Chapter 2

Applications and Layered Architectures

Layering

- Decomposition of overall networking into modules
- Simplifies design
- Risk of ignoring implementation decisions of other layers, particularly layer N +/- 2
Retrieving a Web Page

- Assume computer has an IP address and a defined DNS server and gateway
- Task is to open a web page at:
  - www.comm.utoronto.ca

Client-Server Layout

HTTP client  ─────> HTTP server

Request

<──── Response
How many packets on an Ethernet LAN will it take to retrieve this page?

- 2 images,
- 3 frames,
- ~17 links
Answer: 100 Ethernet Frames

HTTP client

Ephemeral
Port #

TCP

GET 80, #

# 80 STATUS

HTTP server

Port 80

TCP

Port 80
Inter-entity exchange

Layer n Protocol Data Unit (n-PDU)

SDU | H

H | SDU

Figure 2.3

Layer Services

n-SAP

n-SDU

n+1 entity

n+1 entity

n-SDU

n-SAP

n entity

n entity

n-SAP

n-SDU | H

H | n-SDU

n-PDU

Figure 2.4
Segmentation and Blocking

(a) Segmentation

\[ \text{n-SDU} \]
\[ \text{n-PDU} \]
\[ \text{n-PDU} \]

(b) Blocking

\[ \text{n-SDU} \]
\[ \text{n-SDU} \]
\[ \text{n-SDU} \]

Reassembly

\[ \text{n-SDU} \]
\[ \text{n-PDU} \]
\[ \text{n-PDU} \]

Unblocking

\[ \text{n-SDU} \]
\[ \text{n-SDU} \]
\[ \text{n-SDU} \]

OSI Reference Model

What Kind of Devices Are These?
An Example Internetwork

- Single network means single routing protocol
- Is this Connectionless or Connection-Oriented?
- What difference would it make?

Networks of Networks
I.i.e. Internetworks or Subnets

- Multiple networks may involve multiple routing protocols. Internetworking protocols must be used to ensure proper exchange
Encapsulation

![Encapsulation Diagram](image)

TCP/IP Network Architecture

Maps to top 3 OSI layers

- HTTP is really Application layer of TCP/IP stack. Layering not strictly enforced in TCP/IP
- What are the attributes of the Internet Layer?
Application & Transport Layers

- Maps to top 3 layers of OSI stack
- Layering not strictly enforced in TCP/IP -- Application may sit directly on Network layer
- Transport provides two basic services
  - TCP and UDP
  - TCP provides flow control and error recovery, but requires explicit set-up and tear-down - a type of connection-oriented service
  - UDP is best-efforts

![Diagram of Application & Transport Layers]

Internet and Network Interface Layers

![Diagram of Internet and Network Interface Layers]
TCP/IP Protocol Graph

HTTP → SMTP → DNS → RTP

TCP → IP

IP → Network Interface 1 → Network Interface 2 → Network Interface 3

Small Internet

Server → Router → PC

Ethernet → PPP

HTTP → TCP → IP → Net Interface

Ethernet → PPP

HTTP → TCP → IP → Net Interface
IP Encapsulation

- IP Header
- Ethernet Header
  - Source and destination physical addresses
  - Network protocol type

PDU Encapsulation

- HTTP Request
- TCP Header
  - Source and destination port numbers
- IP Header
  - Source and destination IP addresses
  - Transport protocol type
- Ethernet Header
  - Source and destination physical addresses
  - Network protocol type

Frame Check Sequence
Berkeley API Socket

Application 1

socket interface

user

kernel

Socket

Underlying communication Protocols

Communications network

Application 2

socket interface

user

kernel

Socket

Underlying communication Protocols

Connection-Oriented Socket

Create TCP Socket

Server

socket()

bind()

listen()

accept()

blocks until server receives a connect request from client

connect negotiation

read()

write()

close()

Client

socket ()

connect ()

write()

read()

close ()

Active Open

Figure 2.16

Figure 2.17
Connection-Oriented Socket

*Server*
- `socket()`
- `bind()`
- `recvfrom()`
  - blocks until server receives data from client
- `sendto()`
- `close()`

*Client*
- `socket()`
- `bind()`
- `sendto()`
- `recvfrom()`
- `close()`

---

Other Useful C Commands

- `gethostbyname` - takes domain name, returns address
- `gethostbyaddr` - takes IP address, returns hostname
- `inet_ntoa` - takes 32-bit address in network byte order and returns dotted decimal
- `inet_addr` - takes dotted-decimal and returns 32-bit network byte address
FTP Protocol Transfer

User PI
User DTP
User FTP

Server PI
Server DTP
Server FTP

Control Connection
Data Connection

Pi = Protocol interpreter
DTP = Data transfer process

Traceroute and Ping

- Useful for “discovering” a network
- Traceroute utility explores by issuing UDP datagram with TTL=1, invalid port
- First router sets TTL=0, discards, returns ICMP Time Exceeded message
- Incrementally increasing TTL discovers all paths in route
- See www.visualroute.com