• Let’s imagine there are 2 networks, network A which will contain an address space of 10.1.1.0/24 and network B will contain an address space of 192.168.1.0/24. Router A sits between network A and network B. At network A’s interface it has assigned an IP of 10.1.1.1 and at network B’s interface, it has assigned an IP of 192.168.1.2.

• Imagine a computer sitting at network A’s interface which is assigned with an IP address of 10.1.1.100. This computer will be acting as a client in this scenario and let’s name it computer 1. And also there’s computer 2 present at network B’s interface which will be acting as a server in this scenario and this computer 2 is assigned with an IP address of 192.168.1.100 and has an FTP server listening on port 21.

• Now the end-user sitting in front of computer 1 opens up a web browser and enters an ftp://192.168.1.100:21, the web browser communicates with the local networking stack which is the part of the operating system responsible for handling networking functions.

• The networking stack examines its own subnet and finds that it leaves on 10.1.1.0/24 which means that the destination is on another network.

• By now computer 1 knows that it’ll have to send any of the data to its gateway for routing purposes. And it has been configured with a gateway address of 10.1.1.1.

• Next computer 1 looks at its ARP table and doesn’t find the MAC address of its gateway 10.1.1.1. So it creates an ARP request and sends it to the hardware broadcast address of all F’s. This ARP message is sent to every node on the local network.

• When router A receives this ARP request it recognizes it’s the computer connected to its gateway and sends back the MAC address of 00:11:22:33:44:55.

• Computer 1 receives this ARP response and now knows the hardware address of its gateway.

• This means that it’s ready to start constructing the outbound packet.

• Computer 1 knows that the web browser is requesting to form an outbound TCP connection, which means it’ll need the outbound TCP port.

• The operating system identifies the ephemeral port of 60000 is available and opens a socket connecting the web browser to this port.

• Since this a TCP connection, the networking stacks know that before it can transmit data it needs to establish a TCP connection.

• The networking stack starts to build a TCP segment and fills in the appropriate fields in the header including source port of 60000 and destination port of 21. A sequence number is chosen and filled in the sequence number field. Finally, the SYN flag is set and a checksum is calculated and written in the checksum field.

• The newly constructed TCP segment is now passed along to the IP layer of the networking stack.

• This layer constructs an IP header. This header is filled in with source IP, destination IP, and TTL of 64, which is a standard value for this field. Next TCP segment is inserted as data payload for the IP datagram. And a checksum is calculated for the whole thing.

• Now that the IP datagram is constructed, computer 1 needs to get this data to its gateway, which it now knows has a MAC address of 00:11:22:33:44:55, so an Ethernet datagram is constructed. All the relevant fields are filled in with appropriate data, most notably Source (which is computer 1) and Destination (which is Router A) MAC address. Finally, the IP datagram is inserted as Data Payload in the Ethernet frame and another checksum is calculated. Now the Ethernet frame is ready to be sent across the physical layer.

• The network interface connected to computer 1 sends this binary data as modulations of the voltage of an electrical current running across CAT6 cable, which is connected between it and network switch. This switch receives the frame and inspects the destination MAC address, the switch knows which of its interface this MAC address is attached to, and forwards the frame only across the cable connected to its interface.

• At the other end of the link Router A receives the frame and recognizes its own hardware address as the destination. Now Router A takes the entire frame and calculates a checksum against it and compares it with the checksum field of the Ethernet header and sees that they match. Meaning all of the data has been received in one piece. Next Router A strips away the Ethernet frame, leaving it with an IP datagram, again router A calculates a checksum and compares it with the checksum field and it finds that it matches, meaning all of the data is correct. It inspects the destination IP address and performs a lookup of this destination in its routing table. Router A finds that the destination computer 2 or 192.168.1.100 is connected on a locally connected network. It decrements the TTL value by 1 and creates a new IP datagram, calculates the checksum, and fills it the checksum field of the IP datagram. Now it creates a new Ethernet frame filling it with appropriate fields, consisting of Source MAC address as Router A and Destination MAC address as Computer 2. Again the checksum is calculated and places it in the checksum field and sends the Ethernet frame out to network B.

• Network B recognizes that the destination MAC address is connected on one of its interfaces and only sends the frame across the cable connected to it.

• Finally computer 2 receives the Ethernet frame identifies its own MAC address as the destination address and knows that it is intended for itself. Computer 2 strips away the Ethernet frame leaving it with an IP datagram. It performs the CRC and recognizes that the data has been delivered intact. It examines the destination IP address as its own. Next computer 2 strips away the IP datagram leaving it with the TCP segment. Again the checksum for this layer is examined and everything checks out. Next computer 2 examines the destination port, which is 21. The networking stack on computer 2 checks to ensure that there is an open socket on port 21, which there is. It’s in listen state and held by a running FTP web server. Computer 2 then sees that this packet has an SYN flag set. It examines the sequence number and stores that since it’ll need to put that number in its acknowledgment field once it crafts the response.

• After all these things a single TCP segment containing an SYN flag is sent from computer 1 and received by computer 2.

• Now everything would happen all over again for computer 2 to send SYN-ACK response to computer1. Then everything would happen again for computer 1 to send an ACK response back to computer 2, and so on.

• This is how all the layers work together in a 5 layer TCP/IP network model to get the job done.