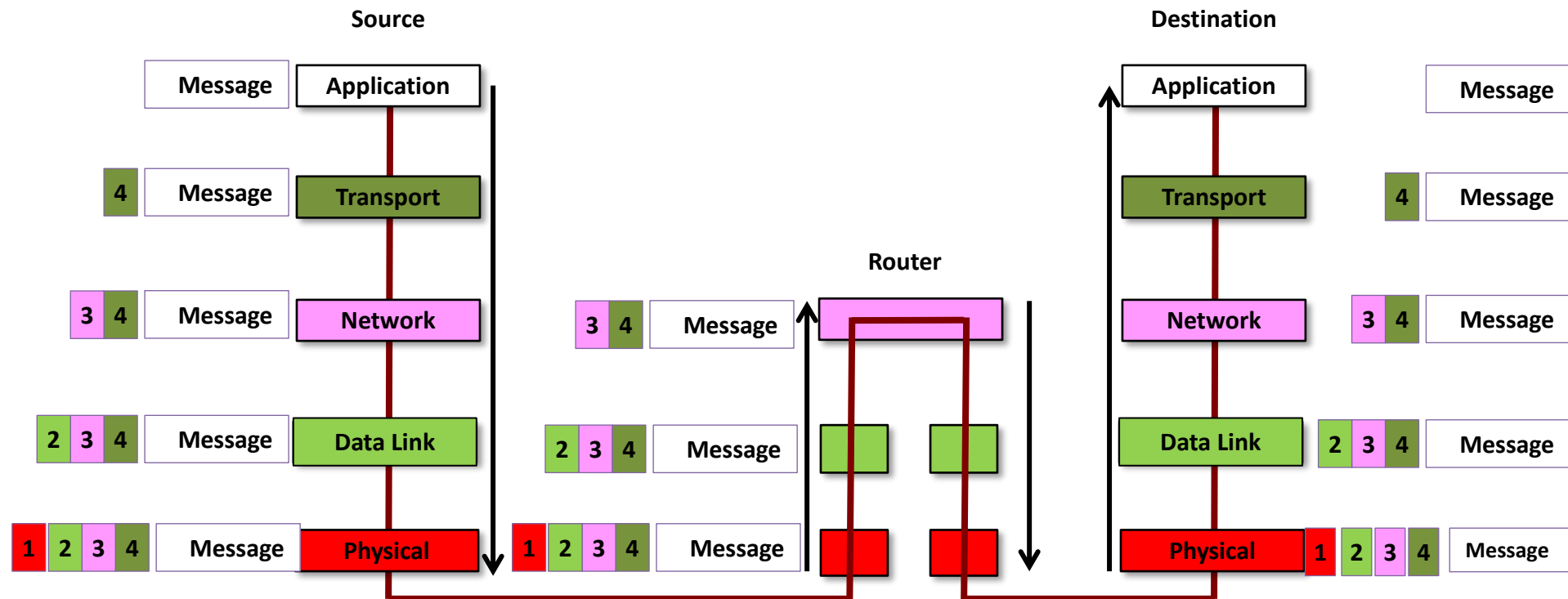
# ISO/OSI reference model

- ❖ **presentation**: allow applications to interpret meaning of data, e.g., encryption, compression, machine-specific conventions
- ❖ **session**: synchronization, checkpointing, recovery of data exchange
- ❖ Internet stack “missing” these layers!
  - these services, *if needed*, must be implemented in application
  - needed?



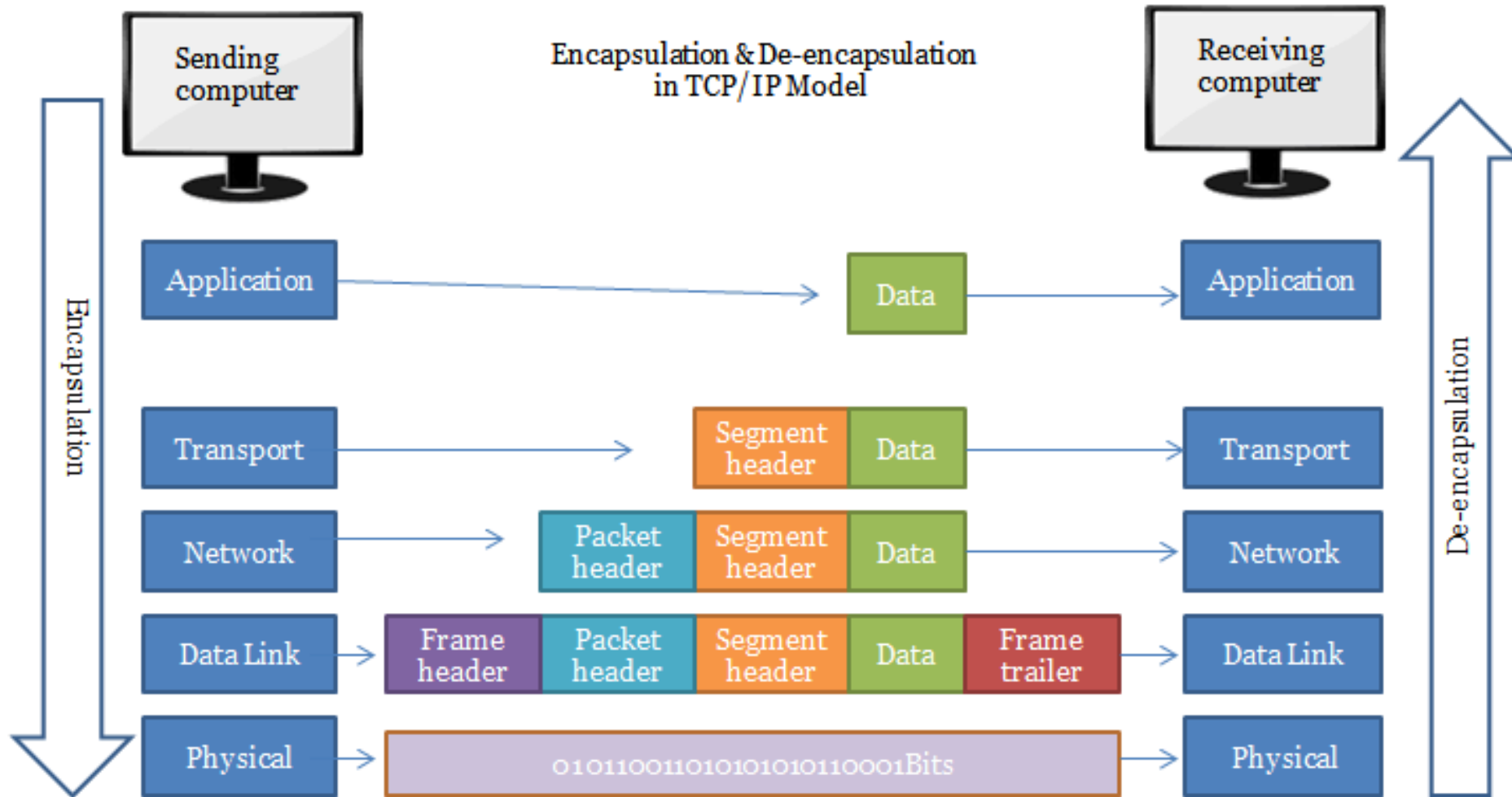
# Encapsulation/Decapsulation

- the process of adding/removing control information as a message passes through the layered model





# Encapsulation and de-encapsulation in TCP/IP model.



# Encapsulation

- ❖ Wraps data with the necessary protocol information before sending to next layer and for network transit.
- ❖ Data receives headers (control information), trailers, and other information
- ❖ Information - Data – Segments – Packets(datagram) – Frames – Bits
  - THIS IS IMPORTANT (defines datagram at each layer)

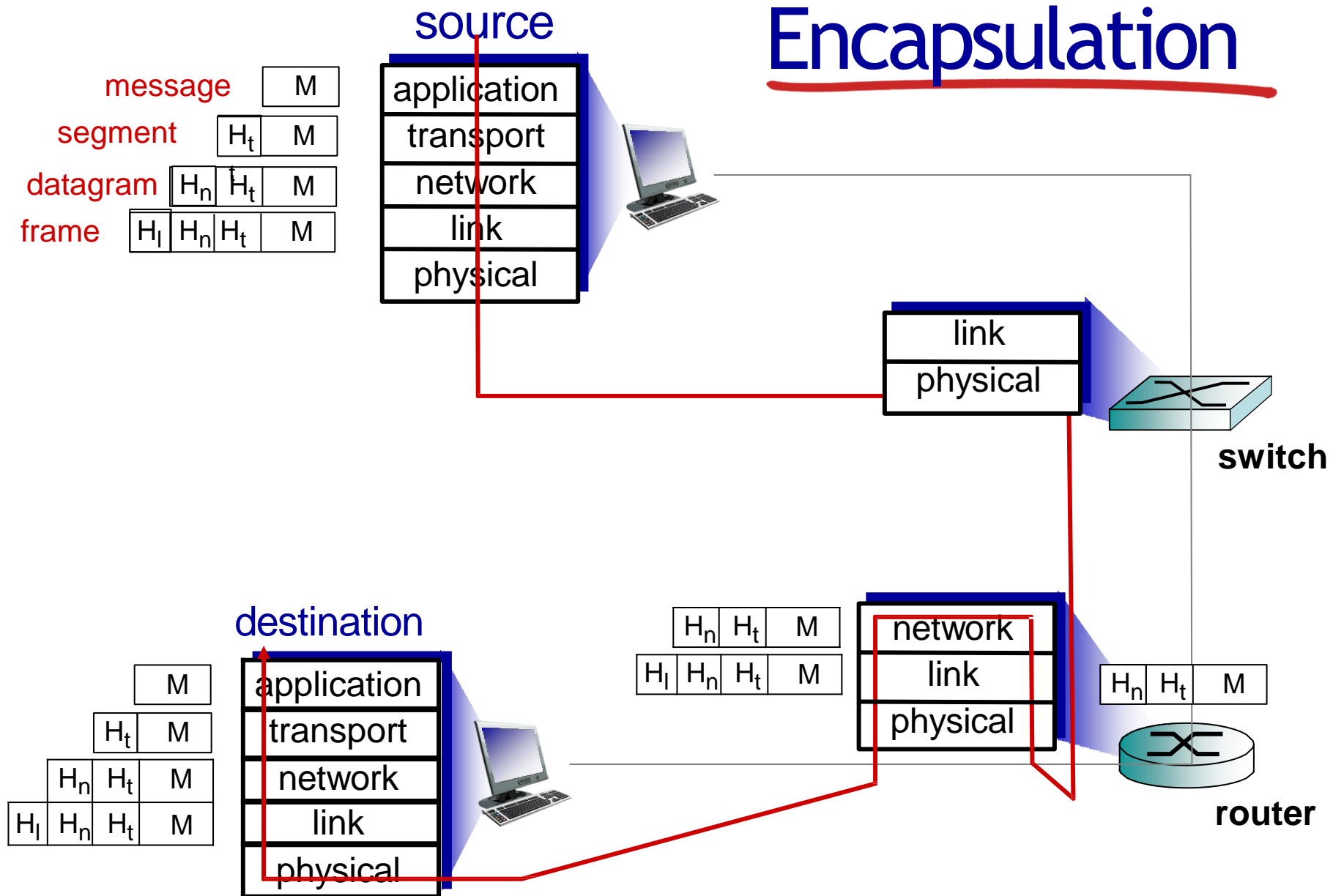
## Headers

- ❖ Control information placed before the data
- ❖ Adds address information, et. al.
- ❖ Exact format depends on layer

## Layers and Encapsulation

- ❖ Presentation – Data
- ❖ Transport – Segments
- ❖ Network – Packets
- ❖ Data Link – Frames
- ❖ Physical - Bits

# Encapsulation



# Network security

- ❖ **field of network security:**
  - how bad guys can attack computer networks
  - how we can defend networks against attacks
  - how to design architectures that are immune to attacks
- ❖ **Internet not originally designed with (much) security in mind**
  - *original vision:* “a group of mutually trusting users attached to a transparent network” 😊
  - Internet protocol designers playing “catch-up”
  - security considerations in all layers!

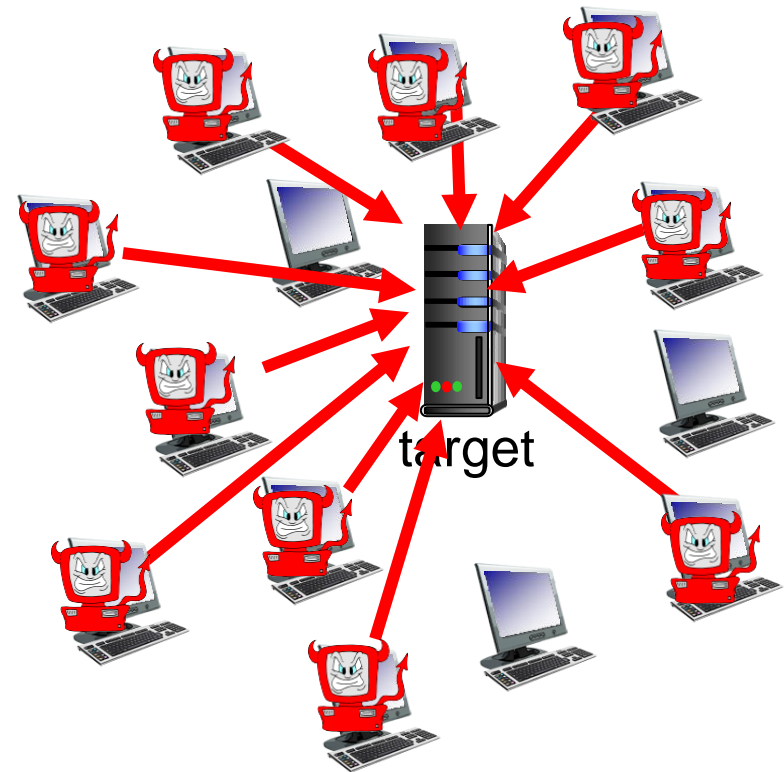
# Bad guys: put malware into hosts via Internet

- ❖ malware can get in host from:
  - *virus*: self-replicating infection by receiving/executing object (e.g., e-mail attachment)
  - *worm*: self-replicating infection by passively receiving object that gets itself executed
- ❖ **spyware malware** can record keystrokes, web sites visited, upload info to collection site
- ❖ infected host can be enrolled in **botnet**, used for spam. DDoS attacks

# Bad guys: attack server, network infrastructure

*Denial of Service (DoS):* attackers make resources (server, bandwidth) unavailable to legitimate traffic by overwhelming resource with bogus traffic

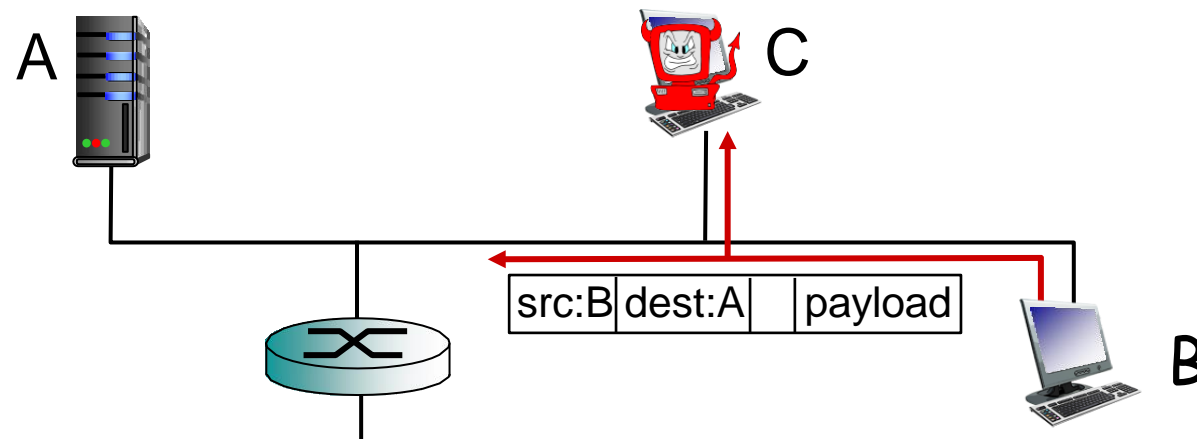
1. select target
2. break into hosts around the network (see botnet)
3. send packets to target from compromised hosts



# Bad guys can sniff packets

## *packet “sniffing”* :

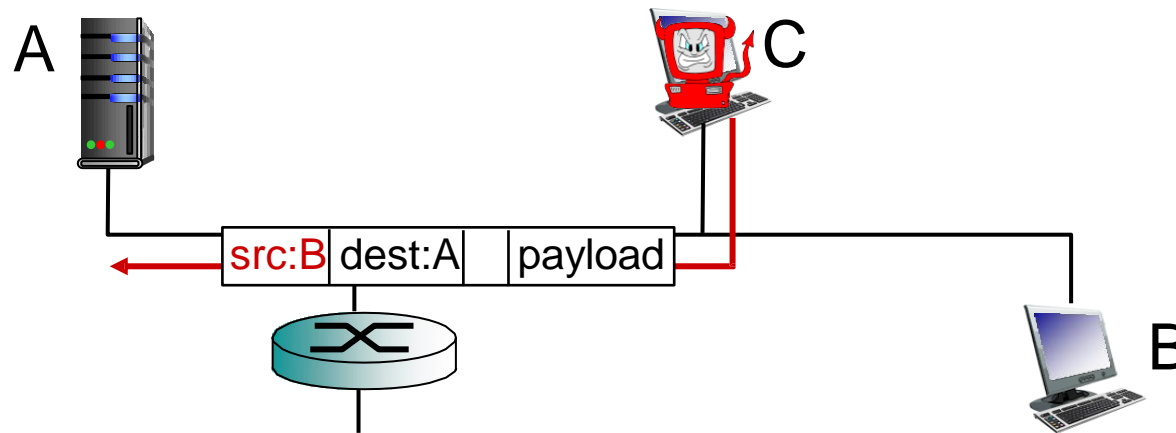
- broadcast media (shared ethernet, wireless)
- promiscuous network interface reads/records all packets (e.g., including passwords!) passing by



- ❖ wireshark software used for end-of-chapter labs is a (free) packet-sniffer

# Bad guys can use fake addresses

*IP spoofing*: send packet with false source address



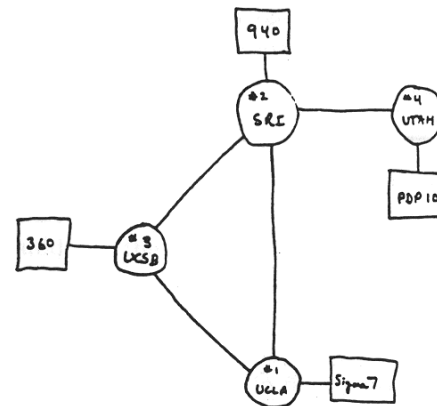
... *lots more on security (throughout, Chapter 8)*



# Internet history

## *1961-1972: Early packet-switching principles*

- ❖ **1961:** Kleinrock - queueing theory shows effectiveness of packet-switching
- ❖ **1964:** Baran - packet-switching in military nets
- ❖ **1967:** ARPAnet conceived by Advanced Research Projects Agency
- ❖ **1969:** first ARPAnet node operational
- ❖ **1972:**
  - ARPAnet public demo
  - NCP (Network Control Protocol) first host-host protocol
  - first e-mail program
  - ARPAnet has 15 nodes



THE ARPA NETWORK

# Internet history

## *1972-1980: Internetworking, new and proprietary nets*

- ❖ **1970:** ALOHAnet satellite network in Hawaii
- ❖ **1974:** Cerf and Kahn - architecture for interconnecting networks
- ❖ **1976:** Ethernet at Xerox PARC
- ❖ **late70' s:** proprietary architectures: DECnet, SNA, XNA
- ❖ **late 70' s:** switching fixed length packets (ATM precursor)
- ❖ **1979:** ARPAnet has 200 nodes

### **Cerf and Kahn' s internetworking principles:**

- minimalism, autonomy - no internal changes required to interconnect networks
- best effort service model
- stateless routers
- decentralized control

**define today' s Internet  
architecture**

# Internet history

## *1980-1990: new protocols, a proliferation of networks*

- ❖ **1983:** deployment of TCP/IP
- ❖ **1982:** smtp e-mail protocol defined
- ❖ **1983:** DNS defined for name-to-IP-address translation
- ❖ **1985:** ftp protocol defined
- ❖ **1988:** TCP congestion control
- ❖ new national networks: Cset, BITnet, NSFnet, Minitel
- ❖ 100,000 hosts connected to confederation of networks

# Internet history

## *1990, 2000' s: commercialization, the Web, new apps*

- ❖ early 1990' s: ARPAnet decommissioned
- ❖ 1991: NSF lifts restrictions on commercial use of NSFnet (decommissioned, 1995)
- ❖ early 1990s: Web
  - hypertext [Bush 1945, Nelson 1960' s]
  - HTML, HTTP: Berners-Lee
  - 1994: Mosaic, later Netscape
  - late 1990' s: commercialization of the Web
- late 1990' s - 2000' s:
  - ❖ more killer apps: instant messaging, P2P file sharing
  - ❖ network security to forefront
  - ❖ est. 50 million host, 100 million+ users
  - ❖ backbone links running at Gbps

# Internet history

## *2005-present*

- ❖ ~750 million hosts
  - Smartphones and tablets
- ❖ Aggressive deployment of broadband access
- ❖ Increasing ubiquity of high-speed wireless access
- ❖ Emergence of online social networks:
  - Facebook: soon one billion users
- ❖ Service providers (Google, Microsoft) create their own networks
  - Bypass Internet, providing “instantaneous” access to search, email, etc.
- ❖ E-commerce, universities, enterprises running their services in “cloud” (eg, Amazon EC2)

# Introduction: summary

*covered a “ton” of material!*

- ❖ Internet overview
- ❖ what's a protocol?
- ❖ network edge, core, access network
  - packet-switching versus circuit-switching
  - Internet structure
- ❖ performance: loss, delay, throughput
- ❖ layering, service models
- ❖ security
- ❖ history

*you now have:*

- ❖ context, overview, “feel” of networking
- ❖ more depth, detail *to follow!*

*Thank you for listening*

**Taqwa Altameemi**