

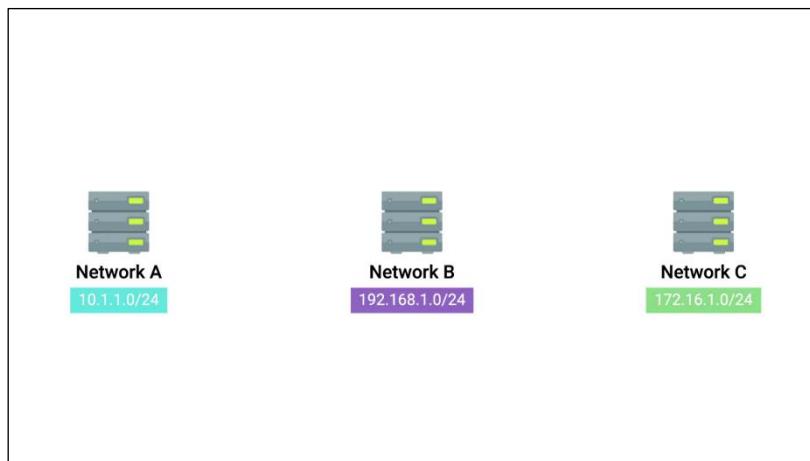
How data are sent from one network to another.

The 5-layer model

1 Definition of the network

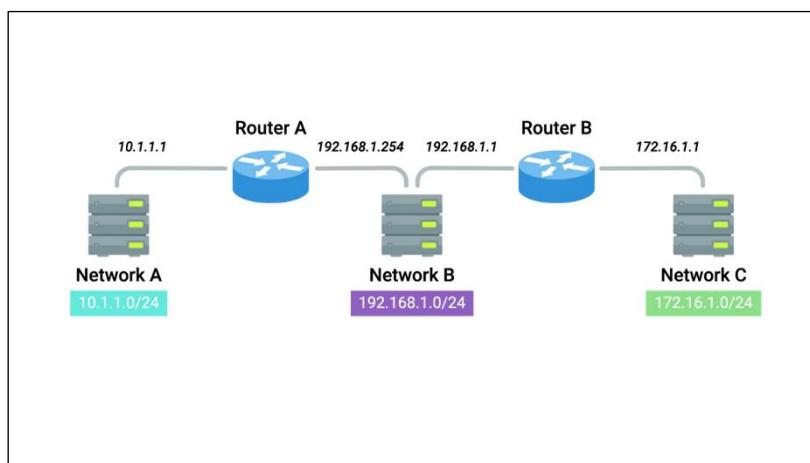
1.1 Networks

Imagine three networks, network A will contain address space 10.1.1.0/24. Network B Will contain address space 192.168.1.0/24, and network C will be 172.16.1.0/24.



1.2 Routers

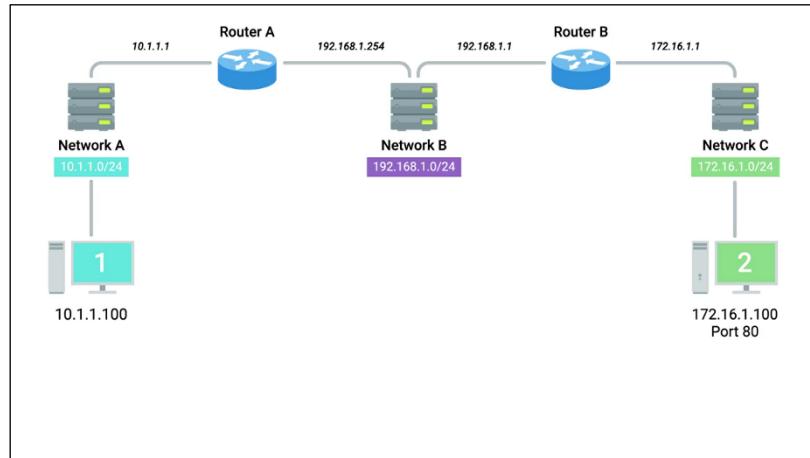
Router A sits between network A and network B. With an interface configured with an IP of 10.1.1.1 on network A, and an interface at 192.168.1.254 on network B. There's a second router, router B, which connects networks B and C. It has an interface on network B with an IP address of 192.168.1.1, and an interface on network C with an IP address of 172.16.1.1.



1.3 The computers

Now let's put a computer on one of the networks. Imagine it's a desktop, sitting on someone's desk at the workplace. It'll be our client in this scenario, and we'll refer to it as computer 1. It's part of Network A and has been assigned an IP address of 10.1.1.100.

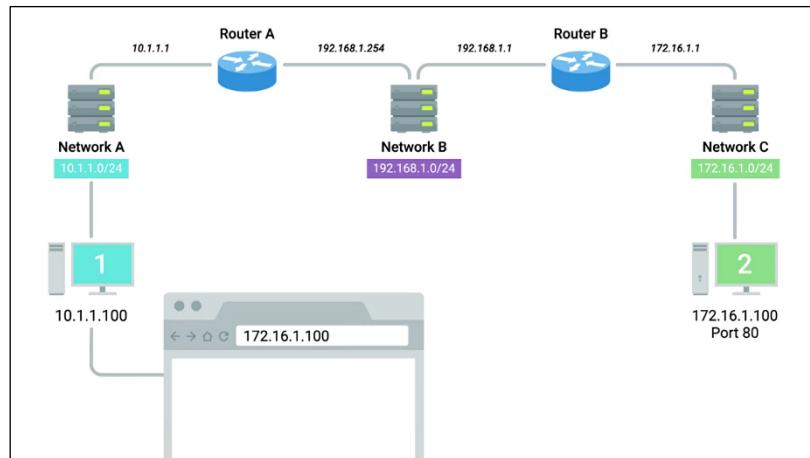
Now, let's put another computer on one of our other networks. This one is a server in a data center, it'll act as our server in this scenario, and we'll refer to it as computer 2. It's part of network C, and has been assigned an IP address of 172.16.1.100, and has a web server listening on port 80.



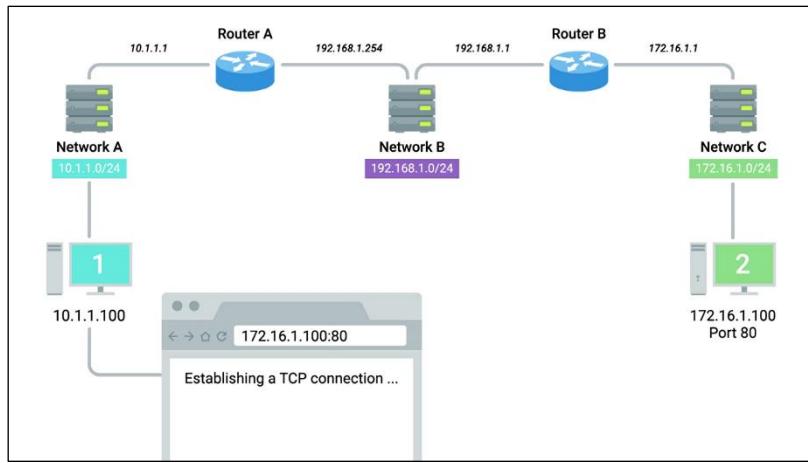
2 The client request

2.1 Client request from the client computer

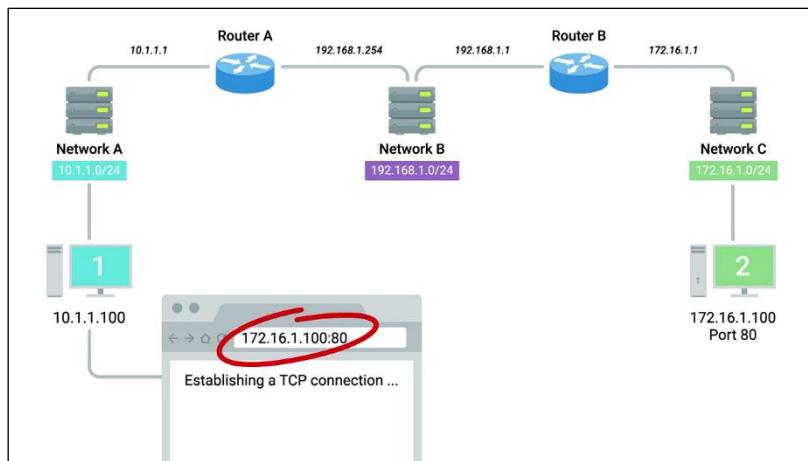
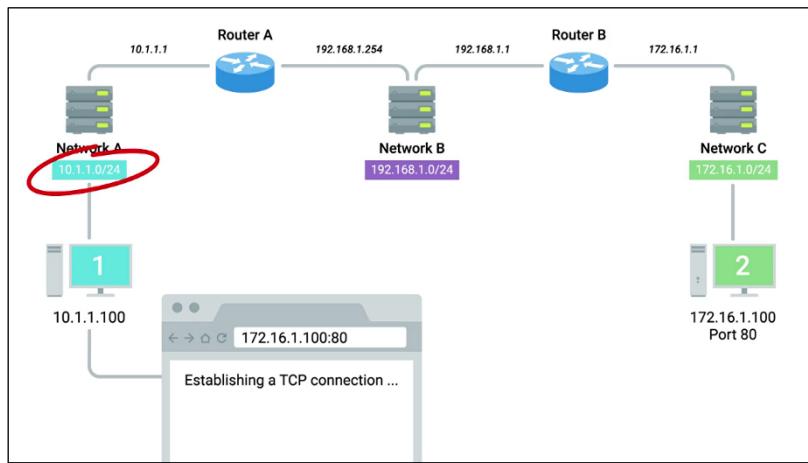
An end user sitting at computer 1 opens up a web browser and enters 172.16.1.100 into the address bar, let's see what happens.



The web browser running on computer 1 knows it's been ordered to retrieve a web page from 172.16.1.100. The web browser communicates with the local networking stack, which is the part of the operating system responsible for handling networking functions. The web browser explains that it's going to want to establish a TCP connection to 172.16.1.100, port 80.



The networking stack will now examine its own subnet. It sees that it lives on the network 10.1.1.0/24, which means that the destination 172.16.1.100 is on another network.

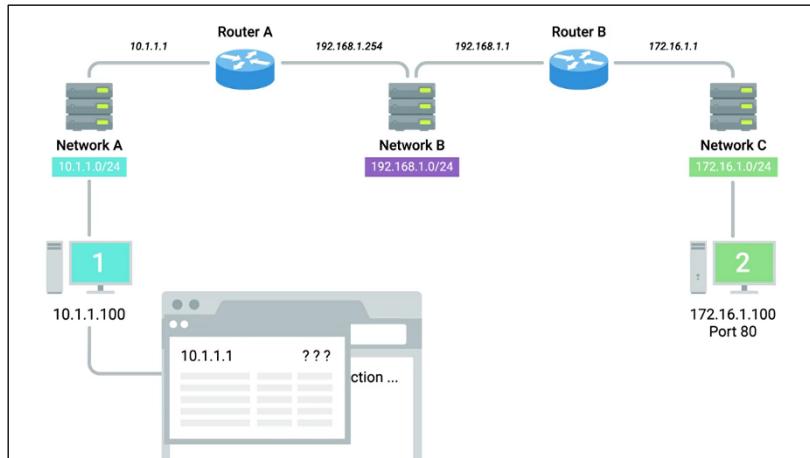


At this point, computer 1 knows that it'll have to send any data to its gateway for routing to a remote network. And it's been configured with a gateway of 10.1.1.1.

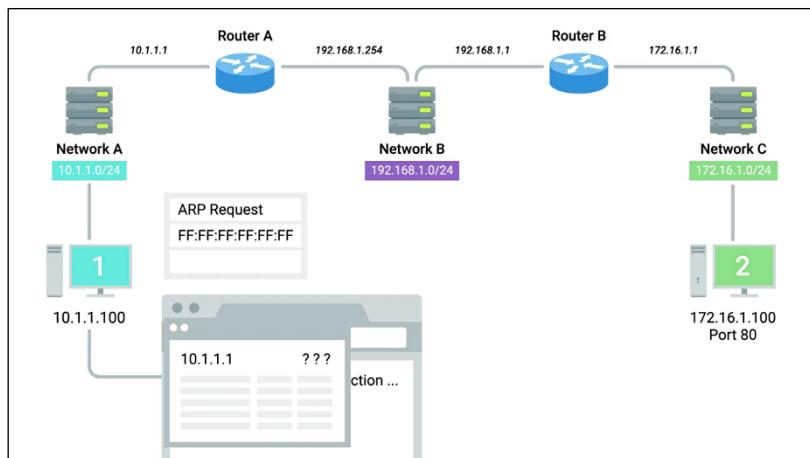
2.2 Check ARP table

2.2.1 Check if MAC address is in the current network

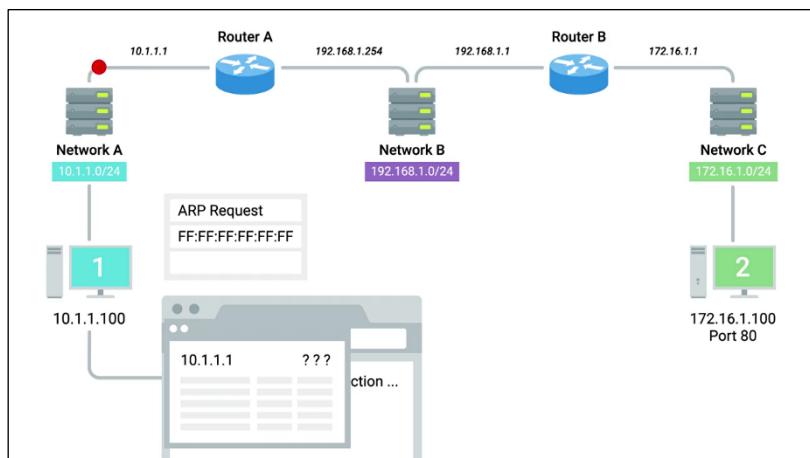
Next, computer 1 looks at its ARP table to determine what MAC address of 10.1.1.1 is, but it doesn't find any corresponding entry.



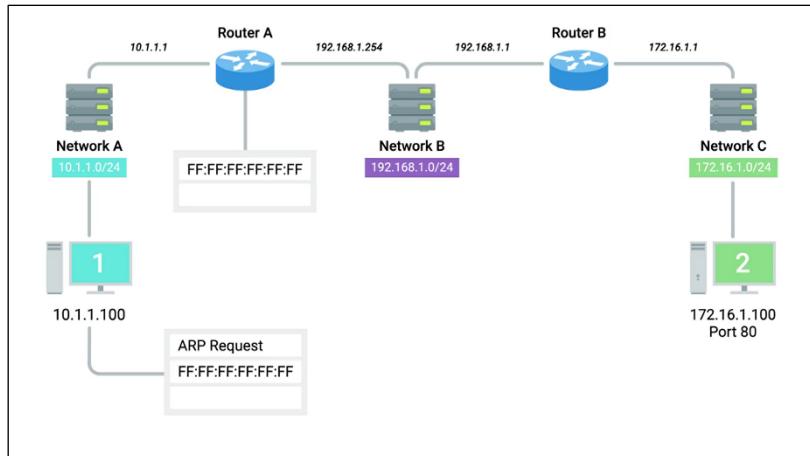
Uh-oh, it's okay, computer 1 crafts an ARP request for an IP address of 10.1.1.1, which it sends to the hardware broadcast address of all Fs.



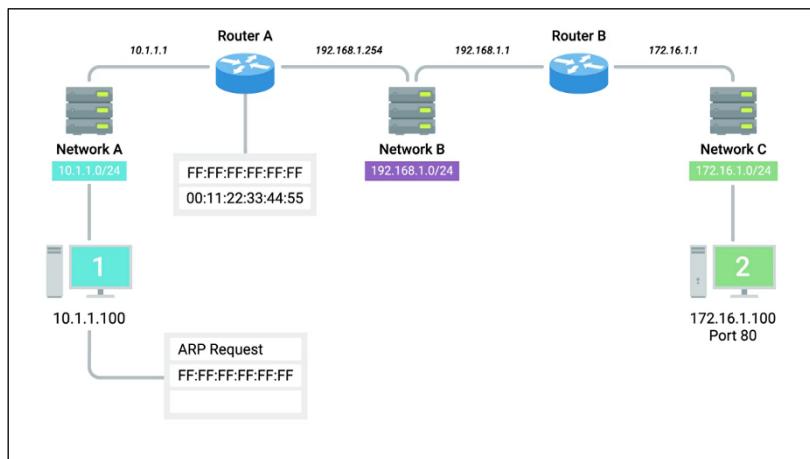
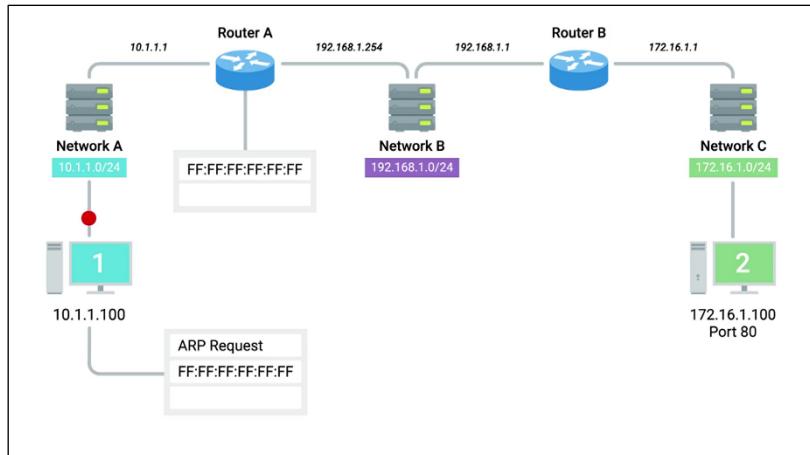
This ARP discovery request is sent to every node on the local network.



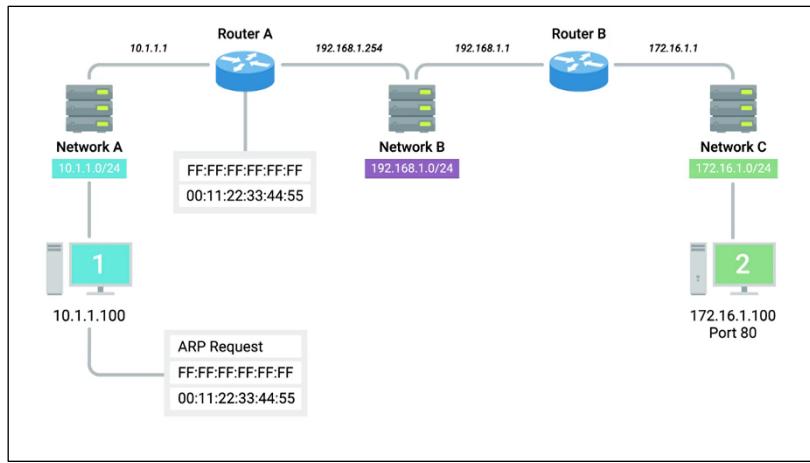
When router A receives this ARP message, it sees that it's the computer currently assigned the IP address of 10.1.1.1.



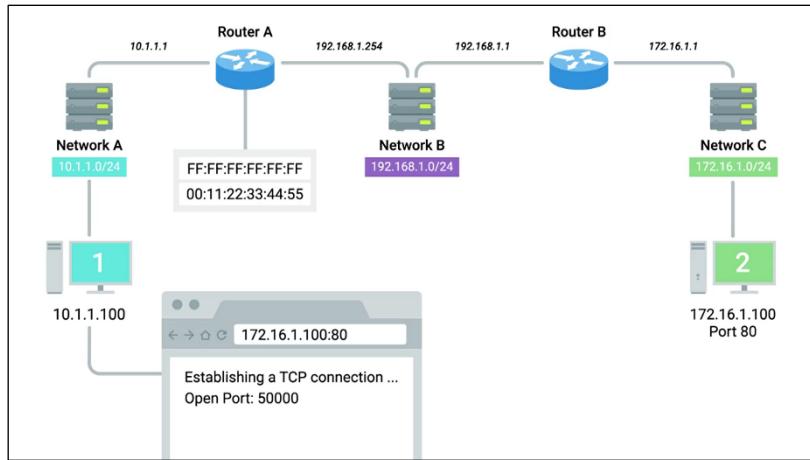
So it responds to computer 1 to let it know about its own MAC address of 00:11:22:33:44:55.



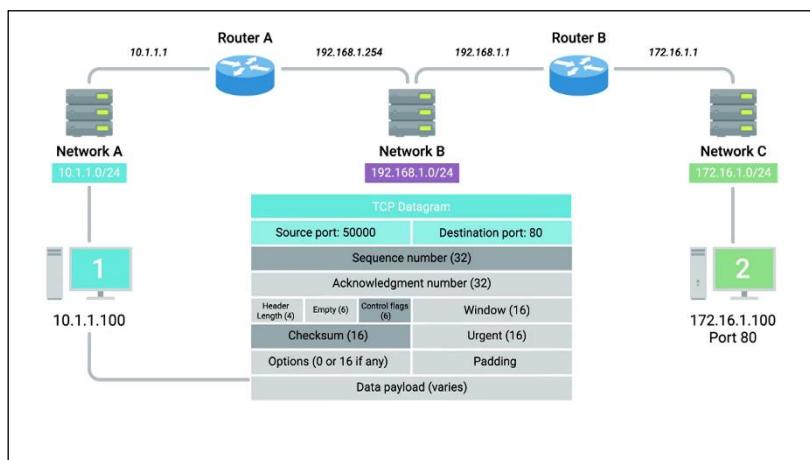
Computer 1 receives this response and now knows the hardware address of its gateway. This means that it's ready to start constructing the outbound packet.



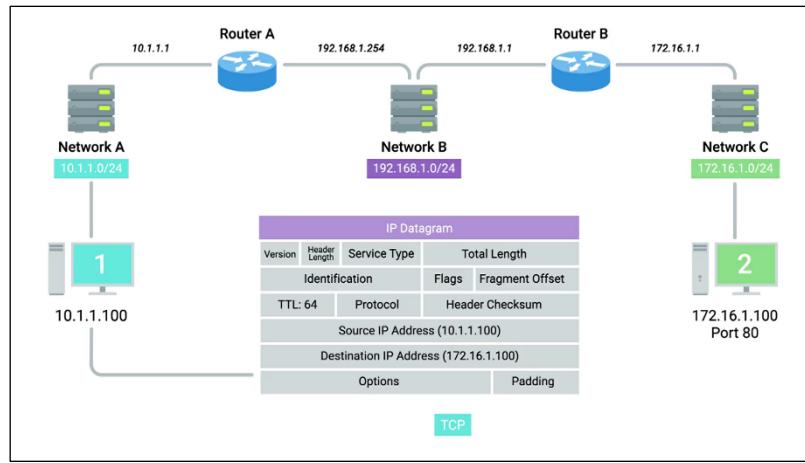
Computer 1 knows that it's being asked by the web browser to form an outbound TCP connection, which means it'll need an outbound TCP port. The operating system identifies the ephemeral port of 50000 as being available, and opens a socket connecting the web browser to this port. Since this is a TCP connection, the networking stack knows that before it can actually transmit any of the data the web browser wants it to, it'll need to establish a connection.



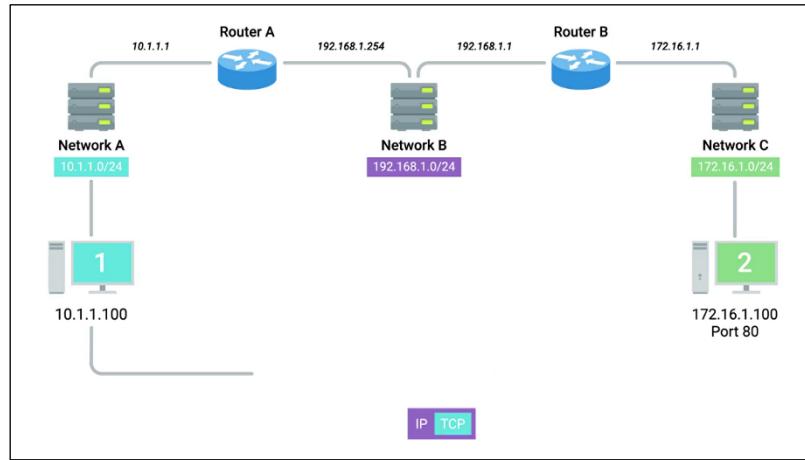
The networking stack starts to build a TCP segment. It fills in all the appropriate fields in the header, including a source port of 50000 and a destination port of 80. A sequence number is chosen and is used to fill in the sequence number field. Finally, the SYN flag is set, and a checksum for the segment is calculated and written to the checksum field.



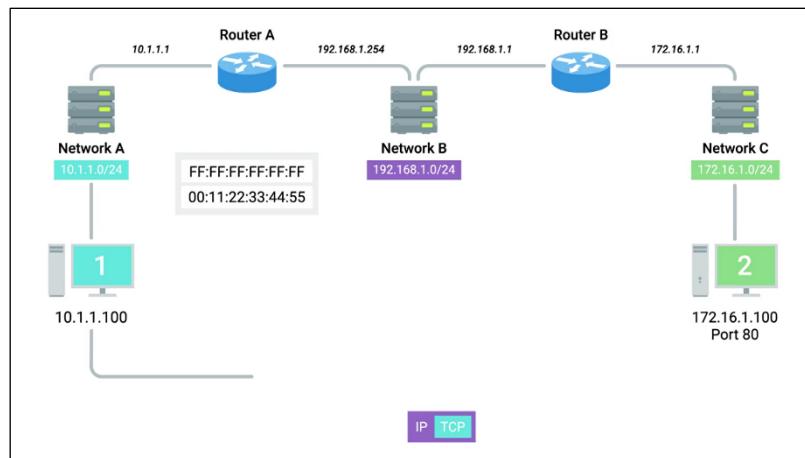
Our newly constructed TCP segment is now passed along to the IP layer of the networking stack.

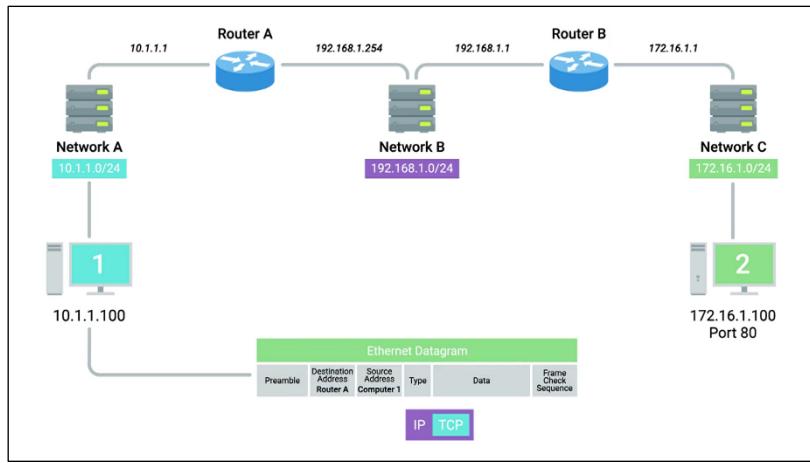


This layer constructs an IP header. This header is filled in with the source IP, the destination IP, and a TTL of 64, which is a pretty standard value for this field.



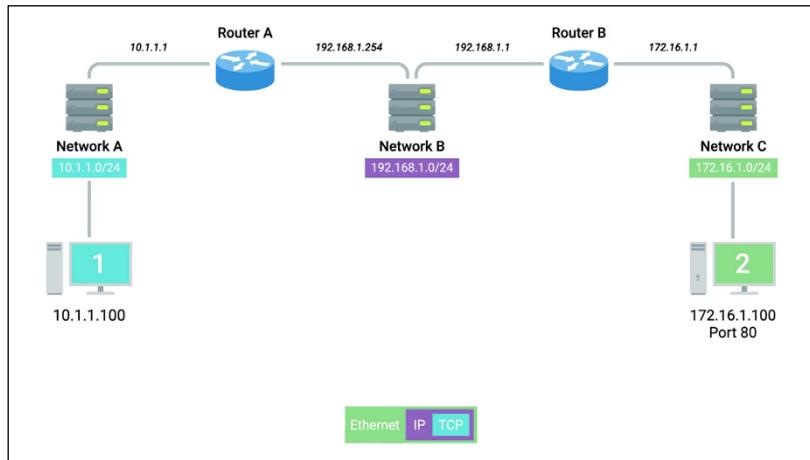
Next, the TCP segment is inserted as the data payload for the IP datagram. And a checksum is calculated for the whole thing. Now that the IP datagram has been constructed, computer 1 needs to get this to its gateway, which it now knows has a MAC address of 00:11:22:33:44:55, so an Ethernet Datagram is constructed.





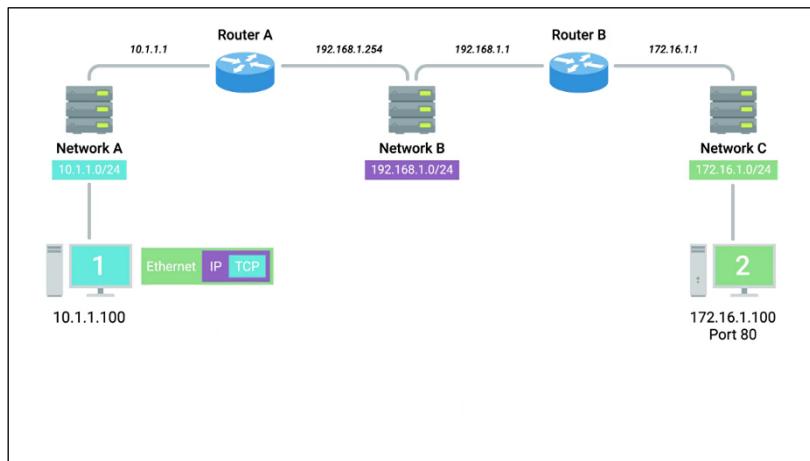
All the relevant fields are filled in with the appropriate data, most notably, the source and destination MAC addresses.

Finally, the IP datagram is inserted as the data payload of the Ethernet frame, and another checksum is calculated.

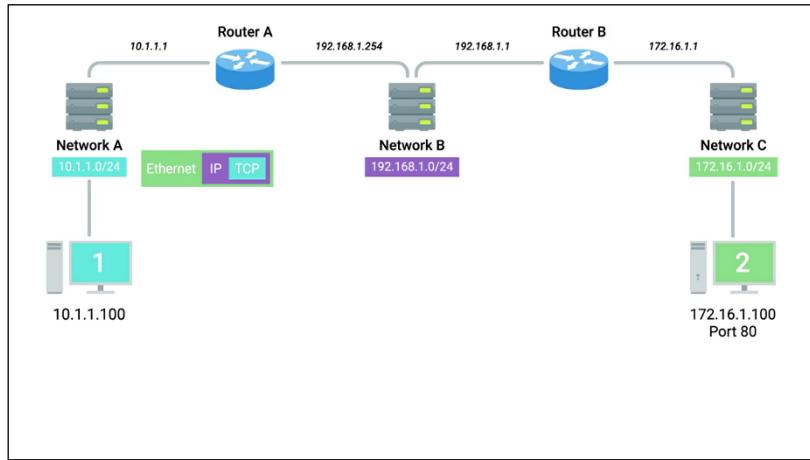


Now we have an entire Ethernet frame ready to be sent across the physical layer.

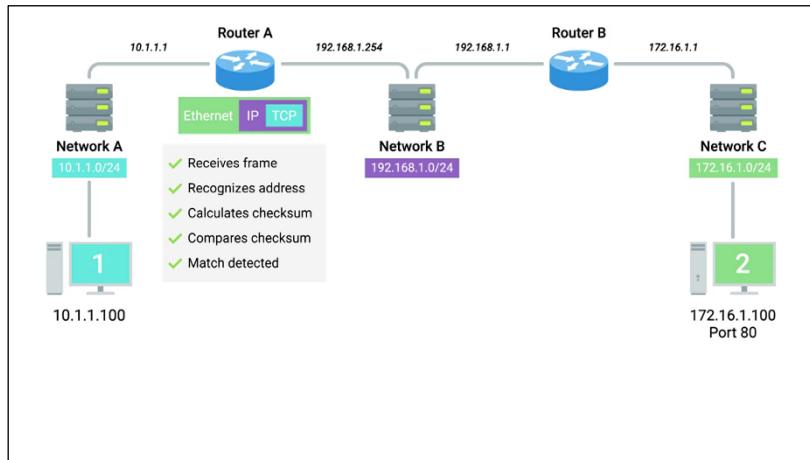
The network interface connected to computer 1 sends this binary data as modulations of the voltage of an electrical current running across a CAT6 cable that's connected between it and a network switch.



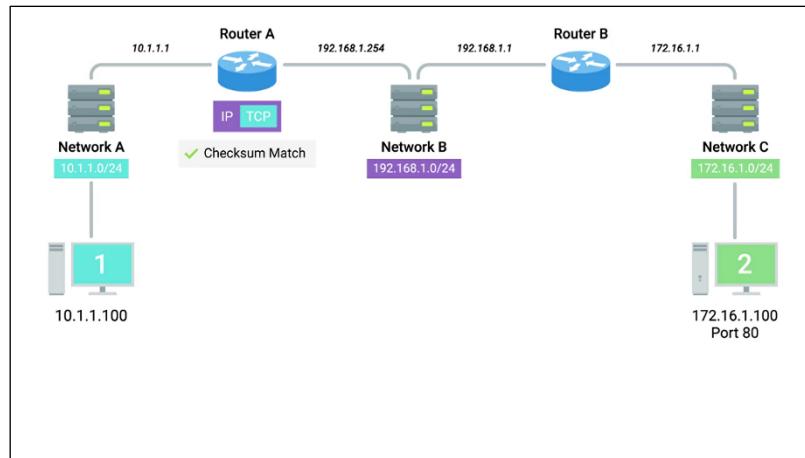
This switch receives the frame and inspects the destination MAC address. The switch knows which of its interfaces this MAC address is attached to, and forwards the frame across only the cable connected to this interface.



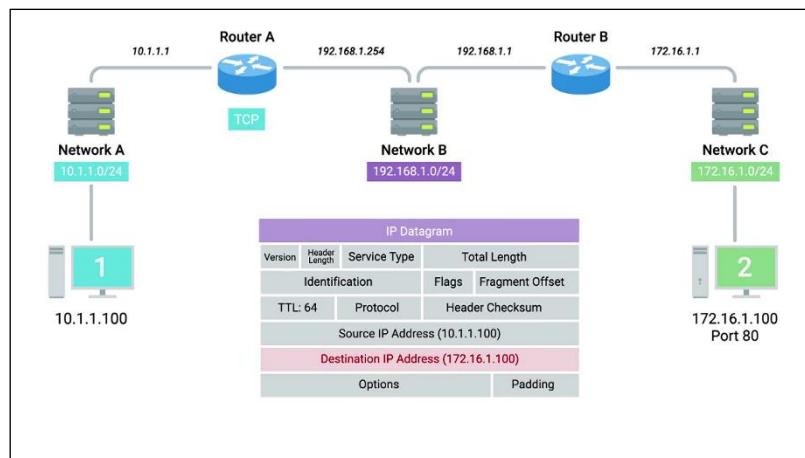
At the other end of this link is router A, which receives the frame and recognizes its own hardware address as the destination. Router A knows that this frame is intended for itself. So it now takes the entirety of the frame and calculates a checksum against it. Router A compares this checksum with the one in the Ethernet frame header and sees that they match. Meaning that all of the data has made it in one piece.



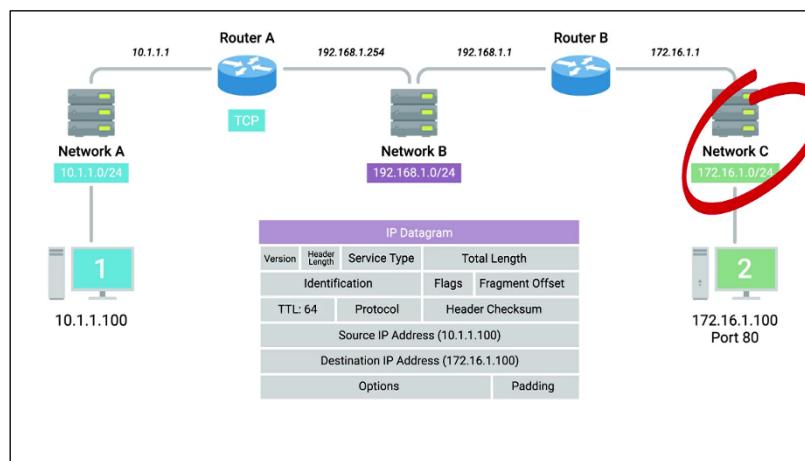
Next, Router A strips away the Ethernet frame, leaving it with just the IP datagram. Again, it performs a checksum calculation against the entire datagram. And again, it finds that it matches, meaning all the data is correct.

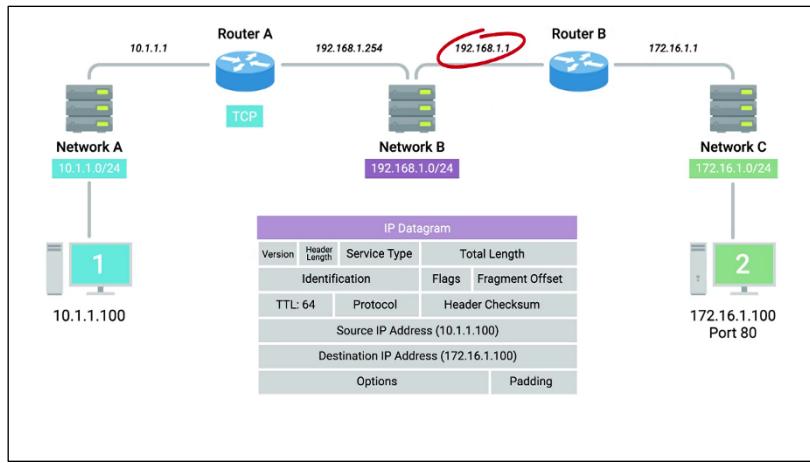


It inspects the destination IP address and performs a lookup of this destination in its routing table.

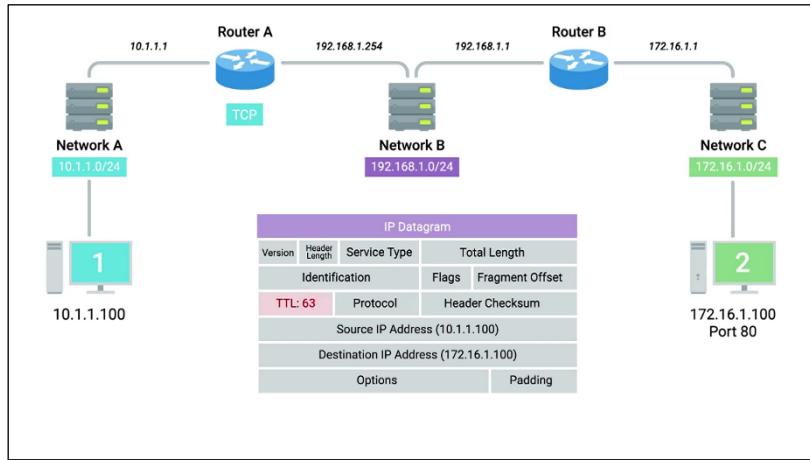


Router A sees that in order to get data to the 172.16.1.0/24 network, the quickest path is one hop away via Router B, which has an IP of 192.168.1.1.

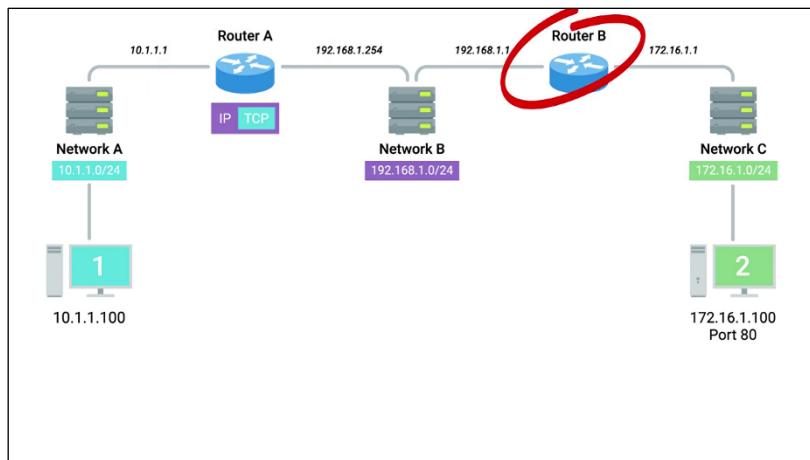




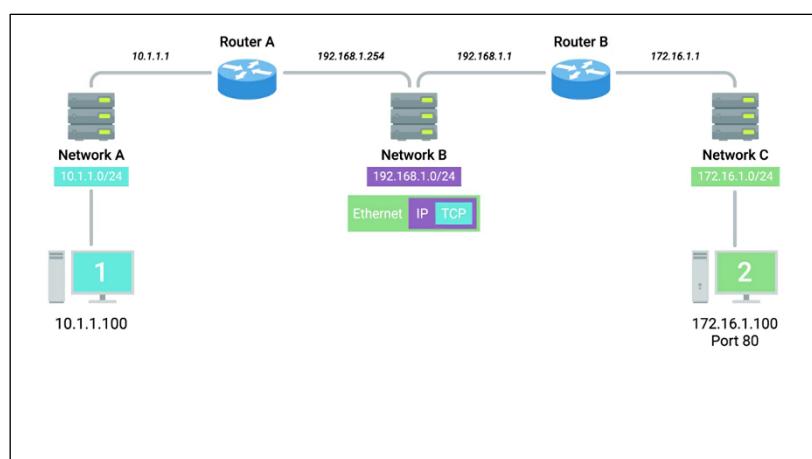
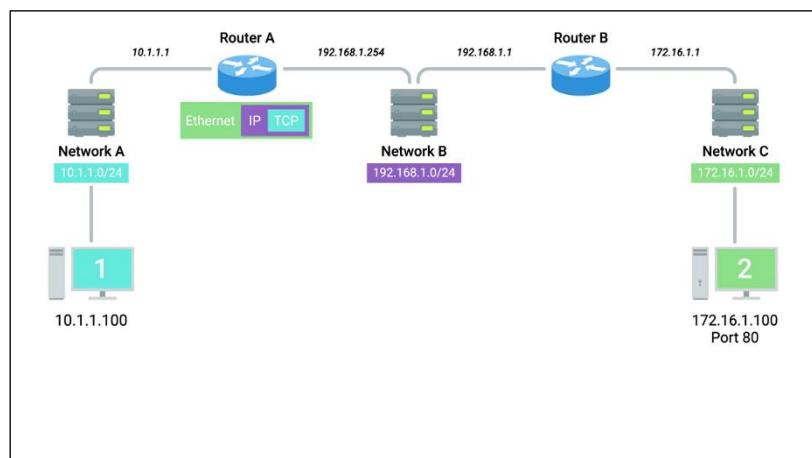
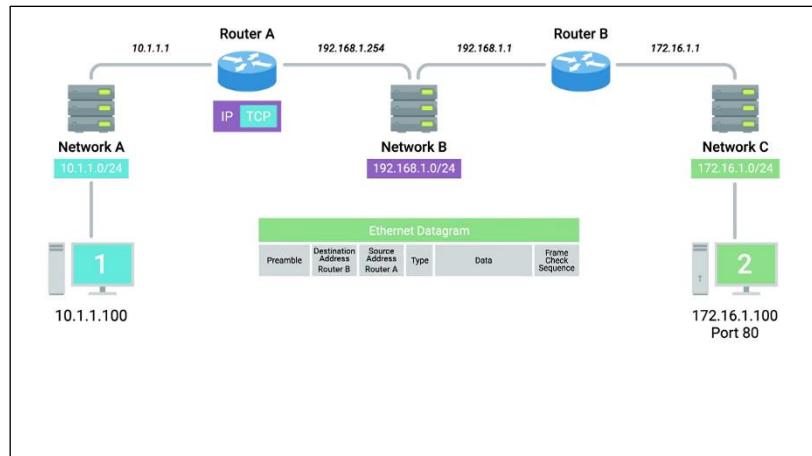
Router A looks at all the data in the IP datagram, decrements the TTL by 1, calculates a new checksum reflecting that new TTL value, and makes a new IP datagram with this data.



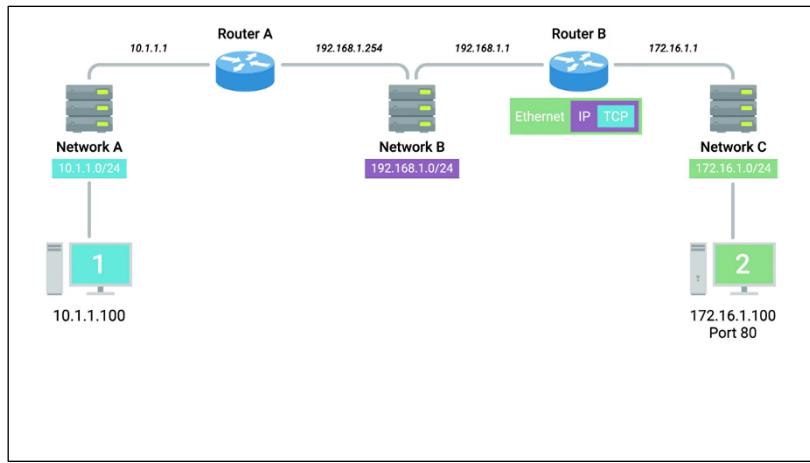
Router B knows that it needs to get this datagram to router B, which has an IP address of 192.168.1.1. It looks at its ARP table, and sees that it has an entry for 192.168.1.1.



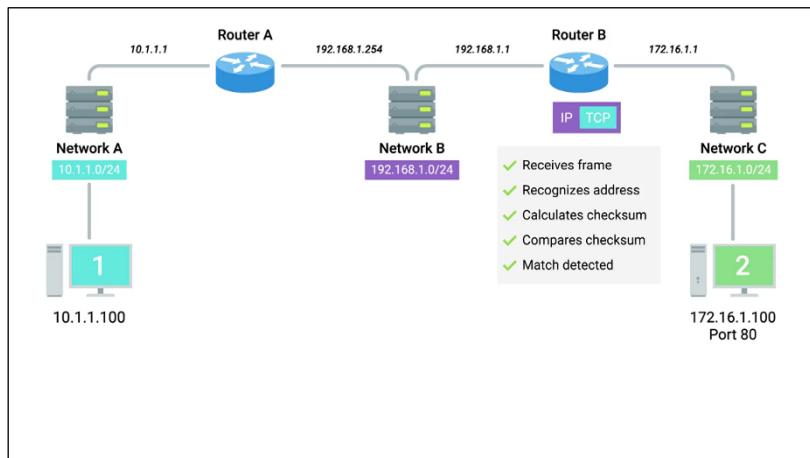
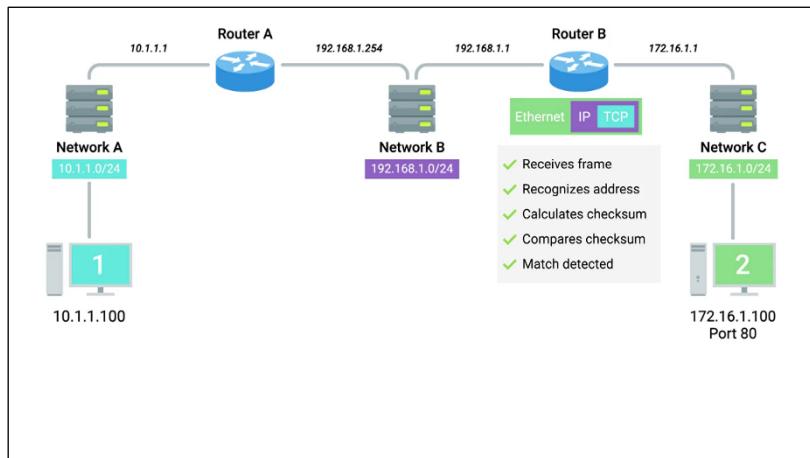
Now router A can begin to construct an Ethernet frame with the MAC address of its interface on network B as the source. And the MAC address on router B's interface on network B as the destination. Once the values for all fields in this frame have been filled out, router A places the newly constructed IP datagram into the data payload field. Calculates a checksum, and places this checksum into place, and sends the frame out to network B.

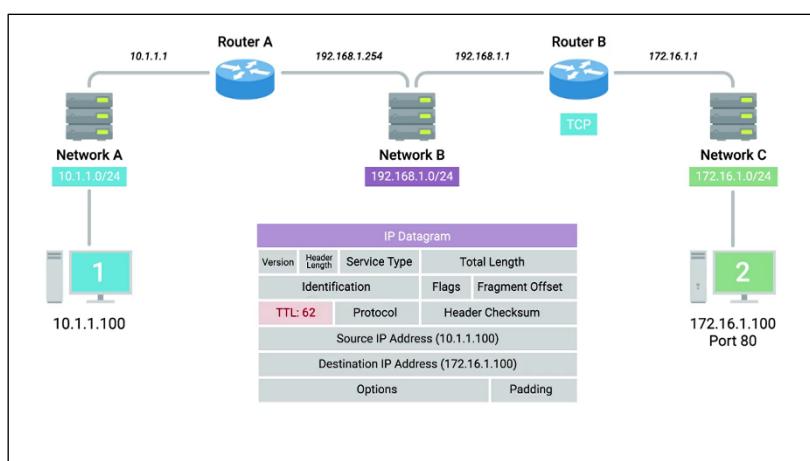
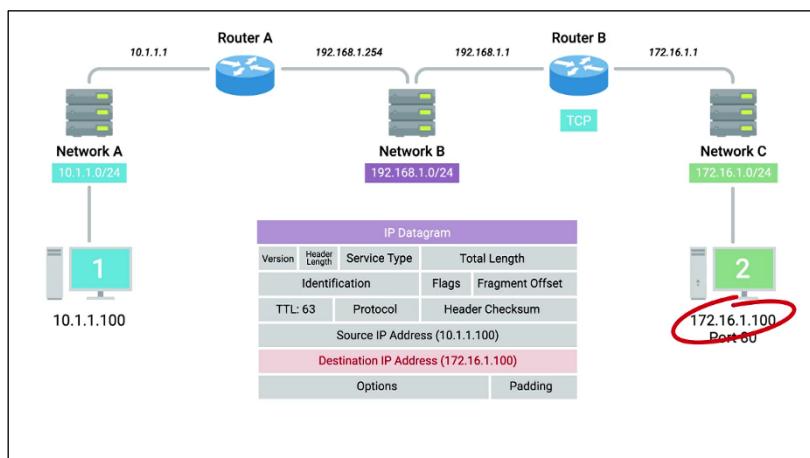
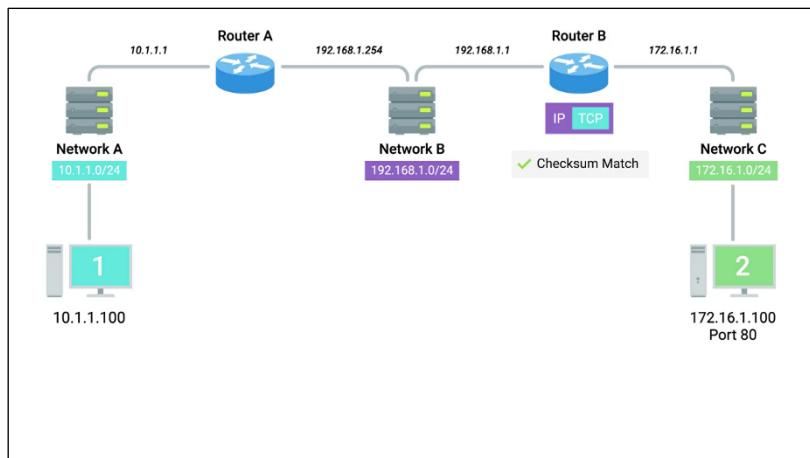


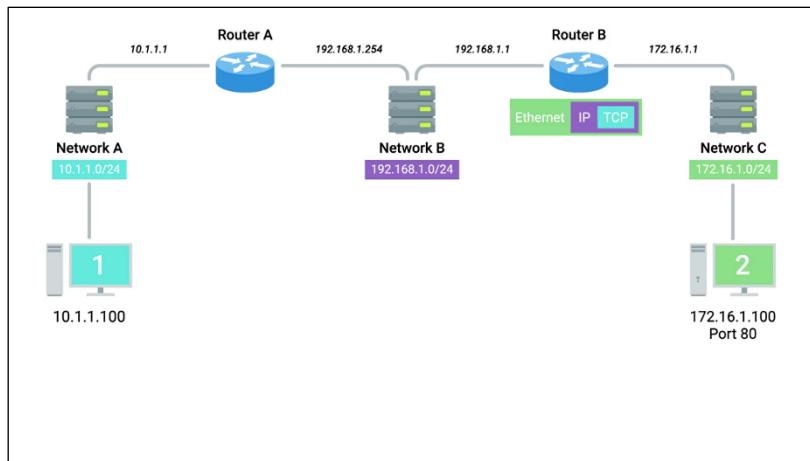
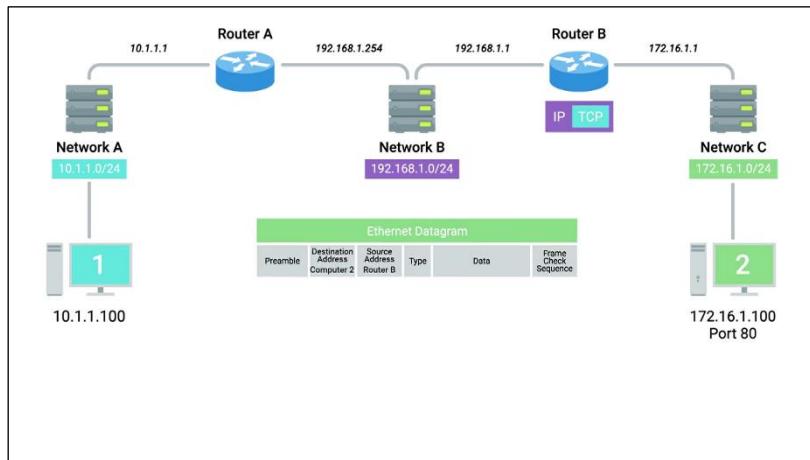
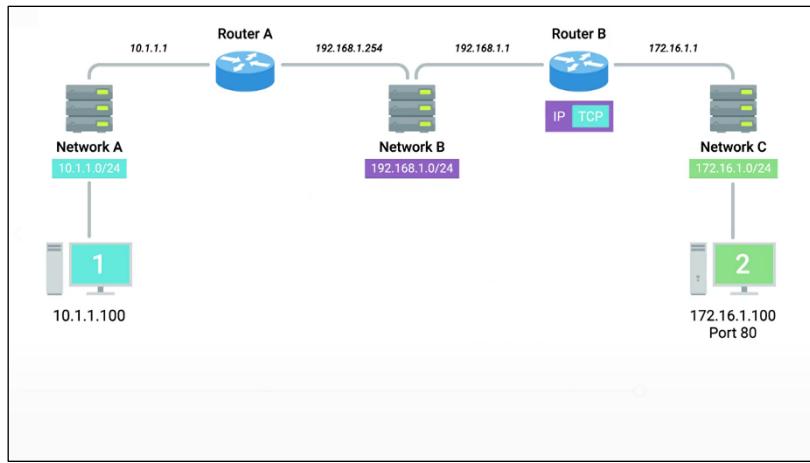
Just like before, this frame makes it across network B, and is received by router B.



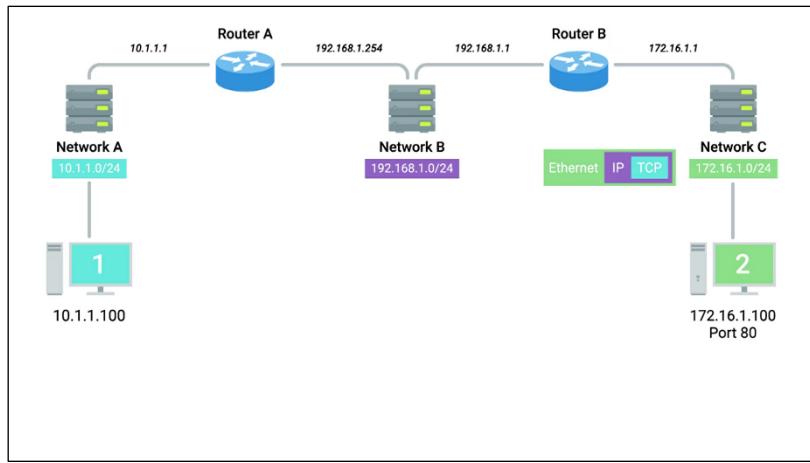
Router B performs all the same checks, removes the the Ethernet frame encapsulation, and performs a checksum against the IP datagram. It then examines the destination IP address. Looking at its routing table, router B sees that the destination address of computer 2, or 172.16.1.100, is on a locally connected network. So it decrements the TTL by 1 again, calculates a new checksum, and creates a new IP datagram. This new IP datagram is again encapsulated by a new Ethernet frame. This one with the source and destination MAC address of router B and computer 2. And the whole process is repeated one last time.



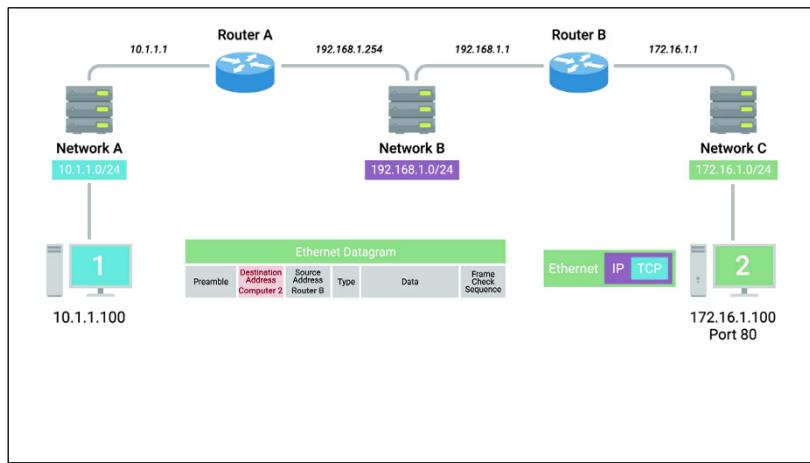




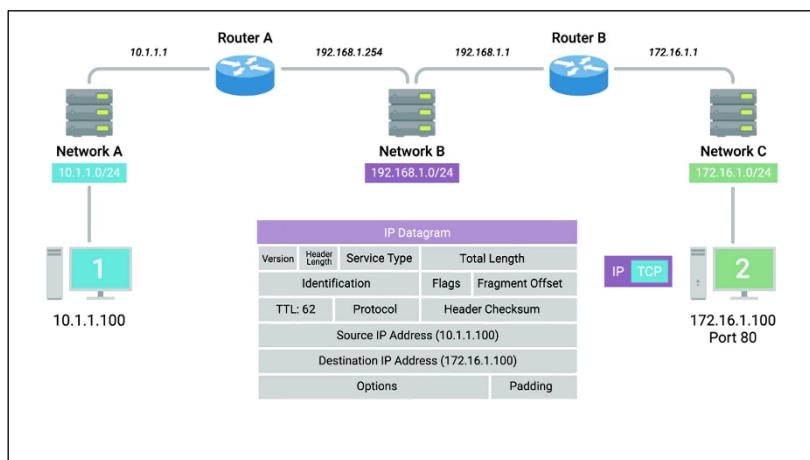
The frame is sent out onto network C, a switch ensures it gets sent out of the interface that computer 2 is connected to.



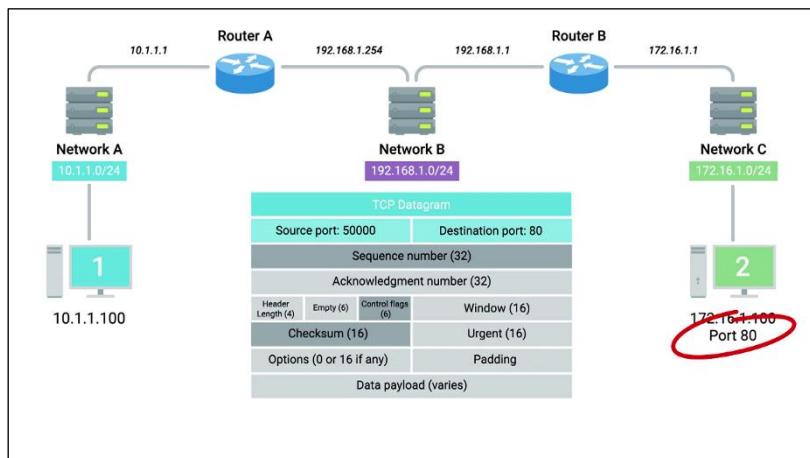
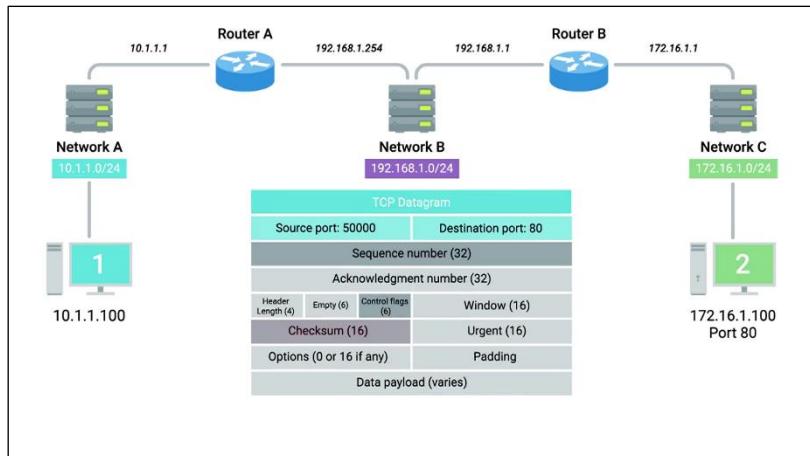
Computer 2 receives the frame, identifies its own MAC address as the destination, and knows that it's intended for itself.



Computer 2 then strips away the Ethernet frame, leaving it with the IP datagram. It performs a CRC and recognizes that the data has been delivered intact. It then examines the destination IP address and recognizes that as its own.



Next, computer 2 strips away the IP datagram, leaving it with just the TCP segment. Again, the checksum for this layer is examined, and everything checks out.



Next, computer 2 examines the destination port, which is 80. The networking stack on computer 2 checks to ensure that there's an open socket on port 80, which there is. It's in the listen state, and held open by a running Apache web server.

Computer 2 then sees that this packet has the SYN flag set. So it examines the sequence number and stores that, since it'll need to put that sequence number in the acknowledgement field once it crafts the response.

After all of that, all we've done is get a single TCP segment containing a SYN flag from one computer to a second one. Everything would have to happen all over again for computer 2 to send a SYN-ACK response to computer 1. Then everything would have to happen all over again for computer 1 to send an ACK back to computer 2, and so on and so on.

Looking at all of this end to end hopefully helps show how all the different layers of our networking model have to work together to get the job done. I hope it also gives you some perspective in understanding how remarkable computer networking truly is. Even more remarkable than that, you [LAUGH] for making it through this module.

Now it's time to apply your new knowledge to the next assessment. When you're done, I'll see in the next video, but first, another quiz, you got this. But even if you don't, just review the material until you get more comfortable with this stuff.