**Computer Networks**

Smart Attendance using LAN

**Problem statement:**

Employees of a workspace(office) can give their attendance from there registered devices with the office or the cabin PC in office.

**Why Networking is required for the application:**

* Office Attendance using Biometrics has a waiting time on peak hours.
* Easy Access and more secured.
* Low cost and high-performance wireless fingerprint attendance system.
* Late comers will have a chance to save their salary cut.

**How WIFI works in office?**

● Data transmission in between the employees of the respective office.

● Best performance can be obtained from the wireless LAN.

● Office data transmission in the WLAN frequency band (up to 5.9 GHz transmission)

● WLAN band is better than ISM band, which enables higher WIFI speeds and high availability.

**Benefits of computer networks in office:**

● File sharing - you can easily share data between different users, or access it remotely.

● Resource sharing - using network-connected peripheral devices like printers, scanners and copiers, or sharing software between multiple users, saves money.

● Sharing a single internet connection - it is cost-efficient and can help protect your systems if you properly secure the network.

● Increasing storage capacity - you can access files and multimedia, such as images and music, which you store remotely on other machines.

● It also improves communication.

**Protocols in office:**

● Transmission Control Protocol (TCP)

● User Datagram Protocol (UDP)

● Internet Protocol (IP)

● File Transfer Protocol (FTP))

● Simple mail transport Protocol (SMTP), Post office Protocol (POP)

● Hyper Text Transfer Protocol (HTTP)

**1. Type of Network:**

Local Area Network

Local Area Network within the campus of organization is used to ensure the physical presence. Both wired and wireless Lan can be used.

2**. Client Configuration:**

Windows 10, Linux, Mac

Minimum Configuration required to run the respective operating system.

For Windows 10

* **Processor:** 1 gigahertz (GHz) or faster processor or SoC
* **RAM:** 1 gigabyte (GB) for 32-bit or 2 GB for 64-bit
* **Hard disk space:** 16 GB for 32-bit OS or 20 GB for 64-bit OS
* **Graphics card:** DirectX 9 or later with WDDM 1.0 driver
* **Display:** 800 x 600

**3. Server Configuration:**

* **Operating System:** Windows Server.
* **Processor:** Quad Core processor or higher recommended for optimum performance.
* **RAM:** 16GB+ recommended for optimum performance.
* **Hard disk space:** 100GB Operating System space plus Overhead space more than 1 tera byte.

**4. Types of Servers:**

* SQL Server: To record the data of attendance into the database on server.
* Application Server: Acts as middle ware between database and end user.
* Client Server: Client Server is a program that awaits and fulfils requests from client programs in the same or other computers.

5**. Network Cable and Topology:**

Star topology.

Coax, twisted-pair, or fibre optic cabling can be used as network cable.

**Networking devices used in office:**

**1) in Head office:**

a) Office Management Computer - navigational computers run on various RTOS

b) Office Router - Routers are an essential part of any larger TCP/IP network.

c) Access Points (AP) - Access Points are used to connect wirelessly connected devices to wired network.

**2) inter office dept:**

a) Office Data processing systems - all information related to the employees.

b) Access control system - recognizes authenticates and authorizes entry of a person to enter into the office.

c) Client - a normal desktop computer, or a laptop which an end user uses to perform his normal duties in the office.

d) Server - A server is a high configuration special computer, which serves resources to clients in the office.

**Why measure network performance?**

The demands on networks are increasing every day, and the need for proper network performance measurement is more important than ever before. Effective network performance translates into improved user satisfaction, whether that be internal employee efficiencies, or customer-facing network components such as an e-commerce website, making the business rationale for performance testing and monitoring self-evident.

When delivering services and applications to users, bandwidth issues, network down time, and bottlenecks can quickly escalate into IT crisis mode. Proactive network performance management solutions that detect and diagnose performance issues are the best way to guarantee ongoing user satisfaction.

The performance of a network can never be fully modelled, so measuring network performance before, during, and after updates are made and monitoring performance on an ongoing basis are the only valid methods to fully ensure network quality. While measuring and monitoring network performance parameters are essential, the interpretation and actions stemming from these metrics are equally important.

**Common Network Performance Challenges**

A common challenge that we have encountered while trying to determine network performance was the lack of a real-time provision that enables the instant detection of problems in transmission, routing, network paths, servers, bandwidth, etc. This meant that IT professionals had to conduct network measurement half-blind until they stumble upon the problems halfway. Most of the time, data gathered is never complete, as slight errors in latency or packet loss might not be detected, thus leading to technical oversights that can lead to IT crisis in the long run.

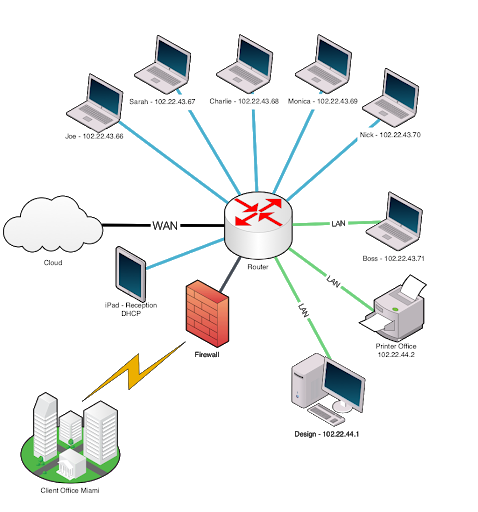
**How to Measure Network Performance**

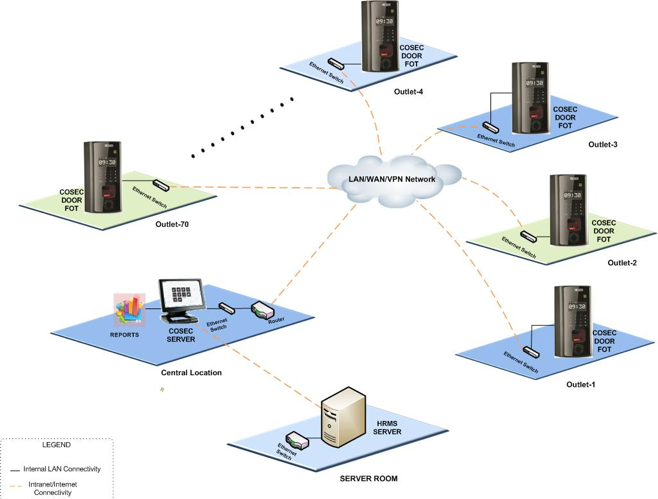
When optimizing network performance there are important metrics that must be measured. Some common metrics used to measure network performance include latency, packet loss indicators, jitter, bandwidth, and throughput.

**Performance parameters:**

|  |  |  |
| --- | --- | --- |
| **Parameter** | **Meaning** | **Formula** |
| **Bandwidth** | Bandwidth is the capacity of a wired or wireless network communications link to transmit the maximum amount of data from one point to another over a computer network or internet connection in a given amount of time | Expressed as [bits](https://web.archive.org/web/20190816003233/https:/whatis.techtarget.com/definition/bit-binary-digit) per second ([bps](https://web.archive.org/web/20190816003233/https:/searchnetworking.techtarget.com/definition/bits-per-second)), modern network links have greater capacity, which is typically measured in millions of bits per second ([megabits per second](https://web.archive.org/web/20190816003233/https:/searchnetworking.techtarget.com/definition/Mbps), or Mbps) or billions of bits per second ([gigabits per second](https://web.archive.org/web/20190816003233/https:/whatis.techtarget.com/definition/Gbps-billions-of-bits-per-second), or Gbps). |
| **Throughput** | Throughput measures the percentage of data packets that are successfully being sent; a low throughput means there are a lot of failed or dropped packets that need to be sent again. |  |
| **Packet Loss** | Packet loss occurs when one or more packets of data travelling across a computer network fail to reach their destination. Due to network congestion | Efficiency = 100% \* (transferred - retransmitted) / transferred  Network Loss = 100 - Efficiency |
| **Transmission time** | The time required for transmission of a message depends on the size of the message and the bandwidth of the channel. | Transmission time=Message size / Bandwidth |
| **Propagation Time** | Propagation time measures the time required for a bit to travel from the source to the destination. The propagation time is calculated by dividing the distance by the propagation speed. | Propagation time = Distance /Propagation speed |
| **Processing Delay** | Time taken by the processor to process the data packet is called processing delay. |  |
| **Queuing Delay** | Time spent by the data packet waiting in the queue before it is taken for execution is called queuing delay. |  |
| **Jitter** | Jitter is defined as the variation in time delay for the data packets sent over a network. This variable represents an identified disruption in the normal sequencing of data packets. Jitter is related to latency, since the jitter manifests itself in increased or uneven latency between data packets, which can disrupt network performance and lead to packet loss and network congestion. Although some level of jitter is to be expected and can usually be tolerated, quantifying network jitter is an important aspect of comprehensive network | Latency=sum of all delays    To measure Jitter, we take the difference between samples, then divide by the number of samples (minus 1). |

**Architecture diagram:**





**Analytical questions:**

10 questions and answers for various analytical questions regarding office Attendance:

1) What if employee can’t connect to server to give his/her attendance?

A) permanent fail: For back-up the hardware bio-metrics can be linked to server.

B) temporary fail: emp can try connecting/ or try from cabin PC.

2) If lots of employees are connecting for marking their attendance at the same time?

A) Each Significantly big department can have a server.

B) server can split the load between other servers

C) the connections are non-persistent.

3) How is wired network spread across department?

A) It is hybrid topology of linear and star.

B) better resource sharing and web cache management.

4) How does an employee give attendance from his personal devices?

A) That device should be registered with the office database.

B) emp can only access and give attendance from the office network

5) How does an emp give inter-dept attendance?

A) that request will go to the main server.

6) How to manage if the centralized server is crashed?

A) a backup on company’s cloud platform will be there.

7) How does the employee give the attendance for leaving the premise?

A) emp will get have to show to the security guard that emp has marked.

B) emp will get an OTP to show at the main door.

8) How will an emp get to know he is late?

A) after the attendance marked, he will get a message at what time his attendance was marked.

B) if late a warning message will pop up.

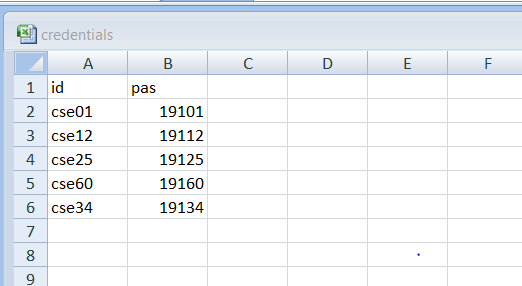
**SOCKET PROGRAMMING:**

Server Socket Programming🡪

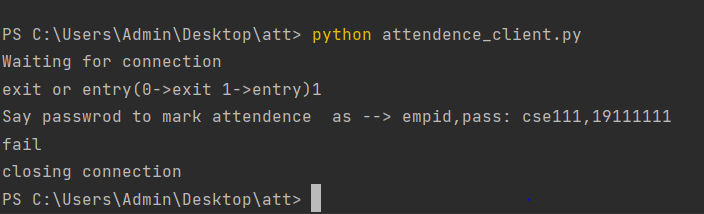
import socket  
import os  
import csv  
import datetime as d  
from \_thread import \*  
import random  
ServerSocket = socket.socket()  
host = '127.126.45.1'  
port = 1233  
ThreadCount = 0  
try:  
 ServerSocket.bind((host, port))  
except socket.error as e:  
 print(str(e))  
  
print('Waitiing for a Connection..')  
ServerSocket.listen(5pyth)  
head = ['empIP','password','timestamp','marked']  
with open("entry.csv", 'a') as csvfile:  
 w=csv.DictWriter(csvfile,fieldnames=head)  
 w.writeheader()  
csvfile.close()  
with open("exit.csv", 'a') as csvfile:  
 w=csv.DictWriter(csvfile,fieldnames=head)  
 w.writeheader()  
csvfile.close()  
  
def threaded\_client(connection):  
 connection.send(str.encode('Welcome to the Server\n'))  
  
 f=0  
 inout=connection.recv(1024)  
 inout=inout.decode('utf-8')  
  
 if inout=='1':  
 data = connection.recv(2048)  
 data=data.decode('utf-8')  
 id,pas=data.split(",")  
 print(id,pas)  
 with open('credentials.csv','r') as csvfile:  
 r=csv.reader(csvfile)  
 for x in r:  
 if(id==x[0] and pas==x[1]):  
 with open('entry.csv','a')as cf:  
 w=csv.writer(cf)  
 w.writerow([id,pas,d.datetime.now(),'p'])  
 cf.close()  
 f=1  
 ok = 'Marked present'  
 reply = 'Server Says: ' + ok  
  
 if inout=='0':  
 data = connection.recv(2048)  
 data=data.decode('utf-8')  
 id,pas=data.split(",")  
 print(id,pas)  
 with open('credentials.csv','r') as csvfile:  
 r=csv.reader(csvfile)  
 for x in r:  
 if(id==x[0] and pas==x[1]):  
 with open('exit.csv','a')as cf:  
 w=csv.writer(cf)  
 w.writerow([id,pas,d.datetime.now(),'l'])  
 cf.close()  
 f=1  
 otp=random.randrange(1001,9000,1)  
 otp=str(otp)  
 ok = 'Marked exit\n'  
 reply = 'Server Says: ' + ok+" OTP for exit is :> "+otp  
  
  
  
 if f==0:  
 oops="fail"  
 connection.send(str.encode(oops))  
  
 connection.send(str.encode(reply))  
 connection.close()  
  
while True:  
 Client, address = ServerSocket.accept()  
 print('Connected to: ' + address[0] + ':' + str(address[1]))  
 start\_new\_thread(threaded\_client, (Client, ))  
 ThreadCount += 1  
 print('Thread Number: ' + str(ThreadCount))  
ServerSocket.close()  
#py server.py

Client Socket Programming🡪

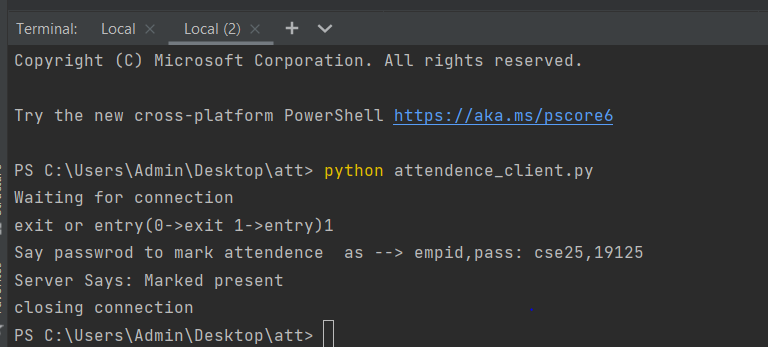
import socket  
ClientSocket = socket.socket()  
host = '127.126.45.1'  
port = 1233  
print('Waiting for connection')  
try:  
 ClientSocket.connect((host, port))  
except socket.error as e:  
 print(str(e))  
Response = ClientSocket.recv(1024)  
cate=input("exit or entry(0->exit 1->entry)")  
ClientSocket.send(str.encode(cate))  
Input = input('Say passwrod to mark att: as empid,pass: ')  
ClientSocket.send(str.encode(Input))  
Response = ClientSocket.recv(1024)  
print(Response.decode('utf-8'))  
print("closing connection")  
ClientSocket.close()  
#py multi\_c.py

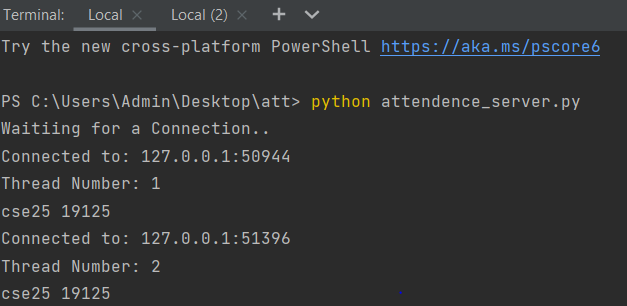


If credentials are wrong the entering of attendance fails🡪

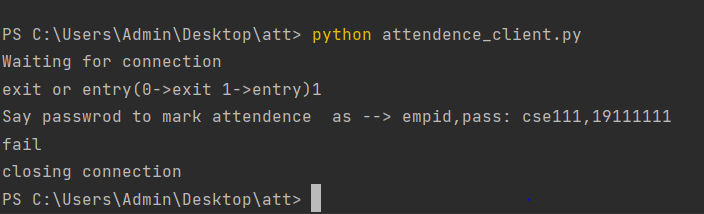


If credentials are correct🡪

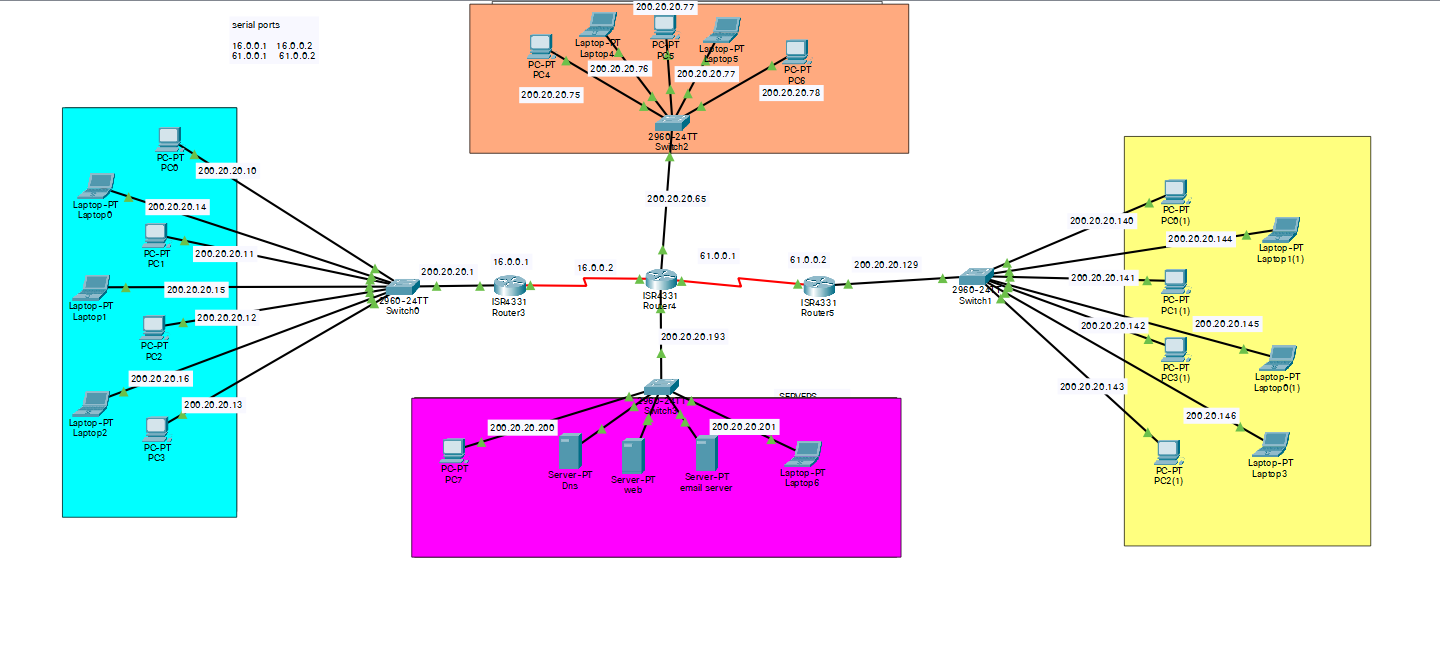




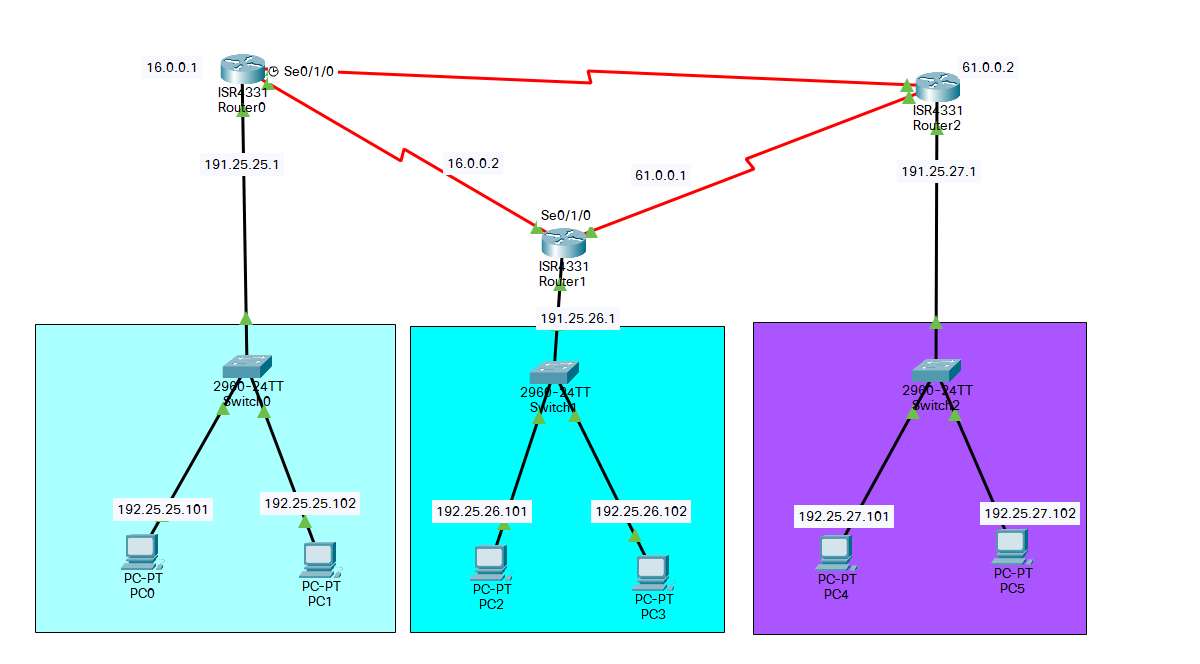
If credentials are wrong the entering of attendance fails🡪



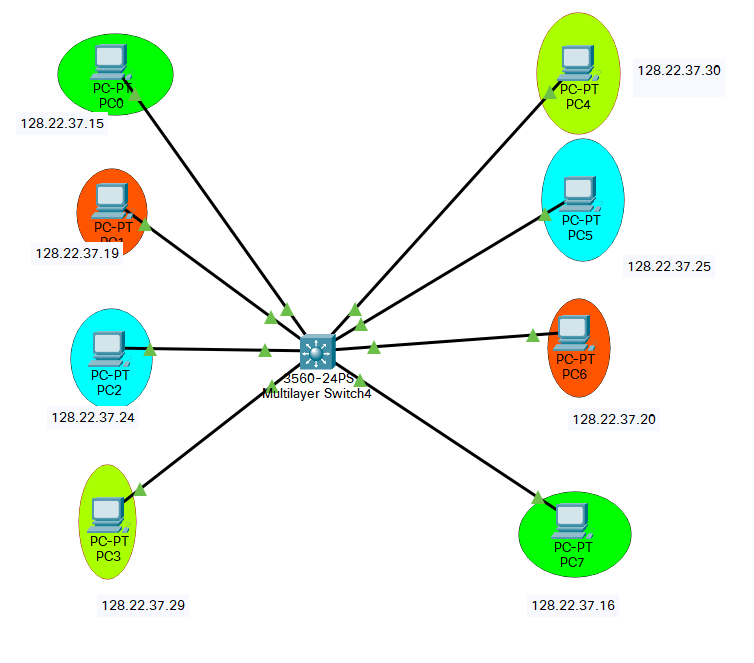
**OSPF PROTOCOL:**

****

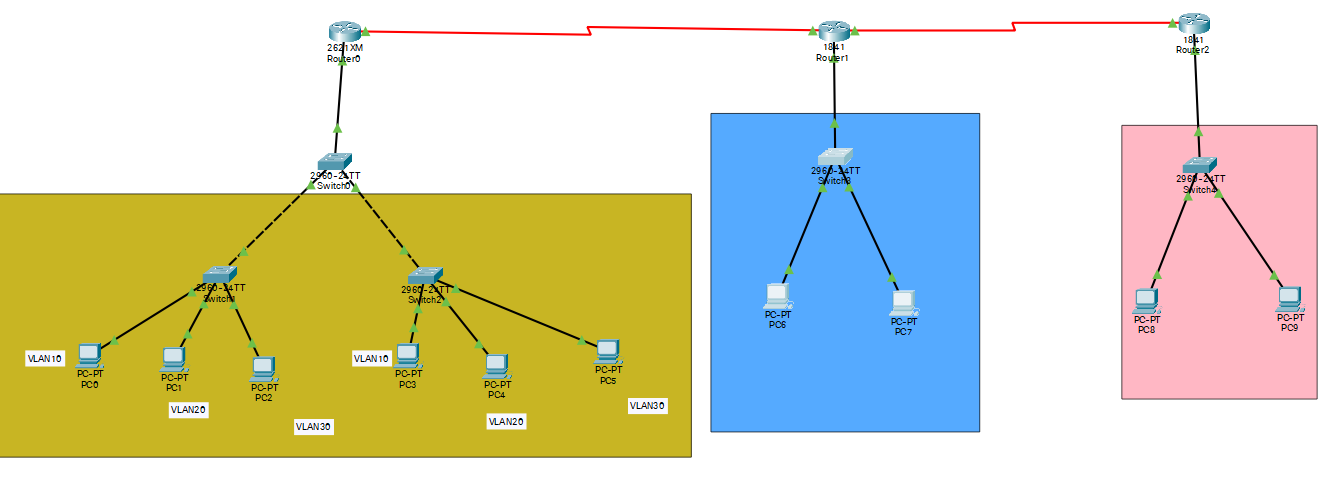
**RIP PROTOCOL:**

****

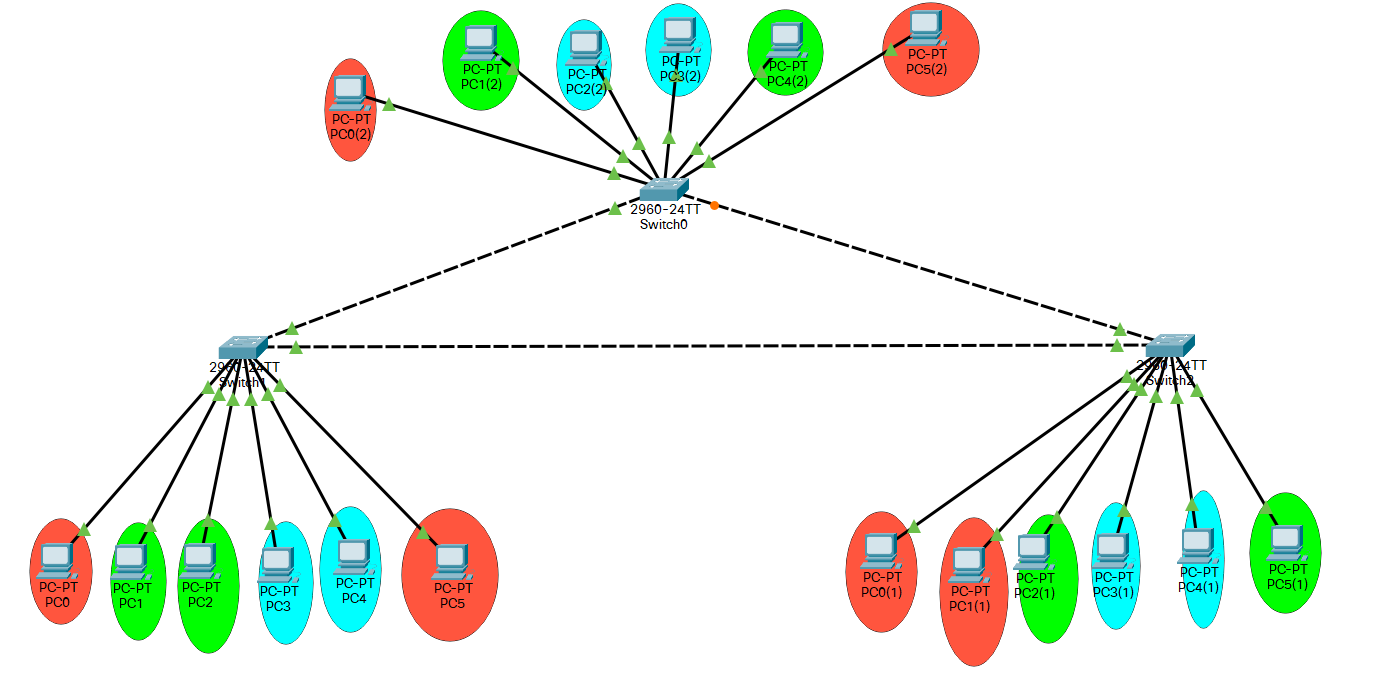
**VLAN:**

****

**INTERVLAN:**

****

**VLAN MULTIPLIER SWITCH:**

****