Distributed Systems

COMP90015 2021 Semester 1
Tutorial 02

Today’s Agenda

- Questions/ discussion on Sockets, UDP, TCP
- Code demonstration
    - Sockets /client server – UDP and TCP

Sockets

Q1. Briefly discuss three aspects of the Socket interface.

Sockets

Q1. Briefly discuss three aspects of the Socket interface.

- Bound to a local port.
  - Socket address = IP + Port number
- Sockets are used for communication between two networked processes.
  - sending and receiving data
- Each socket is associated with a protocol (UDP or TCP).

- Acts as programming interface to application code and transport layer
- Socket handle is mostly like file handle

Q2. Briefly explain three possible failures that can happen when using UDP for communication.

Q2. Briefly explain three possible failures that can happen when using UDP for communication.

- Data Corruption.
- Omission failures (No guaranteed delivery).
- Order.

Q3. Briefly explain aspects of TCP that address issues not addressed by UDP.

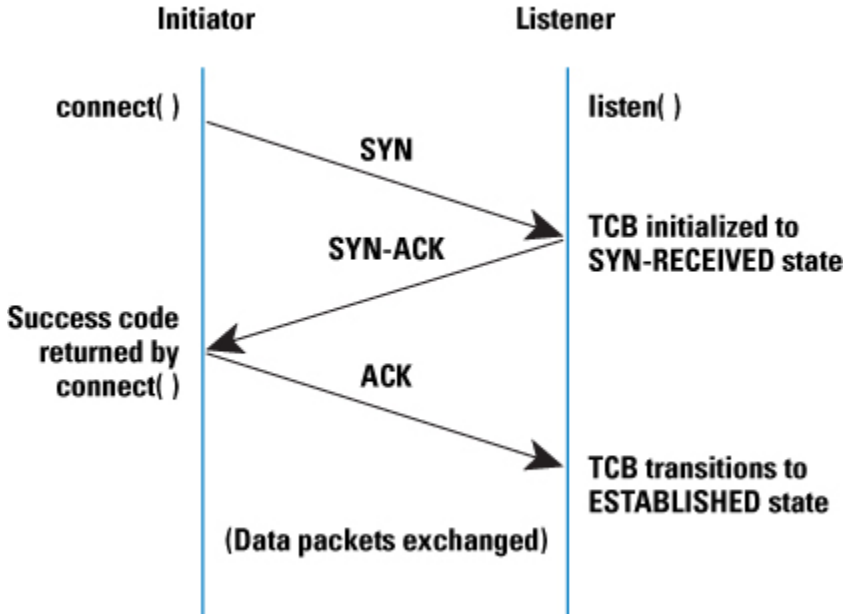
Q3. Briefly explain aspects of TCP that address issues not addressed by UDP.

- **Connection oriented:** The communicating processes establish a connection before communicating. The connection involves a connect request from the client to the server followed by an accept request from the server to the client.

- **Message sizes:** There is no limit on data size applications can use.
- **Lost messages:** TCP uses an acknowledgment scheme unlike UDP. If acknowledgments are not received the messages are retransmitted.
- **Flow control:** TCP protocol attempts to match the speed of the process that reads the message and writes to the stream.
- **Message duplication or ordering:** Message identifiers are associated with IP packets to enable the recipient to detect and reject duplicates and reorder messages in case messages arrive out of order.

Q4. List the steps involved at the client and at the server to establish a TCP stream socket connection.

Q4. List the steps involved at the client and at the server to establish a TCP stream socket connection.



UDP vs TCP

UDP vs TCP

UDP: User Datagram Protocol

- Provides a message passing abstraction.
- Is the simplest form of connectionless Interprocess Communication (IPC).
- Transmits a single message (called as datagram) to the receiving process.
- Example use cases- Video stream (real time applications), DNS, NTP (query response)

TCP: Transmission Control Protocol

- Provides an abstraction for a two-way stream (called as packets).
- Streams do not have message boundaries.
- Stream provide the basis for producer/consumer communication.
- Data sent by the producer are queued until the consumer is ready to receive them.
- The consumer must wait when no data is available.
- Example use cases – http, ftp

Client - Server Demo

1. UDP
    - Client and Server
2. TCP
    - Interactive Client and Server