**INDEX**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Sl no | Date of Experiment | Name of Experiment | Page No | Remarks |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

**Experiment No: 1**

**AIM OF THE EXPERIMENT:** TO STUDY ABOUT DIFFERENT HARDWARES, TOOLS USED FOR NETWORKING.

**THEORY:**

**NETWORKING**: A computer network connects various nodes (Computers, mobile phones and other devices) which allows them to communicate and share data and resources.

These connections/links can be established using cables/wires or optical fibres or via wireless medium like Wi-Fi.

Network computer devices that originate, route and terminate the data are called network nodes. Nodes are identified by network addresses, and can include hosts such as personal computers, phones, and servers, as well as networking hardware such as routers and switches.

**COMMUNICATION:** In computer networks, communication refers to transmission of this digital data between two or more computers and a computer network or data network is a telecommunications network that allows computers to exchange data. In computer networks, the data is passed in the form of packets.

**TYPES OF COMPUTER NETWORKS:**

Various types of networks are:

1. **Personal Area Network (PAN):** It is the most basic type of network, a PAN is made up of a wireless modem, a computer or two, phones, printers, tablets, etc., and revolves around one person in one building. These types of networks are typically found in small offices or residences, and are managed by one person or organization from a single device.

2. **Local Area Network (LAN**): It is usually a small network that is restricted to a small geographic area. A computer network available only to the residents within a building or between groups of two or three buildings in close proximity to each other can be called a LAN.

3. **Wide Area Network (WAN):** These networks cover a broad range of geographic area. WANs are used to connect LANs and other types of networks together so that users and computers can communicate with computers in other regions. An example of a WAN is internet.

4. **Metropolitan Area Network (MAN):** MAN is a network that connects the users with computer resources in a geographic area that is larger than LAN but not quite as large as WAN. An example would a connection over a large city.

**COMPONENTS OF A COMPUTER NETWORK:**

1. **Servers** - Servers are computers that hold shared files, programs, and the network operating system. There are many different kinds of servers like file servers, print servers, mail servers, communication servers, database servers, print servers, fax servers and web servers.

2. **Clients** - Clients are computers that access and use the network and shared network resources. Client computers are basically the customers (users) of the network, as they request and receive services from the servers.

3. **Transmission Media** - Transmission media are the facilities used to interconnect computers in a network, such as twisted-pair wire, coaxial cable, and optical fibre cable. Transmission media are sometimes called channels, links or lines.

4. **Shared data** - Shared data are data that file servers provide to clients such as data files, printer access programs and e-mail.

5. **Shared printers and other** **peripherals** - Shared printers and peripherals are hardware resources provided to the users of the network by servers. Resources provided include data files, printers, software, or any other items used by clients on the network.

6. **Network Interface Card** - Each computer in a network has a special expansion card called a network interface card (NIC). The NIC prepares (formats) and sends data, receives data, and controls data flow between the computer and the network

7. **Local Operating System** - A local operating system allows personal computers to access files, print to a local printer, and have and use one or more disk and CD drives that are located on the computer.

8. **Network Operating System** - The network operating system is a program that runs on computers and servers, and allows the computers to communicate over the network.

9. **Hub** - Hub is a device that splits a network connection into multiple computers. It is like a distribution centre. When a computer request information from a network or a specific computer, it sends the request to the hub through a cable. The hub will receive the request and transmit it to the entire network. Each computer in the network should then figure out whether the broadcast data is for them or not.

10. **Switch** - Switch is a telecommunication device grouped as one of computer network components. It uses physical device addresses in each incoming messages so that it can deliver the message to the right destination or port.

**TYPES OF A COMMUNICATION/TOPOLOGIES:**

**1. Point to Point communication:** A direct communication is established between the two communicating computers.

**2. Bus Topology:** There is a main cable to which all of the computers are connected. Here, the data flows in a single direction. It is cost effective and used in small networks. The whole network goes down if the main cable fails.

**3. Ring Topology:** The computers are connected in a ring structure. It is easy to install and reconfigure.

**4. Star Topology:** All the computer are connected to a central hub/switch. Here, the whole network goes down if the hub fails. It is less expensive and robust.

**5. Mesh Topology:** In this topology, all the computers are interconnected with each other. So, there are n\*(n-1)/2 bus needed. It is highly private and has secure connection. More no of cabling and I/O ports are required.

**6. Hybrid Topology:** Combination of two or more types of topology is called Hybrid Topology.

**PHYSICAL EQUIPMENTS:**

**1. NIC Card:** A network interface card is a hardware used to connect a node into network. It has an inbuilt unique MAC address of 48 bits in IPV4.

**2. Hub:** A hub is a hardware device that relays communication data. A hub sends data packets (frames) to all devices on a network, regardless of any MAC addresses contained in the data packet. A Hub is used to communicate only within a local area but not outside their own network.

**3.** **Switch:** A switch is a high-speed device that receives incoming data packets and redirects them to their destination using their physical/MAC address on a local area network (LAN). It is smarter than a hub as it sends the packets only to the receiver node.

**4. Router:** Router is an interconnecting device which is based on ip and make the communication between different networks possible. Routers perform the traffic directing functions on the Internet. Routers have the additional ability to forward packets between different networks, whereas a switch is limited to node-to-node communication on the same network.

**5. Firewall:** A firewall is a network security system that monitors and controls incoming and outgoing network traffic based on predetermined security rules. A firewall typically establishes a barrier between a trusted internal network and untrusted external network, such as the Internet.

**6. Socket:** A network socket is an internal endpoint for sending or receiving data within a node on a computer network. Sockets allow communication between two different processes on the same or different machines. There are 2 sockets for a connection: Server socket and Client socket.

**7. Ports:** A port is an endpoint of communication. Physical as well as wireless connections are terminated at ports of hardware devices. Ports range from 0 to 65,534 where 0-1023 are well known ports and 1024- 49,151 are registered ports maintained by IANA. The dynamic or private ports are those from 49152 through 65,535. One common use for this range is for ephemeral/short-lived ports.

**Submitted by:**

Name- Atul Kumar Agrawal

Branch- Computer Science and Engineering

Roll no: 1602040031

**Experiment No: 2**

**AIM OF THE EXPERIMENT:** Write a program in java to establish peer to peer connection between 2 computers.

**PROGRAM:**

**/\* Server Program \*/**

import java.net.\*; //Socket,ServerSocket

import java.io.\*; //DataInputStream,BufferedInputStream,IOException

public class Server{

private Socket socket\_var=null;

private ServerSocket server\_var=null;

private DataInputStream in\_var=null;

public Server(int port){ // constructor with port passed through main()

try{

server\_var=new ServerSocket(port);

System.out.println("Server Started!");

System.out.println("Waiting for a client..");

socket\_var=server\_var.accept();

System.out.println("Client Accepted");

in\_var=new DataInputStream(new BufferedInputStream(socket\_var.getInputStream())); //read primitive Java data types

String line\_var="";

while(!line\_var.equals("Over")){

try{

line\_var=in\_var.readUTF(); //reads the string from client

System.out.println(line\_var);

}

catch(IOException i){

System.out.println(i);

}

}

System.out.println("Closing Connection!");

socket\_var.close();

in\_var.close();

}

catch(IOException i){

System.out.println(i);

}

}

public static void main(String args[]){

Server server=new Server(5000);

}

}

**/\* Client Program \*/**

import java.net.\*;

import java.io.\*;

public class Client

{

private Socket socket\_var=null;

private DataInputStream input\_var=null;

private DataOutputStream out\_var=null;

public Client(String address\_input,int port\_input){

try{

socket\_var=new Socket(address\_input,port\_input); //why both address and port here but in server only port??

System.out.println("Connected");

input\_var=new DataInputStream(System.in);

out\_var=new DataOutputStream(socket\_var.getOutputStream());

}

catch(UnknownHostException u){

System.out.println(u);

}

catch(IOException i){

System.out.println(i);

}

String line\_input="";

while(!line\_input.equals("Over")){

try{

line\_input=input\_var.readLine(); //read input string from user

out\_var.writeUTF(line\_input); //send it to server

}

catch(IOException i){

System.out.println(i);

}

}

try{

input\_var.close();

out\_var.close();

socket\_var.close();

}

catch(IOException i){

System.out.println(i);

}

}

public static void main(String args[]){

Client client\_var=new Client("127.0.0.1",5000);

}

}

**OUTPUT**

**Server Side Client Side**

Hello Hello

I made my first socket connection I made my first socket connection

Over Over

Closing connection

**Submitted by:**

Name- Atul Kumar Agrawal

Branch- Computer Science and Engineering

Roll no: 1602040031

**Experiment No: 3**

**AIM OF THE EXPERIMENT:** Write a program in java to implement Stop and Wait protocol.

**ALGORITHM:**

Step 1: Start the program.

Step 2: Generate a random that gives the total number of frames to be transmitted.

Step 3: Transmit the first frame.

Step 4: Receive the acknowledgement for the first frame.

Step 5: Transmit the next frame

Step 6: Find the remaining frames to be sent.

Step 7: If an acknowledgement is not received for a particular frame retransmit that frame

alone again.

Step 8: Repeat the steps 5 to 7 till the number of remaining frames to be send becomes

zero.

Step 9: Stop the program.

**CODE:**

#include<stdio.h>

#include<conio.h>

#include<stdlib.h>

void main()

{

int i,j,noframes,x,x1=10,x2;

clrscr();

for(i=0;i<200;i++)

rand();

noframes=rand()/200;

i=1;

j=1;

noframes = noframes / 8;

printf("\n number of frames is %d",noframes);

getch();

while(noframes>0)

{

printf("\nsending frame %d",i);

srand(x1++);

x = rand()%10;

if(x%2 == 0)

{

for (x2=1; x2<2; x2++)

{

printf("waiting for %d seconds\n", x2);

sleep(x2);

}

printf("\nsending frame %d",i);

srand(x1++);

x = rand()%10;

}

printf("\nack for frame %d",j);

noframes-=1;

i++;

j++;

}

printf("\n end of stop and wait protocol");

getch();

}

**OUTPUT:**

No of frames is 6

Sending frame 1

Acknowledgement for frame 1

Sending frame 2

Acknowledgement for frame 2

Sending frame 3

Acknowledgement for frame 3

Sending frame 4

Acknowledgement for frame 4

Sending frame 5

Waiting for 1 second

Retransmitting frame 5

Acknowledgement for frame 5

Sending frame 6

Waiting for 1 second

Sending frame 6

Acknowledgement for frame 6

End of stop and wait protocol

**Submitted by:**

Name- Atul Kumar Agrawal

Branch- Computer Science and Engineering

Roll no: 1602040031

**Experiment No: 4**

**AIM OF THE EXPERIMENT:** Write a program in java to implement Leaky bucket algorithm.

**ALGORITHM:**

**Initialize a counter to n at the tick of the clock.**

1. If n is greater than the size of the packet, send the packet and decrement the counter by the packet size. Repeat this step until n is smaller than the packet size.
2. Reset the counter and go to step 1.

Example – Let n=1000

Packet=  
Since n> front of Queue i.e. n>200  
Therefore, n=1000-200=800  
Packet size of 200 is sent to the network.  
  
Now Again n>front of the queue i.e. n > 400  
Therefore, n=800-400=400  
Packet size of 400 is sent to the network.  
  
Since n< front of queue  
Therefore, the procedure is stop.  
Initialize n=1000 on another tick of clock.  
This procedure is repeated until all the packets are sent to the network.

**PROGRAM:**

import java.io.\*;

import java.util.\*;

class LeakyBucket{

public static void main(String args[]){

int no\_of\_queries,storage,output\_pkt\_size;

int input\_pkt\_size,bucket\_size,size\_left;

storage=0;

no\_of\_queries=5;

bucket\_size=10;

input\_pkt\_size=4;

output\_pkt\_size=1;

for(int i=0;i<no\_of\_queries;i++){

size\_left=bucket\_size-storage;

if(input\_pkt\_size<=(size\_left)){

storage+=input\_pkt\_size;

System.out.println("Buffer size="+storage+" out of bucket size="+bucket\_size);

}

else{

System.out.println("Packet loss="+(input\_pkt\_size-(size\_left)));

storage=bucket\_size;

System.out.println("Buffer size="+storage+" out of bucket size="+bucket\_size);

}

storage-=output\_pkt\_size;

}

}

}

**OUTPUT**

Buffer size=4 out of bucket size=10

Buffer size=7 out of bucket size=10

Buffer size=10 out of bucket size=10

Packet loss=3

Buffer size=10 out of bucket size=10

Packet loss=3

Buffer size=10 out of bucket size=10

**Submitted by:**

Name- Atul Kumar Agrawal

Branch- Computer Science and Engineering

Roll no: 1602040031