Assignment 1

Network Simulator Documentation

CSC 677

Kathryn Reardon

September 20, 2020

Network Scenario and Networking and Programming Concepts

The following concepts and scenarios were used in the development of this network simulator.

Scenario: Node-to-Node via Two Routers

The simulator’s design is based on the simple scenario of transmitting a message from one node to another through a link with two routers.

Concept of Peers

1. Network Topology

There are four devices in the network: the sending host H1, two routers R1 and R2, and the receiving host H2. The network has a point-to-point topology with three links between H1 and R1, R1 and R2, and R2 and H2.

1. Layering

The simulator’s design represents the OSI model of the layered network system framework as the presentation and session layers are separated from the application layer. The logical connections between each layer are represented by their horizontal alignment, background color, and identical objects (e.g., clicking on either transport layer after the message has moved into or past displays the same TCP segment). The scope of this simulator covers the transport layer.

1. Protocols

The simulator follows the Transmission Control Protocol (TCP) in the transport layer to transform the user’s input message and generate a segment. This is represented by the pop-up TCP segment which includes the calculated checksum field.

1. Transmission

The simulator animates the user’s input message moving down through the network layers as it is transformed into a segment, transmitted as a bit sequence via the link, and transformed again as it travels up through the corresponding network layers of the receiving host. Because only the transport layer is implemented, the encapsulation and decapsulation process is not as obvious but can be seen in how the transport layer adds the pseudoheader and header to the data of the message.

1. Peer-to-Peer Transmission

The simulator shows a message sent from H1 to H2. The link between the two computers is assumed to be half-duplex so that while H1 is sending the message, H2 is open to receive. While the channel is in use, H1 cannot receive any transmissions, and H2 cannot send any transmissions.

Concept of Segments

1. Reliable Network Services

Since the simulator uses TCP in the transport layer, it simulates a reliable, connection-oriented service. However for this assignment, flow, error, and congestion control are not implemented (although the checksum for TCP is calculated).

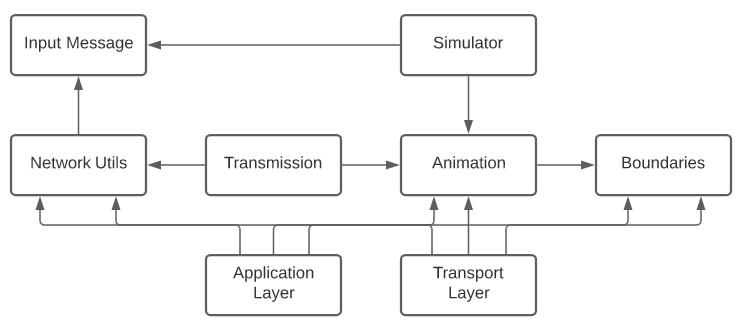
1. Encapsulation

As mentioned before, the simulator shows the encapsulation of the user’s message at H1’s transport layer with the addition of the TCP header and creation of the TCP segment. Decapsulation occurs at H2’s transport layer and the message is delivered to the receiving process.

1. Structural, Functional, and Logical Characteristics

The structural characteristics of a TCP segment are shown very clearly in the simulator’s pop-up window. The appearance and labels are based on Figure 24.9 on page 750 of *Data Communications and Networking*. The functional characteristics of the TCP segment that are included in the simulator are randomly generated Class C IP addresses, randomly generated port numbers, a randomly generated window size, and a checksum. The logical connection between segments is shown when clicking on either the H1 or H2 transport layer displays the same TCP segment once the message has reached the layer.

Object Modeling



An arrow from one object to another denotes the object is dependent on the other.

User Manual

Portability

The program is built using Java 8, and the jar file NetworkSimulator.jar is the executable file.

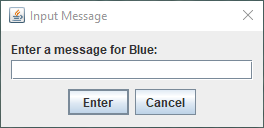
Interaction

When the program starts, the user is asked to enter a message for the blue envelope. Clicking on the application layer, transport layer, or link before the envelope has passed into it gives a notification to that effect. Clicking on the application layer, transport layer, or link after the envelope has passed into it displays the user’s input message, TCP segment, or bit sequence respectively. The animation is paused while the pop-up windows are visible and resumes when the user closes them. The user may also click the pause and play buttons to pause and play the animation.

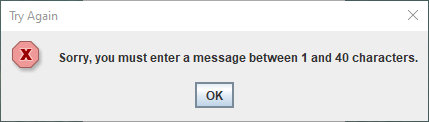
Segments

The TCP segment for the blue envelope is displayed at the transport layer. It consists of the blue pseudoheader, white header, and yellow data. Its values are in hexadecimal.

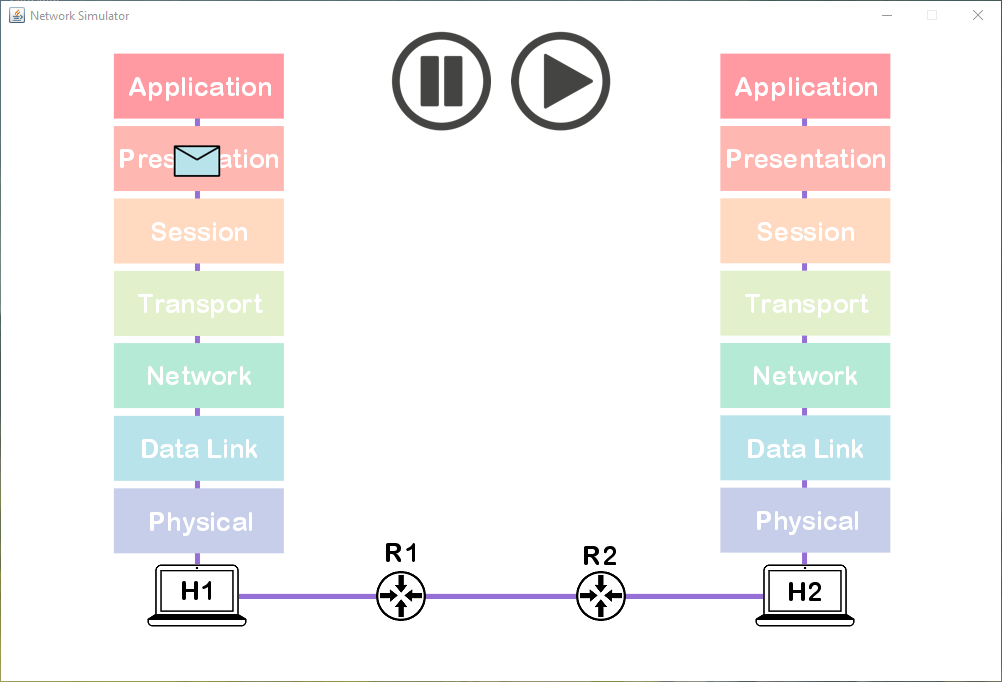
Screen Shots



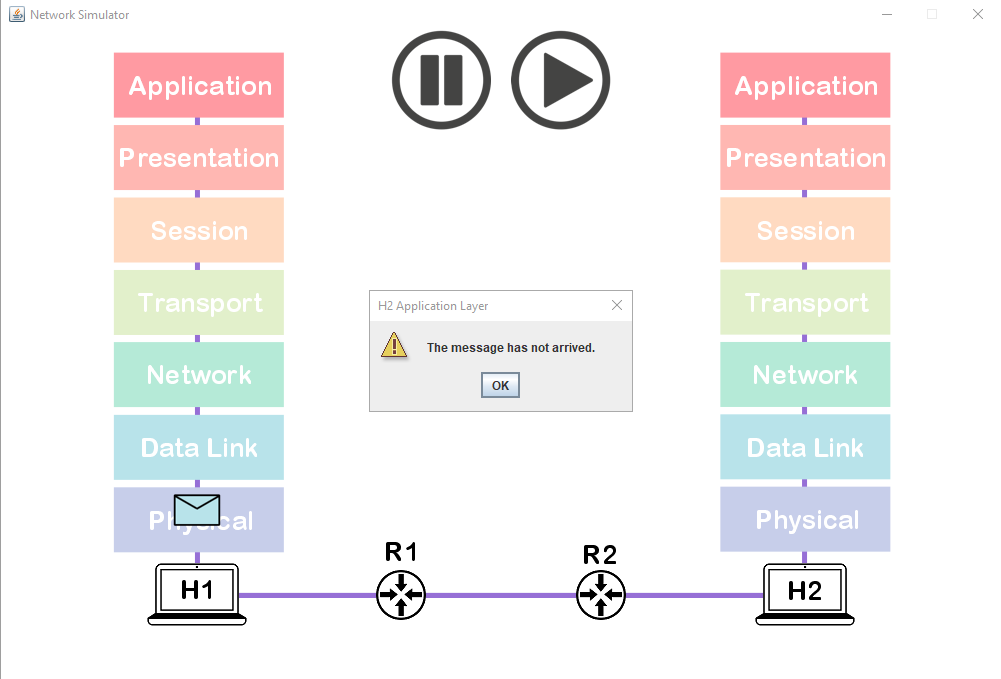
This window appears when you run the program.



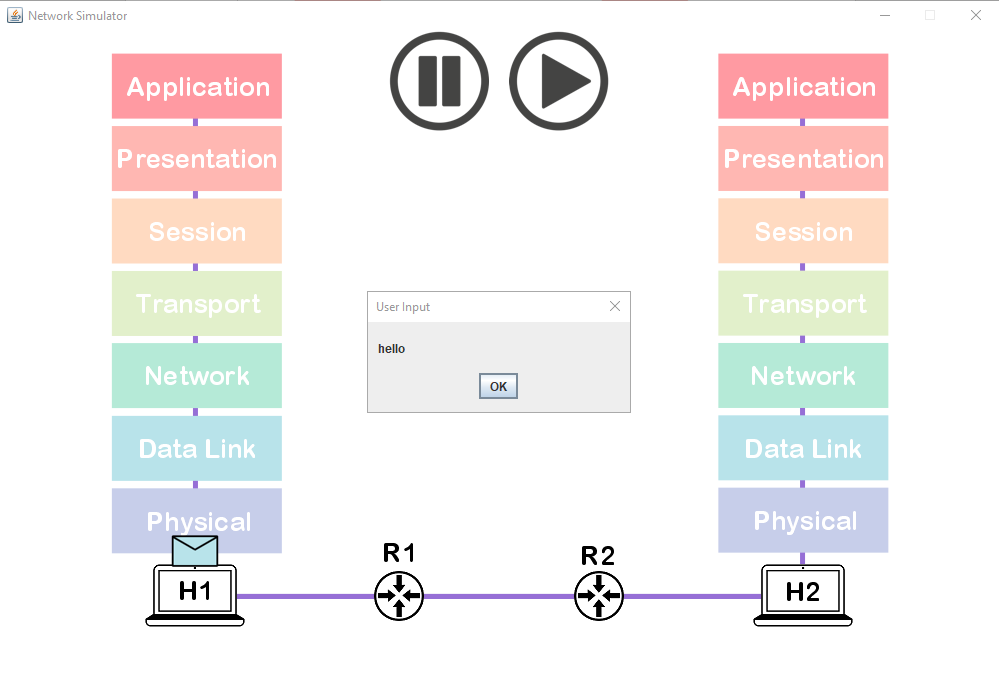
The animation will not begin unless the character limit is met.



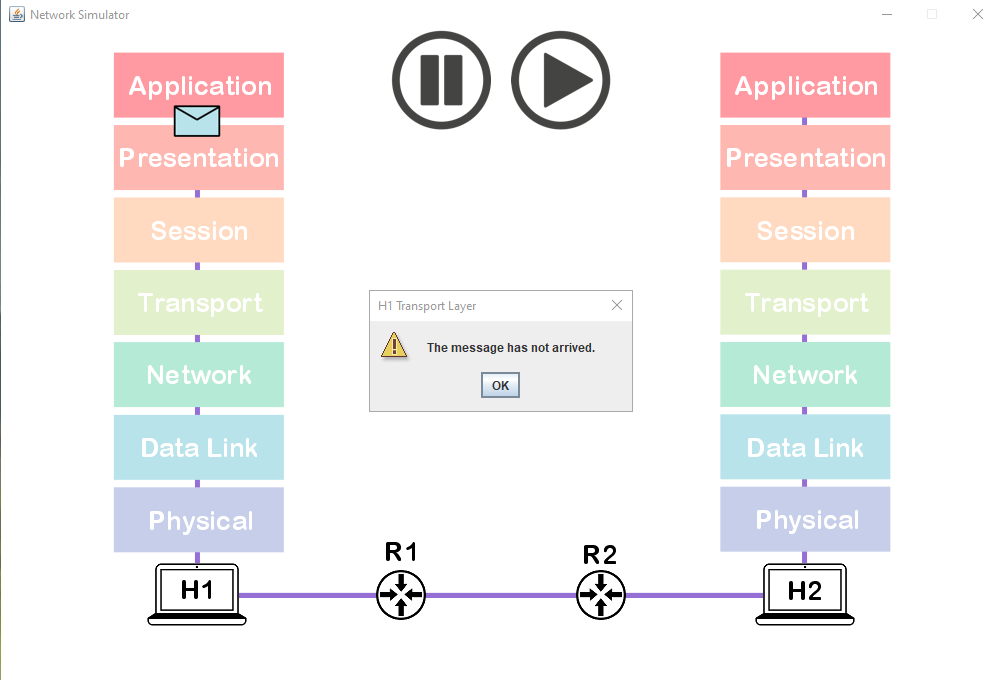
The animation begins.



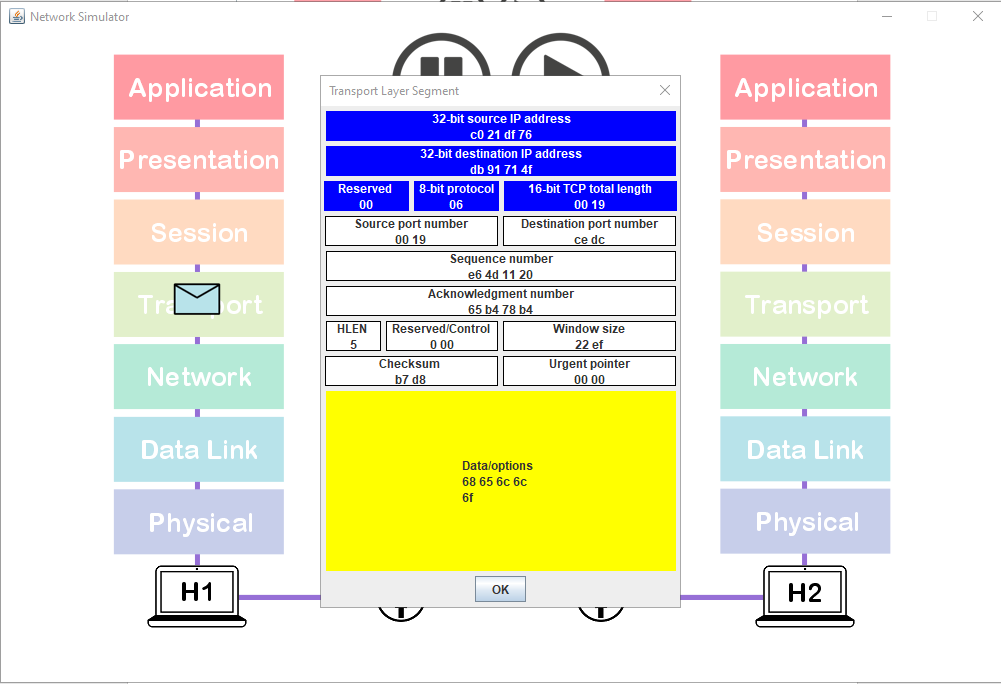
Clicking on either application layer before the envelope has passed into it pauses the animation and gives a notification.



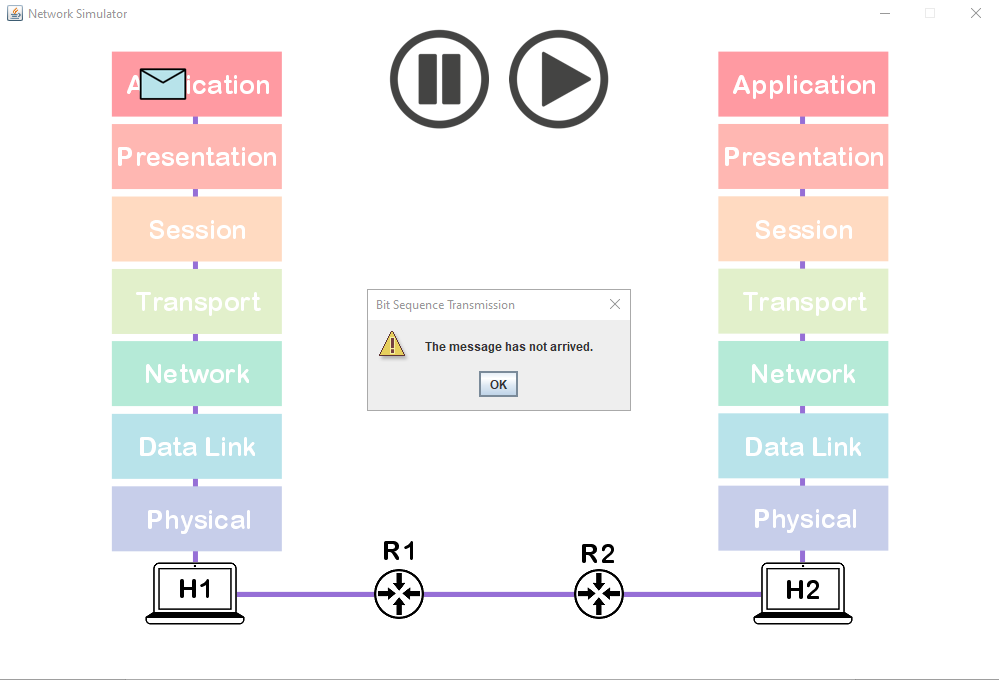
Clicking on either application layer after the envelope has passed into it displays the user’s input message.



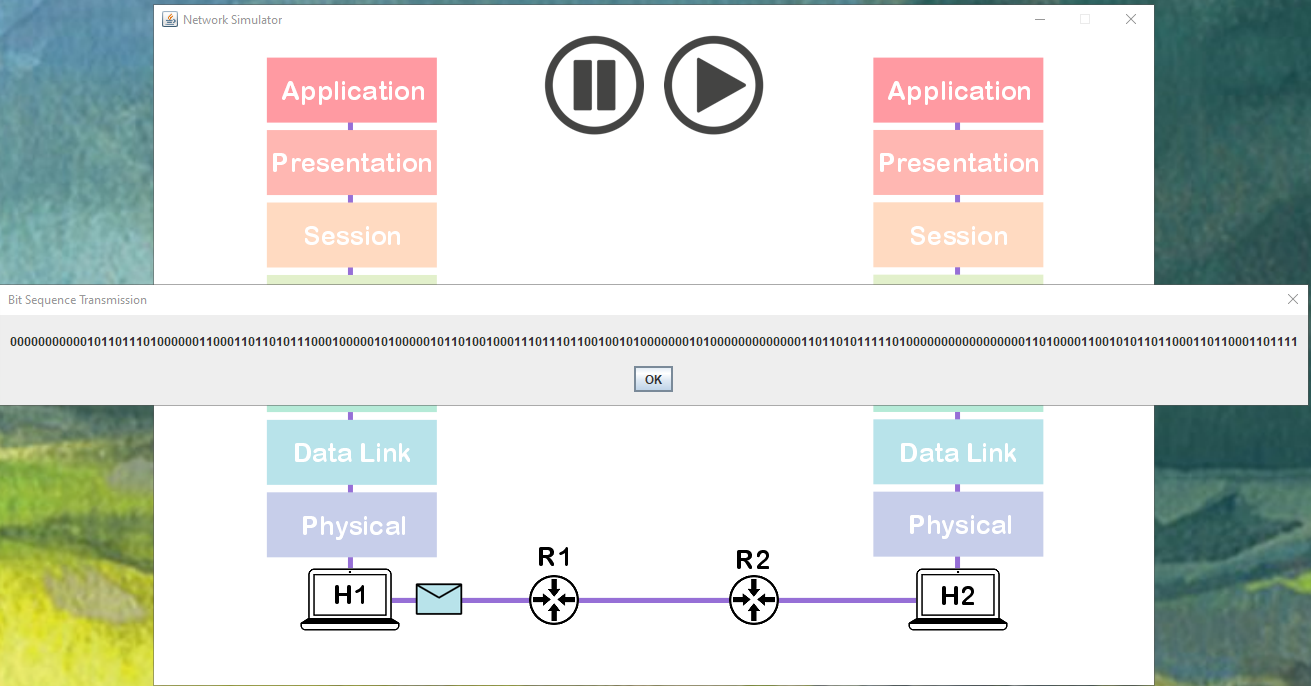
Clicking on either transport layer before the envelope has passed into it pauses the animation and gives a notification.



Clicking on either transport layer after the envelope has passed into it displays the TCP segment.



Clicking anywhere on the link before the envelope has passed into it pauses the animation and gives a notification.



Clicking anywhere on the link after the envelope has passed into it displays the bit sequence.