

**19CSE301-
COMPUTER NETWORKS
Case Study
R.NO:CB.EN.U4CSE19156
V S S K CHAITANYA**

Title: Railway Zone Management

Department Name	Purpose	Network Details (No of nodes,servers,Protocols)
Tourism And Catering Department	Catering Department is used to serve food in trains.	35 nodes, 4 servers,RIP,OSPF,FTP
Railway Hospital Department	Railway Hospital is used for emergency surgeries and for Railway employees treatment.	22 nodes, 4 servers,RIP,OSPF,FTP
Railway Reservation Department	Railway Reservation Department manages ticket bookings and special bookings like tatkal.	30 nodes, 4 servers,RIP,OSPF,FTP
Accounts Department	Manages employee salary and bills.	20 nodes, 4 servers,RIP,OSPF,FTP

ROLL Number	Name	Contribution in Case Study
CB.EN.U4CSE19153	T. Krishna Chaitanya	Tourism And Catering Department
CB.EN.U4CSE19155	U. Yeshwanth	Railway Hospital Department
CB.EN.U4CSE19156	V. S. S. K. Chaitanya	Railway Reservation Department
CB.EN.U4CSE19165	P. Prem Sai	Accounts Department

Topic	Page No in the document	Supporting file name
Problem Statement	3	
Objective of the case study	3	
Network Architecture Diagram	23	
Performance Parameters	9	
Department Details in the case study with description	3	
Subnet ip scheme followed [Solving the IP address allocation process for each subnetwork]	23	
Socket programming	13	File Description [File Name] Include text files
Cisco packet tracer – Application layer protocols	25	
Cisco packet tracer – Virtual Local Area Network	26	
Cisco packet tracer – OSPF	25	
Cisco packet tracer – RIP	25	
GoBack N and Selective Repeat protocol	28	
Cloud concepts	29	
Cloud and Networking	29	
How is cloud related to your application?	30	

Name: V.S.S.K.Chaitanya

Roll Number : CB.EN.U4CSE19156

Problem Statement:

To design a computer network that organize and manage various railway departments within a railway zone.

Objective of the case study:

To design a network architecture for Railway Zone Management

Description of the case study:

Railway zones play a very crucial part in maintenance of Indian Railway system

various data that need to be managed between the stations coming under a particular zone such as maintaining the record of passenger data, health system, amount sanctioned to the station for development etc. All this data must be collected from various stations falling under a particular zone must be managed by the zone headquarters

So In order to operate effectively exchange of data among railway stations located in various place is crucial and needs a proper architecture. Our case study is aimed to demonstrate how different departments within a railway zone headquarter manage this effectively.

Railway Reservation Department:

In this department railway bookings, cancellation and many other services that are provided to the customers are managed.

In the network along with sub networks connection with same department across other branch is also shown.

Railway Zone Network

NETWORKS

Tourism And Catering Department – Wide Area Network

Railway Reservation – Wide Area Network

Railway Hospital – Local Area Network

Accounts Department – Virtual Private Network

Client configuration

Intel(R) Core 4005U Series Processor

8 GB RAM

8 MB Cache

1 TB

Dual External Storage Connectivity

DVD ROM Drive

Server Configuration

2 Intel(R) Itanium 2 9100 Series Processors

23.98 GB RAM

12 MB Cache

4.5 TB Storage (EMC Box)

Dual External Storage Connectivity

DVD ROM Drive

SERVERS

Database Server

Proxy Server

Mail Server

Application Server

Client Server

CABLES

Coaxial Fibre

Twisted Fibre

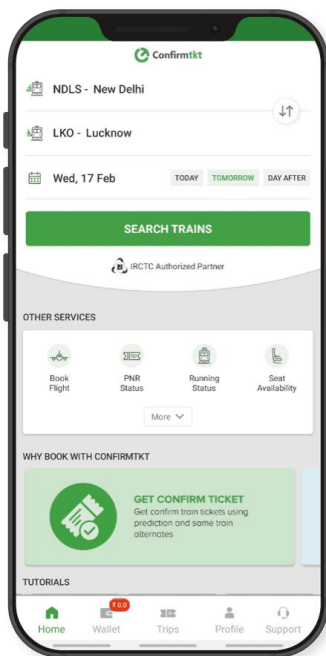
Optical Fibre

TOPOLOGY

Reservation, Accounts - Star Topology

Medical - Bus Topology

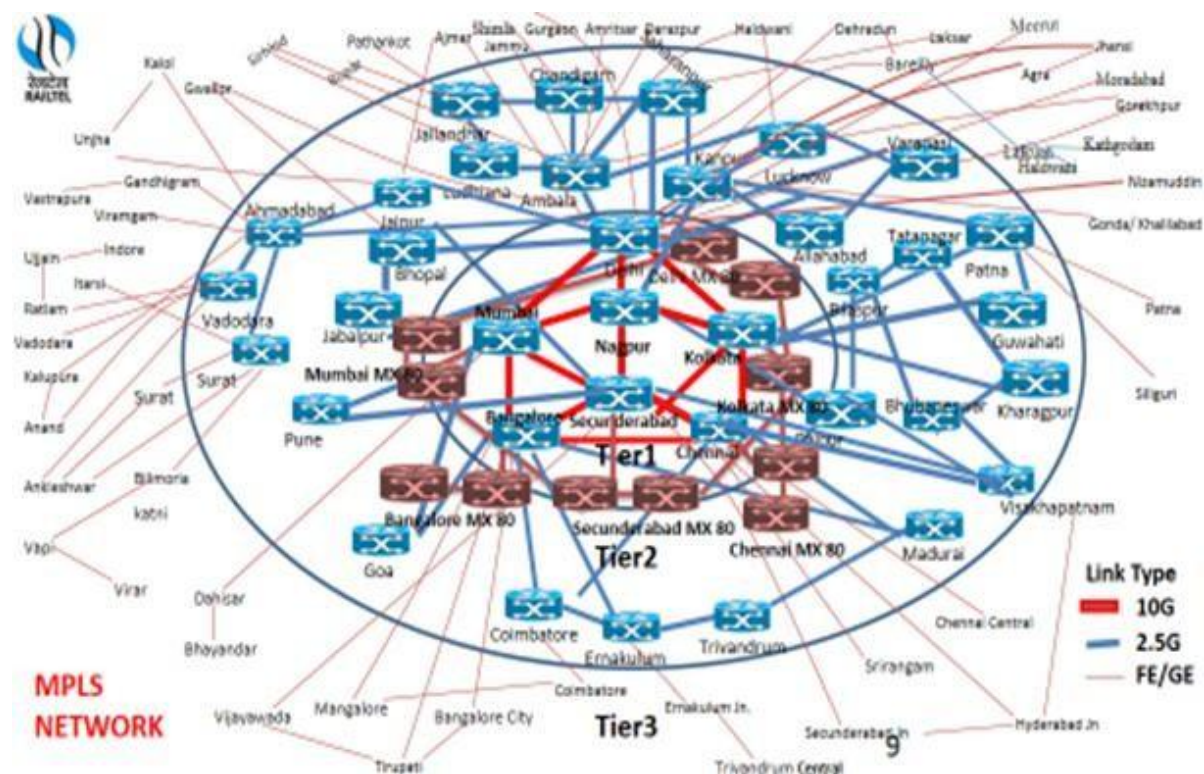
Tourism And Catering Department -





Why is Networking required?

As we saw before railway zone headquarters needs to collect and send info to various railway stations and should be able to manage the passenger data. So computer network plays a vital role to establish a proper connection between various departments. A proper architecture is needed because various devices like printer, ip phone, pcs are to be connected. So that a zone headquarter can manage all the railway stations effectively.



List of Network performance parameters:

Bandwidth:

bandwidth is defined as the maximum number of bits that can flow through a network connection in a given period of time. The fundamental unit of network bandwidth is bits per second.

Formula:

Expressed as bits per second (bps), modern network links have greater capacity, which is typically measured in millions of bits per second (megabits per second , or Mbps) or billions of bits per second (gigabits per second ,or Gbps).

Possible Value In General Scenario.

normally speeds ranging from 10–1,000 Mbps

Throughput:

The purpose of throughput is measuring the number of messages successfully transmitted per unit time.

Formula:

F = Frames per minute

A = Average of bits each frame carries

Through Put = $((F \times A)/\text{Unit Time})$

Possible Value In General Scenario:

16.2 mbps

Latency (Delay):

Latency meaning in networking is best thought of as the amount of time it takes for a packet of data to be captured, transmitted, processed through multiple devices, then received at its destination and decoded.

Formula:

Propagation time = Distance/Propagation speed

Transmission time=Message size / Bandwidth

Queuing Time = Directly Proportional to the congestion in the network

Processing Delay = Directly proportional to processing speed of the routers.

Latency = Propagation Time + Transmission Time + Queuing Time + Processing Delay

Possible Value In General Scenario:

1ms to 20ms

Bandwidth-Delay Product:

Bandwidth and delay are two performance measurements of a link. However, what is significant in data communications is the product of the two, the bandwidth-delay product.

Formula:

Bandwidth x Delay

Possible Value In General Scenario:

500kb

Example is Residential Cable internet

Jitter:

Jitter is used to describe the amount of inconsistency in latency across the network, while latency measures the time it takes for data to reach its destination and ultimately make a round trip

Formula:

To measure Jitter, we take the difference between samples, then divide by the number of samples (minus-1)

Possible Value In General Scenario:

below 30ms

Railway reservation department manages the passenger reservation count of trains based on different classes like sleeper class, 3A class, 2A class.

Description of CSV file:

It consists of five columns

Train_id: Train Id is an unique id which is given to each train to avoid confusion with trains having same names.

Train name: Name of the Train.

Sleeper/3a/2a : Passenger count of respective classes on a particular day.

b. List of operations completed with the File

1.Adding new train details.

2.updating existing train details.

3.Retrieving the particular train details.

4.Viewing all train details on a particular Day.

CODE:
SERVER.PY

```
import socket
import threading
import pandas as pd
df=pd.read_csv('train_list.csv')
df=df.set_index('train_id')
print(df)
s=socket.socket(socket.AF_INET,socket.SOCK_STREAM)
port=6000
host=socket.gethostname()
s.bind((host,port))
s.listen()
print('Server is listening ...Passenger_Reservation__Department')
```

```

data=con.recv(1024)
f=repr(data.decode())
print(f)
f=f[1:]
f=f[:len(f)-1]
f=f.split('_')
print(f)

if f[0]=="1":p=f[1:]

df2=pd.DataFrame.from_records([{'train_id':p[0],'train':p[1],'sleeper':int(p[2]),'3a':int(p[3]),'2a':int(p[4])}],index='train_id ')
print(df2)
df=df.append(df2)
print(df)
df.to_csv('train_list.csv')

if f[0]=="2":
    p=f[1:]
    df.at[int(p[0]),p[1]]=p[2]

if f[0]=="3":
    d=get_info(f[1:],df)
    print(d)
    con.send(str(d).encode())

if f[0]=="4":
    c=False

if f[0]=="5":

```

```
        dfi=str(df)
        con.send(dfi.encode())

def get_info(p,df):
    id=df.index
    if (p[1] not in df):
        return 'Not exist'
    return df.at[int(p[0]),p[1]]

while True:

    con,addr=s.accept()
    print('Connection established from ',addr)
    thread=threading.Thread(target=handle_client,args=(con,df,addr))
    thread.start()
```


CLIENT.PY

```
import socket

s=socket.socket(socket.AF_INET,socket.SOCK_STREAM)

host=socket.gethostname()

port=6000

s.connect((host,port))

print("WELCOME TO RAILWAY PASSNGER RESERVATION DEPARTMENT")

s1=0

while s1!=4:

    print("Select the actions below")

    print('1-add_info')

    print('2-update_info')

    print('3-get_info')

    print('4-stop')

    s1=int(input())

    if s1==1:

        train_id=input('enter train id:')

        train=input('enter train name:')

        sleeper=input('enter number of passengers in sleeper:')

        ac3=input('enter no of passengers in 3A:')

        ac2=input('enter no of passengers in 2A:')

        p=str(s1)+'_'+str(train_id)+'_'+str(train)+'_'+str(sleeper)+'_'+str(ac3)+'_'+str(ac2)

        s.send(p.encode())

        print('sent')

    if s1==2:
```

```
train_id=input('enter train id:')
sleeper=input('enter class(sleeper/3a/2a:')
num=input('enter no of passengers to update:')
p=str(s1)+'_'+str(train_id)+'_'+str(sleeper)+'_'+str(num)
```

```
p=str(p)
s.send(p.encode())
print('updated')
```

```
if s1==3:
```

```
train=input('enter train id:')
sleeper=input('enter class(sleeper/3a/2a:')

```

```
p=str(s1)+'_'+str(train)+'_'+str(sleeper)
```

```
p=str(p)
```

```
s.send(p.encode())
```

```
print('sent')
```

```
data=s.recv(1024)
```

```
print('no of pasengers are',data.decode())
```

```
if s1==4:
```

```
break
```

Client 3.py:

```
import socket
import pandas as pd
s=socket.socket()
host=socket.gethostname()
port=6000
s.connect((host,port))
print("WELCOME TO RAILWAY PASSNGER RESERVATION DEPARTMENT")
s1=0
while s1!=4:

    print("Select the actions below")
    print('5-view data')
    print('4-stop')
    s1=int(input())
    if s1==5:

        p=str(s1)

        s.send(p.encode())
        print('sent')
        data=s.recv(1024)
        l=data.decode()
        d=""
        for i in l:
            if i!="\n" and i!="\r":
                if i==" ":
                    d+=" "
```

```
    else:  
        d+=i  
elif i=="\n":  
    print(d)  
    d=""
```

```
if s1==4:  
    break
```

Output screen shorts:

SERVER:

```
(venv) C:\Users\Chait\Desktop\books\computer networks>python res_server.py
      train sleeper 3a 2a
train_id
11      rajadhani      45 33 22
12      vishaka      56 34 21
13      pinakini      72 45 34
14      kerala      67 45 34
Server is listening ...Passenger_Reservation_Department
Connection established from ('192.168.56.1', 6420)
'2_13_2a_33'
['2', '13', '2a', '33']
Connection established from ('192.168.56.1', 6421)
'3_14_sleeper'
['3', '14', 'sleeper']
67
'1_15_nizamuddin_70_56_45'
['1', '15', 'nizamuddin', '70', '56', '45']
      train sleeper 3a 2a
train_id
15      nizamuddin      70 56 45
      train sleeper 3a 2a
train_id
11      rajadhani      45 33 22
12      vishaka      56 34 21
13      pinakini      72 45 33
14      kerala      67 45 34
15      nizamuddin      70 56 45
```

CLIENT 1:

```
(venv) C:\Users\Chait\Desktop\books\computer networks>python res_client.py
WELCOME TO RAILWAY PASSNGER RESERVATION DEPARTMENT
Select the actions below
1-add_info
2-update_info
3-get_info
4-stop
2
enter train id:13
enter class(sleeper/3a/2a:2a
enter no of passengers to update:33
updated
Select the actions below
1-add_info
2-update_info
3-get_info
4-stop
```

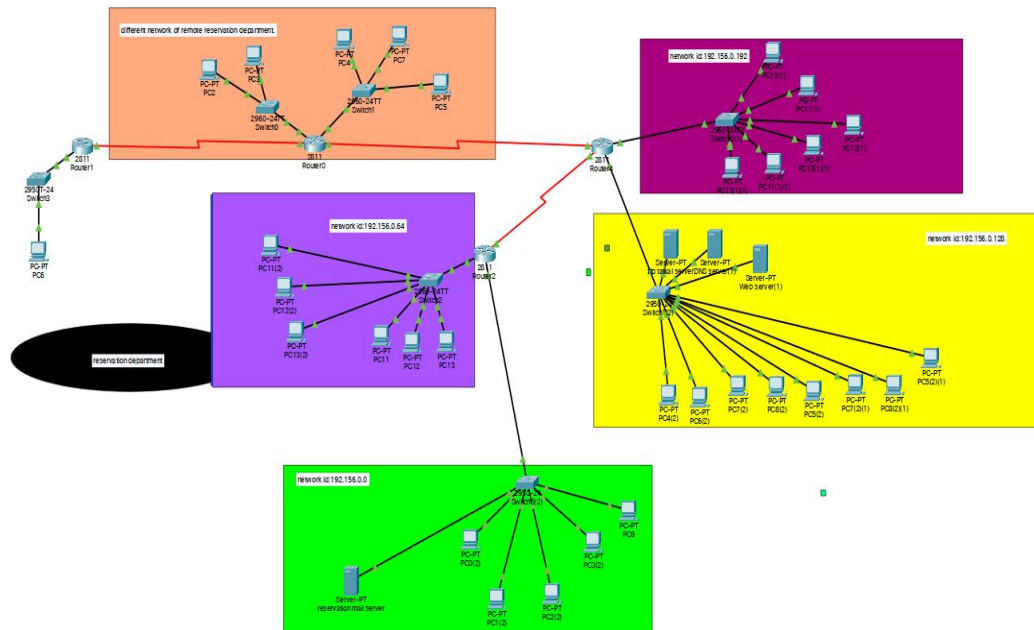
CLIENT 2:

```
(venv) C:\Users\Chait\Desktop\books\computer networks>python res_client.py
WELCOME TO RAILWAY PASSNGER RESERVATION DEPARTMENT
Select the actions below
1-add_info
2-update_info
3-get_info
4-stop
3
enter train id:14
enter class(sleeper/3a/2a:sleeper
sent
no of pasengers are 67
Select the actions below
1-add_info
2-update_info
3-get_info
4-stop
1
enter train id:15
enter train name:nizamuddin
enter number of passengers in sleeper:70
enter no of passengers in 3A:56
enter no of passengers in 2A:45
sent
Select the actions below
1-add_info
2-update_info
3-get_info
4-stop
```

Client 3:

```
(venv) C:\Users\Chait\Desktop\books\computer networks>python res_client_2.py
WELCOME TO RAILWAY PASSNGER RESERVATION DEPARTMENT
Select the actions below
5-view data
4-stop
5
sent
train_id train sleeper 3a 2a
11 rajadhani 45 33 22
12 vishaka 56 34 21
13 pinakini 72 45 33
14 kerala 67 45 34
15 nizamuddin 70 56 45
Select the actions below
5-view data
4-stop
```


Cisco Packet Tracer Design(RAILWAY RESERVATION):

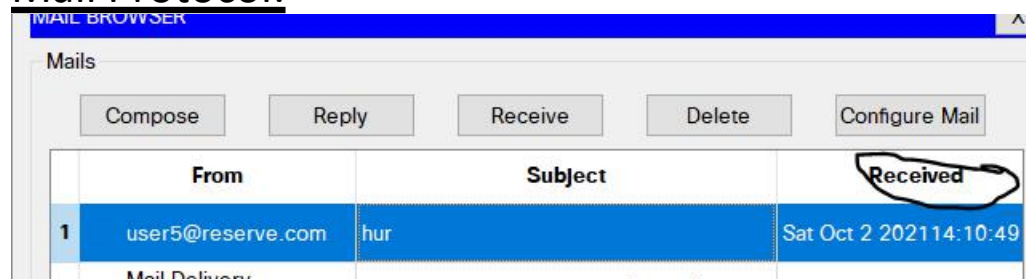


Subnet Allocation:

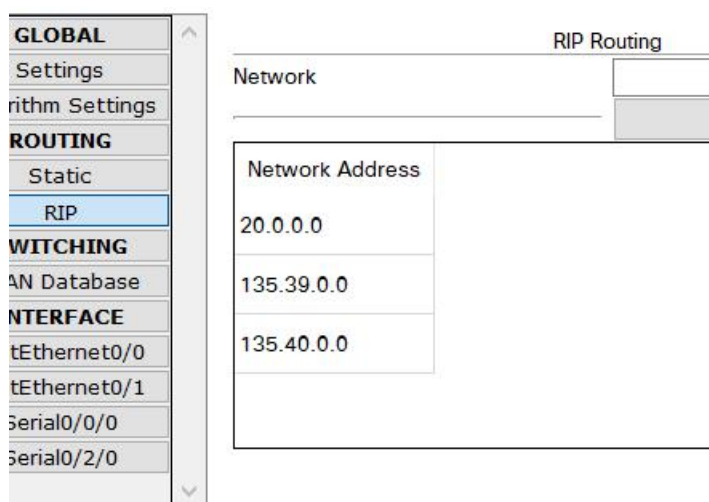
Subnetting with range of IP addresses possible of 64 and starting from subnet address as 196.156.0.0 for first subnet and following for other three mentioned in table.

Reservation Department Local Office	Reservation	Range of IP address: 192.156.0.1-192.156.0.63 Subnet address: 192.156.0.0 Subnet mask :255.255.255.192 Protocols configured:ospf	Organizing reservation info.
	Tatkal	Range of IP address: 192.156.0.65-192.156.0.127 Subnet address: 192.156.0.64 Subnet mask :255.255.255.192 Protocols configured:ospf	Managing Tatkal operations
	Management	Range of IP address: 192.156.0.129-192.156.0.191 Subnet address: 192.156.0.128 Subnet mask :255.255.255.192 Protocols configured:ospf	Updating reservation info about any changes
	Subnetwork name	Range of IP address: 192.156.0.193-192.156.0.255 Subnet address: 192.156.0.192 Subnet mask :255.255.255.192 Protocols configured:ospf	Back office management
Remote Reservation Offices		Showing how remote networks are connected.	
	FTP server	Category: FTP	
	Email server	Category: Email	
	Application server	Category: Web	
	OSPF	Routing protocol	Between subnets and outer network
	RIP	Routing protocol	Between remote office and local office.

Mail Protocol:



RIP:



OSPF:

```
Router#SHOW IP OSPF NEIGHBOR
```

```
Neighbor ID    Pri  State           Dead Time   Address      Interface
192.167.12.1    0    FULL/ -         00:00:34    30.0.0.2     Serial0/0/0
Router#
```

```
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial0/0/0, changed state to up
```

```
00:00:20: %OSPF-5-ADJCHG: Process 1, Nbr 192.156.0.204 on Serial0/0/0 from LOADING to FULL, Loading Done
```

VLAN:

Switch#SHOW VLAN BRIEF

VLAN Name	Status	Ports
-----	-----	-----
1 default	active	Fa0/5, Fa0/6, Fa0/7, Fa0/8 Fa0/9, Fa0/10, Fa0/11, Fa0/12 Fa0/13, Fa0/14, Fa0/15, Fa0/16 Fa0/17, Fa0/18, Fa0/19, Fa0/20 Fa0/21, Fa0/22, Fa0/23, Fa0/24 Gig0/1, Gig0/2
10 VLAN0010	active	Fa0/1, Fa0/4
20 VLAN0020	active	Fa0/2

Interface	IP-Address	OK?	Method	Status
Protocol				
FastEthernet0/0	135.39.12.224	YES	manual	up
FastEthernet0/1	unassigned	YES	manual	up
FastEthernet0/1.10	192.167.12.1	YES	manual	up
FastEthernet0/1.20	135.40.0.1	YES	manual	up
Serial0/0/0	30.0.0.2	YES	manual	up
Serial0/2/0	20.0.0.1	YES	manual	up
Vlan1	unassigned	YES	unset	administratively down
Router#				

Go back N and Selective Repeat:

Server side:

```
<terminated> go_back [Java Application] C:\Program Files\Java\jdk-7.0.2\bin
waiting for connection
The number of packets sent is:11
```

```
package snippet;

import java.io.DataInputStream;
import java.io.DataOutputStream;
import java.io.IOException;
import java.net.ServerSocket;
import java.net.Socket;
import java.net.SocketException;

public class go_back {

    static ServerSocket Serversocket;
    static DataInputStream dis;
    static DataOutputStream dos;

    public static void main(String[] args) throws
    SocketException {

        try {
            int a[] = { 100,30, 40, 50, 60, 70, 80, 90,
100, 110,40 };
            Serversocket = new ServerSocket(8011);
            System.out.println("waiting for
connection");
            Socket client = Serversocket.accept();
            dis = new
DataInputStream(client.getInputStream());
            dos = new
DataOutputStream(client.getOutputStream());
            System.out.println("The number of packets
sent is:" + a.length);
            int y = a.length;
            dos.write(y);
            dos.flush();

            for (int i = 0; i < a.length; i++) {
                dos.write(a[i]);
                dos.flush();
            }

            int k = dis.read();

            dos.write(a[k]);
            dos.flush();

        } catch (IOException e) {
            System.out.println(e);
        } finally {
            try {
                // ...
            }
        }
    }
}
```

Client Side:

```

Localhost/127.0.0.1
No of frame is:11
100
30
40
50
60
70
80
90
100
110
40
Received frame is: 100
Received frame is: 30
Received frame is: 40
Received frame is: -1
Received frame is: 60
Received frame is: 70
Received frame is: 80
Received frame is: 90
Received frame is: 100
Received frame is: 110
Received frame is: 40
Request to retransmit packet no 4 again!!

Received frame is: 50
quitting

```

<	
sd Instruct...	Total Instructio...

```

package snippet;

import java.lang.System;
import java.net.*;
import java.io.*;

public class client {
    static Socket connection;

    public static void main(String a[]) throws SocketException
    {
        try {
            int v[] = new int[11];
            //int g[] = new int[8];
            int n = 0;
            InetAddress addr =
InetAddress.getByName("Localhost");
            System.out.println(addr);
            connection = new Socket(addr, 8011);
            DataOutputStream out = new DataOutputStream(
                connection.getOutputStream());
            DataInputStream in = new DataInputStream(
                connection.getInputStream());
            int p = in.read();
            System.out.println("No of frame is:" + p);

            for (int i = 0; i < p; i++) {
                v[i] = in.read();
                System.out.println(v[i]);
                //g[i] = v[i];
            }
            v[3] = -1;
            for (int i = 0; i < p; i++)
            {
                System.out.println("Received frame is: " + v[i]);
            }

            for (int i = 0; i < p; i++)
                if (v[i] == -1) {
                    System.out.println("Request to retransmit packet no
"
                        + (i+1) + " again!!");
                    n = i;
                    out.write(n);
                    out.flush();
                }
            System.out.println();
            v[n] = in.read();
            System.out.println("Received frame is: " + v[n]);
            System.out.println("quitting");

```

Cloud Virtualization:

Now a days it is very difficult to bring every employee around the world to work at same place where all the data centers of a company are located or it will be very costly for a small organization to purchase heavy hardware material for limited amount of work.

To answer all these problems , CLOUD concept comes into play where you can virtually use hardware or software present in some other parts of the world right from your home.

Having physical devices or copies of these things can be difficult to move or copy when you need to move something. A virtual server or desktop can be easily moved and accessed from a different location.

In addition, physical devices are limited by the specific set of hardware they're installed on, while virtual devices can easily be allocated resources as necessary. Physical hardware can be consolidated, and a virtual machine could access resources from multiple pieces of hardware.

Cloud and Networking:

Normally with the help of internet you can login to your cloud provider. After that within the cloud you can create your virtual private cloud and access the storage and software facilities so that your data is protected from accessing by some other clients. Cloud providers will provide separate network ids and subnets to work on.

Cloud is not different from normal network but a part of network where the hardware or software that needed to be installed in different geographical locations are offered as a remote service that can be accessed by people belonging to different areas.

Cloud In Railway Zone Management:

Railway requires huge software support in order to provide services relentlessly . So software that is being used can be a part of cloud and data that is needed to be accessed within departments also uses cloud.