Date ______
Page ______

Assignment A5

Title: UDP Socket Programming

Problem Statement:

Write a program in C/C++ using UDP sockets to enable file transfer (script, text, audio and video) between two machines. Demonstrate the packets captured traces using wireshark Packet Analyser tool for peer to peer modes.

Software and Hardware Requirements:

Fedora 20 with Pentium IV and above 1 GB RAM, Eclipse IDE, Wireshark Packet analyzer tool installed

Theory:

TCP guarantees the delivery of packets, and preserves their order on destination. Sometimes the features are not required and since they do not come without performnance costs, it would be better to have a lighter transfer protocol.
This kind of service is accomplished by the UDP protocol which conveys datagram packets.
Datagram packets are used to implement a connectionless packet delivery service supported by the UDP protocol.
Each message is transferred from the source machine to the destination based on information contained within the packet ie each packet needs to have destination address and might be routed differently and might arrive in any order.

2020/09/21 23:32

Packet delivery is not guaranteed.
The format of datagram packet is

| Msg | length | Host | serverPort |

Java supports datagram communication through the following classes:

1) DatagrAm Packet
2) Datagram Socket

The class datagram packet contains several constructors that can be used for creating packet object.
One of them is DatagramPacket (byte [] buf, int length, Inet Address, address, int port);

The key methods of DatagramPacket class are: byte[] getData(.) Returns the data buffer int getLength() Returns the length of the data received. void SetData (byte [] buf) Sets the data buffer for this packet. void setLength (int length) sets the length for this packet.

DatagramSocket class supports various methods the two key methods are void send(Datagram Packet p) Sends a datagram packet from this Socket void receive (Datagram Packet p) receives a datagram packet from this Socket.

2020/09/21 23:33

A simple UDP server program that waits for client's requests and then accepts the message (datagram) and sends back the same message is given below.

1) int socket (int domain, int type, int protocol)
creates an unbound socket in the specified domain
returns socket file descriptor
Arguments -
· domain - specifies the communication.
(AF_INET for IPv4 / AF_INET6 for IPv6)
type - type of socket to be created
(SOCK_STREAM for TCP / SOCK_DIAGRAM for UDP)

2) Protocol - protocol to be used by socket
0 means use default protocol for the address family

int bind (int sockfd, const struct sockaddr *addr,
socklen_t addrlen)
assigns address to the unbound socket.

3) ssize_t sendto (int socfd, const void *buf, size_t len,
int flags, const struct sockaddr *dest_addr,
socklen_t addrlen)
send a message to the socket

4) ssize_t recvfrom (int socfd, void *buf, size_t len, int
flags, struct sockaddr *src_addr, socklen_t *addrl
Receive a message from the socket

5) int close (int fd)
Close a file descriptor

Conclusion:

We have succesfully implimented UDP socket programming to enable file tranfer between two machines.