31341

Assignment 9

Problem Statement: Study of any network simulation tools - To create a network with three nodes and establish a TCP connection between node 0 and node 1 such that node 0 will send TCP packet to node 2 via node 1.

Objectives:

1. To study network simulation tools
2. To create a network and establish TCP connection to send TCP packet

Learning outcomes: Students will be able to

1. Demonstrate knowledge of various network simulation tools
2. Create a network of nodes and establish a TCP connection to send TCP packet

Requirements:

1. Open source linux based OS
2. NS2 Network simulator

Theory:

A network simulator is a software that predicts the behaviour of a computer network. Since communication networks have become too complex for traditional analytical methods to provide an accurate understanding of system behaviour, network simulators are used. In simulators, the computer network is modelled with devices, links, applications, etc. and the performance is analysed. Simulators come with support for popular technologies and networks in use such as WLANs, LTE, IOT, etc.

Categorisation of network simulators:

1. Commercial and open source simulators

Commercial simulators require users to pay to get the license to use their software. These generally have up-to-date documentation and are consistently maintained by staff. OPNET is a commercial simulator.

31341

.

Open source simulators are flexible and the code is available for everyone to use and improve. However, they may lack in systematic and complete documentations and support. NS2, NS3, J-Sim, etc. are typical open source network simulators.

2. Simple vs Complex

Minimally, a network simulator should enable users to represent a network topology, defining scenarios, specifying the nodes on the network, the links and traffic between those nodes. Complicated systems even allow protocol specification, visualization, intuitive interfaces, etc.

TCP Connection Establishment

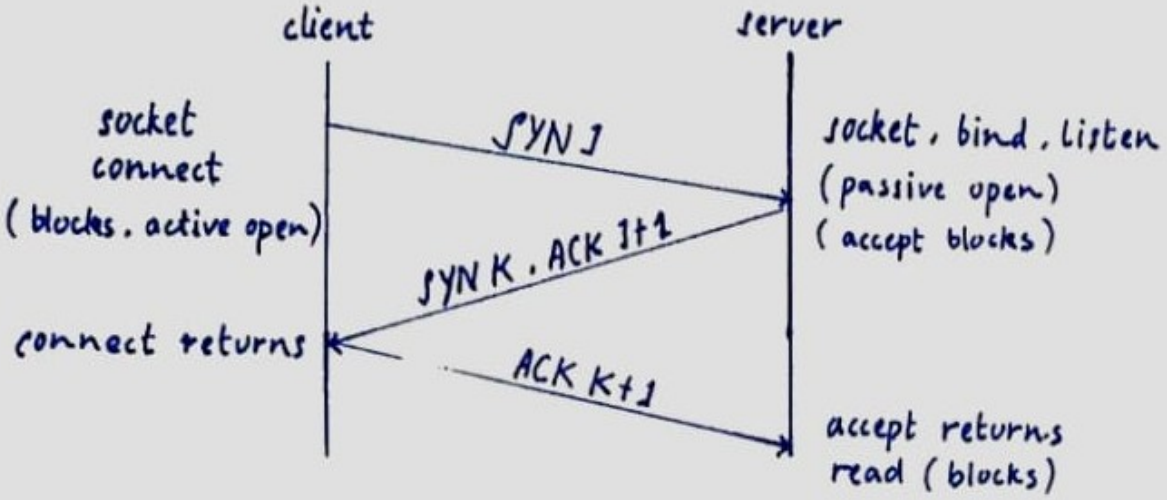
Three-Way Handshake

1. The server must be prepared to accept an incoming connection. This is normally done by calling socket, bind, and listen and is called a passive open.
2. The client issues and active open by calling connect. This causes the client TCP to send a synchronize (SYN) segment, which tells the server the client's initial sequence number. This only contains an IP header, a TCP header and possible TCP options.
3. The server must acknowledge (ACK) the client's SYN and the server must also send its own SYN containing the initial sequence number for the data that the server will send on the connection. The server sends these in a single segment.
4. The client must acknowledge the server's SYN.

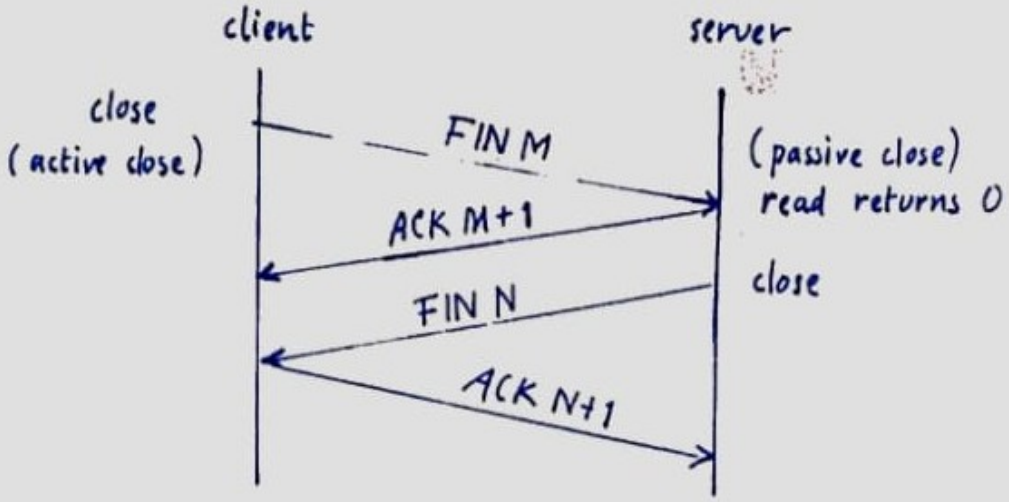
We show the client's initial sequence number as J and the server's initial number as K. A SYN occupies 1 byte and it's ACK is initial number plus one. Same applies for FIN as well.

TCP Connection Termination

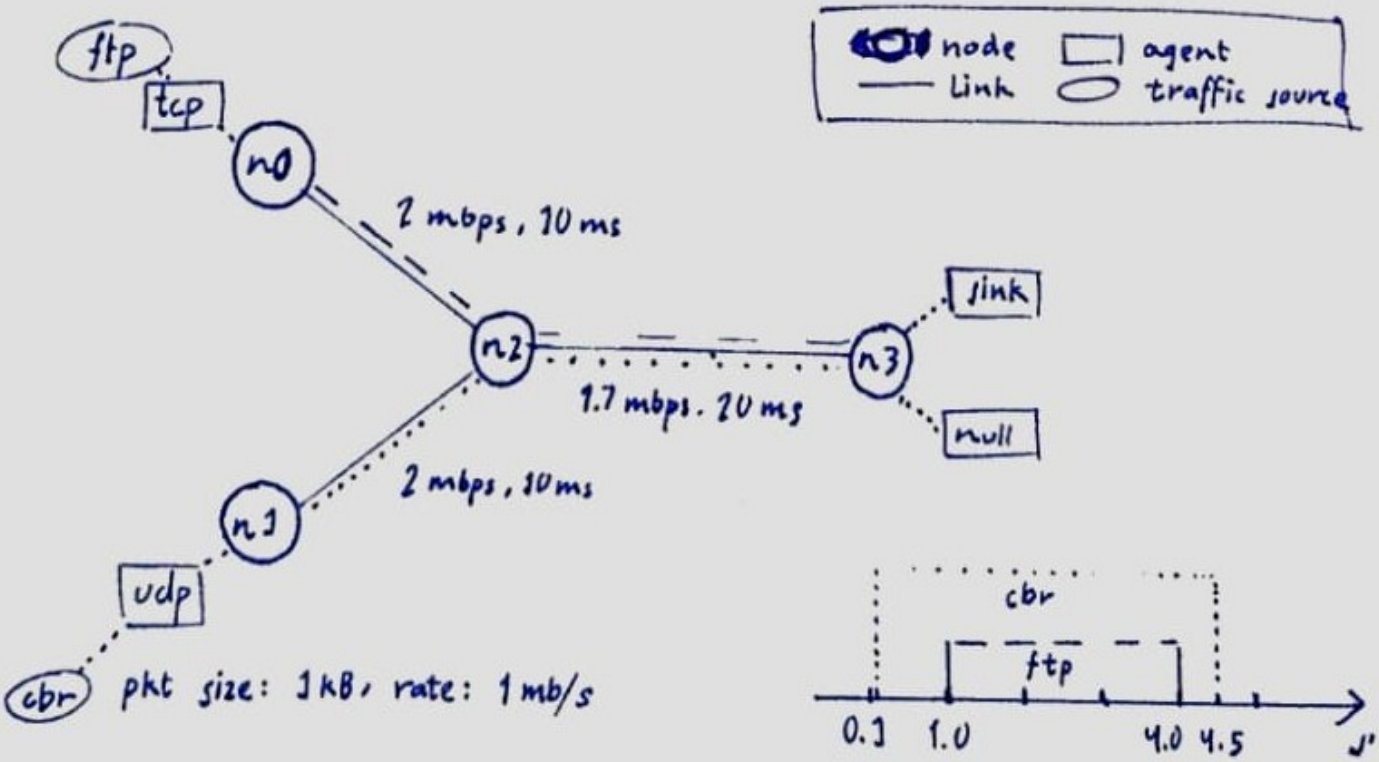
1. One application calls close first, and we say that end performs the active close.



TCP Connection Establishment



TCP Connection Termination



31341

This sends a FIN segment, which means it is finished segment data.

2. The other end that receives the FIN performs the passive close. The received FIN is acknowledged by TCP. The receipt of the FIN is also passed to the application as an end-of-file since the application will not receive any additional data on the connection.

3. Sometime later, the application that received the end-of-file will close its socket. This causes its TCP to send a FIN.

4. The TCP on the system that receives this final FIN acknowledges the FIN.

Conclusion:

We have successfully studied NS2 network simulation tool and implemented a TCP connection to send packets.