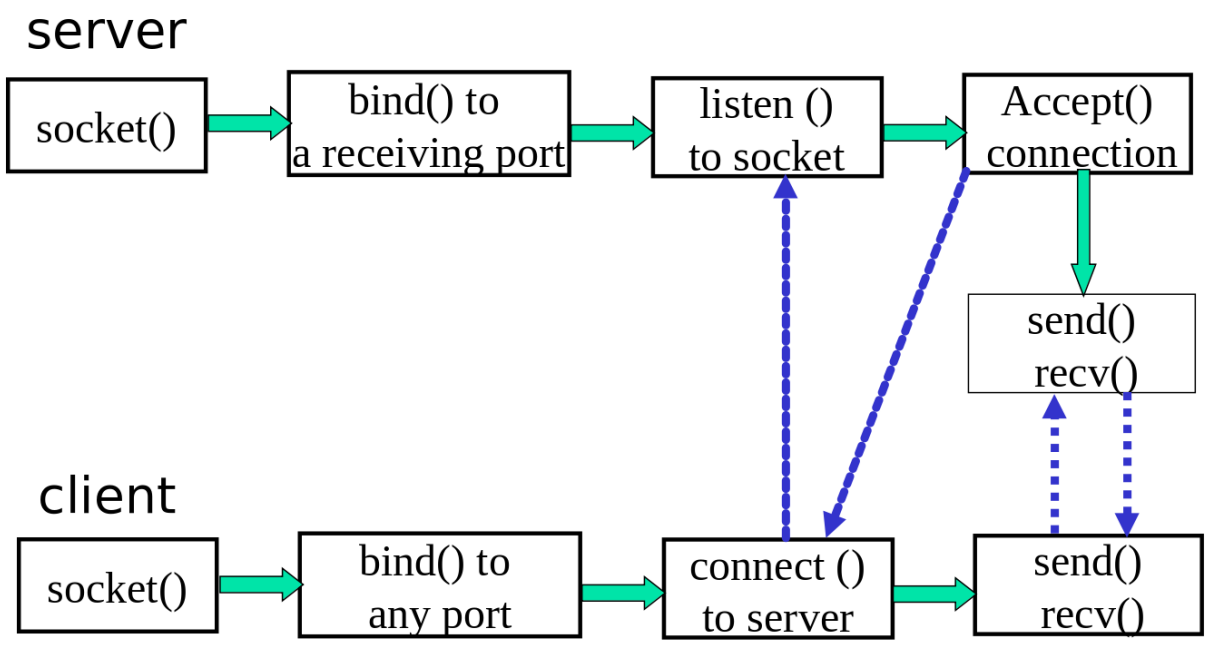
Selected protocol to handle packet drop: Go-Back-N

1. **TCP is a connection-oriented protocol, whereas UDP is a connectionless protocol.**

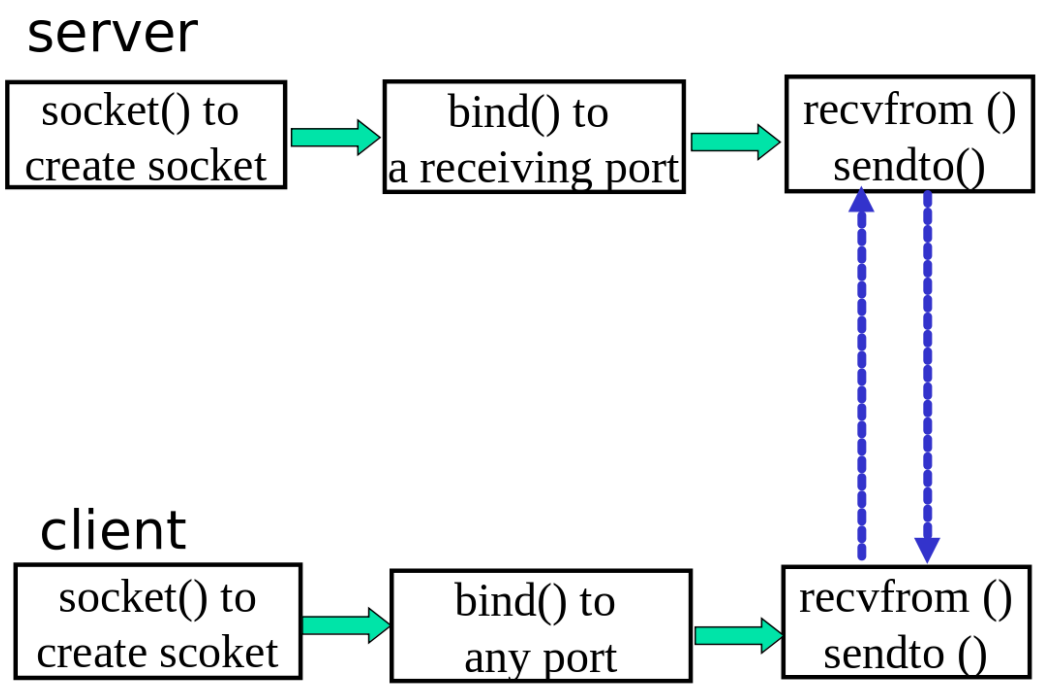
Basis	Transmission control protocol (TCP)	User datagram protocol (UDP)
Type of Service	TCP is a connection-oriented protocol. Connection-orientation means that the communicating devices should establish a connection before transmitting data and should close the connection after transmitting the data.	UDP is the Datagram-oriented protocol. This is because there is no overhead for opening a connection, maintaining a connection, and terminating a connection. UDP is efficient for broadcast and multicast types of network transmission.
Reliability	TCP is reliable as it guarantees the delivery of data to the destination router.	The delivery of data to the destination cannot be guaranteed in UDP.
Error-checking mechanism	TCP provides extensive error-checking mechanisms. It is because it provides flow control and acknowledgment of data.	UDP has only the basic error checking mechanism using checksums.
Acknowledgment	An acknowledgment segment is present.	No acknowledgment segment.
Sequence	Sequencing of data is a feature of Transmission Control Protocol (TCP). this means that packets arrive in order at the receiver.	There is no sequencing of data in UDP. If the order is required, it has to be managed by the application layer.
Speed	TCP is comparatively slower than	UDP is faster, simpler, and more

	UDP.	efficient than TCP.
Retransmission	Retransmission of lost packets is possible in TCP, but not in UDP.	There is no retransmission of lost packets in the User Datagram Protocol (UDP).
Header Length	TCP has a (20-60) bytes variable length header.	UDP has an 8 bytes fixed-length header.
Weight	TCP is heavy-weight.	UDP is lightweight.
Handshaking Techniques	Uses handshakes such as SYN, ACK, SYN-ACK	It's a connectionless protocol i.e. No handshake
Broadcasting	TCP doesn't support Broadcasting.	UDP supports Broadcasting.
Protocols	TCP is used by HTTP, HTTPs, FTP, SMTP and Telnet.	UDP is used by DNS, DHCP, TFTP, SNMP, RIP, and VoIP.
Stream Type	The TCP connection is a byte stream.	UDP connection is message stream.
Overhead	Low but higher than UDP.	Very low.

TCP:


server
socket()
bind() to
a receiving port
listen ()
to socket
Accept()
connection
send()
recv()
client
socket()
bind() to
any port
connect ()
to server
send()
recv()

UDP:


server
socket() to
create socket
bind() to
a receiving port
recvfrom ()
sendto()
client
socket() to
create scoket
bind() to
any port
recvfrom ()
sendto ()

2. در هر ردیف در جدول پایین، ستون خانه ای که سبز شده است مشخص می‌کند که کدام الگوریتم مزیت دارد. ردیف های زرد شده یعنی هر دو الگوریتم در معیار ردیف یکسان هستند. باقی معیار ها صرفا برای مقایسه دو روش هستند.

مورد مقایسه	Selective-repeat	Go-Back_N
Window size	For sender: N For reciever: N	For sender: N For reciever: 1
Complexity	The receiver window must sort the frames so it has higher complexity.	Easier implementation since we don't need to sort the received data. (window size = 1)
Efficiency	Efficiency = N/(1+2a) a = propagation delay/transmission delay N = number of packets sent	Efficiency = N/(1+2a) a = propagation delay/transmission delay N = number of packets sent
Acknowledgement	individual	cumulative
Out-of-Order packets	Out-of-order packets will be accepted and sorted on the sender side.	Out-of-order packets won't be accepted and the whole window for sender has to be sent again.
Minimum Sequence Number	2N	N+1
Bandwidth consumed for data packets	Doesnt use extra bandwidth since it only resends the lost packets.	Uses high rate of bandwidth since for every lost packet, the whole window has to be resent, so if the error-rate decreases, most of the bandwidth would be used for

		duplicate data.
Bandwidth consumed for ack packets	Since all acks are sent individually, one ack is sent for each packet and more bandwidth is used.	Since all acks are sent cummolatively, less number of acks are sent and less bandwidth is used for acks.