



TCP/IP

The language of the Internet

UNIX Network Administration

CIS 68C2-01

TCP/IP

- What is TCP/IP?
 - ✗ TCP/IP is a suite of protocols
 - ✗ Protocol: formal rules of behavior
 - ✗ A sampling of the protocols defined in TCP/IP
 - ✗ UDP, TCP, ARP, IPv4, IPv6, ICMP, RARP, ...
 - ✗ TCP/IP is TCP over IP
 - ✗ Transmission Control Protocol
 - ✗ Internet Protocol

TCP/IP

□ Features of TCP/IP

✗ Open protocol standards

- ✗ Most defined in Request for Comments (RFCs)
- ✗ Others defined in MIL STD and IEN (abandon)

✗ Hardware independent

- ✗ TCP/IP can be carried over Ethernet, token ring, dial-up, serial lines, FDDI, etc.

✗ Common addressing scheme

- ✗ Each host is uniquely identifiable worldwide

✗ Standardized high-level protocols

- ✗ Allows user services

ISO's OSI Networking Model

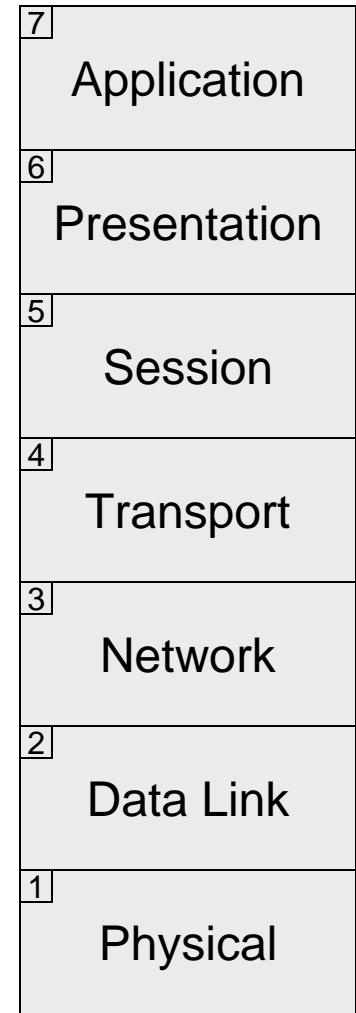
□ OSI Reference Model

✘ Open Systems Interconnect

- ✘ Defined by International Standards Organization (ISO)

✘ Defines 7 Functional Layers

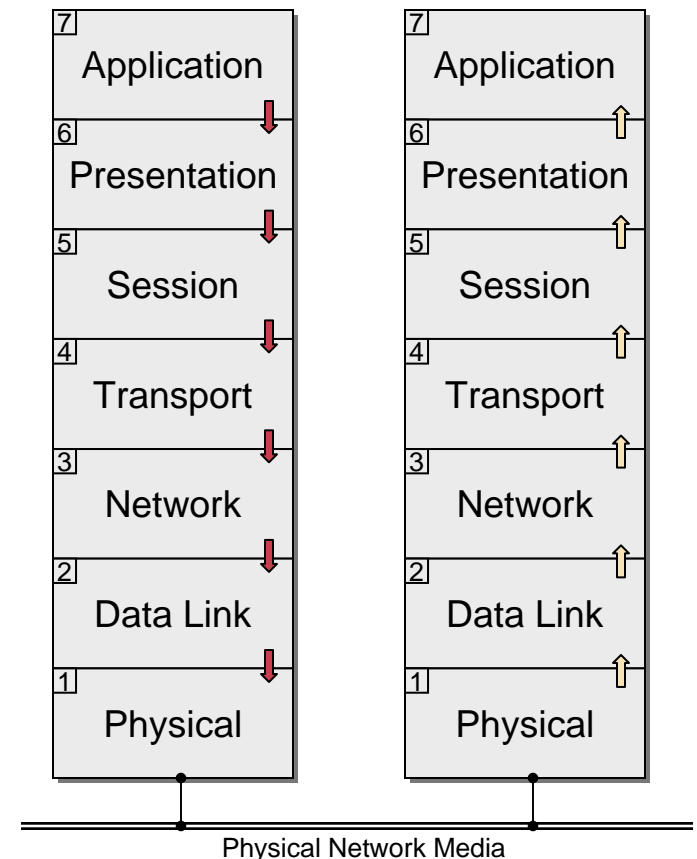
- ✘ Each layer defines a function performed when data is exchanged between cooperating applications across a network
- ✘ Layers are data communication functions
 - ✘ Not individual protocols
 - ✘ A layer's function may be performed by many protocols



ISO's OSI Networking Model

□ The 7 Layer Stack

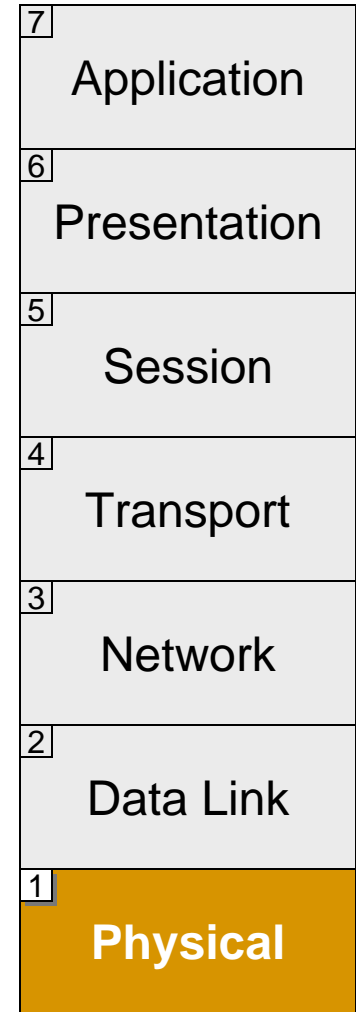
- ✗ Data travels...
 - ✗ ...down the stack to be transmitted to the network
 - ✗ ...up the stack when received from the network
- ✗ Each layer knows how to pass data to the layer above and below itself
- ✗ Each protocol communicates with its peer across network
 - ✗ Eg. session layer to session layer



ISO's OSI Networking Model

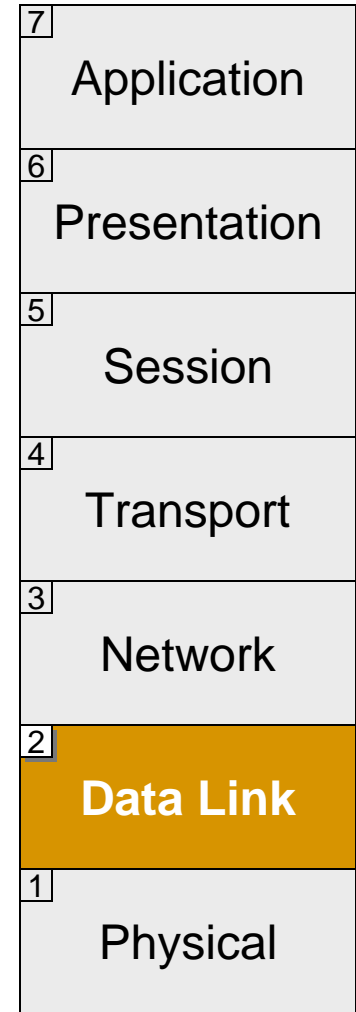
□ Layer 1 – Physical Layer

- ✘ Defines hardware characteristics of physical media
- ✘ Uses exiting hardware media standards
 - ✘ IEEE 802.3
 - ✘ RS232C
 - ✘ V.35



ISO's OSI Networking Model

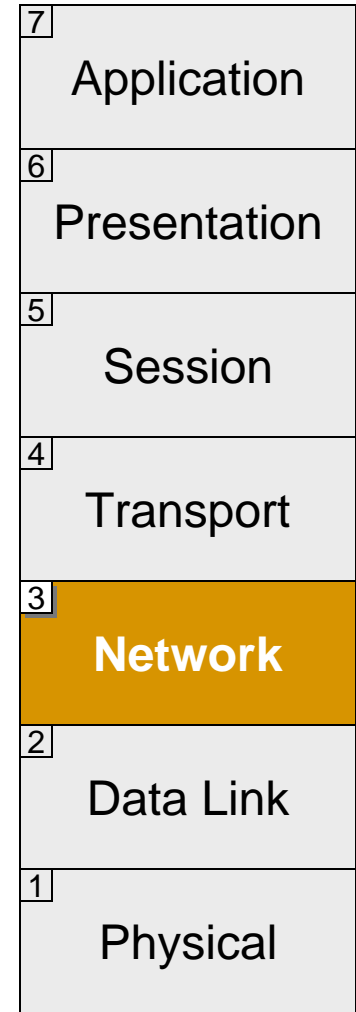
- Layer 2 – Data Link Layer
 - ✗ Reliable delivery of data across physical network
 - ✗ *Reliable* does not mean *guaranteed*
 - ✗ TCP/IP generally does not use this layer directly



ISO's OSI Networking Model

□ Layer 3 – Network Layer

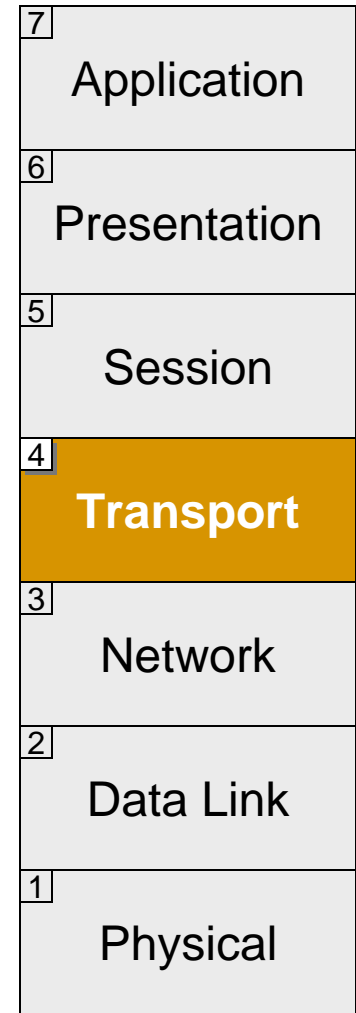
- ✗ Manages connections across network
- ✗ Performed in IP of TCP/IP
 - ✗ Isolates upper layers from underlying physical network
 - ✗ Handles addressing and data delivery
 - ✗ TCP/IP's Network Layer



ISO's OSI Networking Model

□ Layer 4 – Transport Layer

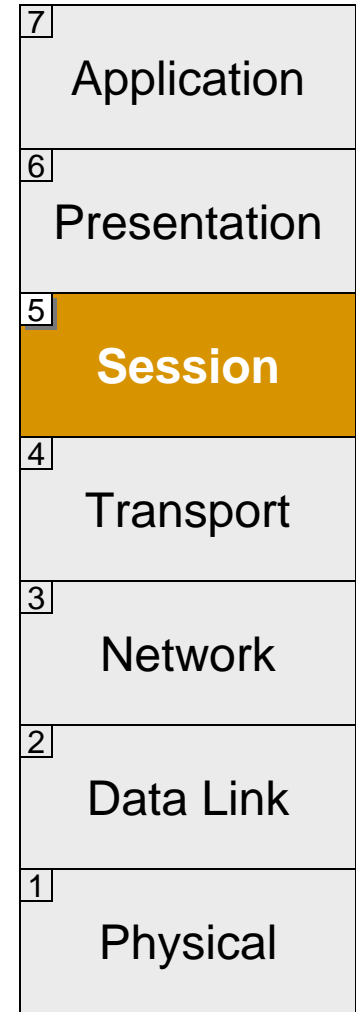
- ✗ Guarantees the accurate transmission of data across a network
- ✗ Performed in TCP of TCP/IP
- ✗ TCP/IP also has UDP
 - ✗ No reliability
 - ✗ Connectionless



ISO's OSI Networking Model

□ Layer 5 – Session Layer

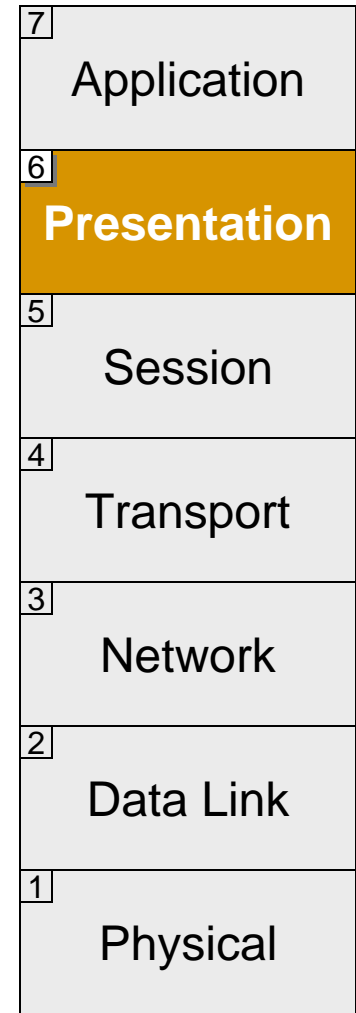
- ✘ Manages sessions (connections) between applications across network
- ✘ Implemented in TCP/IP in Transport Layer
 - ✘ Connections are called sockets and ports
 - ✘ Not sessions



ISO's OSI Networking Model

□ Layer 6 – Presentation Layer

- ✗ Describes how data is represented between two applications
- ✗ Required to allow applications to exchange data
- ✗ Standard data presentation routines are defined
- ✗ Often handled within applications
 - ✗ Also handled via TCP protocols such as XDR or MIME



ISO's OSI Networking Model

□ Layer 7 – Application Layer

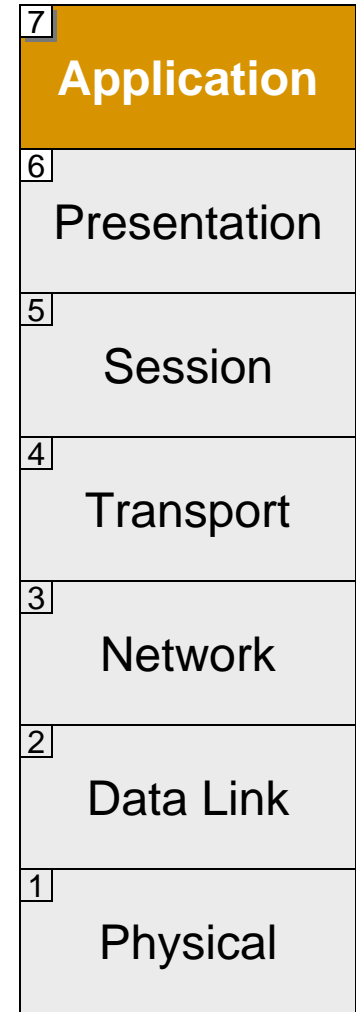
✘ Applications

- ✘ Network applications and utilities

- ✘ Daemons providing network services

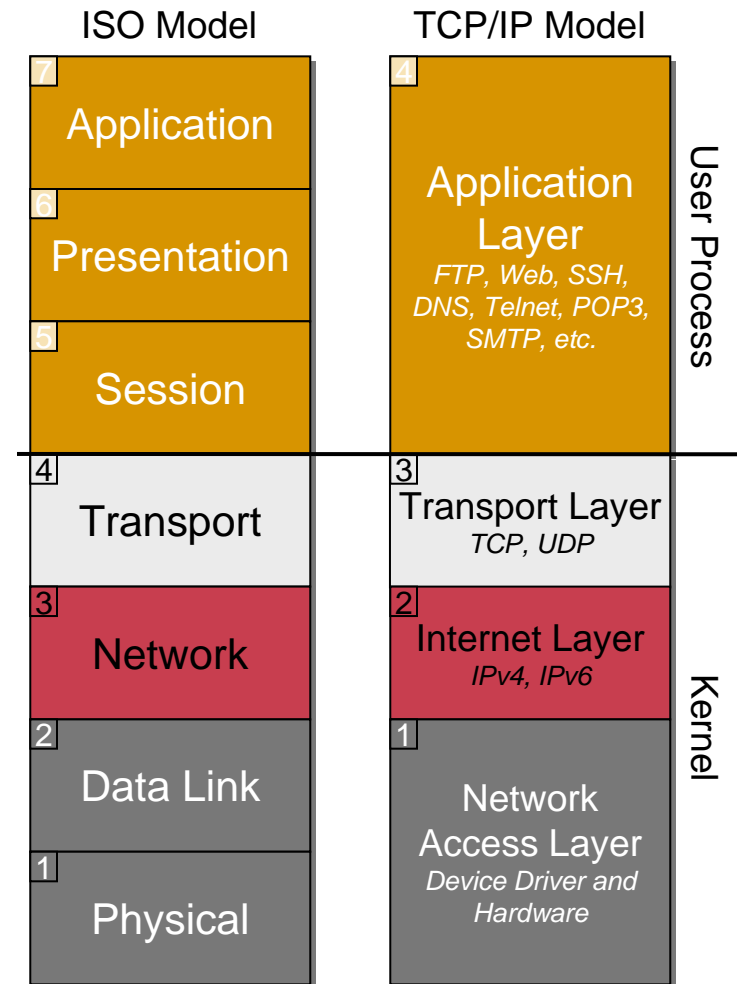
✘ TCP/IP Application

- ✘ Considered anything above Transport Layer



TCP/IP Protocol Architecture

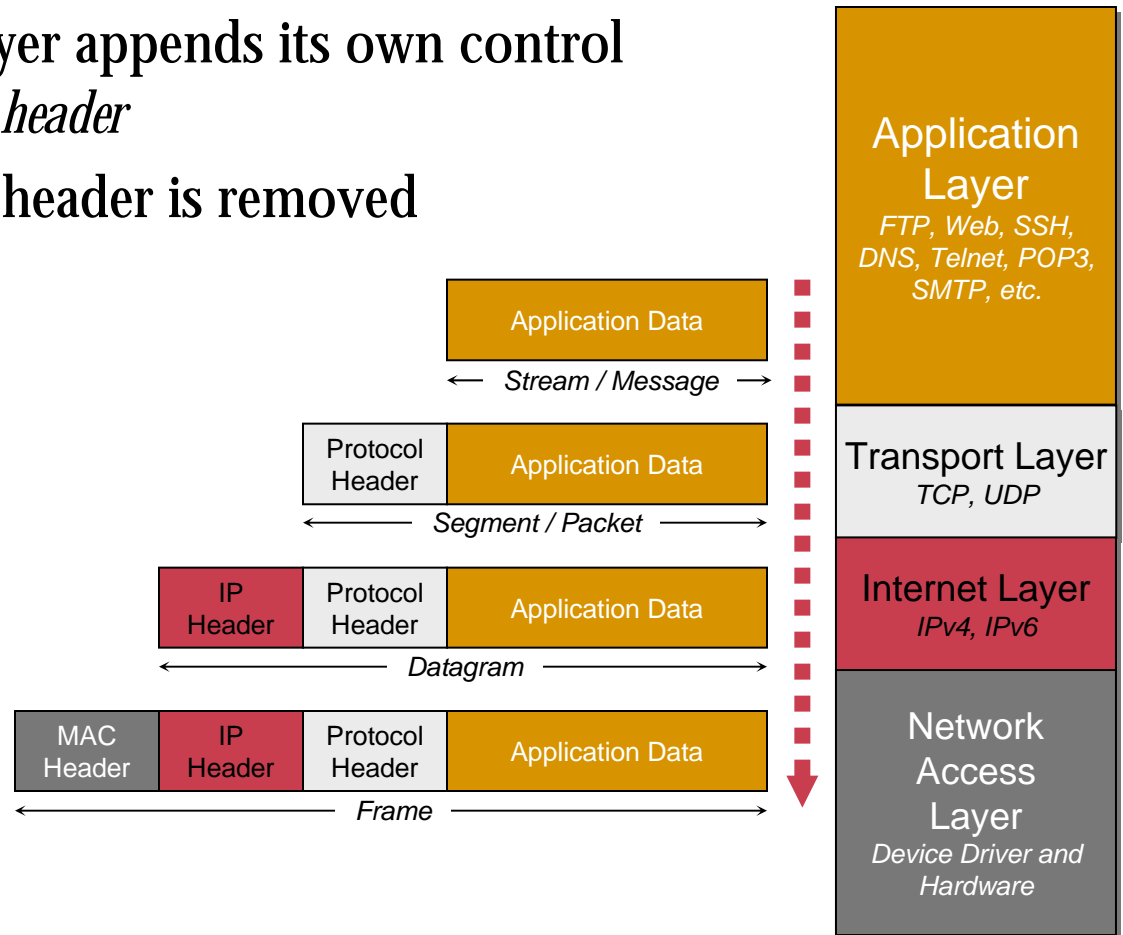
- TCP/IP Architecture
 - ✗ OSI is considered overkill
 - ✗ Complex and inefficient
 - ✗ TCP/IP is described with a simpler model
 - ✗ TCP/IP has 3 to 5 layers
 - ✗ No universal agreement on exact layer definitions



TCP/IP Protocol Architecture

□ Data Encapsulation

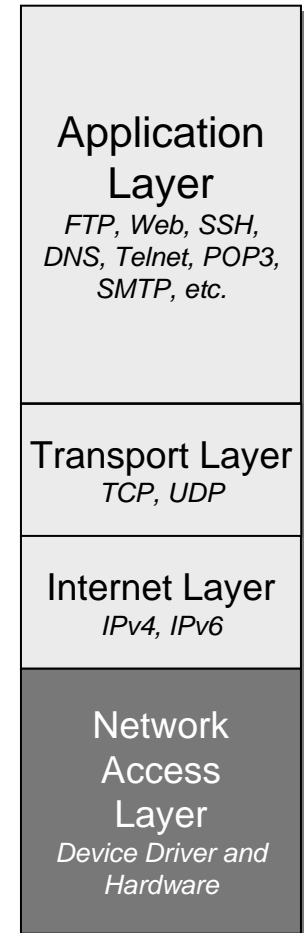
- ✗ On transmit, each layer appends its own control information called a *header*
- ✗ On receive, a layer's header is removed



TCP/IP Protocol Architecture

□ Network Access Layer

- ✗ Maps IP address into physical address
 - ✗ Physical address scheme depends on media
 - ✗ RFC 826: Address Resolution Protocol (ARP)
 - ✗ Specifies IP address to Ethernet address translation
- ✗ Encapsulates IP datagram
 - ✗ RFC 894: Specifies IP datagram encapsulation for Ethernet networks
 - ✗ MAC header
- ✗ Frame delivery to/from the physical network media



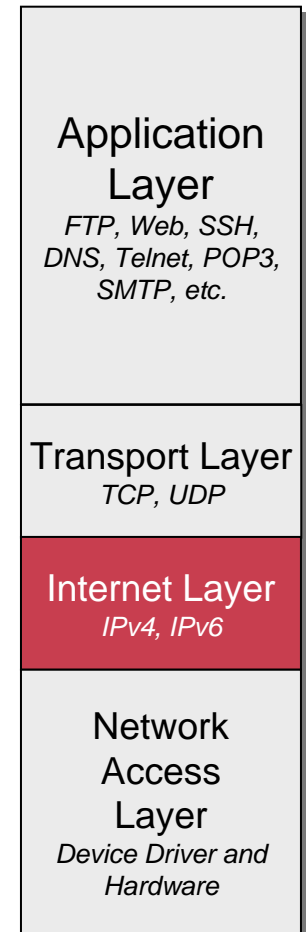
TCP/IP Protocol Architecture

□ Internet Layer

✗ RFC 791: Internet Protocol (IP)

✗ Defines ...

- ✗ The datagram
 - ✗ Contains the destination IP address
- ✗ The internet addressing scheme
- ✗ How data is moved between the network access and transport layers
- ✗ Routing of datagrams to hosts
- ✗ Datagram fragmentation and re-assembly
- ✗ IP is a connectionless protocol



TCP/IP Protocol Architecture

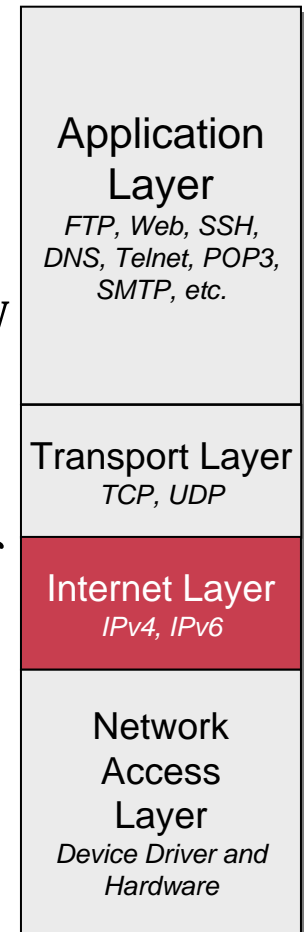
□ Internet Layer *(continued)*

✗ Routing

- ✗ Systems can only deliver packets to hosts directly attached to same physical network
- ✗ Packets are switched from one network to another by routers (also called gateways)

✗ Fragmentation and re-assembly

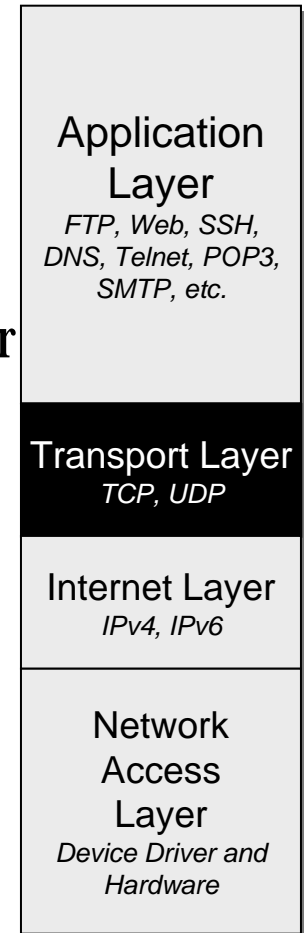
- ✗ IP ensures frames are sized according to the needs of the underlying physical media
- ✗ Frame sizes (Maximum Transmission Unit)
 - ✗ Ethernet: 1500 bytes
 - ✗ FDDI: 4500 bytes



TCP/IP Protocol Architecture

□ Transport Layer

- ✗ Delivers data between Internet Layer and Application Layer
- ✗ Data delivered to a particular *service*
 - ✗ Specified by the destination port in the transport layer header
- ✗ Two most common protocols
 - ✗ UDP Protocol
 - ✗ Best-effort, connectionless datagrams
 - ✗ TCP Protocol
 - ✗ Reliable, connection-oriented byte-stream



TCP/IP Protocol Architecture

□ Transport Layer *(continued)*

✗ RFC 768: UDP Protocol

- ✗ Minimal overhead

- ✗ Connectionless datagram delivery

- ✗ Delivery is not guaranteed

✗ RFC 793: TCP Protocol

- ✗ Reliable

- ✗ PAR – Positive Acknowledgment with Re-transmission

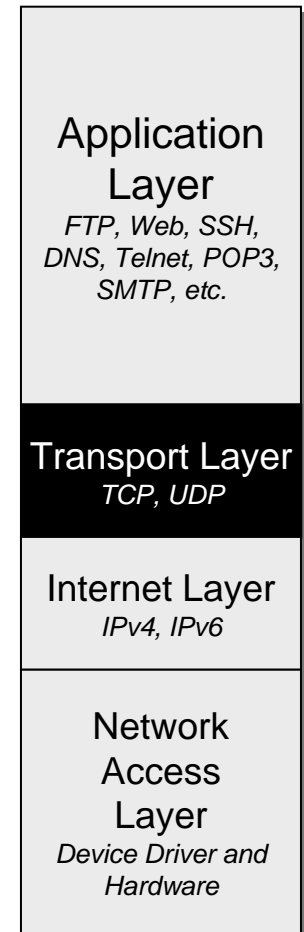
- ✗ Checksums used to ensure correct data delivered

- ✗ Connection oriented (End-to-end connection)

- ✗ Established with three-way handshake

- ✗ Byte Stream

- ✗ TCP data is continuous stream of bytes



TCP/IP Protocol Architecture

□ Application Layer

- ✘ Many networking applications and *services*
 - ✘ User programs: telnet, ftp, http, etc.
 - ✘ Daemons: DNS, NFS, NIS, etc.
- ✘ Applications/servers listen to a particular TCP/UDP port number
 - ✘ Packet directed using the destination port in Transport Layer's header

