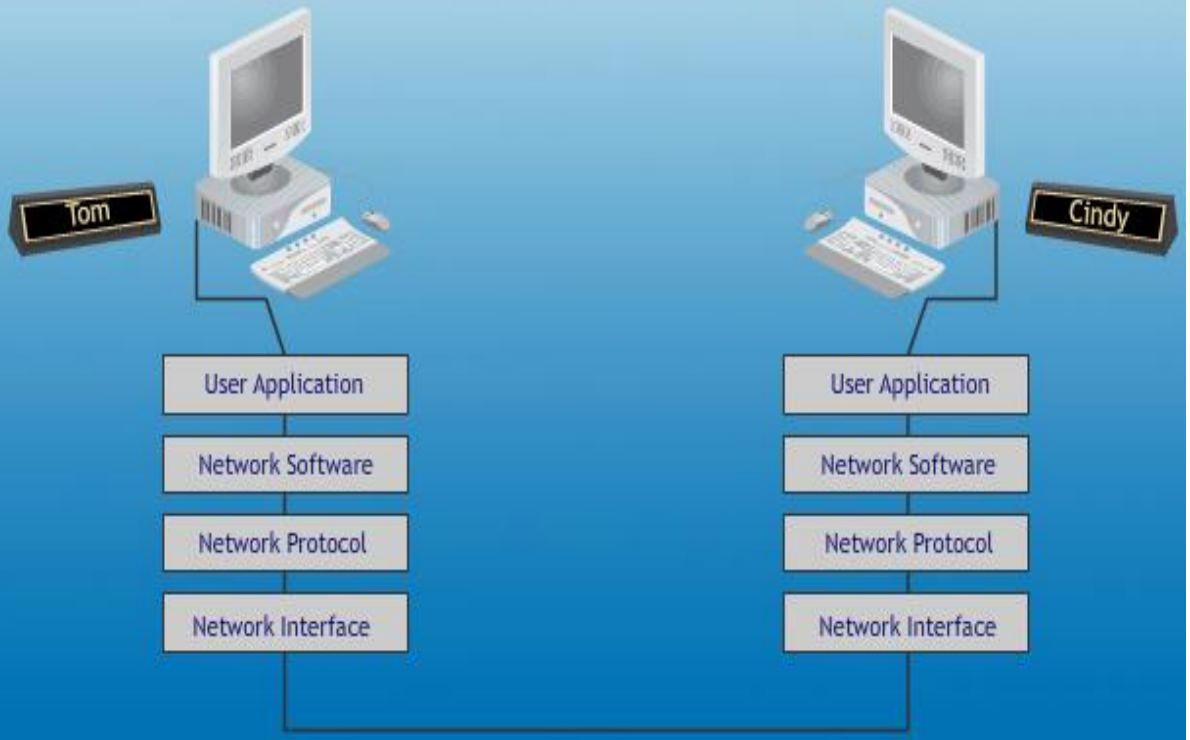
Layers of the Network Communication Process

This simulation demonstrates how networking works in layers. Tom and Cindy are using a chat program. Tom at Computer 1 types a message to Cindy. The chat program represents the user application. The user application sends the message to the network software. The network software processes the message and packages the message suitable for the network protocol to receive. The message is then passed to the network protocol which adds addressing information and packages it suitable for the network interface. The network interface, which includes the NIC driver and NIC, completes the packaging of the message and sends it across the medium as bits.

When the destination computer receives the message, the reverse process occurs.

1. The bits are received and assembled.
2. The information added by the network interface on the sending side is verified and removed and the resulting message is sent to the network protocol.
3. The information added by the network protocol on the sending side is verified and removed and the resulting message is sent to the network software.

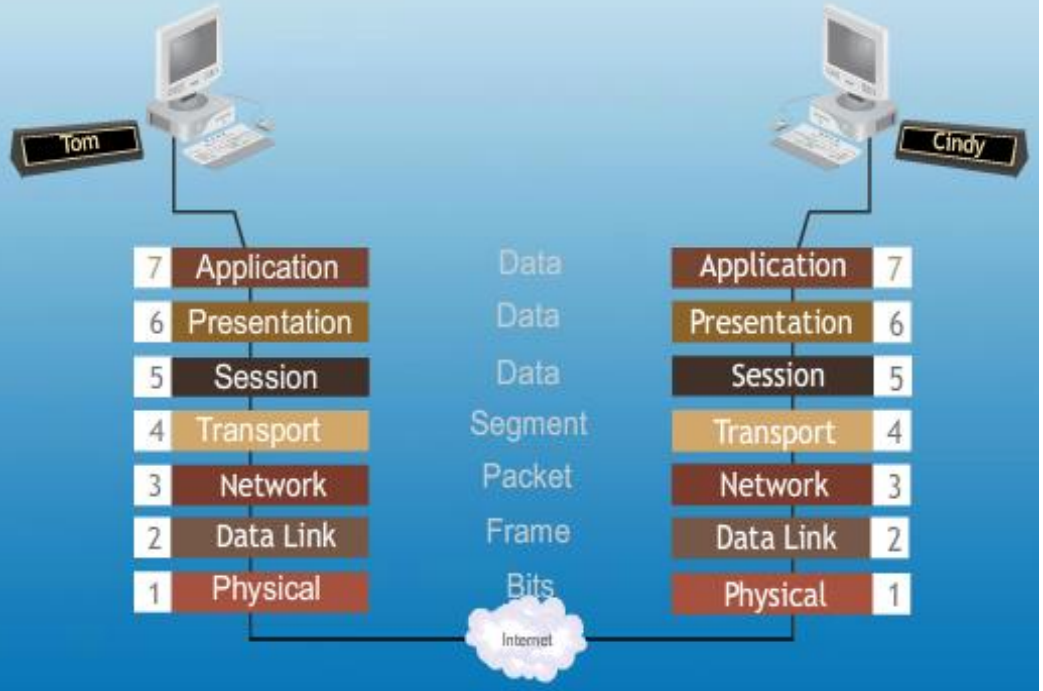


Peer Communication with the OSI Model

The OSI Model uses peer communication between its 7 layers residing on the source and destination computers. Recall that each layer in the OSI model provides services to the next higher layer until you get to the Application layer, which has the job of providing services to user applications. In the layered approach, each layer on one computer behaves as though it were communicating with its counterpart on the other computer. This means each layer on the receiving computer sees network data in the same format as its counterpart on the sending computer. This simulation is similar to Simulation 1 - Layers of the Networking Process except that more details are provided.

Tom and Cindy are using a chat program. Tom types a message to Cindy. The Chat program communicates with the Application layer to request network services. The protocol data unit, or PDU, at this layer is just called Data. The Application layer sends the data to the Presentation layer for any necessary formatting. The Session layer, if needed, maintains the communication session. The PDU is still referred to as Data at the Presentation and Session layers. The Transport layer breaks the Data PDU into smaller Segments, if necessary, and adds port and sequencing information as needed. The Segment is transferred to the Network layer which adds logical addressing information to create a Packet. The Packet is sent to the Data Link layer which adds physical address information - the source and destination MAC addresses, and a CRC to create a Frame. The Data Link layer also gains access to the media. The Frame is passed down to the Physical layer which transfers the frame across the medium as bit signals.

When the message arrives at the destination computer, the bits are read in by the Physical layer and transferred up to the Data Link layer as a Frame.



The Data Link layer verifies the Destination MAC address and CRC and strips the Frame header and trailer and hands off the resulting Packet to the Network Layer. The Network layer verifies the logical address in the Packet header, strips the Packet header and sends the resulting Segment up to the Transport layer. The Transport layer strips off the Segment header, assembles multiple segments if necessary, uses the port number to determine which Application layer service should receive the message and sends the resulting Data to the Session layer. The Session layer passes the data to the Presentation layer and finally to the Application layer. The Application layer sends the original message to the chat program and the message is received.

Communication Between Two Computers

In this simulation we look at the communication process between two computers.

A user at Computer A uses the Ping program to send Computer B a Ping message. The ping message is created and the network protocol packages the message by adding the IP address of Computer B as the destination address, and Computer A's address as the source or return address.

The network protocol also acquires the MAC address of Computer B - using a process called ARP that is described in Chapter 5 of Networking Essentials. The network interface then adds the MAC address of both computers to the message. The network interface now sends the ping message onto the medium as bits.

When Computer B receives the message, it creates a reply. Since both the IP address and MAC address of Computer A are contained in the original message, Computer B simply adds Computer A's IP and MAC address to the ping message and sends it. Computer A successfully receives the reply.

Destination MAC	Source MAC	Dest. IP	Src. IP	Data	Error Check
00:0C:21:44:15:3C	00:0C:21:35:48:16	10.1.1.2	10.1.1.1	Ping Message	CRC
Destination MAC	Source MAC	Dest. IP	Src. IP	Data	Error Check
00:0C:21:35:48:16	00:0C:21:44:15:3C	10.1.1.1	10.1.1.2	Ping Reply	CRC

Basic Operation of a Hub

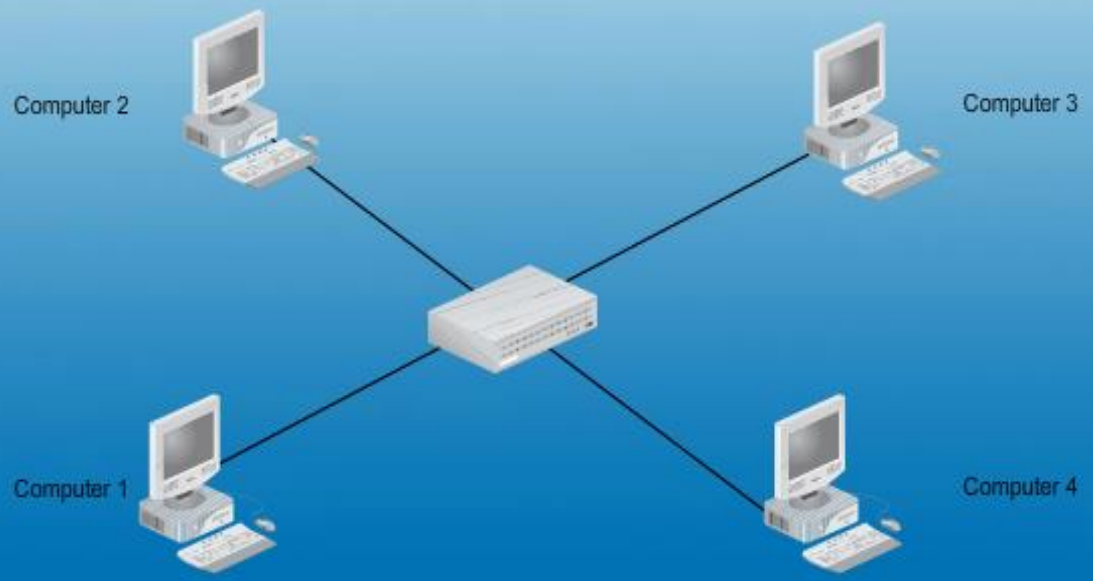
This simulation illustrates the basic operation of a hub, or multiport repeater.

Computer 1 sends a ping message to Computer 3. When the ping reaches the hub, it doesn't see the actual data in the message. It only sees bit signals. It strengthens the signals, cleans them up and sends them out all connected ports.

When Computers 2 and 4 receive the bit signals they reassemble them into frames, check the destination address, and discard the frame because it is not addressed to them. Computer 3 receives the frame successfully and sends a reply.

Once the reply reaches the hub, the hub again forwards the bit signals out all connected ports.

Computers 2 and 4 discard the frame. Computer 1 successfully receives the ping reply.



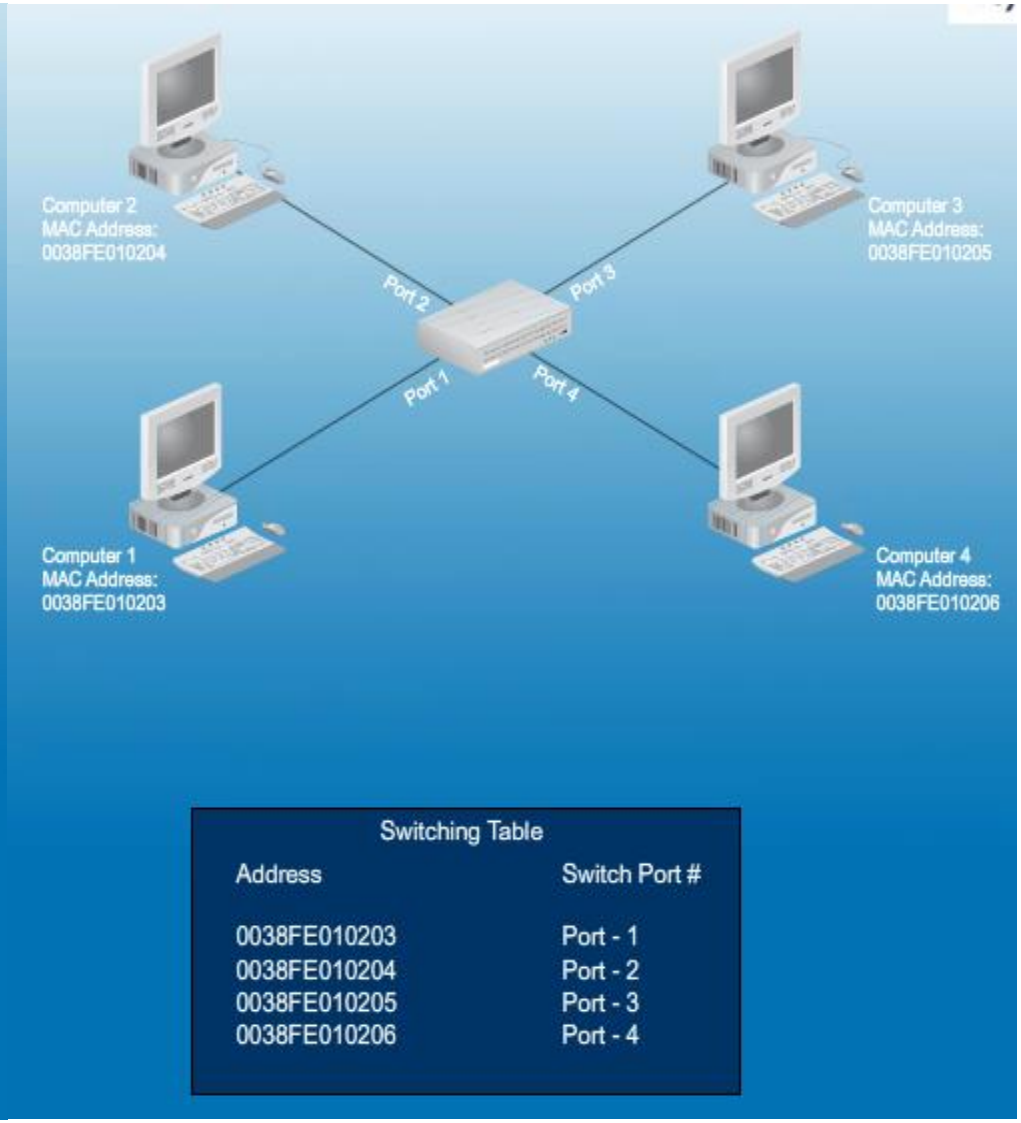
Basic Operation of a Switch

In this simulation we look at the basic operation of a switch.

Computer 1 sends a ping to Computer 3
When the ping frame reaches the switch, the switch unlike a hub reads the data in the frame; in particular, it reads the source and destination MAC addresses.

The switch looks up the destination MAC address in its switching table. In this case, it finds that Computer 3 is connected to port 3 so the switch forwards the frame out port 3. Computers 2 and 4 never see the ping frame.

When Computer 3 receives the ping frame, it sends a ping reply back to Computer 1. When the frame reaches the switch, again the switch consults the switching table and finds Computer 1 is connected to port 1 – the frame is forwarded out port 1 and is received by Computer 1.



How a NIC Works

This simulation gives you an idea of how a NIC and its driver works.

For incoming data, bit signals travel along the medium and are received by the NIC.

The received bits are formatted into a frame.

The CRC is calculated and compared to the the CRC in the frame trailer. If they don't match, the frame was damaged or changed and the frame is discarded. This situation is rare but can happen in electrically noisy environments or if the media is poorly terminated.

If the CRC is okay, the destination MAC address is checked. If it matches the NIC's burned in address or is a broadcast, the frame is processed; otherwise it is discarded.

Once the MAC address is verified, the frame header and trailer are stripped creating a packet which is sent to the network protocol for further processing. The NIC's job is done.

Router Operation in a Simple Internetwork

This simulation looks at the operation of a router.

Computer 1 wants to send data to Computer 2. The packet header destination address is filled in with the IP address of Computer 2 and the source is filled in with the IP address of Computer 1. Then the frame is created.

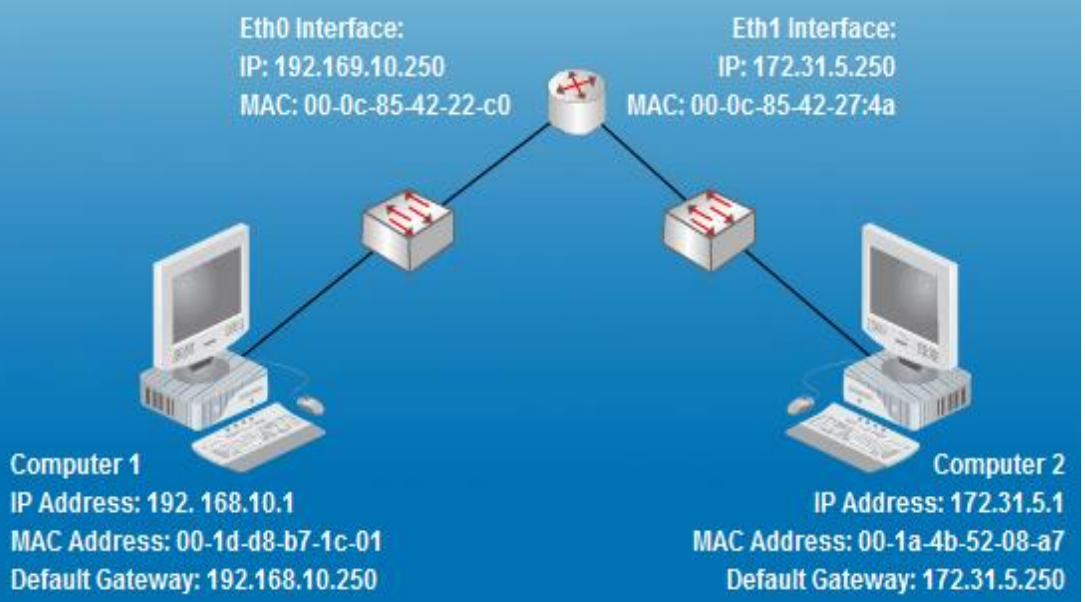
The frame is going to be delivered initially to the router's Eth0 interface, so the destination MAC address in the frame header is the MAC address of the router's Eth0 Interface. Computer 1 fills in the source MAC address with its own MAC address and sends the frame.

When the frame reaches the router, the router reads the destination IP address. The router consults its routing table and finds if it has a match for the destination network.

In this case, the router determines that the destination network, 172.31.0.0 is directly connected to its Eth1 interface. The router creates a new frame using Computer 2's MAC address as the destination and the MAC address of its Eth1 interface as the source and forwards it out the Eth1 interface.

Computer 2 successfully receives the frame.

Routing Table	
Network	Interface
192.168.10.0	Eth0
172.31.0.0	Eth1



Ethernet Operation Using CSMA/CD

This simulation shows how Ethernet works using CSMA/CD media access control in a network that uses a hub.

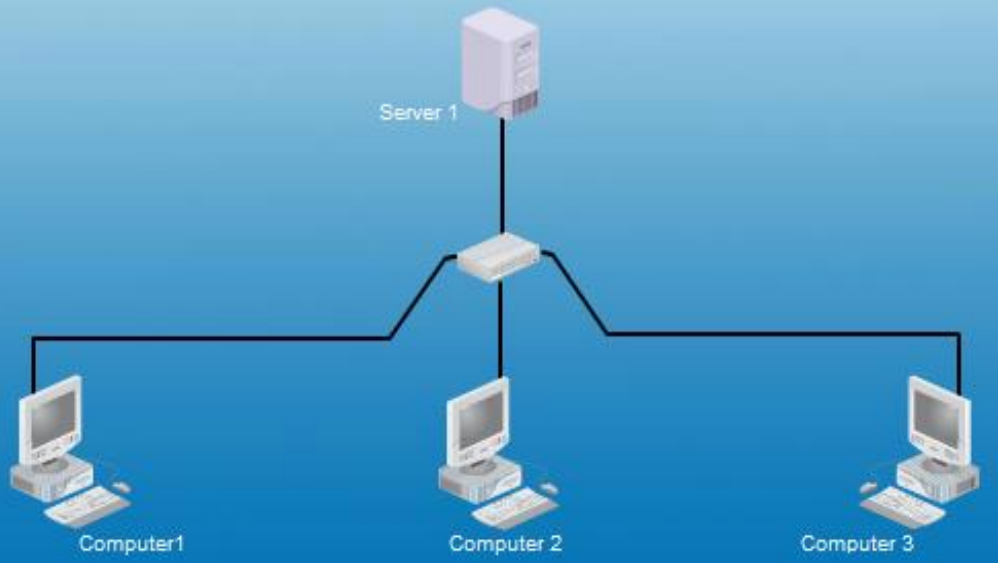
Computer 1 wants to send a frame to Server 1. Computer 1 must first listen to the media to determine whether another station is transmitting - this is the Carrier Sense part of CSMA/CD. No network activity is detected so Computer 1 sends the frame and it is received by Server 1.

When Server 1 receives the frame, it creates a reply. However, Computer 3 also has data to send to Server 1. Computer 3 and Server 1 both listen to the media and detect no activity, so both computers sends the frame.

The rules of CSMA/CD say that only one computer at a time can send data when using a hub. Computer 3 and Server 1 continue to listen to the media as they transmit their frames. If they detect network activity before they finish sending their frames, a collision has occurred. The collision is detected by all computers in the collision domain, causing all computers to pause for a random amount of time before attempting to send any more frames.

In this case, Server 1 tries first to re-send its frame. Since no activity is detected, the frame is sent successfully. Computer 3 is now able to send its frame.

Remember that collisions rarely occur in networks that use switches, and not at all if all computers operate in full-duplex mode. So, while you may not need to be concerned about collisions in most networks, the obvious performance degradation that comes with collisions helps explain why switches were developed to replace hubs.



Wireless LAN Operation

W-Fi operation uses the Carrier Sense/Multiple Access with Collision Avoidance (CSMA/CA) access method. When Computer 1 wants to send a data frame to Computer 4, it first listens, or senses the media to be sure it's free. If the media is busy, the station must wait. If the media is free, Computer 1 transmits the frame to Computer 4.

The access point receives the message and in a similar manner to a hub, repeats it. Computer 4 receives the frame while computers 2 and 3 discard it. Computer 4 then sends an acknowledgement frame back to Computer 1 to verify the frame was received successfully.

An advanced operational WiFi mode uses handshaking. This mode is usually only supported by high-end wifi NICs and is used when stations are located far apart, causing the hidden node problem. In this mode, Computer1 again must first listen to verify the media is not in use. If the media is free, Computer1 must in a sense, ask permission to transmit by sending a request to send (RTS) message to the access point. The access point replies with a Clear to Send (CTS) message if the media remains free.

The CTS message is heard by all other stations, notifying them that another station is about to transmit and that they must wait before attempting to transmit. The amount of time to wait is part of the CTS message.

Computer 1 now sends the data frame to Computer 4. Computer 4 receives the frame and then sends an acknowledgement frame back to Computer 1.

The Changing Frame Header

This simulation shows how the frame header changes as a packet travels through the internetwork.

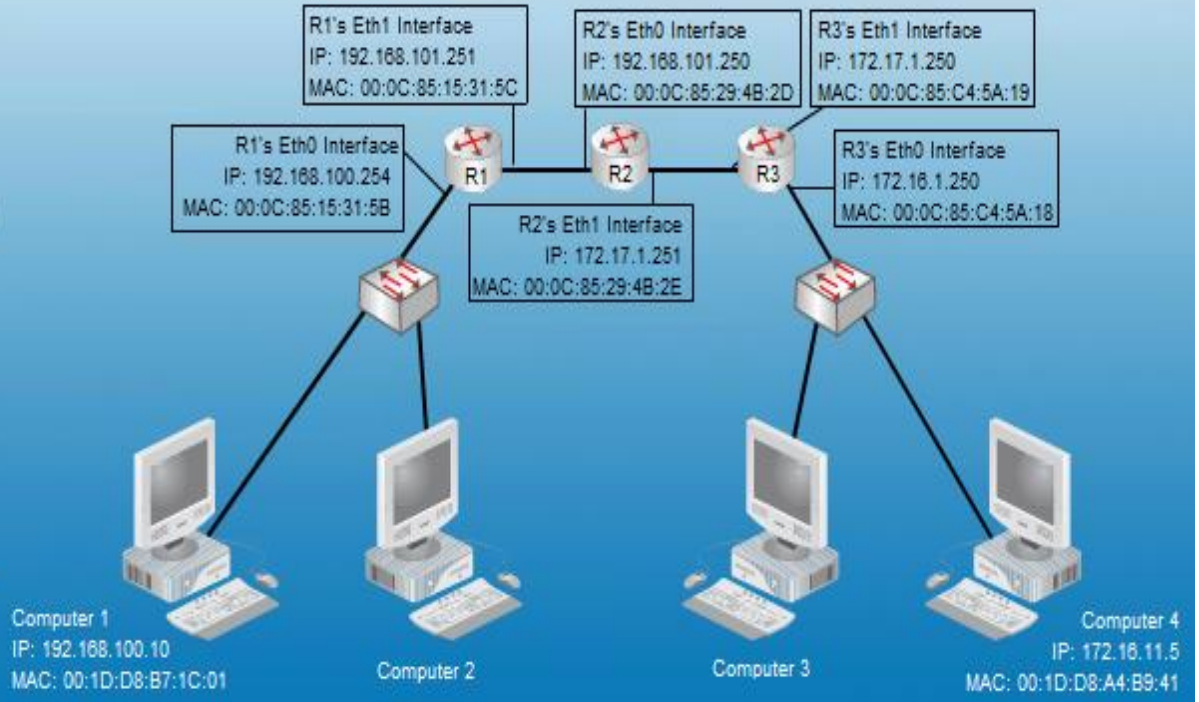
Computer 1 sends a Ping message to Computer 4. The ping packet header contains the source and destination IP addresses, which remain the same throughout the packet's journey.

Computer 1 determines that the destination IP address is located on another network, so the frame must be delivered to Computer1's default gateway, which is Router 1's Eth0 interface. Computer 1 checks its ARP cache to see if Router 1's Eth0 MAC address can be found. Computer 1 does not have Router 1's MAC address in its ARP cache, so it sends an ARP request to retrieve it.

Router 1 responds with an ARP reply containing the MAC address of its Eth0 interface. The ping frame is constructed using Router1's Eth0 MAC address as the destination MAC and Computer1's MAC address as the source. Computer 1 sends the ping frame which is received by Router 1.

Router 1 determines that the ping packet should be forwarded to Router 2. To do so, the frame header must be changed with Router 1s Eth1 Interface as the source address and Router 2s Eth0 interface as the destination. Router 1 finds Router 2's MAC address in its ARP cache. Router 1 constructs the new frame header, calculates a new CRC and sends the frame.

Router 2 receives the frame, and determines that the ping packet should be forwarded to Router 3. Router 2 finds Router 3s MAC address in its ARP cache, constructs the new frame and sends it.



When Router 3 receives the frame, it determines that the destination IP address in the ping packet is in the network connected to its Eth0 interface. Router 3, checks its ARP cache and finds Computer 4's IP address in its ARP cache. Router 3 constructs the new frame and delivers it to Computer 4.

As you can see, the frame header and CRC are changed at each router along the route to the final destination; but the source and destination IP addresses stay the same.

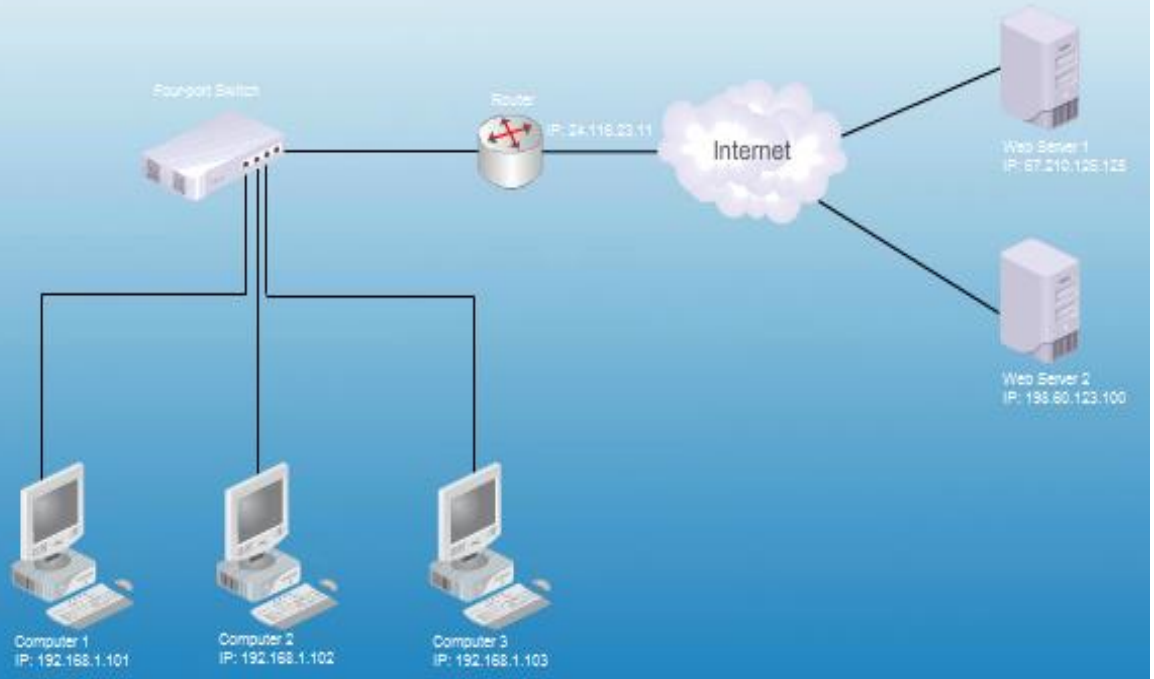
Demonstrating NAT/PAT

This simulation demonstrates how Network Address Translation (NAT) / Port Address Translation (PAT) work. This configuration demonstrates PAT as it is implemented in a typical small office/home office (SOHO) network; but the process is essentially the same in large enterprise networks.

Computer 1 wants to request a Web page from Web Server 1. The request is formatted with the source and destination IP addresses in the IP header and the source and destination port numbers in the TCP header. Web servers use port 80 and is the destination port. The source port is basically a random number assigned to the open Web browser window making the request. The packet is sent to the router for delivery out to the Internet. Since the network is using private IP addresses for host assignment, the source address must be translated to a public address.

The router performs the address translation using the address of its interface that's connected to the Internet. The router maintains a NAT table that keeps a record of active translations. The source IP address and port number in the outgoing packet are replaced with a public address and port number. While the port number is usually the same in the original and translated address, it doesn't have to be. The packets is then sent out onto the Internet with its new source IP address and port and is delivered to the destination Web server.

The Web server sends a reply. When the reply reaches the router, the translation is reversed. The packet is delivered to Computer 1. Each packet that is sent from Computer1's Web browser to Web Server1 uses the same translation.



Now, suppose Computer2 requests a Web page from Web Server 2. The same process occurs, except that Computer 2's packets are assigned a different source port number. Web server2 replies and once again the translation is reversed when the reply packet reaches the router.

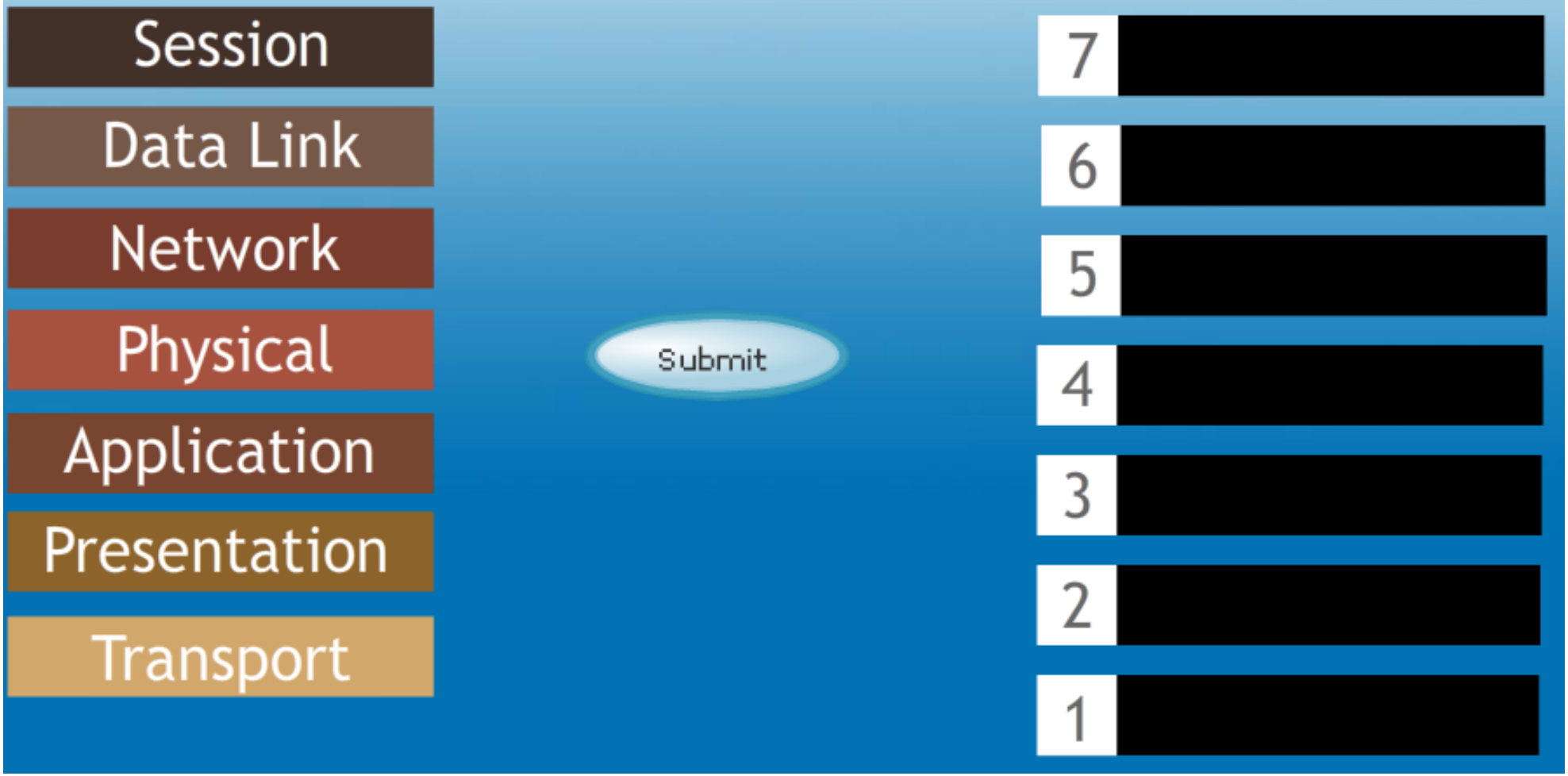
Each communication session between a private computer and an Internet computer is translated in this manner, so a single computer might have several translations that the NAT router must keep track of. For example, each Web browser window or tab requires a different port number and therefore has a unique translation assigned to it.

After a while, if no packets between the source and destination computer are transmitted, the entry in the NAT table is deleted.

OSI Model: Layer Names Activity

Instructions:

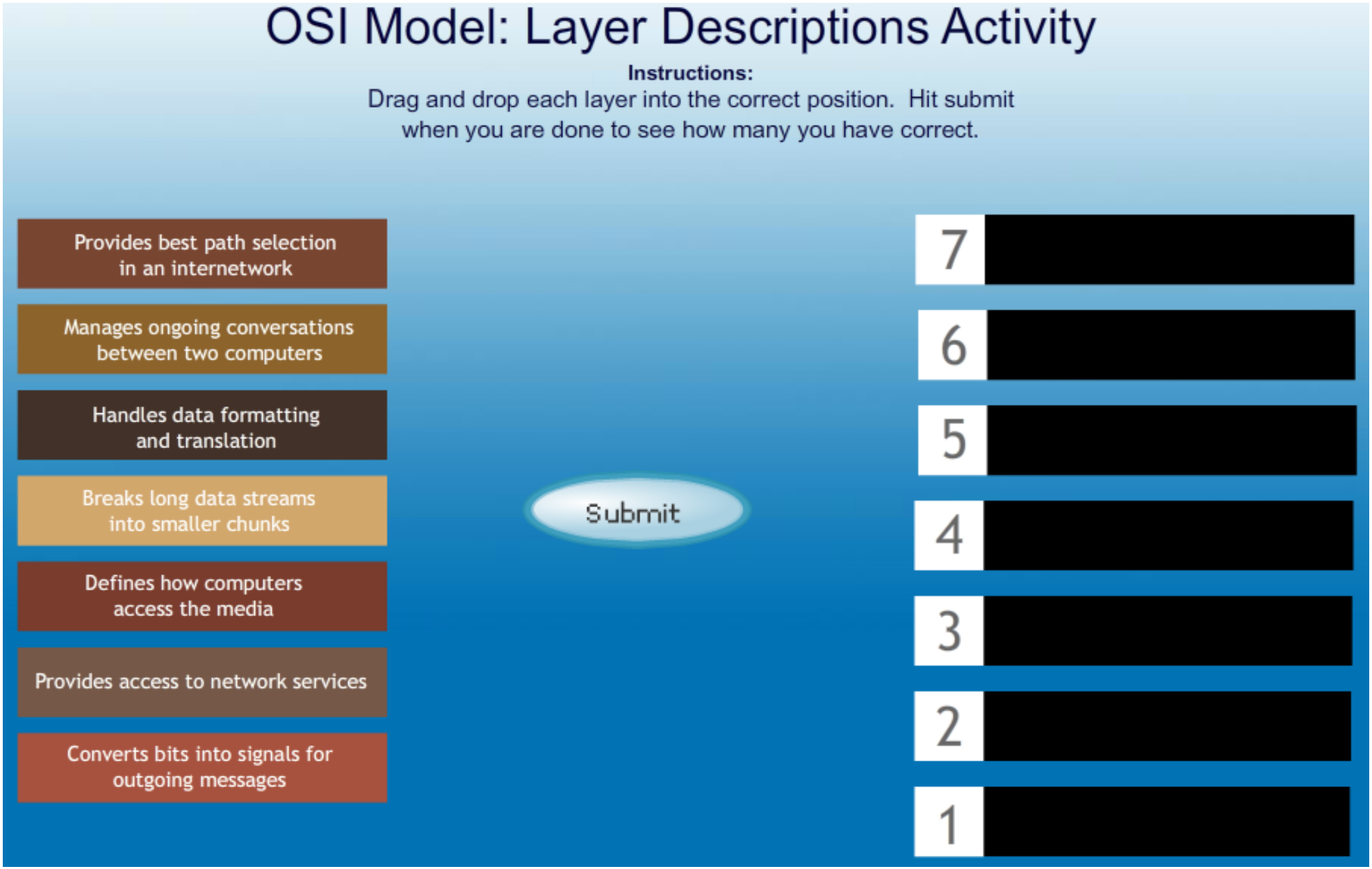
Drag and drop each layer into the correct position. Hit submit when you are done to see how many you have correct.



OSI Model: Layer Descriptions Activity

Instructions:

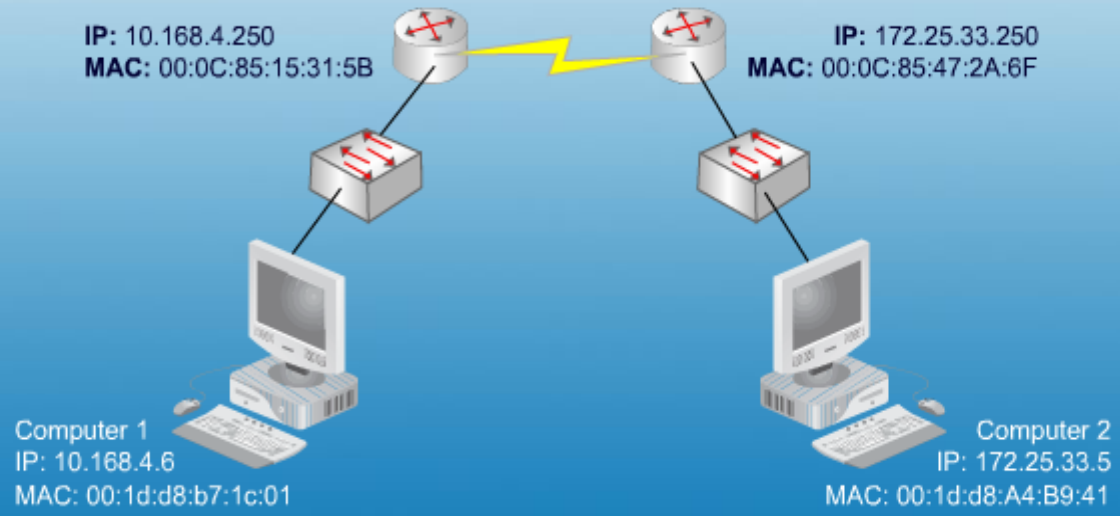
Drag and drop each layer into the correct position. Hit submit when you are done to see how many you have correct.



Build a Data Frame Activity

Frame Headers

- Data
- Dst: 172.25.33.5
  Src: 172.25.33.250
- Dst: 00:1d:d8:A4:B9:41
  Src: 00:1d:d8:b7:1c:01
- Dst Port: 80
  Src Port: 4498
- Dst: 10.168.4.250
  Src: 10.168.4.6
- Dst: 00:0C:85:15:31:5B
  Src: 00:1d:d8:b7:1c:01
- Dst: 172.25.33.5
  Src: 10.168.4.6
- Dst: 00:1D:D8:A4:B9:41
  Src: 00:0C:85:47:2A:6F
- Dst Port: 25
  Src Port: 7199
- CRC



Instructions:
Drag and drop the frame headers into the spaces below. Create the frame that Computer 1 sends to Computer 2 to request a Web page. Click submit. to see how you did.

Submit