The Five-Layer Network Model

A node on one network establishes a TCP connection with a node on another network via the same router.

Node 1 which is a client, want to establish a connection with node 2 which is a server. Let's say Node 1 which is on network A, while Node 2 is on Network B and both are connected to the same router. Node 1 is seeking to establish a TCP connection with node 2.

First, it starts with the physical layer, which makes the connections possible through cabling, computer, routers, and server hardware, and this means that the physical layer consists of the electronic circuit transmission technologies of a network. The physical layer or layer 1 is the first and lowest layer. The physical layer defines the means of transmitting raw bits. Through this layer, nodes can send data back and forth between network A and network B.

Node 1 then request information from Node 2, which it seeks to establish a TCP connection by communicating with the local networking stack, which is part of the operating system responsible for handling network functions. The networking stack then examines its subnet at the data link layer level.

The data link layer is responsible for the node to node delivery of the message. The primary function of this layer is to make sure data transfer is error free from one node to another, over the physical layer. By checking the subnet node, 1 sees that the IP address destination for its TCP connection lives on another network. Node 1 then knows that it'll have to send any data to its gateway for routing to a remote network. The node then examines the gateway configuration number that is present between network A and the router. It checks its ARP table to determine MAC address that aligns with the configuration number, if it doesn't find a number that matches the ARP table, it then sends out an ARP discovery request to every node on the local network. Once the router receives node 1 ARP request, it sees the client currently assigned the IP address, which is an address having information about how to reach a specific host. The IP address is a 32-bit unique address. The router then responds to the client to let it know its MAC address. The MAC address is a unique 48-bits hardware number of a computer, which is embedded into the network during the time of manufacturing. MAC Address is also known as Physical Address of a network device.

The client receives the response and becomes aware of the hardware address of the client's router. The client is now ready to build the outbound packet, the server then receives it.

The client now knows that it's being asked to form an outbound TCP connection, which means it'll need an outbound TCP port, an TCP port is The Operating System identifies the ephemeral port that is available and opens a socket connecting the client to this port. Because this is a TCP connection, it needs to establish a connection before the networking stack can transmit any data. We now enter the network layer.

The network layer works for the transmission of data from one host to the other located in different networks. It also takes care of packet of routing, i.e., selection of the shortest path to transmit the packet, from the number of routes available. The send and receiver's IP address are therefore placed in the header by network layer.

*Segment in Network layer can be referred to as Packet.

A TCP segment is built. It then fills all the appropriate fields in the header, including source ports and destination port. A sequence number is then chosen and is used to fill in the sequence number field. The SYN flag is set, and a checksum for the segment is calculated and written to the checksum field. The checksum is an error-detection method; it computes the numerical value according to the number of set or unset bits. The constructed TCP segment is now passed along to the IP layer of the networking stack. In this layer, an IP header is constructed. This header is filled source and destination IP and a TTL of 64. The TTL is the number of hops that a packet is permitted to travel before being discarded by a router. This layer is also considered the Transport Layer. So, what is the transport layer?

The transport layer provides services to the application layer and takes services from the network layer. The data in the transport layer is referred to as Segments. It is responsible for the End to End delivery of the complete message, it also provides the acknowledgment of successful data transmission and retransmits the data if an error occurred.

Now the TCP segment is inserted as the data payload for the IP datagram. Moreover, a checksum is calculated for the whole thing. When the IP datagram is constructed, the client needs to get this to its gateway, which now has a MAC address. An Ethernet Datagram is constructed. All the relevant fields are filled in with the appropriate data.

then the IP datagram is inserted as a data payload of the Ethernet Frame, and another checksum is calculated. The Ethernet frame is then ready to be sent across the physical layer. Now the Ethernet frame is converted into binary data that is transmitted via modulating electrical signals that run across the CAT6 cabling which connects the client to a network switch of network A. When the switch receives the Ethernet frame, it inspects the destination MAC address, which is

the router that is specified. The switch knows which of its interface this MAC address is attached to and forwards the frame across only the cable connected to the router.

When the router receives the Ethernet frame, it then performs a checksum calculation because it recognizes that its own MAC address is specified as the destination, it there knows it intended to receive it. The checksum of a result of the router can now be compared against the checksum of the Ethernet frame; the checksum shows that both matches, which mean all the data within the Ethernet frame have been transferred to the router. Now the router strips away the ethernet frame, leaving just one IP datagram, it now can perform another checksum to compare against the IP datagram checksum. It is now confirmed by the checksum that all the IP datagram data has arrived in one piece.

It then examines the destination IP address. It looks at its routing table, which is a data table stored in a router or a networked computer that lists the routes to network destinations, and in some cases, metrics associated with those routes. The router sees that the destination address of the server is on a locally connected network. So, it decrements the TTL by 1 again and calculates a new checksum and creates a new IP datagram. A new Ethernet frame again encapsulates this new IP datagram. The new Ethernet frame is specified by the router's MAC address as the source MAC address and server MAC address as the destination MAC address. Before being sent across network B to computer 2. The new datagram is inserted as the data payload of the newly constructed ethernet frame, and a checksum is completed.

The ethernet frame then arrives at network B. Once it's arrived, the network B switch inspects the destination MAC address and sees that it belongs to the server. It will also see that the server is also connected to the switch, it then can forward the Ethernet frame across the cable. When the server receives the frame, it immediately recognizes that its own MAC address was specified as the destination. The server then strips away the ethernet frame to inspect the IP datagram and performs a checksum to determine that the data arrived in one piece. It now starts the check the IP datagram with has an address specified within it, it sees that its IP address was specified confirming that it was for the server.

The server then strips away the IP datagram layer, leaving just the TCP segment of the data sent. A checksum against the TCP layer is performed, once it confirmed that all the data within the TCP segment arrived intact. The server then inspects the destination ports which is specified within the TCP segment, which is 80. The networking stack on the server checks to ensure that there's an open socket on port 80, which there is. It's in the listening state by the server. The server then sees that his packet has the SYN flag set. So, it's examined the sequence number and stores that, because it'll need to put that sequence number in the acknowledgment field once it crafts the response.

Once all of it has been completed, we get a single TCP segment containing an SYN flag from one node to another one. Everything is going to happen again, for the server to send an SYN-ACK response to the client. Then everything would have to happen all over again for the client to send an ACK back to the server, and over and over and over again.