CENGAGE Learning

TOPIC 05
TCP/IP

Objectives

- Describe the purpose of a network protocol and the layers in the TCP/IP architecture
- Describe TCP/IP Application-layer protocols
- Describe TCP/IP Transport-layer protocols
- Describe TCP/IP Internetwork-layer protocols
- Describe TCP/IP Network access-layer protocols

TCP/IP's Layered Architecture

- **Protocols** are rules and procedures for communication and behavior
  - Computers must "speak" the same language and agree on the rules of communication
- When a set of protocols works cooperatively it is called a **protocol suite** (or "protocol stack")
- The most common protocol stack is **Transmission Control Protocol/Internet Protocol (TCP/IP)**
- TCP/IP is composed of more than a dozen protocols operating at different levels of the communication process

TCP/IP's Layered Architecture

Layer name	TCP/IP protocols			
Application	HTTP	FTP	DHCP	TFTP
	SMTP	POP3	DNS	SNMP
Transport	TCP		UDP	
Internetwork	ICMP	ARP	IPsec	
	IPv4 and IPv6			
Network access	Ethernet, token ring, FDDI, WAN technologies			

Figure 5-1 The TCP/IP layered architecture

TCP/IP's Layered Architecture

- Example of how the layers work together:
  - You start your Web browser and your home page is *http://www.cengage.com*
  - The web browser formats a request for your home page by using the Application layer protocol HTTP
  - The request looks something like:

    get the cengage.com home page

  - The unit of information the Application layer works with is simply called "data"

TCP/IP's Layered Architecture

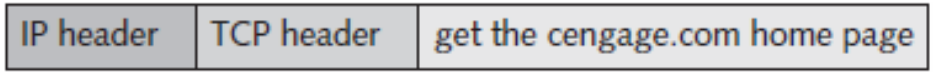
- Example continued:
  - The Application-layer protocol HTTP passes the request down to the Transport-layer protocol (TCP)
  - TCP adds a header to the request that looks like:

TCP header	get the cengage.com home page

  - The unit of information the Transport layer works with is called a segment
  - TCP passes the segment to the Internetwork layer protocol (IP)

TCP/IP's Layered Architecture

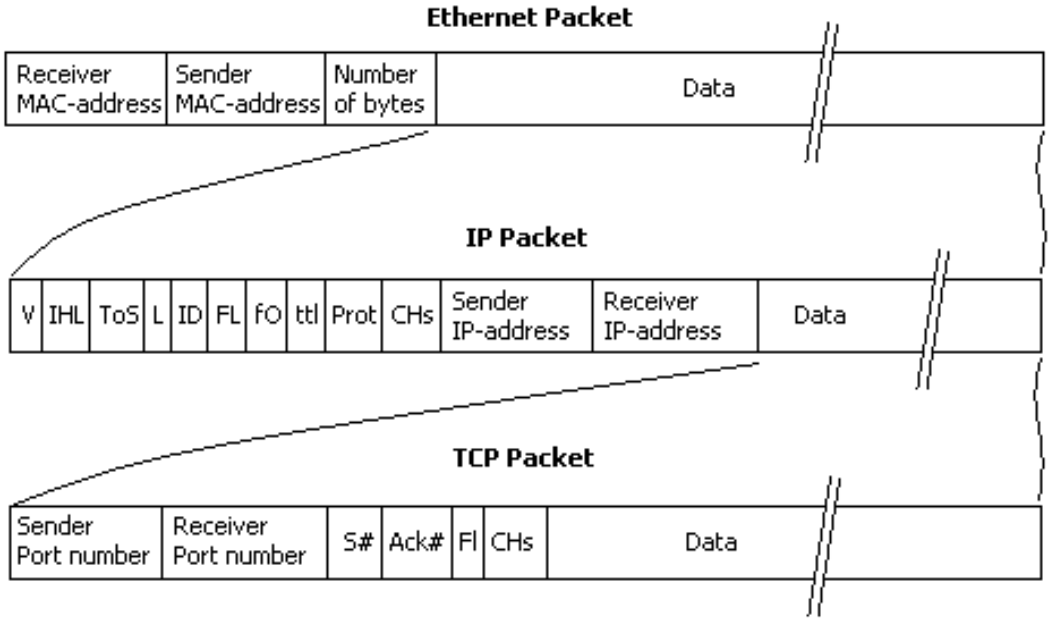
- Example continued:
  - IP places its header on the segment:



  - The unit of information is now called a packet
  - The packet is passed down to the Network access layer, where the NIC operates
  - A frame header and trailer are added

  - The frame is delivered to the network medium as bits
    - on its way to the *www.cengage.com* server
  - The web server processes it and returns a Web page

Packet details

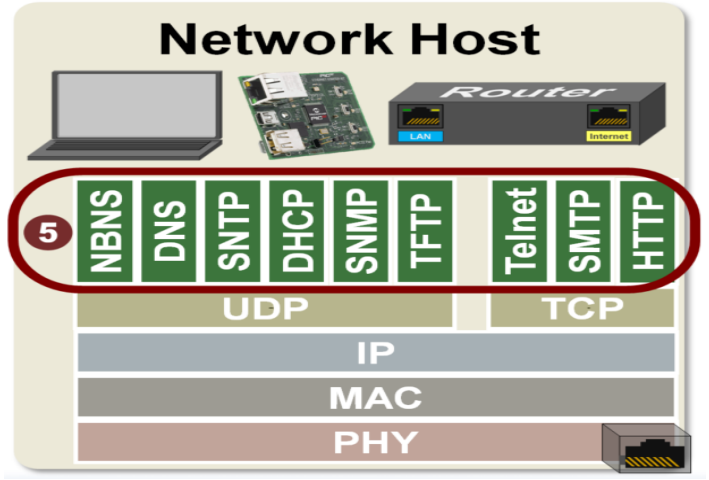


- The data part of an Ethernet packet can hold up to 1500 bytes. MAC-addresses (48bits) are 6 bytes wide each and the Number Of Bytes field is 2 byte wide. That gives the maximum size of an Ethernet frame to be 1514 bytes.

Application Layer

Layer name	TCP/IP protocols			
Application	HTTP	FTP	DHCP	TFTP
	SMTP	POP3	DNS	SNMP
Transport	TCP		UDP	
Internetwork	ICMP	ARP	IPsec	
	IPv4 and IPv6			
Network access	Ethernet, token ring, FDDI, WAN technologies			

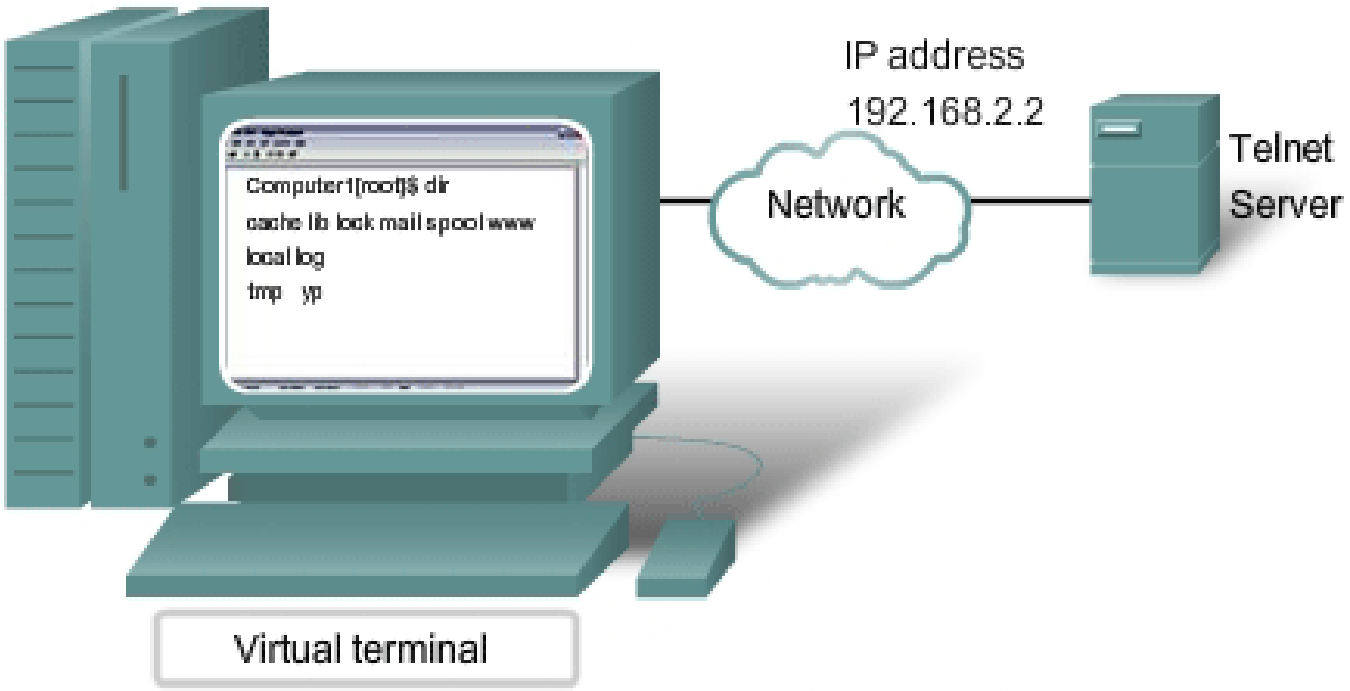
Figure 5-1 The TCP/IP layered architecture



Remote Desktop Protocol

- Remote Desktop Protocol (RDP) is used to access a Windows computer remotely by using the Windows GUI
  - Used to run Windows applications remotely and network administrators use it to manage Windows workstations and servers remotely

Telnet



Telnet provides a way to use a computer, connected via the network, to access a network device as if the keyboard and monitor were directly connected to the device.

Telnet and SSH

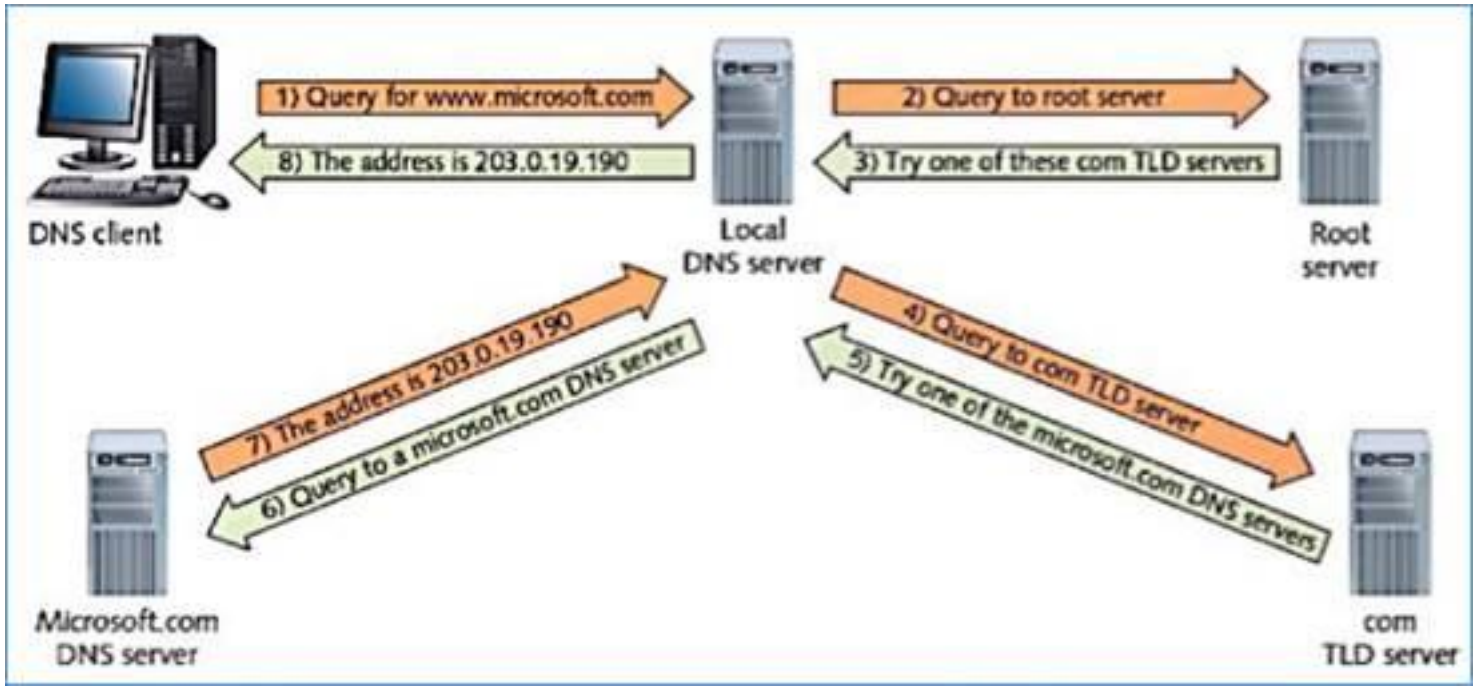
- Telnet and Secure Shell (SSH)
  - Used to connect to a device across a network via a command-line interface
  - Example: use to connect to a managed switch or router
- Telnet uses TCP port 23
  - Is not a secure protocol
- SSH uses TCP port 22
  - Provides an encrypted channel between the client and server

Domain Name System

- DNS is a name-to-address resolution protocol that keeps a list of computer names and their IP addresses
- Using DNS a user can use a computer's name instead of using it's IP address
- Example:
  - When you enter www.cengage.com in your Web browser, the DNS Client service contacts the DNS server specified in your OS's IP configuration and requests that the name be resolved to an IP address
  - Once the IP address for the website is returned, your computer can contact Web server to request a Web page
- DNS uses UDP because DNS messages usually consist of a single packet of data

DNS Explained

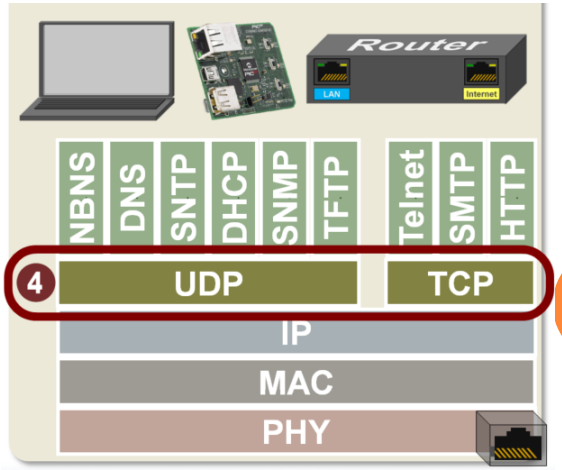
DNS Server



TCP
Transport-Layer Protocols

Layer name	TCP/IP protocols			
Application	HTTP	FTP	DHCP	TFTP
	SMTP	POP3	DNS	SNMP
Transport	TCP		UDP	
Internetwork	ICMP	ARP	IPsec	
	IPv4 and IPv6			
Network access	Ethernet, token ring, FDDI, WAN technologies			

Figure 5-1 The TCP/IP layered architecture



Transport-Layer Protocols

- Transport-layer protocols are used with most Application-layer protocols because they:
  - Supply a header field to identify the Application layer
  - Provide reliability and flow control for applications that typically transfer a large amount of data

Dst Port: nnnn Src Port: nnnn	Application Data For example - HTTP Request GET HTTP 1.1 www,course.com
TCP/UDP Header	Applicaation Data
Server Port (Well Known): 80 - Web Server (HTTP) 25 - SMTP 110 - POP3 143 - IMAP 25 - DNS 69 - DHCP 20/21 - FTP 22 - SSH 23 - Telnet **Client Port: Random**	

TCP Header

Offset (bits)	0–15			16–31
0	Source			Destination
32	Sequence Number			
64	Acknowledgement Number			
96	Data Offset	Reserved	Flags	Window
128	Checksum			Urgent
160	Options			

Role of the Transport Layer

- Transport layer has two protocols:
  - Transmission Control Protocol (TCP)
    - Connection oriented and designed for reliable transfer of information in complex internetworks
  - User Datagram Protocol (UDP)
    - Connectionless and designed for efficient communication of generally small amounts of data
  - TCP vs UDP
  - Both:
    - Work with segments or datagrams
    - Provide a means to identify the source and destination applications involved in a communication
    - Protect data with a checksum

Working with Segments and Datagrams

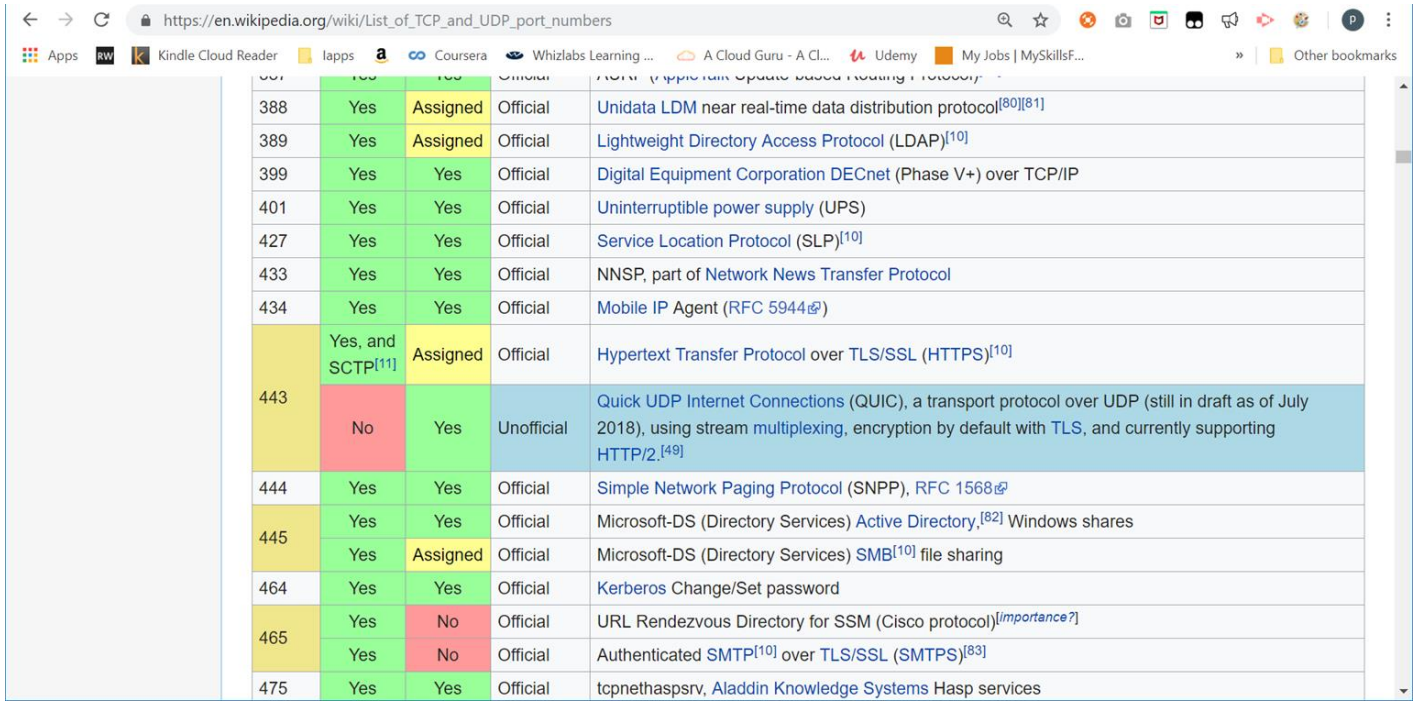
- Transport-layer protocols work with units of data called segments (TCP) or datagrams (UDP)
- Both TCP and UDP add a header to data
- The Transport-layer protocol then passes the segment to the Internetwork protocol (IP)
- With incoming data, the Transport-layer receives the segment from the Internetwork protocol, processes it, de-encapsulates it and sends the resulting data up to the Application layer

Identifying Source and Destination Applications

- How do computers keep track of incoming data when a Web browser, email application, chat and a word processing program are all running at the same time?
- TCP and UDP use **port numbers** to specify the source and destination Application-layer protocols
  - Port numbers are 16-bit values assigned to specific applications running on a computer or network device

Decimal	80
Binary	0000 0000 0101 0000
Hex	50

Port number 443



388	Yes	Assigned	Official	Unidata LDM near real-time data distribution protocol[80][81]
389	Yes	Assigned	Official	Lightweight Directory Access Protocol (LDAP)[10]
399	Yes	Yes	Official	Digital Equipment Corporation DECnet (Phase V+) over TCP/IP
401	Yes	Yes	Official	Uninterruptible power supply (UPS)
427	Yes	Yes	Official	Service Location Protocol (SLP)[10]
433	Yes	Yes	Official	NNSP, part of Network News Transfer Protocol
434	Yes	Yes	Official	Mobile IP Agent (RFC 5944)
443	Yes, and SCTP[11]	Assigned	Official	Hypertext Transfer Protocol over TLS/SSL (HTTPS)[10]
	No	Yes	Unofficial	Quick UDP Internet Connections (QUIC), a transport protocol over UDP (still in draft as of July 2018), using stream multiplexing, encryption by default with TLS, and currently supporting HTTP/2.[49]
444	Yes	Yes	Official	Simple Network Paging Protocol (SNPP), RFC 1568
445	Yes	Yes	Official	Microsoft-DS (Directory Services) Active Directory,[82] Windows shares
	Yes	Assigned	Official	Microsoft-DS (Directory Services) SMB[10] file sharing
464	Yes	Yes	Official	Kerberos Change/Set password
465	Yes	No	Official	URL Rendezvous Directory for SSM (Cisco protocol)[*importance?*]
	Yes	No	Official	Authenticated SMTP[10] over TLS/SSL (SMTPS)[83]
475	Yes	Yes	Official	tcpnethaspsrv, Aladdin Knowledge Systems Hasp services

https://en.wikipedia.org/wiki/List_of_TCP_and_UDP_port_numbers

Decimal	443
Binary	0000 0001 1011 0001
Hex	01BB

Common port Numbers

Common port numbers [edit]

Main article: List of TCP and UDP port numbers

The Internet Assigned Numbers Authority (IANA) is responsible for the global coordination of the DNS Root, IP addressing, and other Internet protocol resources. This includes the registration of commonly used port numbers for well-known Internet services.

The port numbers are divided into three ranges: the *well-known ports*, the *registered ports*, and the *dynamic* or *private ports*.

The well-known ports (also known as *system ports*) are those from 0 through 1023. The requirements for new assignments in this range are stricter than for other registrations,[2] examples include:

- 20: File Transfer Protocol (FTP) Data Transfer
- 21: File Transfer Protocol (FTP) Command Control
- 22: Secure Shell (SSH) Secure Login
- 23: Telnet remote login service, unencrypted text messages
- 25: Simple Mail Transfer Protocol (SMTP) E-mail routing
- 53: Domain Name System (DNS) service
- 80: Hypertext Transfer Protocol (HTTP) used in the World Wide Web
- 110: Post Office Protocol (POP3)
- 119: Network News Transfer Protocol (NNTP)
- 123: Network Time Protocol (NTP)
- 143: Internet Message Access Protocol (IMAP) Management of digital mail
- 161: Simple Network Management Protocol (SNMP)
- 194: Internet Relay Chat (IRC)
- 443: HTTP Secure (HTTPS) HTTP over TLS/SSL

Protecting Data with a Checksum

- To protect data integrity, TCP and UDP provide a checksum similar to the CRC
- Intermediate devices don't recalculate the checksum in the Transport layer so if data corruption occurs during the transmission, the final receiving station detects the checksum error and discards the data

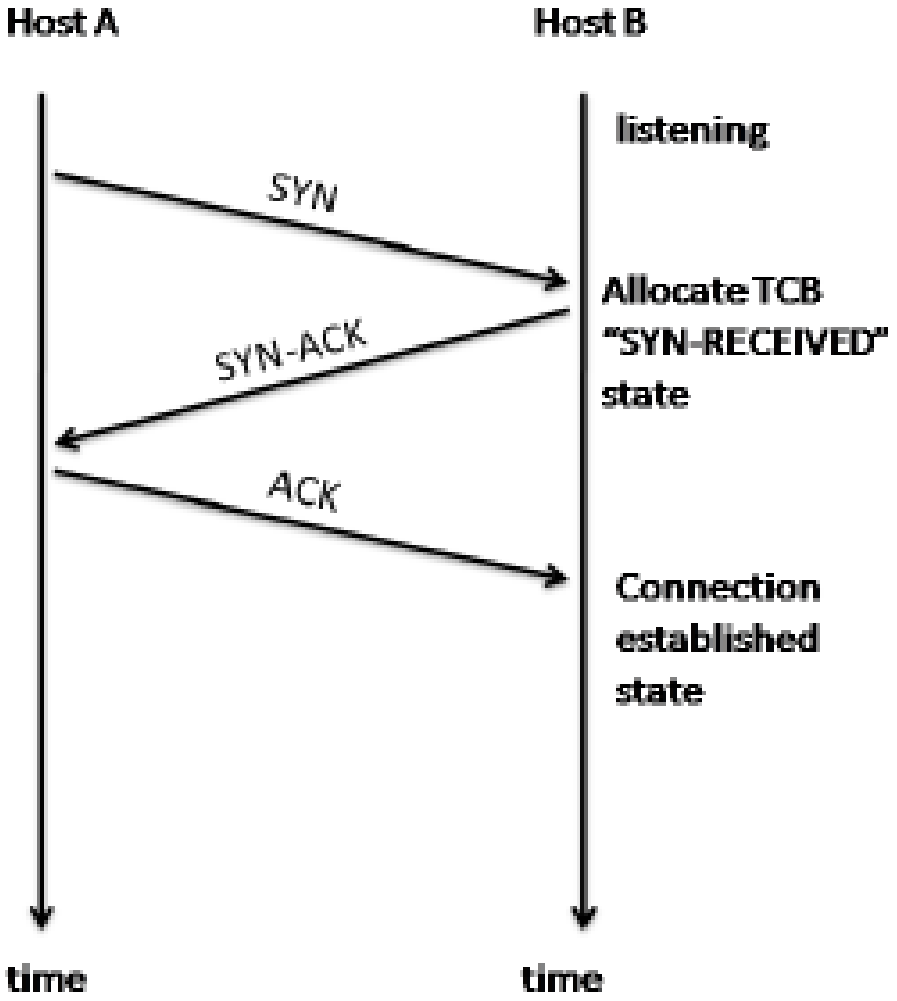
TCP: The Reliable Transport Layer

- If an application requires reliable data transfer, it uses TCP as the Transport-layer protocol
- How does TCP guarantee data delivery?
- TCP provides reliability by using these features:
  - Establishing a connection
  - Segmenting large chunks of data
  - Ensuring flow control with acknowledgements
- TCP is a connection-oriented protocol
  - It establishes a connection with the destination, data is transferred, and the connection is broken

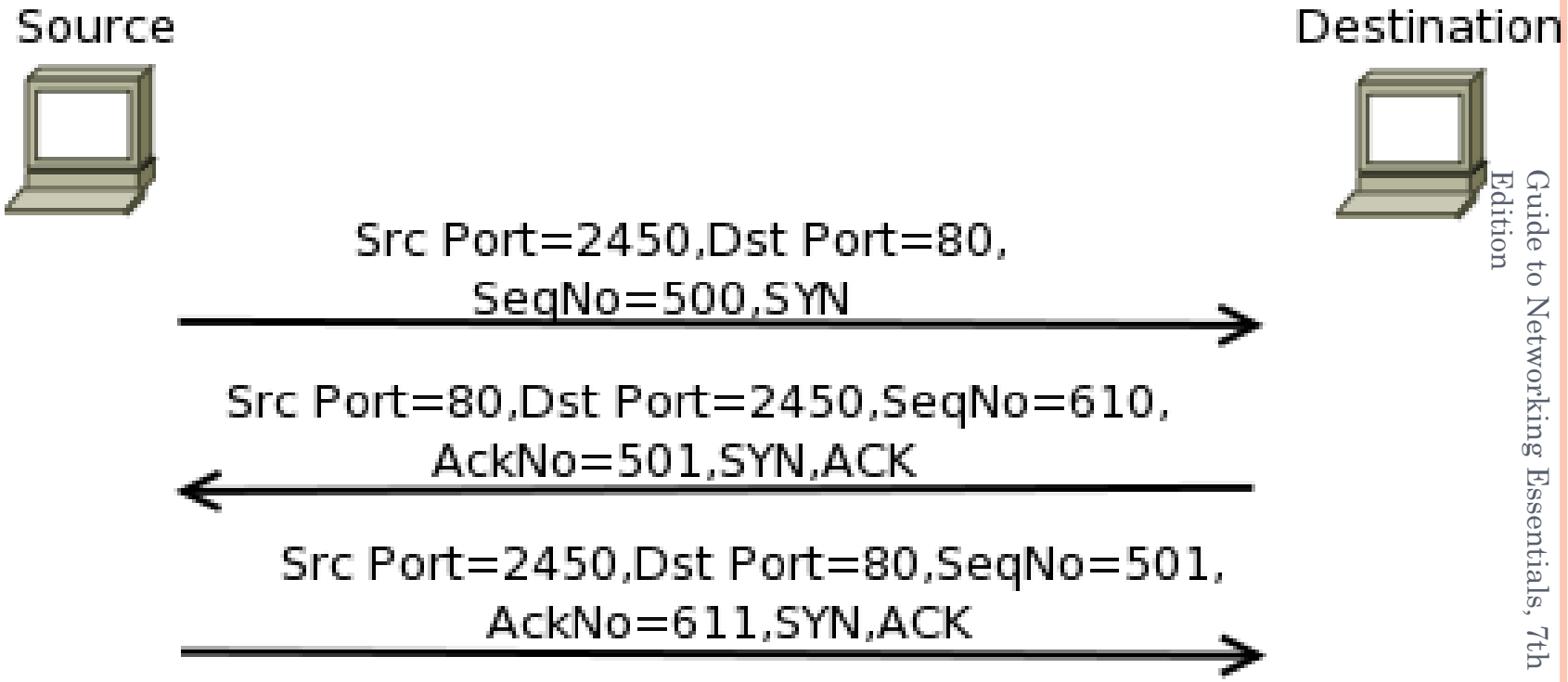
Establishing a Connection: The TCP Handshake

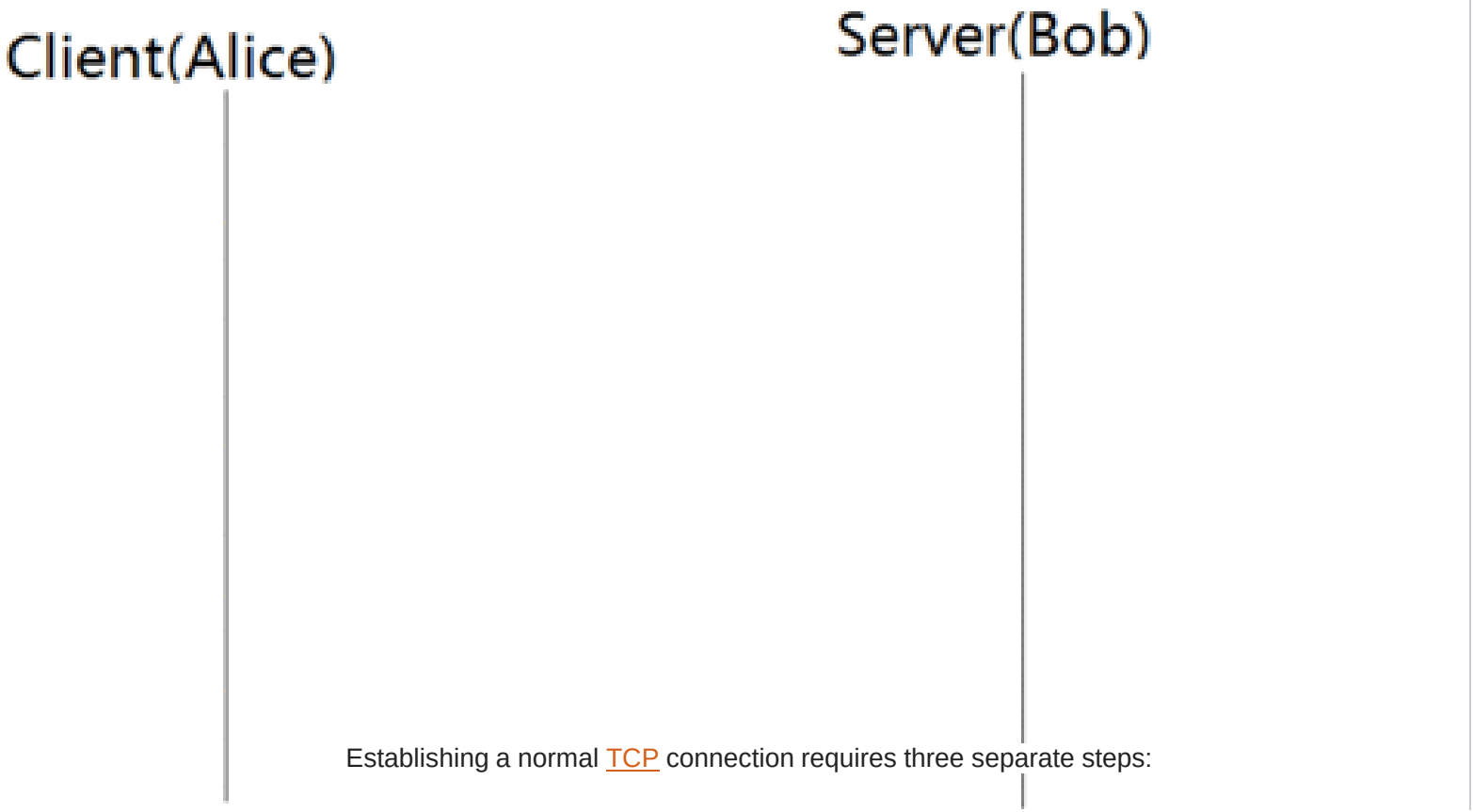
- A client sends a TCP synchronization (SYN) segment to the destination device, usually a server
  - A destination port is specified and a source port is assigned dynamically
- When the server receives the SYN segment, it responds by sending either an acknowledgement-synchronization (ACK-SYN) segment or a reset connection (RST) segment
  - RST is sent when the server refused the request to open the session
  - If an ACK-SYN is returned, the client completes the **three-way handshake** by sending an ACK segment back to the server

Establishing a Connection: The TCP Handshake



3 Way Hand shake Example





Establishing a normal TCP connection requires three separate steps:

1. The first host (Alice) sends the second host (Bob) a "synchronize" (SYN) message with its own sequence number which Bob receives.

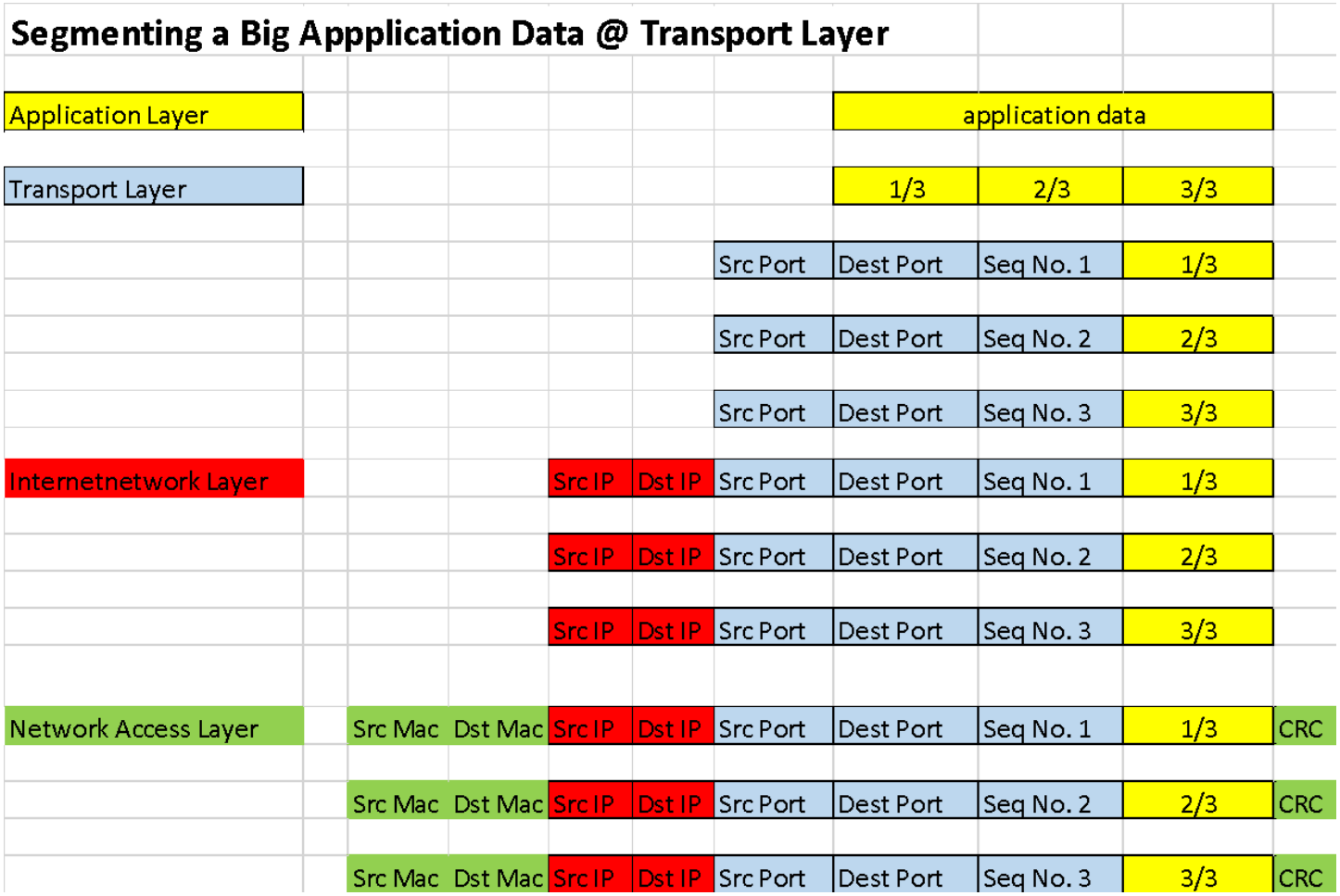
2. Bob replies with a synchronize-acknowledgment (SYN-ACK) message with its own sequence number and acknowledgement number, which Alice receives.

3. Alice replies with an acknowledgment (ACK) message with acknowledgement number , which Bob receives and to which he doesn't need to reply.

Segmenting Data

- When TCP receives data from the Application layer, the size might be too large to send in one piece
- TCP breaks the data into smaller segments (max frame sent by Ethernet is 1518 bytes)
- Each segment is labeled with a sequence number so that if segments arrive out of order they can be reassembled in the correct order

Segmentation



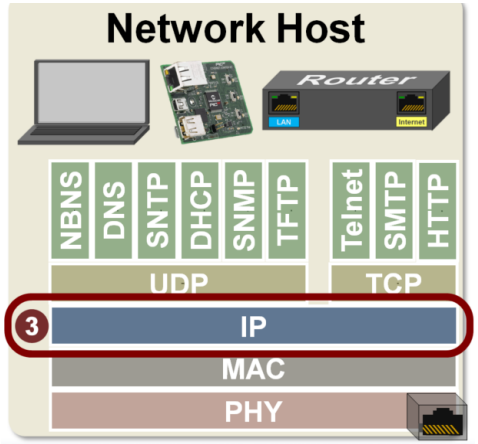
Ensuring Flow Control with Acknowledgements

- Flow control prevents a destination from becoming overwhelmed by data, resulting in dropped packets
- TCP establishes a maximum number of bytes, called the “window size”, that can be sent before the destination must acknowledge the receipt of data
- If no acknowledgement is received within a specified period of time, the sending station will retransmit from the point at which an acknowledgement was last received

IP

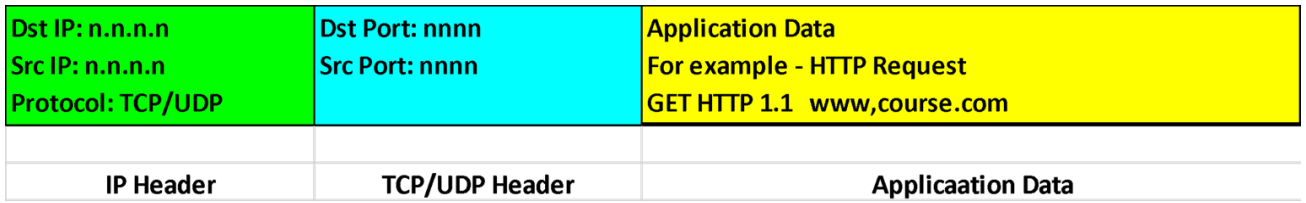
Layer name	TCP/IP protocols			
Application	HTTP	FTP	DHCP	TFTP
	SMTP	POP3	DNS	SNMP
Transport	TCP		UDP	
Internetwork	ICMP	ARP	IPsec	
	IPv4 and IPv6			
Network access	Ethernet, token ring, FDDI, WAN technologies			

Figure 5-1 The TCP/IP layered architecture

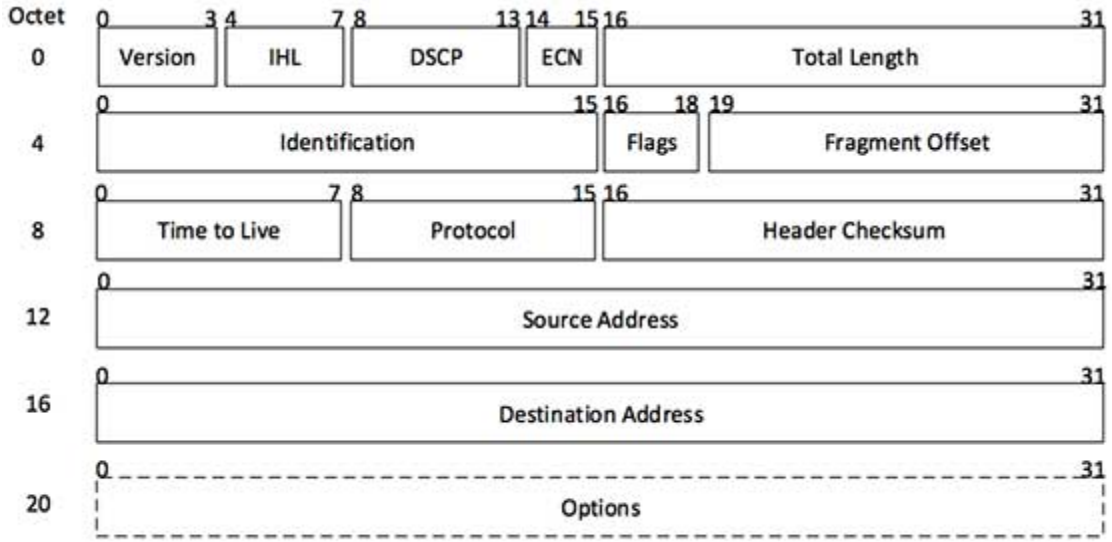


Internetwork-Layer Protocols

- The Internetwork layer is where administrators usually do the most network configuration
- Where the IP protocol operates and is the heart of the TCP/IP protocol suite
- Responsible for four main tasks:
  - Defines and verifies IP addresses
  - Routes packets through an internetwork
  - Resolves MAC addresses from IP addresses
  - Delivers packets efficiently



IP Header



[Image: IP Header]

Source: https://www.tutorialspoint.com/ipv4/ipv4_packet_structure.htm

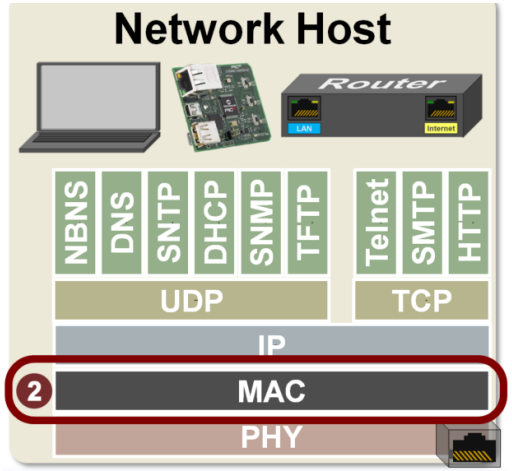
Data link layer

Layer name	TCP/IP protocols			
Application	HTTP	FTP	DHCP	TFTP
	SMTP	POP3	DNS	SNMP
Transport	TCP		UDP	
Internetwork	ICMP	ARP	IPsec	
	IPv4 and IPv6			
Network access	Ethernet, token ring, FDDI, WAN technologies			

Figure 5-1 The TCP/IP layered architecture

Layer 2 is the **Data Link** layer. This layer uses a **Media Access Controller (MAC)** to generate the frames that will be transmitted. As the name suggests, the MAC controls the physical transmission media.

When transmitting data, this layer adds a header containing the source and destination MAC addresses to the packet received from the Network layer (layer 3). The frame it creates will then be forwarded to the Physical layer.



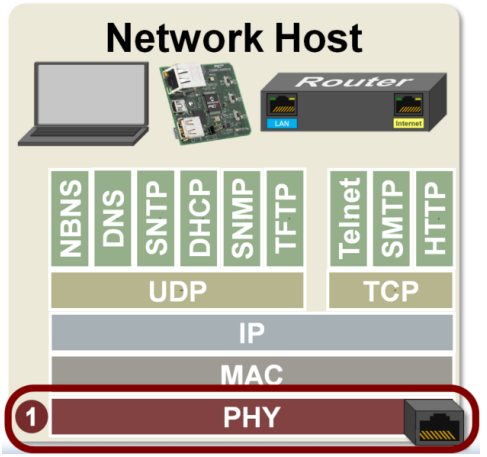
Physical Layer

Layer name	TCP/IP protocols			
Application	HTTP	FTP	DHCP	TFTP
	SMTP	POP3	DNS	SNMP
Transport	TCP		UDP	
Internetwork	ICMP	ARP	IPsec	
	IPv4 and IPv6			
Network access	Ethernet, token ring, FDDI, WAN technologies			

Figure 5-1 The TCP/IP layered architecture

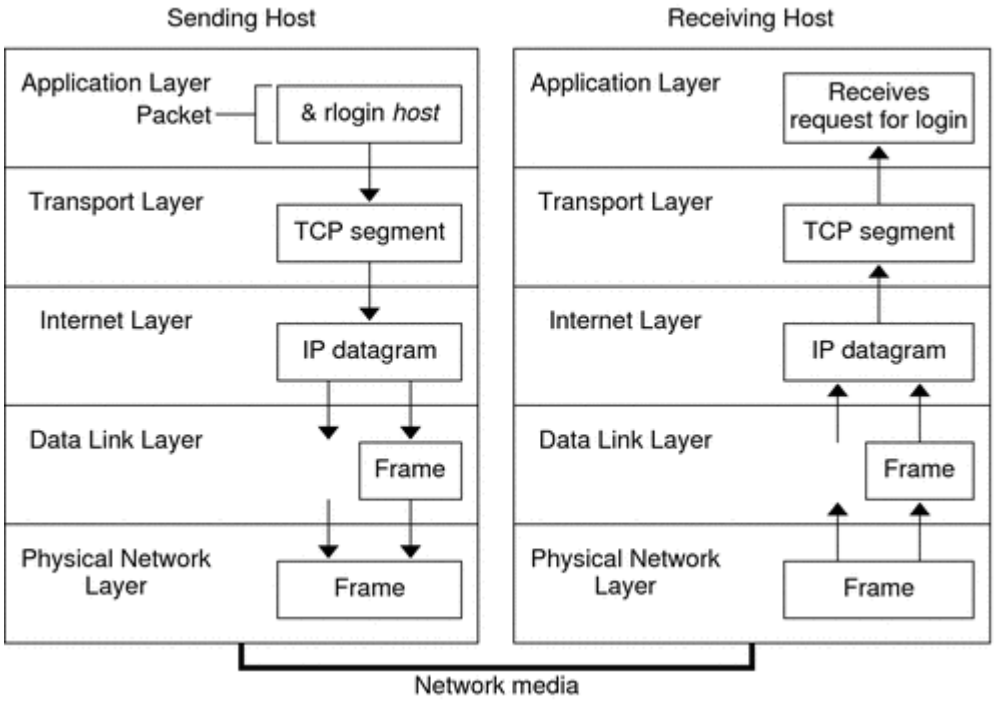
Layer 1 is the **Physical** layer. It sends and receives signals on the physical wire or antenna to transmit the bits found in frames.

There is a PHY found at the end of every network interface (e.g. end of wire or antenna).



Data Encapsulation

When a protocol on the sending system adds data to the packet header, the process is called data encapsulation.



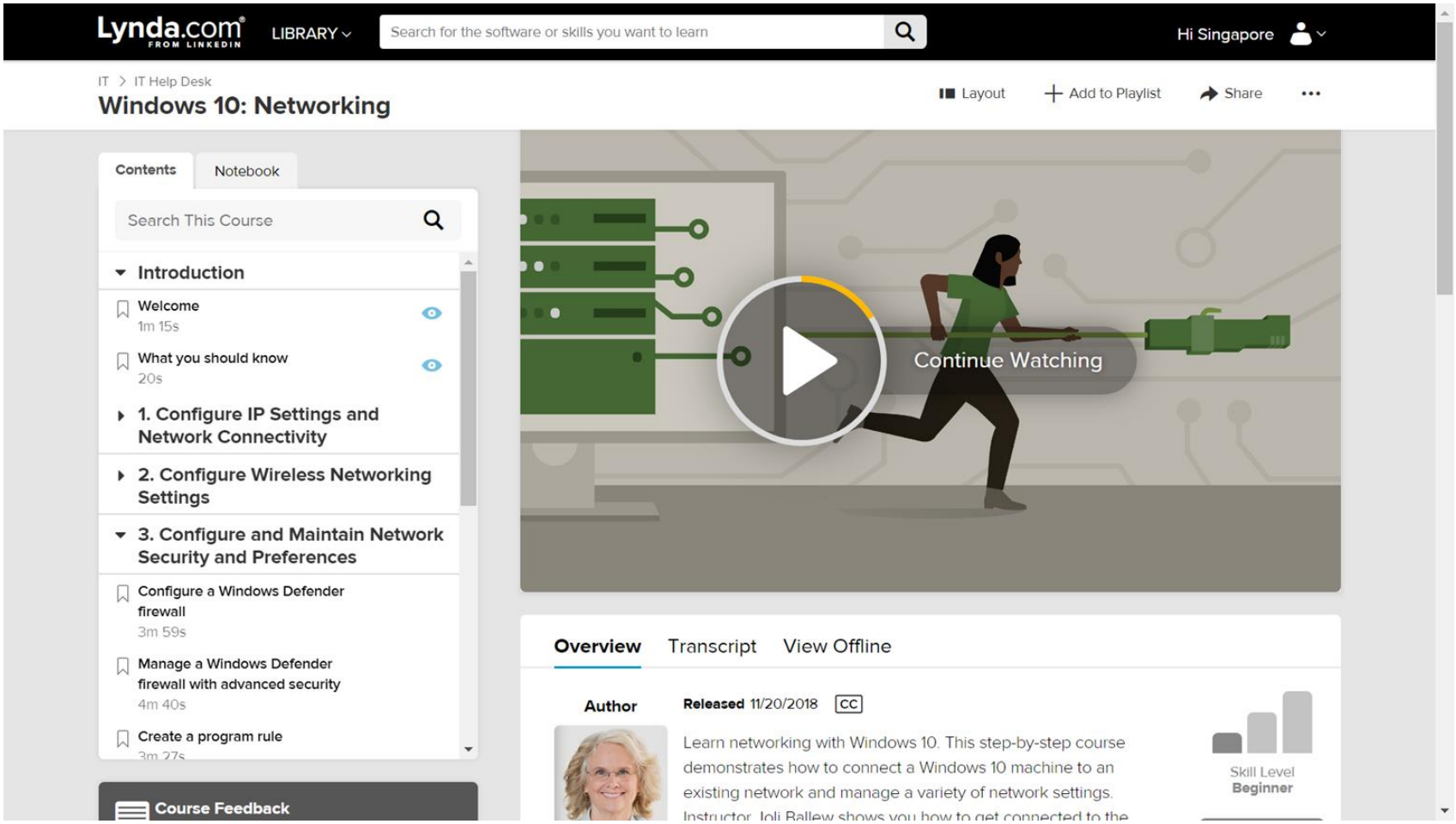
source: https://docs.oracle.com/cd/E19120-01/open.solaris/819-3000/ipov-32/index.html

Summary

- TCP/IP is the main protocol suite used in networks
- The Application layer consists of protocols such as HTTP and DNS and provides an interface for applications to access network services
- The Transport layer provides reliability and works with segments (TCP) and datagrams (UDP)
- The Internetwork layer is where most network configuration occurs and is composed of IP, ICMP, and ARP

Summary

- The Network access layer is composed of network technologies, such as Ethernet and WAN technologies


Lynda.com FROM LINKEDIN
LIBRARY
Search for the software or skills you want to learn
Hi Singapore
IT > IT Help Desk
Windows 10: Networking
Layout
Add to Playlist
Share
Contents
Notebook
Search This Course
Introduction
Welcome
1m 15s
What you should know
20s
1. Configure IP Settings and Network Connectivity
2. Configure Wireless Networking Settings
3. Configure and Maintain Network Security and Preferences
Configure a Windows Defender firewall
3m 59s
Manage a Windows Defender firewall with advanced security
4m 40s
Create a program rule
3m 27s
Course Feedback
Continue Watching
Overview
Transcript
View Offline
Author
Released 11/20/2018
CC
Learn networking with Windows 10. This step-by-step course demonstrates how to connect a Windows 10 machine to an existing network and manage a variety of network settings. Instructor Joli Ballew shows you how to get connected to the
Skill Level
Beginner

Learn More:
Introduction to TCP/IP Video