CprE 530

Lecture 14

Topics

- TCP Layer
  - Responsible for reliable end-to-end transfer of application data.

TCP Services

Multiplexing:

- A process within a host using TCP service is identified with a **port**. A port, when concatenated with an internet address, forms a **Socket**, which is unique throughout the internet. Service provided by TCP is provided by means of a logical connection between a pair of sockets.
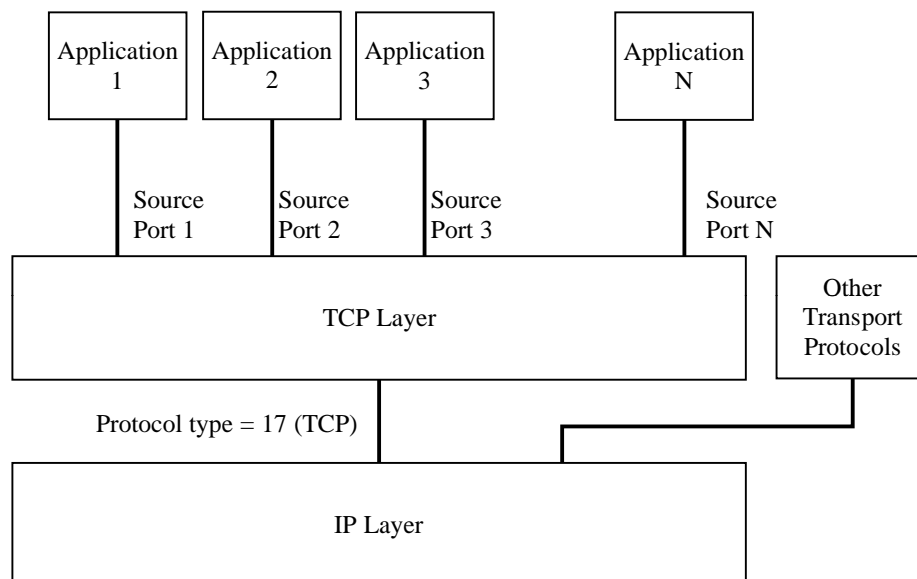
Multiplexing service

| Application 1 | Application 2 | Application 3 | | Application N |

Source Port 1   Source Port 2   Source Port 3   Source Port N

TCP Layer

Other Transport Protocols

Protocol type = 17 (TCP)

IP Layer

Figure 7.1 TCP Multiplexing

TCP port numbers

5	RJE	68	Bootstrap Protocol Client
7	echo	69	Trivial FTP
9	Discard	75	any private dialout service
11	Active Users	77	any Private RJE service
13	daytime	79	FINGER
15	Who is up	101	NIC host name server
17	Quote of the day	102	ISO-TSAP
19	Character Generator	103	X.400
20	FTP (default data)	104	X.400-SND
21	FTP (control)	105	CSnet Name server
23	TELNET	109	Post Office Protocol Ver 2
25	SMTP	113	Authentication Service
37	Time	115	Simple FTP
42	Host name service	119	NNTP
53	Domain name server	123	NTP
67	BOOTP	161	SNMP agent
		162	SNMP management station

TCP Connection Management

Consists of three services:

- Connection Establishment: Allow two TCP users to setup a logical connection between their respective sockets. A connection may be setup if:
- No connection between the two sockets currently exists. From a given socket, it is possible to simultaneously maintain more than one connection, but only one connection to any specific remote socket at a time is permitted.
  - Internal TCP resources are sufficient.
  - Both users have agreed to the connection.
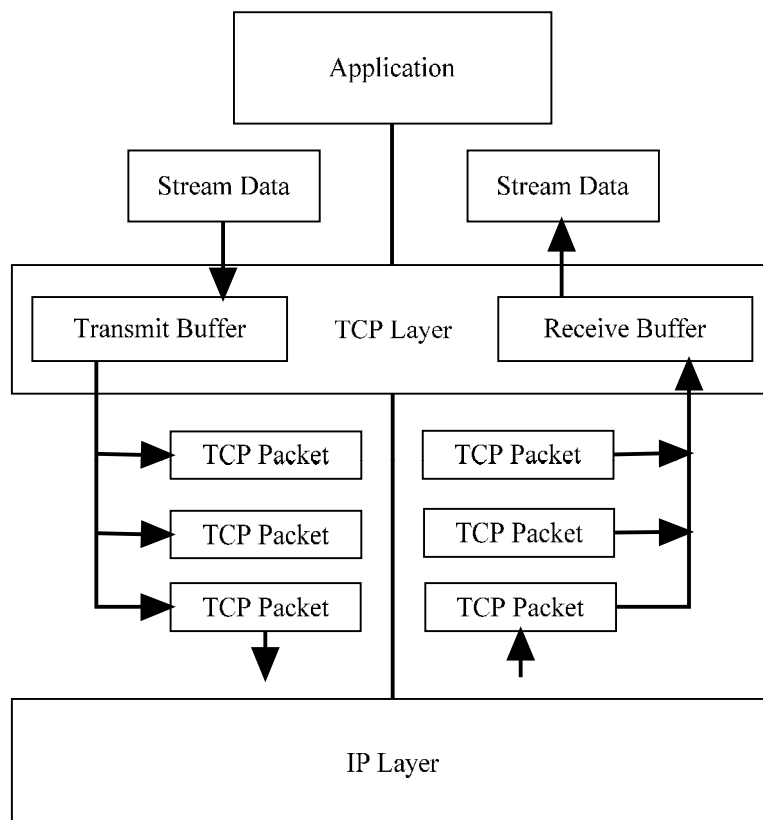
TCP Connection Management

- Connection Maintenance service provides for the exchange of data between the two sockets and supports the data transport (described in the next slide).
- Connection Termination may be either abrupt or graceful. With abrupt termination, data in transit may be lost. A graceful termination prevents either side from shutting down until all data have been received.

TCP Data Transport

- Full Duplex: Both users may transmit at once.

- Timely: The user may request timely delivery of data by associating a timeout with data submitted for transmission. If TCP detects a timeout the connection is abruptly terminated.
- Ordered: TCP is stream oriented. TCP guaranteed that the stream of data presented by one user to TCP will be delivered in the same order to the destination user.
- Labeled: TCP establishes a connection only if the security designation provided by both users match.
- Flow Control: Used to prevent internal TCP congestion
- Error Control: TCP uses a simple checksum.

TCP

- **Stream Orientation -** When two application processes transfer large volumes of data, we can think of the as a stream of bits divided into 8-bit bytes The stream service on the destination passes the same sequence of octets to the receiver that the sender passed to the source machine. Data are not treated as packets but as a stream of data that is passed to the transport entity. The transport entity will divide the data into packets for transmission to the destination. The destination transport entity will pass the data to the user as a stream.

TCP Stream

```
              ┌─────────────────┐
              │   Application   │
              └─────────────────┘
     ┌──────────────┐      ┌──────────────┐
     │ Stream Data  │      │ Stream Data  │
     └──────────────┘      └──────────────┘
 ┌────────────────────────────────────────────┐
 │ ┌───────────────┐  TCP Layer ┌────────────┐ │
 │ │Transmit Buffer│            │Receive Buffer│ │
 │ └───────────────┘            └────────────┘ │
 └────────────────────────────────────────────┘
        ┌────────────┐    ┌────────────┐
        │ TCP Packet │    │ TCP Packet │
        └────────────┘    └────────────┘
        ┌────────────┐    ┌────────────┐
        │ TCP Packet │    │ TCP Packet │
        └────────────┘    └────────────┘
        ┌────────────┐    ┌────────────┐
        │ TCP Packet │    │ TCP Packet │
        └────────────┘    └────────────┘
 ┌────────────────────────────────────────────┐
 │                  IP Layer                   │
 └────────────────────────────────────────────┘
```

Figure 7.2 TCP Stream Service

TCP Special Capabilities

TCP supports two special capabilities associated with the transfer of data

- Data Stream Push: Used to force the delivery of all data waiting to be sent.

- Urgent Data Signaling: Provides a means of informing the destination TCP user that urgent data is in the incoming data stream.

TCP Error Reporting

- TCP will report service failure stemming from catastrophic conditions

TCP Services

- Unspecified Passive open
- Fully Specified Passive Open
- Active Open
- Active Open with data
- Send
- Deliver
- Allocate
- Close
- Abort
- Terminate
- Error

TCP Protocol

Connection Establishment:

- TCP uses a three handshake for connection establishment. We will see TCP defines only one packet format that contains flags to indicate what type of packet it is. The connection packets have the SYN flag set.
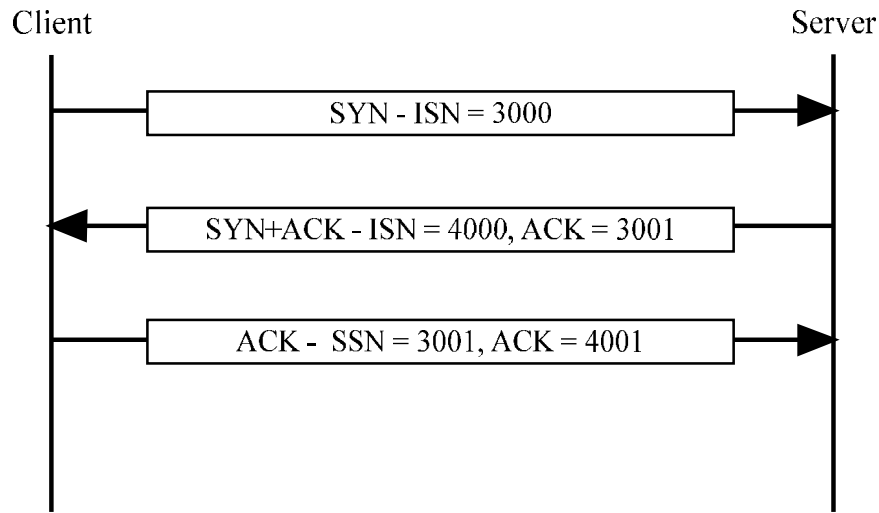
TCP 3-way Handshake

Client   Server

| SYN - ISN = 3000 |

| SYN+ACK - ISN = 4000, ACK = 3001 |

| ACK - SSN = 3001, ACK = 4001 |

Figure 7.3 TCP Connection Establishment

TCP Protocol

Data Transfer:

- Sequence numbers are used for data transfer. The sequence numbers represent the number of bytes not the number of packets. Flow control is handled by using a credit allocation scheme as describe earlier.

TCP Data Transfer

Client   Server

| SSN = 3000, ACK = 500 (1000 bytes) |
| SN1 = 4000, ACK = 500 (1000 bytes) |
| SN2 = 5000, ACK = 500 (500 bytes) |
| SSN = 500, ACK = 4500 (750 bytes) |
| SS3 = 5500, ACK = 1250 (250 bytes) |
| SSN = 1250, ACK = 5000 (0 bytes) |

3000   4000   5000   5500

Packet 1	Packet 2	Packet 3	Packet 4
1000 bytes	1000 bytes	500 bytes	250 bytes

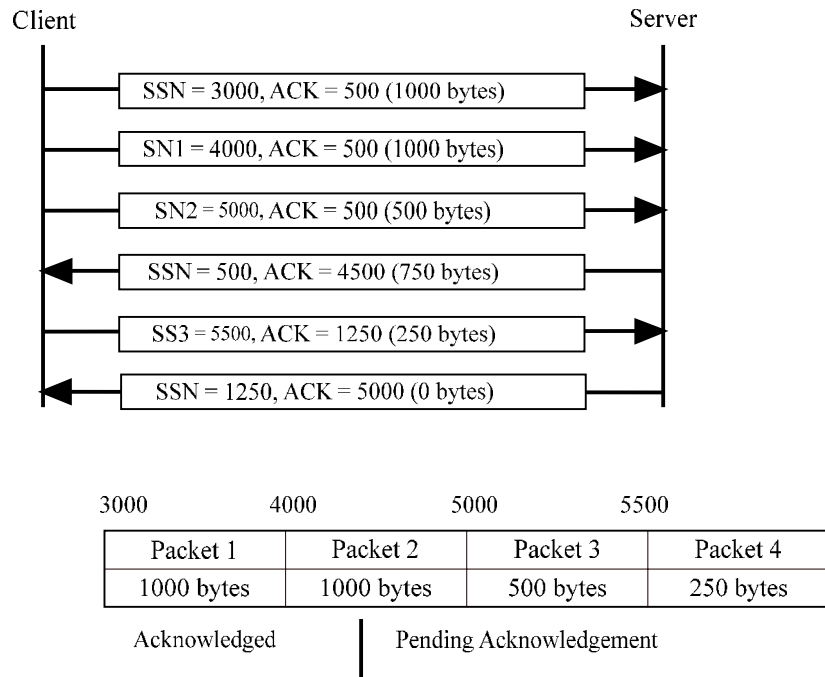Acknowledged   Pending Acknowledgement

Figure 7.4 TCP Data Transfer

TCP Connection Termination

Connection Termination:

- The connection is terminated by sending a packet with the FIN flag set. This packet contains the number of the last packet sent.
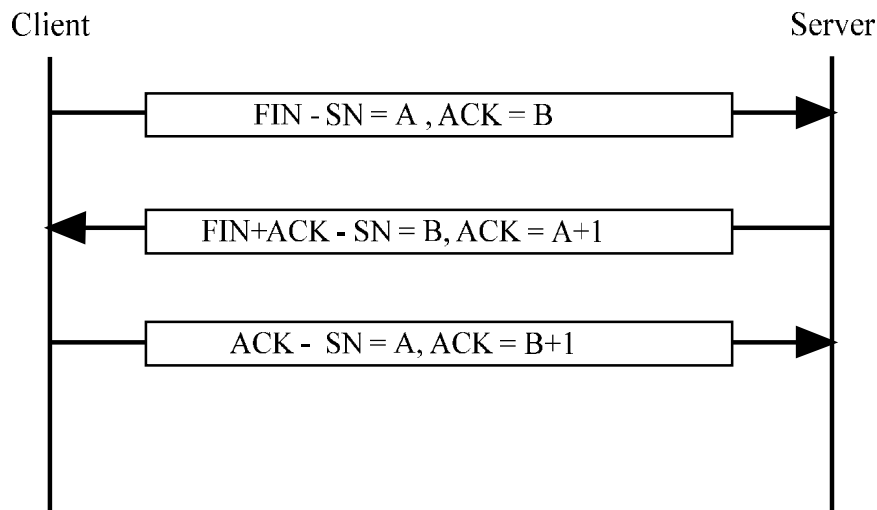
TCP Connection termination

Client   Server

| FIN - SN = A , ACK = B |

| FIN+ACK - SN = B, ACK = A+1 |

| ACK - SN = A, ACK = B+1 |

Figure 7.5 TCP Graceful Termination

TCP Header Format

Source Port	Destination Port
Sequence Number	
Acknowledgement Number	

Hdr-Len	Reserved	Flags	Window Size
Checksum			Urgent Pointer
Options			

Flags

URG	ACK	PSH	RST	SYN	FIN

Flag	Function
URG	Packet contains urgent data
ACK	Acknowledgment number is valid
PSH	Data should be pushed to the application
RST	Reset Packet
SYN	Synchronize packet
FIN	Finish packet

Figure 7.6 TCP Header Format