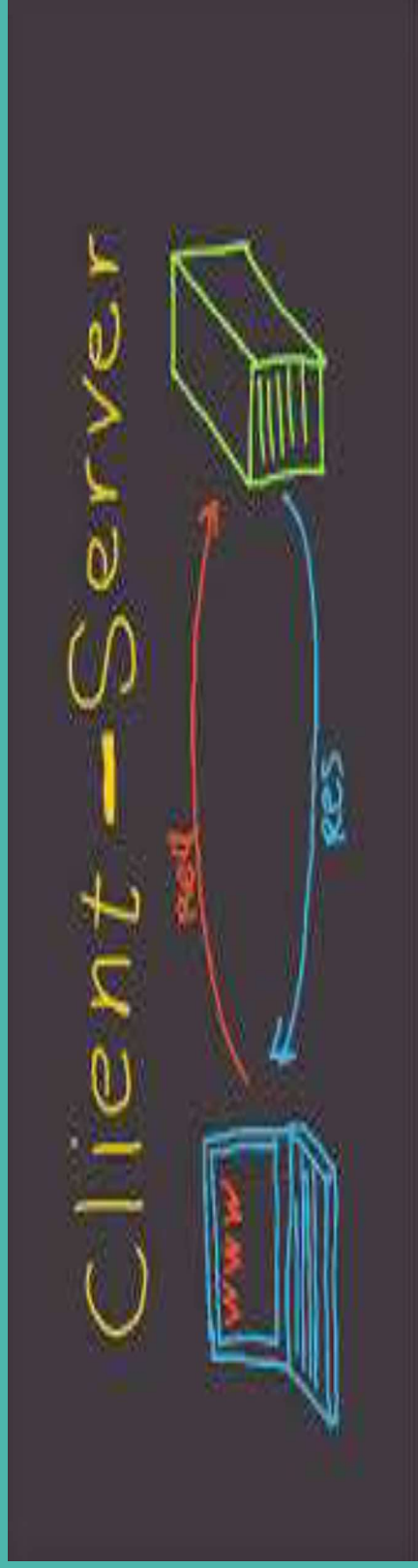


UDP - USER DATAGRAM PROTOCOL



Introduction

- Connectionless
- Unreliable transport protocol
- Located between application layer and network layer in the TCP/IP protocol suite
- Process to process communication using port numbers



Limitations of UDP

- There is no flow control mechanism.
- There is no acknowledgement for received packets.
- Does not provide error control to some extent.



Why would a process want to use UDP when it is powerless?

- It is simple protocol with minimum overhead.
- Application which use small messages to be sent without reliability then UDP is best.
- For small messages less no. of interactions is required between sender and receiver for UDP compared to TCP.

UDP Services



- **Process to process communication** – using sockets with the combination of IP address and port numbers
- **Connectionless services**
 - UDP is an independent datagram
 - The user datagram is not numbered
 - No difference between different user datagram even the source and destination are the same
 - UDP can't send a stream of data. Hence the message should fit in one user datagram (less than 65,507 bytes)

UDP Services Contd.



- **Flow control** –no flow control, no window mechanism so receiver may overflow with incoming messages.
- **Congestion control** – does not provide congestion control and has an assumption that the packets are small and sporadic so they can't create congestion.
- **Error control**
 - no error mechanism except checksum.
 - The sender does not know about the packet lost
 - If the receiver checks the error through checksum then that user datagram is discarded.

UDP Services Contd.

Checksum

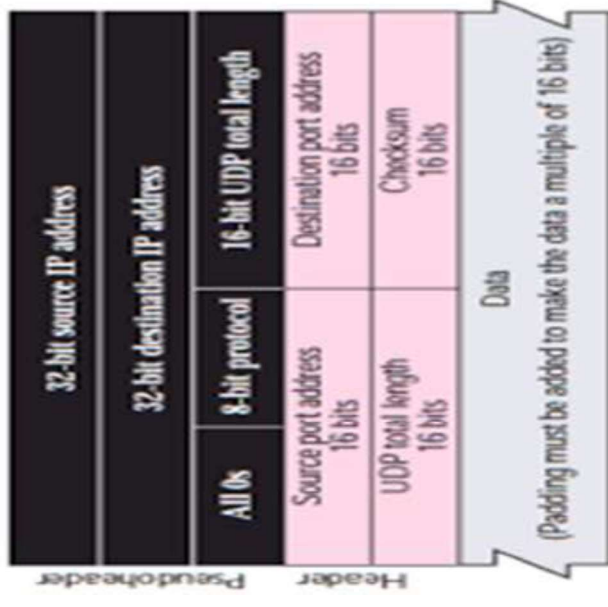
- contain three parts

1. Pseudo header

- It is a part of the header of the IP header
- Encapsulated with some fields with 0's
- Protocol field to differentiate between UDP and TCP
 - The value of the protocol field is 17. If it is changed then the packet gets discarded at receiver end.

2. UDP header

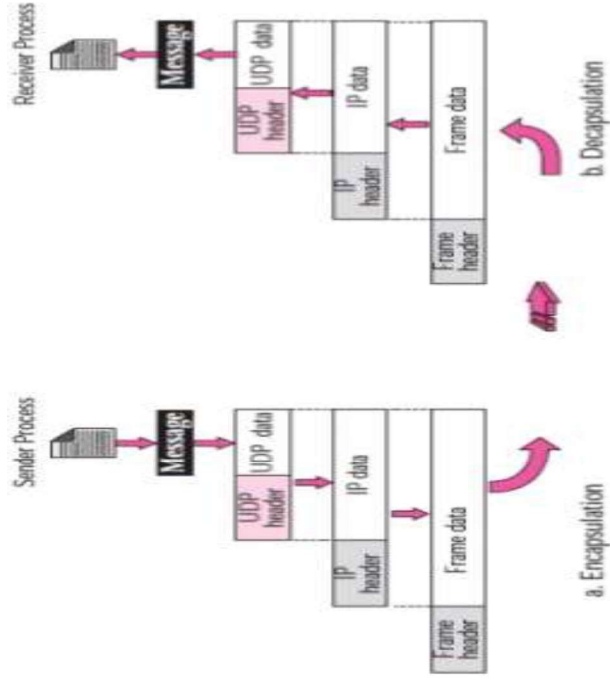
3. **Data** – communicating from the application layer



Source: <https://www.ques10.com/p/10930/how-is-checksum-computed-in-udp-1/>

UDP Services Contd.

- Encapsulation and Decapsulation
 - The process sends the message to the UDP along with a pair of socket address and length of data
 - The UDP then passes to IP adding UDP header
 - The IP adds its own header along with the value 17 to indicate it is a UDP message and sends to data link layer
 - The data link layer adds its own header and passes it to physical layer
 - The physical layer encodes bits to electrical signals and sends to remote machine
- The reverse process happens on the other end for decapsulation



Source:

<https://www.slideshare.net/MelvinCabatuan1/transport-layer-services>

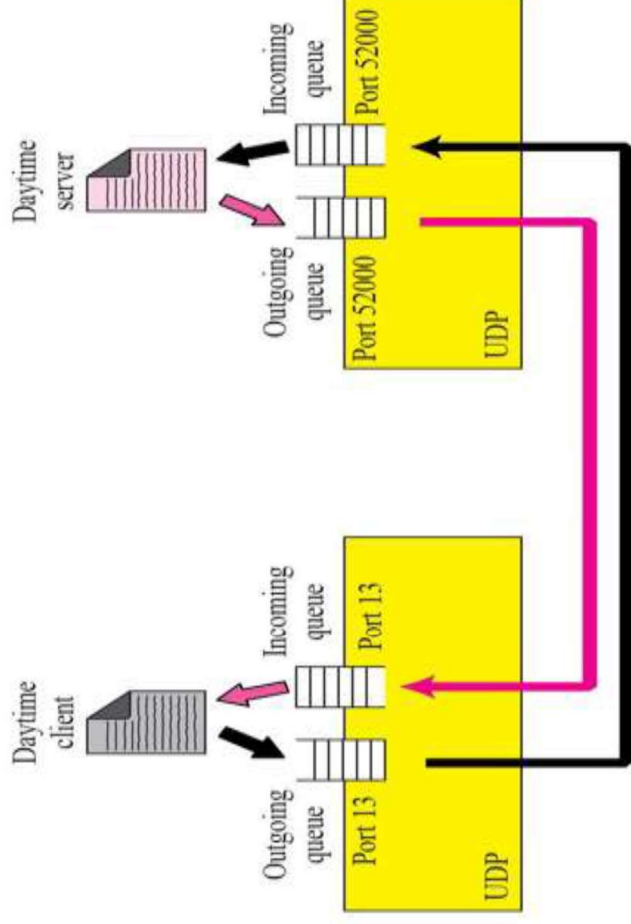
UDP Services Contd.



Queuing

- Port numbers are assigned by OS
- Each process has one port number, one incoming and one outgoing queue.
- When the process terminates the queue is destroyed.
- Client side uses ephemeral port numbers
 - The client sends the message to UDP using the outgoing queue
 - If the client receives the message the UDP checks if there is a incoming queue created. If it is available it will deliver else discards the packet sending a ICMP message “port unreachable” to the server.
- Server side uses well known port numbers.
 - The server side the queue remains open as long as the server is running
 - If the server receives the message the UDP checks if there is a incoming queue created. If it is available it will deliver else discards the packet sending a ICMP message “port unreachable” to the client

UDP Services Contd.



Source:

<http://www.myreadingroom.co.in/notes-and-studymaterial/68-dcn/848-user-datagram-protocol-udp.html>

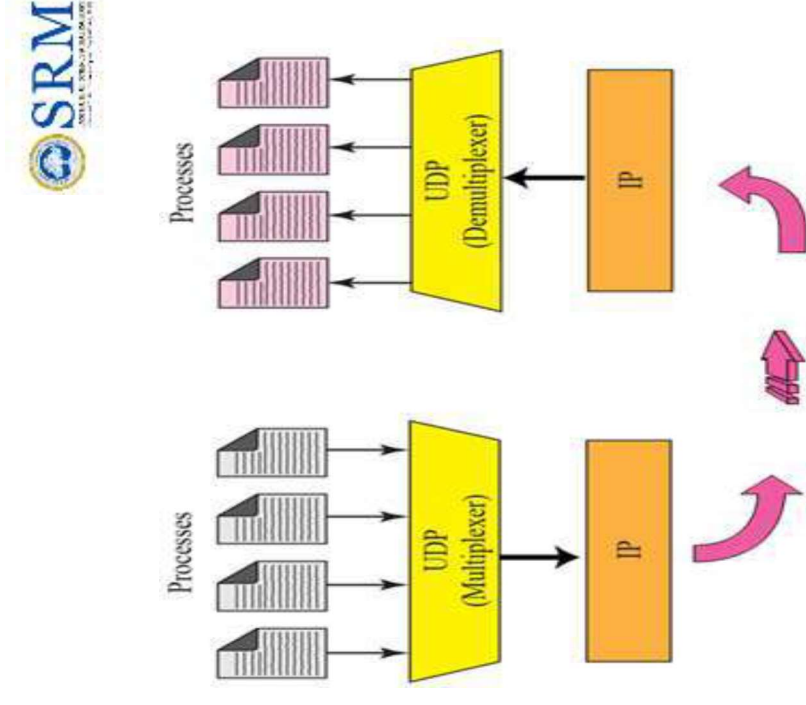
UDP Services Contd.

Multiplexing

- many to one relationship.
- UDP accepts messages from different process and differentiate by port numbers.
- Adds a header and then sends to the IP

Demultiplexing

- One to many relationship
- UDP receives the user datagram from IP and drops the header then sends the message to appropriate process based on port numbers



Source: <https://www.rfwireless-world.com/Terminology/Advantages-and-Disadvantages-of-UDP.html>

UDP Features

- Connectionless services
- Lack of error control
- Lack of congestion control



Connectionless service



- Preferable for small message which fits in a single datagram.
- The overhead to establish and close a connection may be significant whereas in TCP it takes 9 packets for exchanges between client and server to achieve the above goal.
- Provides less delay

Lack of error control



- UDP does not provide error control
- Provides unreliable service
- In reliable service the transport layer needs to take care of the lost packet by resending it. So there will be a uneven delay between different parts of the message delivered.

Lack of congestion control



- UDP does not provide congestion control.
- UDP does not provide additional traffic in error prone network.
- TCP leads to creation of congestion or additional congestion in network by resending packets several times when a packet are lost.

UDP Applications



- Suitable for request- response communication
- Suitable for internal flow and error – control mechanism
- Suitable for multicasting
- Used for management processes such as SNMP
- Used for route updating protocols such as RIP
- Used for real time applications which does not accept uneven delay