St. Lawrence College

Lab Report of CN

Bachelor Of Science in Computer Science

&

Information Technology (BSC.CSIT)

TU Affiliated

Submitted by:

Name: Suyog Rana Magar

Faculty: BSC.CSIT

Submitted to:

Name: Raj Kumar Shrestha

S.N	Contents	Signature
1.	Understanding of Network Equipment, Wiring in Details (CAT6 UTP EIA/TIA 568A/B Straight and Cross-Over Wiring and Testing)	
2.	Practice on Basic Networking Commands (ifconfig/ipconfig, netstat, ip, hostname, route, tracert/traceroute)	
3.	Overview of IP Addressing and Subnetting	
4.	Introduction To Cisco Packet Tracer: Basic Router Configuration	
5.	Static Routing Implementation	
6.	VLAN and VLAN Trunking	
7.	Implementation of RIP, OSPF and BGP	
8.	VMWare Workstation (Creating Virtual Machine and Installation process of Linux)	

LAB 1: Understanding of Network Equipment, Wiring in Details (CAT6 UTP EIA/TIA 568A/B Straight and Cross-Over Wiring and Testing)

Objective(s):

1. To understand the networking equipment (repeater, hub, bridge, switch, router, crimper, UTP, Fiber cable, connectors, patch panel, cable managers, racks, CAT6 straight and crossover wiring standards, LAN meter/tester, RJ-45)
2. To understand the color coding standard of UTP cable
3. To create straight and crossover cable and test/verify its connectivity.

Network Hardware: Crimper/clamper, RJ-45 jack male/female, LAN/Cable tester, UTP, Fiber cable, HUB/Switch/Router/Bridge

a. To understand the networking equipment

Repeaters are simple devices that work at the physical layer of the OSI. They regenerate signals (active hubs does that too).

Hubs are used to build a LAN by connecting different computers in a star/hierarchical network topology, the most common type on LANs now a day. A hub is a very simple (or dumb) device, once it gets bits of data sent from computer A to B, it does not check the destination, instead, it forwards that signal to all other computers (B, C, D….) within the network. B will then pick it up while other nodes discard it. This amplifies that the traffic is shared.
There are mainly two types of hubs:

- Passive: The signal is forwarded as it is (so it doesn't need power supply).
- Active: The signal is amplified, so they work as repeaters. In fact, they have been called multiport repeaters. Hub is a multiport repeater.

Hubs can be connected to other hubs using an uplink port to extend the network. Hubs work on the physical layer (lowest layer). That¡¦s the reason they can¡¦t deal with addressing or data filtering.

Switches on the other hand are more advanced. Instead of broadcasting the frames everywhere, a switch actually checks for the destination MAC address and forwards it to the relevant port to reach that computer only. This way, switches reduce traffic and divide the collision domain into segments, this is very sufficient for busy LANs and it also protects frames from being sniffed by other computers sharing the same segment.
They build a table of which MAC address belongs to which segment. If a destination MAC address is not in the table it forwards to all segments except the source segment. If the destination is same as the source, frame is discarded.

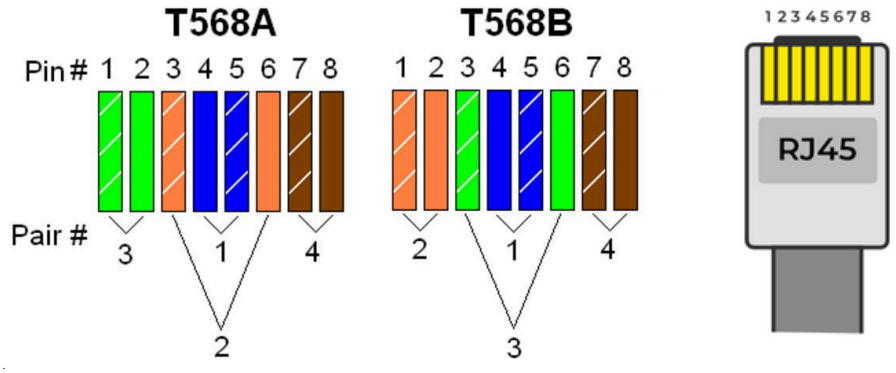
Bridges are used to extend networks by maintaining signals and traffic. Bridges are on the data link layer so in principle they are capable to do what switches do like data filtering and separating the collision domain, but they are less advanced. They are known to be used to extend distance capabilities of networks.

In a comparison with switches, bridges are slower because they use software to perform switching. They do not control broadcast domains and usually come with less number of ports. Multiport bridges are generally termed as switch.
Routers are used to connect different LANs or a LAN with a WAN (e.g. the internet). Routers control both collision domains and broadcast domains. If the packet's destination is on a different network, a router is used to pass it the right way, so without routers, the internet could not function. Routers use NAT (Network Address Translation) in conjunction with IP Masquerading to provide the internet to multiple nodes in the LAN under a single IP Address.

Routers work on the network layer so they can filter data based on IP addresses. They have routing tables to store network addresses and forward packets to the right port.
Gateways are very intelligent devices or else can be a computer running the appropriate software to connect and translate data between networks with different protocols or architecture, so their work is much more complex than a normal router. For instance, allowing communication between TCP/IP clients and IPX/SPX or AppleTalk.
Gateways operate at the network layer and above, but most of them at the application layer.

b. To understand the color coding standard of UTP cable

Background: RJ-45 connectors intended for use with CAT-6 cable are larger than their CAT-5 counterparts.
Working from left to right, the order of the wires shall be set with EIA 568 A or B standard as follows:



568 B standards (wiring sequence)	568 A standards (wiring sequence)
Partial Orange(Orange with white stripe). Solid Orange. Partial Green. Partial Blue. Solid Green. Partial Brown. Solid Brown	Partial Green(Green with white stripe). Solid Green. Partial Orange. Solid Blue. Partial Blue. Solid Orange. Partial Brown. Solid Brown

Remember for normal wiring:

- Odd Number Always holds the partial color while even number holds the solid color.
- Only 1-3, 2-6 pair of number required to be adjusted for A and B standard. Orange and
- Green are interchangeable.
- Color code for number 4, 5, 7 & 8 are always fixed.

- Standard A starts with Green and Standard B starts with Orange.

c. To create straight and crossover cable and test/verify its connectivity.

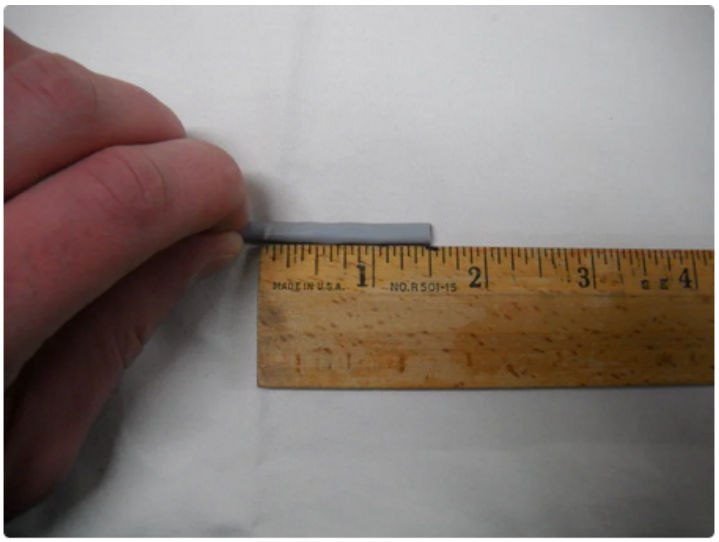
Apparatus: UTP CAT6 cable (1M), Crimper, LAN tester

There are four pairs of wires in an Ethernet cable, and an Ethernet connector (8P8C) has eight pin slots. Each pin is identified by a number, starting from left to right, with the clip facing away from you.

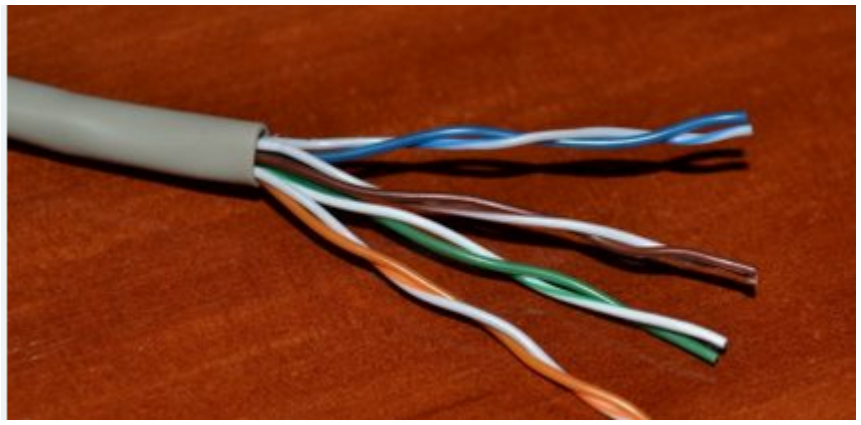
T568A Standard	
Pin 1	White/Green
Pin 2	Green
Pin 3	White/Orange
Pin 4	Blue
Pin 5	White/Blue
Pin 6	Orange
Pin 7	White/Brown
Pin 8	Brown

T568B Standard	
Pin 1	White/Orange
Pin 2	Orange
Pin 3	White/Green
Pin 4	Blue
Pin 5	White/Blue
Pin 6	Green
Pin 7	White/Brown
Pin 8	Brown

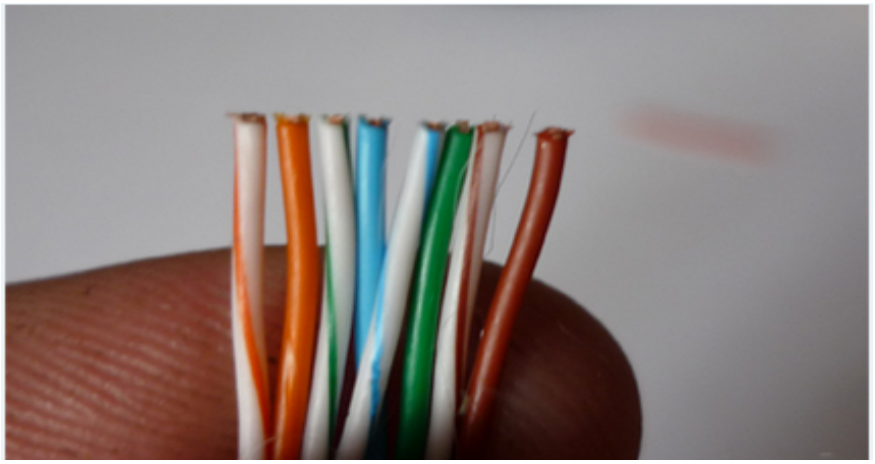
Step 1: Strip the cable jacket about 1.5 inch down from the end.



Step 2: Spread the four pairs of twisted wire apart. For Cat 5e, you can use the pull string to strip the jacket farther down if you need to, then cut the pull string. Cat 6 cables have a spine that will also need to be cut.

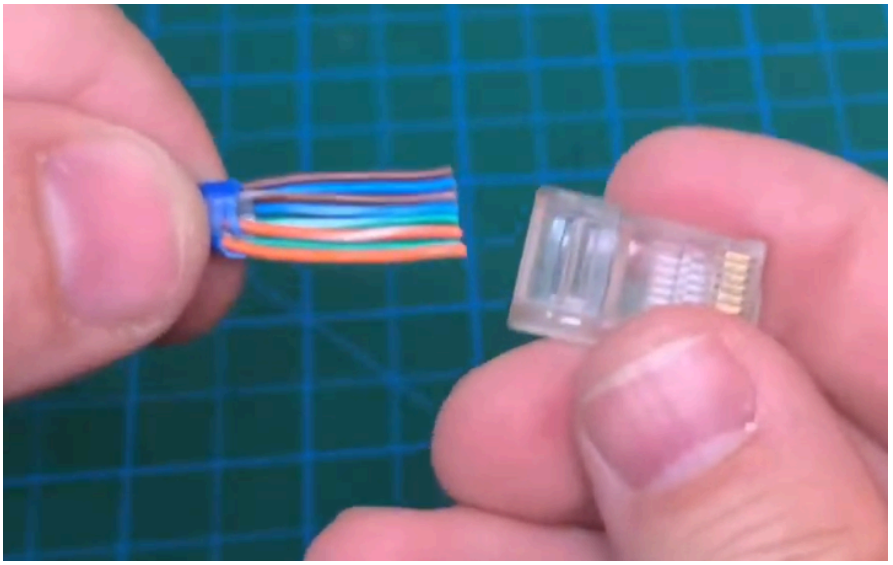


Step 3: Untwist the wire pairs and neatly align them in the T568B orientation. Be sure not to untwist them any farther down the cable than where the jacket begins; we want to leave as much of the cable twisted as possible.



Step 4: Cut the wires as straight as possible, about 0.5 inch above the end of the jacket.

Step 5: Carefully insert the wires all the way into the modular connector, making sure that each wire passes through the appropriate guides inside the connector.

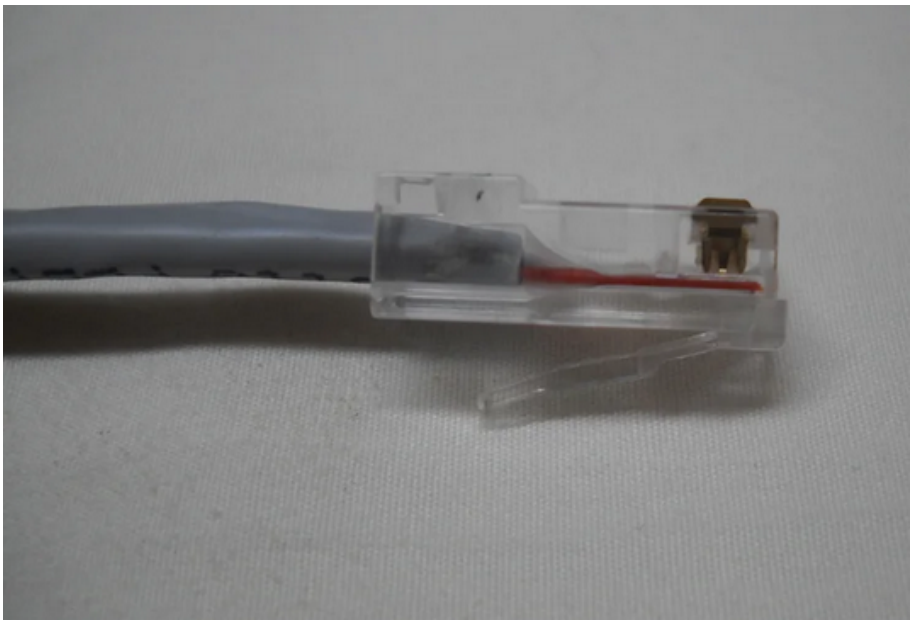


Step 6: Push the connector inside the crimping tool and squeeze the crimper all the way Down.

Step 7: Repeat steps 1-6 for the other end of the cable.

Step 8: To make sure you've successfully terminated each end of the cable, use a cable tester to test each pin.

When you're all done, the connectors should look like this:



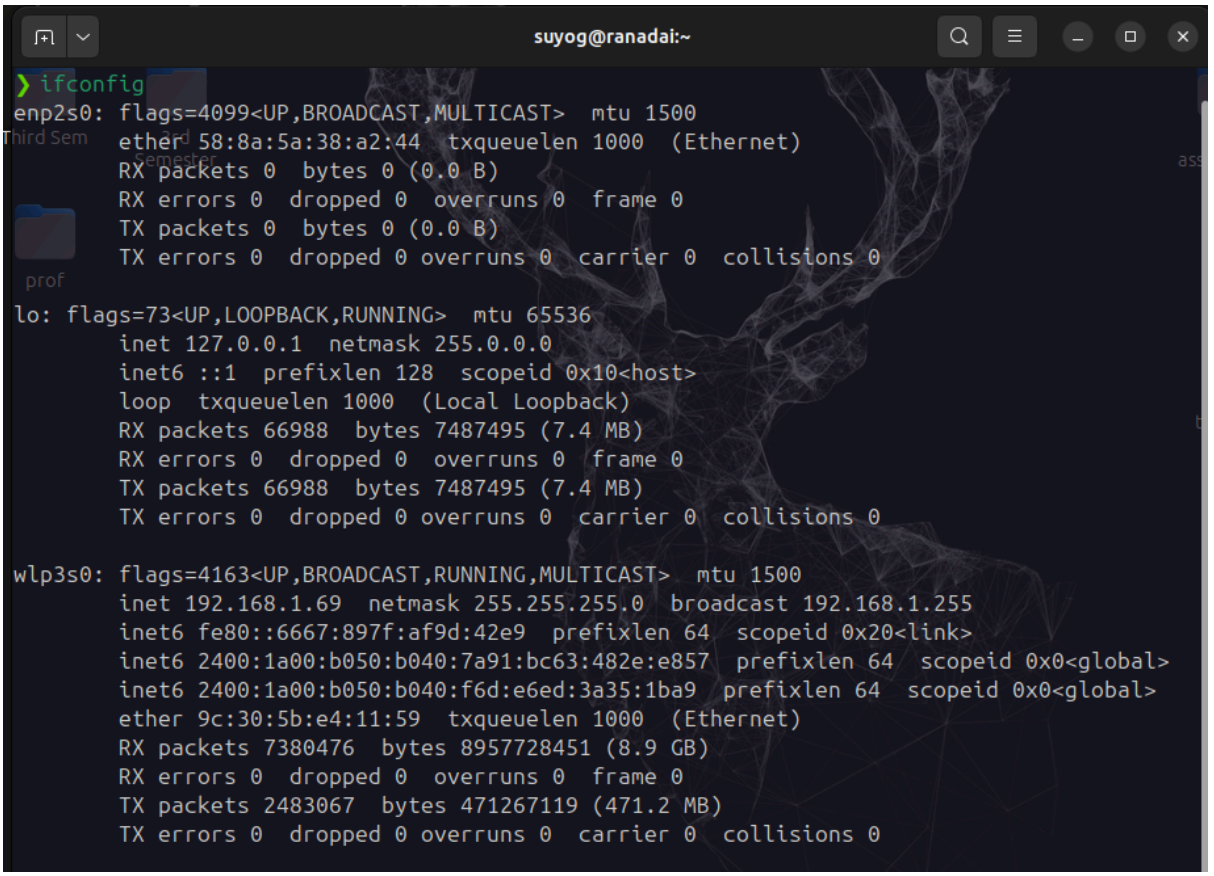
For crossover cables, simply make one end of the cable a T568A and the other end a T568B. Now you can make Ethernet cables of any length, fix broken connectors, or make yourself a crossover cable.

LAB 2: Practice on Basic Networking Commands (ifconfig/ipconfig, netstat, ip, hostname, route, tracert/traceroute)

Objective(s): To understand basic command line operation with Linux operating system and network configuration, testing and verification.

ifconfig

ifconfig is used to configure, or view the configuration of, a network interface. ifconfig stands for "interface configuration." It is used to view and change the configuration of the network interfaces on your system. ifconfig displays information about all network interfaces currently in operation.



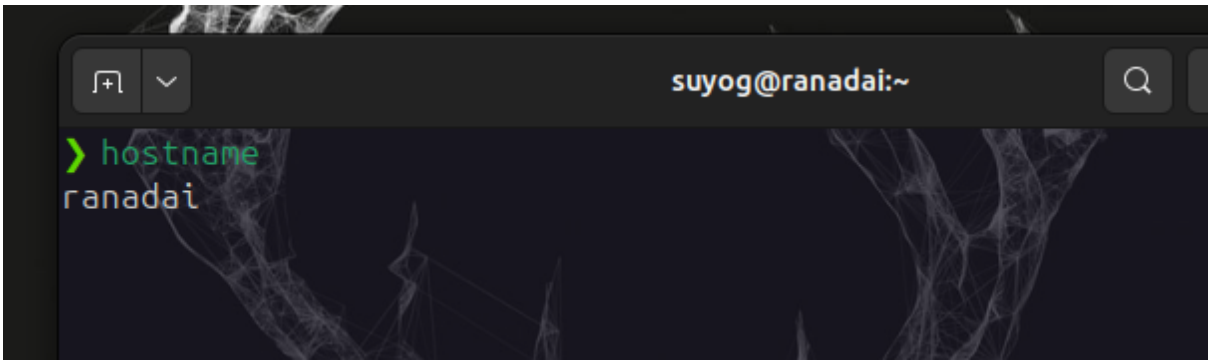
netstat

The netstat command in simple terms, it’s like a window that shows you what’s happening with your computer and the internet. This article will help you learn how to use netstat, exploring different ways to get specific information and giving you a better idea of what’s going on behind the scenes.

```
❯ netstat
Active Internet connections (w/o servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 ranadai:46194           123.208.120.34.bc:https ESTABLISHED
tcp        0      0 ranadai:58628           ec2-13-213-180-20:https ESTABLISHED
tcp        0      0 ranadai:33812           7.5.111.34.bc.goo:https ESTABLISHED
tcp        0      0 ranadai:49704           103.211.150.155:https   ESTABLISHED
tcp        0      0 ranadai:58126           ade9ecc7904667038:https ESTABLISHED
tcp        0      0 ranadai:36482           a12b7a488abeaa9e4:https ESTABLISHED
tcp        0      0 ranadai:52482           ec2-3-216-67-83.c:https ESTABLISHED
tcp        0      0 ranadai:54616           ec2-54-254-153-17:https ESTABLISHED
tcp        0      0 ranadai:44488           del11s09-in-f2.1e:https ESTABLISHED
tcp        0      0 ranadai:38772           67.199.150.82:https     ESTABLISHED
tcp        0      0 ranadai:40340           ec2-100-20-215-16:https ESTABLISHED
tcp        0      0 ranadai:44960           162.159.130.234:https   ESTABLISHED
tcp        0      0 ranadai:55770           88.199.214.35.bc.:https ESTABLISHED
tcp        0      0 ranadai:42784           ec2-54-151-166-24:https ESTABLISHED
tcp        0      0 ranadai:33640           ec2-52-18-140-132:https ESTABLISHED
tcp        0      0 ranadai:57658           server-52-84-45-8:https ESTABLISHED
tcp        0      0 ranadai:47002           ns1016845.ip-15-2:https ESTABLISHED
tcp        0      0 ranadai:53300           69.173.144.165:https    TIME_WAIT
tcp        0      0 ranadai:59436           ec2-44-218-39-160:https ESTABLISHED
```

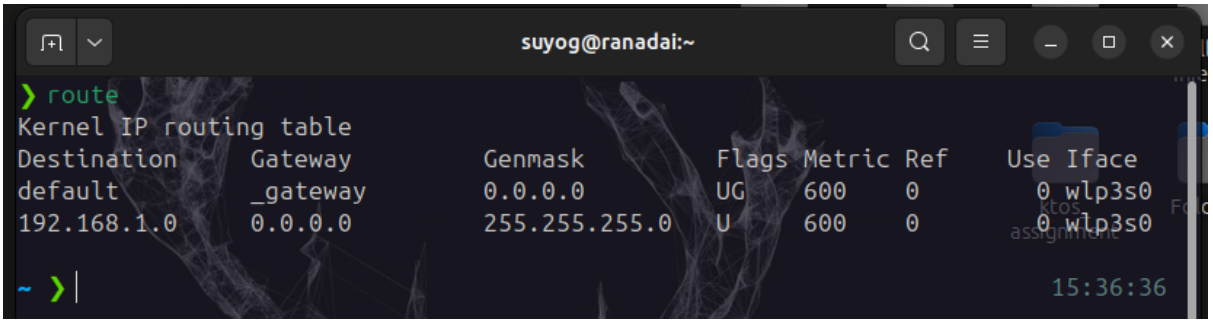
hostname

This command is used to display the hostname of a computer. The hostname is the unique identifier assigned to a device on a network.



route

Route command is used to show/manipulate the IP routing table. It is primarily used to set up static routes to specific host or networks via an interface.



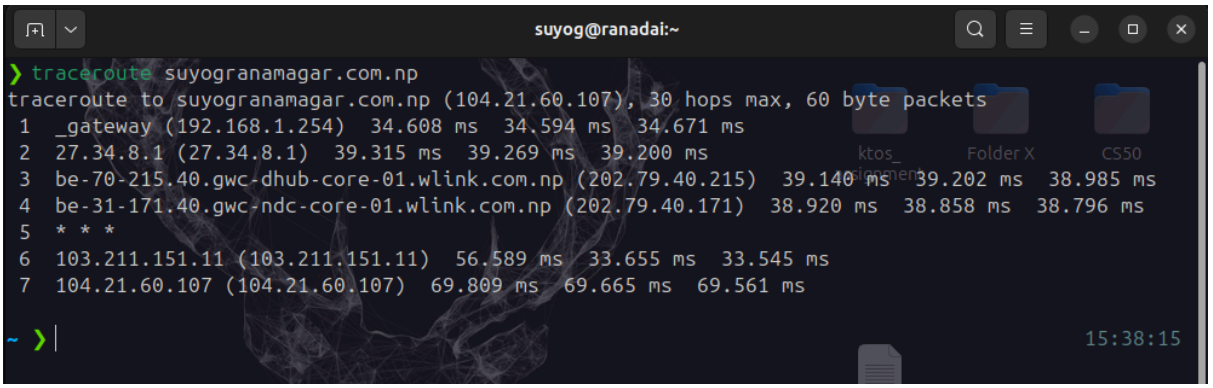
Tracert/Traceroute

It a command-line utility that we can use to trace the path that an Internet Protocol (IP) packet takes to its destination. The TRACERT diagnostic utility determines the route to a destination by sending Internet Control Message Protocol (ICMP) echo packets to the destination. In these packets, TRACERT uses varying IP Time-To-Live (TTL) values. Because each router along the path is required to decrement the packet's TTL by at least 1 before forwarding the packet, the TTL is effectively a hop counter. When the TTL on a packet reaches zero (0), the router sends an ICMP "Time Exceeded" message back to the source computer.

• e.g.

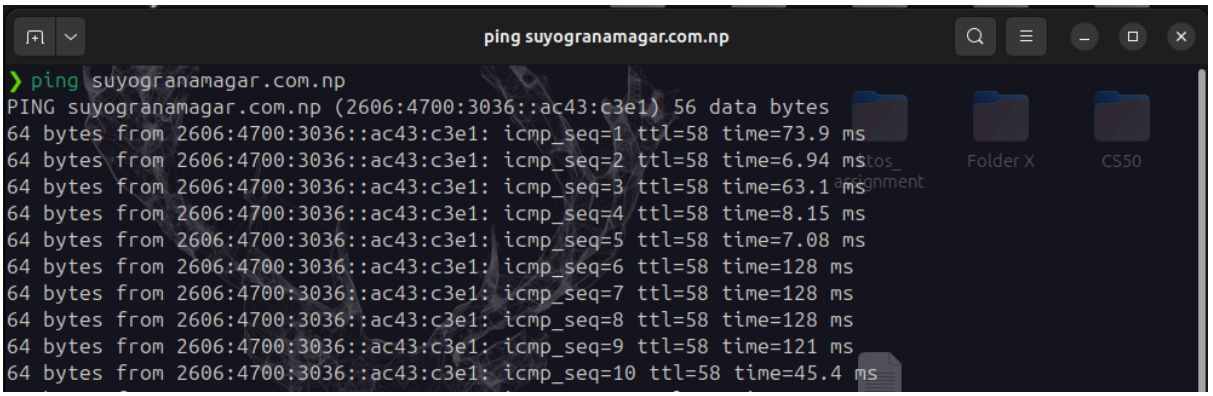
tracert google.com

tracert 216.58.196.131



Ping

The ping command is used to test the reachability of a remote host by sending ICMP (Internet Control Message Protocol) echo request packets and waiting for ICMP echo reply packets. It's commonly used to troubleshoot network connectivity and measure round-trip times.



LAB 3: Overview of IP Addressing and Subnetting

Objective(s):

To understand theoretical knowledge of IPv4 Addressing and Subnetting.

Background:

An IP address (internet protocol address) is a numerical representation that uniquely identifies a specific interface on the network. Addresses in IPv4 are 32-bits long. This allows for a maximum of 4,294,967,296 (232) unique addresses. Addresses in IPv6 are 128-bits, which allows for 3.4 x 1038 (2128) unique addresses. IP addresses are binary numbers but are typically expressed in decimal form (IPv4) or hexadecimal form (IPv6) to make reading and using them easier for humans.

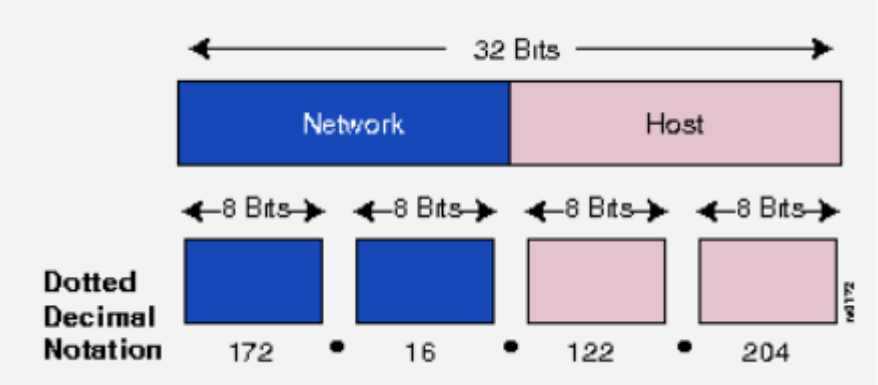
Terminologies

• IPv4 address: a 32-bit number, usually written in dotted decimal form, that uniquely identifies an interface of some computer
• Host Address: another term for IP address of the end device
• Network: a group of hosts, all of which have an identical beginning position of their ip addresses.
• Broadcast Address: a 32-bit number that is used to address all hosts in the network. It can't be assigned as an ip address of a host.
• Subnet: a group of hosts, all of which have an identical portion of their ip addresses, a subnet differs from a network in that a subnet is a further subdivision of a network.
• Sub-netting: the process of subdividing networks into smaller subnets.
• Subnet mask: A 32-bit combination used to describe which portion of an address refers to the subnet and which part refers to the host.

IPv4 Address representations

IPv4 addresses are actually 32-bit binary numbers, consisting of the two identifiers which, identify the network and the host to the network. It is generally represented as 4 octets of numbers from 0-255 represented in decimal form instead of binary form.

The IP address is divided into two main parts; the Network Number and the Host Number. The host number identifies a host in the network and is assigned by the local network administrator.



Subnet Masks
A single IP address identifies both a network, and a unique interface on that network. A subnet mask can also be written in dotted decimal notation and determines where the network part of an IP address ends, and the host portion of the address begins. When expressed in binary, any bit set to one means the corresponding bit in the IP address is part of the network address. All the bits set to zero mark the corresponding bits in the IP address as part of the host address. The bits marking the subnet mask must be consecutive ones. Most subnet masks start with 255. And continue on until the network mask ends.

A Class A network mask is defined as 255.0.0.0.
A Class B network mask is defined as 255.255.0.0.
A Class C network mask would be 255.255.255.0

IP Address Classes

Class	Leading bits	Size of *network number* bit field	Size of *rest* bit field	Number of networks	Addresses per network	Total addresses in class	Start address	End address
Class A	0	8	24	128 (2^7)	16,777,216 (2^{24})	2,147,483,648 (2^{31})	0.0.0.0	127.255.255.255
Class B	10	16	16	16,384 (2^{14})	65,536 (2^{16})	1,073,741,824 (2^{30})	128.0.0.0	191.255.255.255
Class C	110	24	8	2,097,152 (2^{21})	256 (2^8)	536,870,912 (2^{29})	192.0.0.0	223.255.255.255
Class D (multicast)	1110	not defined	not defined	not defined	not defined	268,435,456 (2^{28})	224.0.0.0	239.255.255.255
Class E (reserved)	1111	not defined	not defined	not defined	not defined	268,435,456 (2^{28})	240.0.0.0	255.255.255.255

Before variable length subnet masks allowed networks of any size to be configured, the IPv4 address space was broken into five classes.

Class A Address
The first bit of the first octet is always set to 0 (zero). Thus the first octet ranges from 1 – 127. The default subnet mask for Class A IP address is 255.0.0.0 which implies that Class A addressing can have 126 networks (27-2) and 16,777,214 hosts (224-2). (Note: 0 Octet is forbidden in RFC and 127 is reserved for loopback testing.)

Class B Address
An IP address which belongs to class B has the first two bits in the first octet set to 10. The default subnet mask for Class B is 255.255.0.0. Class B has 16384 (214) Network addresses and 65534 (216-2) Host addresses.

Class C Address
The first octet of Class C IP address has its first 3 bits set to 110. The default subnet mask for Class C is 255.255.255.0. Class C gives 2097152 (221) Network addresses and 254 (28-2) Host addresses.

Class D Address

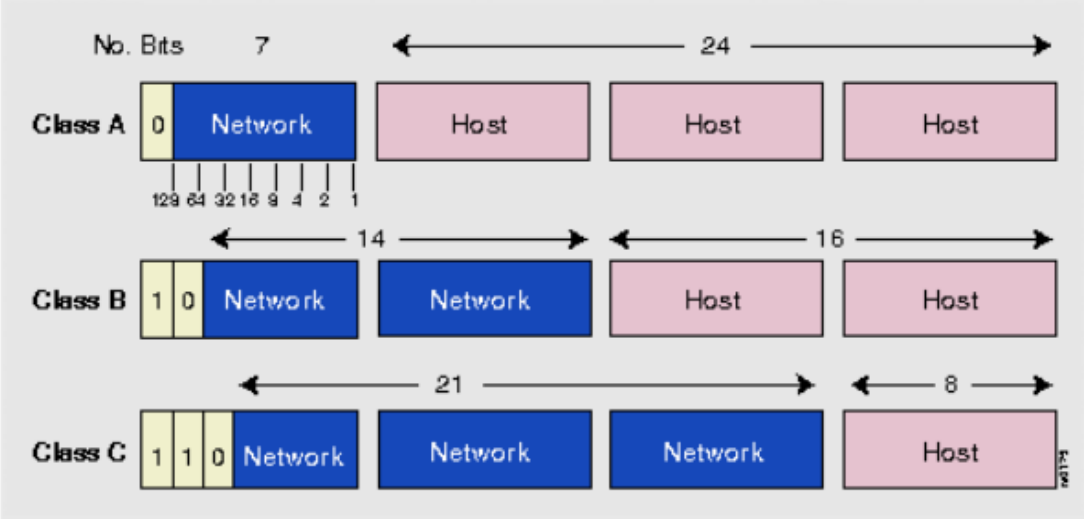
The first four bits of the first octet in Class D IP addresses are set to 1110. Class D has IP address range from 224.0.0.0 to 239.255.255.255. Class D is reserved for Multicasting. In multicasting data is not destined for a particular host, that is why there is no need to extract host address from the IP address, and Class D does not have any subnet mask.

Class E Address

The first four bits of the first octet in Class D IP addresses are set to 1111. This IP Class is reserved for experimental purposes only for R&D or Study. IP addresses in this class ranges from 240.0.0.0 to 255.255.255.254. Like Class D, this class too is not equipped with any subnet mask.

Network and Host Portion:

A Class A address has the first octet as the network portion and the remaining 3 octets as the host portion. A Class B address has the first and second octets as the network portion and the third and fourth octets as the host portion. A Class C address has the first, second, and third octet as the network portion and the last octet as the host portion.



Class A

```
0. 0. 0. 0 = 00000000.00000000.00000000.00000000
127.255.255.255 = 01111111.11111111.11111111.11111111
0nnnnnnn.HHHHHHHH.HHHHHHHH.HHHHHHHH
```

Class B

```
128. 0. 0. 0 = 10000000.00000000.00000000.00000000
191.255.255.255 = 10111111.11111111.11111111.11111111
10nnnnnn.nnnnnnnn.HHHHHHHH.HHHHHHHH
```

Class C

```
192. 0. 0. 0 = 11000000.00000000.00000000.00000000
```

```
223.255.255.255 = 11011111.11111111.11111111.11111111
110nnnnn.nnnnnnnn.nnnnnnnn.HHHHHHHH
```

Class D

```
224. 0. 0. 0 = 11100000.00000000.00000000.00000000
239.255.255.255 = 11101111.11111111.11111111.11111111
1110XXXX.XXXXXXXX.XXXXXXXX.XXXXXXXX
```

Class E

```
240. 0. 0. 0 = 11110000.00000000.00000000.00000000
255.255.255.255 = 11111111.11111111.11111111.11111111
1111XXXX.XXXXXXXX.XXXXXXXX.XXXXXXXX
```

Private addresses

Within the address space, certain networks are reserved for private networks. Packets from these networks are not routed across the public internet. This provides a way for private networks to use internal IP addresses without interfering with other networks. The private networks are

- Class A Private Range: 10.0.0.0 to 10.255.255.255
- Class B Private APIPA Range: 169.254.0.0 to 169.254.255.255
- Class B Private Range: 172.16.0.0 to 172.31.255.255
- Class C Private Range: 192.168.0.0 to 192.168.255.255

Special addresses

Certain IPv4 addresses are set aside for specific uses:

127.0.0.0	Loopback address (the host's own interface)
224.0.0.0	IP Multicast
255.255.255.255	Broadcast (sent to all interface on network)

IPV4 Subnetting

Each IP class is equipped with its own default subnet mask which bounds that IP class to have prefixed number of Networks and prefixed number of Hosts per network. Classful IP addressing does not provide any flexibility of having less number of Hosts per Network or more Networks per IP Class.

CIDR or Classless Inter Domain Routing provides the flexibility of borrowing bits of Host part of the IP address and using them as Network in Network, called Subnet. By using subnetting, one single Class A IP address can be used to have smaller sub-networks which provides better network management capabilities.

Class A Subnets

In Class A, only the first octet is used as Network identifier and rest of three octets are used to be assigned to Hosts (i.e. 16777214 Hosts per Network). To make more subnet in Class A, bits from Host part are borrowed and the subnet mask is changed accordingly.

For example, if one MSB (Most Significant Bit) is borrowed from host bits of second octet and added to Network address, it creates two Subnets (21=2) with (223-2) 8388606 Hosts per Subnet. The Subnet mask is changed accordingly to reflect subnetting.

In case of subnetting too, the very first and last IP address of every subnet is used for Subnet Number and Subnet Broadcast IP address respectively. Because these two IP addresses cannot be assigned to hosts, sub-netting cannot be implemented by using more than 30 bits as Network Bits, which provides less than two hosts per subnet.

Class B Subnets
By default, using Classful Networking, 14 bits are used as Network bits providing (214) 16384 Networks and (216-2) 65534 Hosts. Class B IP Addresses can be subnetted the same way as Class A addresses, by borrowing bits from Host bits.

Class C Subnets
Class C IP addresses are normally assigned to a very small size network because it can only have 254 hosts in a network. Given below is a list of all possible combination
of subnetted Class B IP address

Network Bits	Subnet Mask	Bits Borrowed	Subnets	Hosts/Subnet
24	255.255.255.0	0	1	254
25	255.255.255.128	1	2	126
26	255.255.255.192	2	4	62
27	255.255.255.224	3	8	30
28	255.255.255.240	4	16	14
29	255.255.255.248	5	32	6
30	255.255.255.252	6	64	2

LAB 4 : Introduction to Packet Tracer: Basic Router Configuration

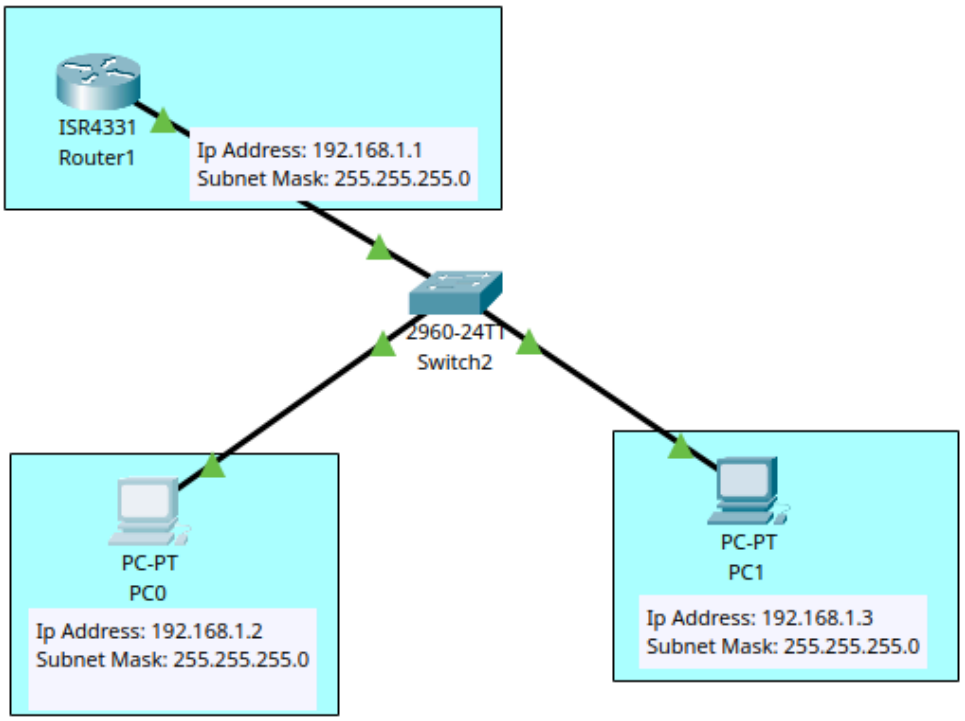
Objective(s)

To Understand Basic Commands for Router Configuration

Background

Packet Tracer is a powerful network simulator that can be utilized in training for network certification like and learning by allowing students to create networks with an almost unlimited number of devices and to experience troubleshooting without having to buy real Suyog routers or switches. The tool is created by Cisco Systems. The purpose of Packet Tracer is to offer students a tool to learn the principles of networking.

Router>	User Mode
Router#	Privileged Mode
Router(config)#	Global configuration mode
Router(config-if)#	Interface mode
Router(ocnfig-subig)#	Subinterface mode
Router(config-line)#	Line mode
Router(config-router)#	Router configuration mode



Configure the following in Router and Switch as illustrated in the figure:

1. Change Hostname (Suyog)
2. Configure passwords (password: Suyog & secret: class)
3. Secure Console Port and Terminal lines (password: Suyog)
4. Encrypt Passwords (service password-encryption)
5. Configure Clock (clock)
6. Configure Banners (banner motd)
7. Configure Interface (IP Address) on Router (interface fa0/0 or fa0/1)
8. Configure VLAN on Switch(interface vlan 1)
9. Save configurations (running-config to startup-config)
10. Use show commands
show running-config,
show startup-config,
show ip interface brief,
show interface vlan 1
11. Configure PCs
12. Verify Connectivity (ping)

Router Configuration

```
Router>enable
Router#config t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#hostname Suyog
Suyog(config)#enable password Suyog
Suyog(config)#enable secret class
Suyog(config)#line console 0
Suyog(config-line)#password Suyog
Suyog(config-line)#login
Suyog(config-line)#line vty 0 4
Suyog(config-line)#password Suyog
Suyog(config-line)#login
Suyog(config-line)#exit
Suyog(config)#service password-encryption
Suyog(config)#exit
Suyog#
%SYS-5-CONFIG_I: Configured from console by console

Suyog#clock ?
  set  Set the time and date
Suyog#clock set ?
  hh:mm:ss  Current Time
Suyog#clock set 11:26:00 ?
  <1-31>  Day of the month
  MONTH   Month of the year
Suyog#clock set 11:26:00 September ?
  <1-31>  Day of the month
```

```
Suyog#clock set 11:26:00 September 21 ?
  <1993-2035>  Year
Suyog#clock set 11:26:00 September 21  2024
Suyog#config t
Enter configuration commands, one per line.  End with CNTL/Z.
Suyog(config)#banner motd $ UNAUTHORISED ACCESS RESTRICTED $
Suyog(config)#interface fastethernet 0/0
%Invalid interface type and number
Suyog(config)#interface fastethernet 0/0
%Invalid interface type and number
Suyog(config)#interface FastEthernet0/1
%Invalid interface type and number
Suyog(config)#
Suyog(config)#
Suyog(config)#interface GigabitEthernet0/0/0
Suyog(config-if)#
Suyog(config-if)#
Suyog(config-if)#exit
Suyog(config)#interface GigabitEthernet0/0/0
Suyog(config-if)#ip address 192.168.1.1 255.255.255.0
Suyog(config-if)#no shutdown

Suyog(config-if)#
%LINK-5-CHANGED: Interface GigabitEthernet0/0/0, changed
state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface
GigabitEthernet0/0/0, changed state to up

Suyog(config-if)#exit
Suyog(config)#exit
Suyog#
%SYS-5-CONFIG_I: Configured from console by console

Suyog#copy running-config startup-config
Destination filename [startup-config]?
Building configuration...
[OK]
Suyog#
```

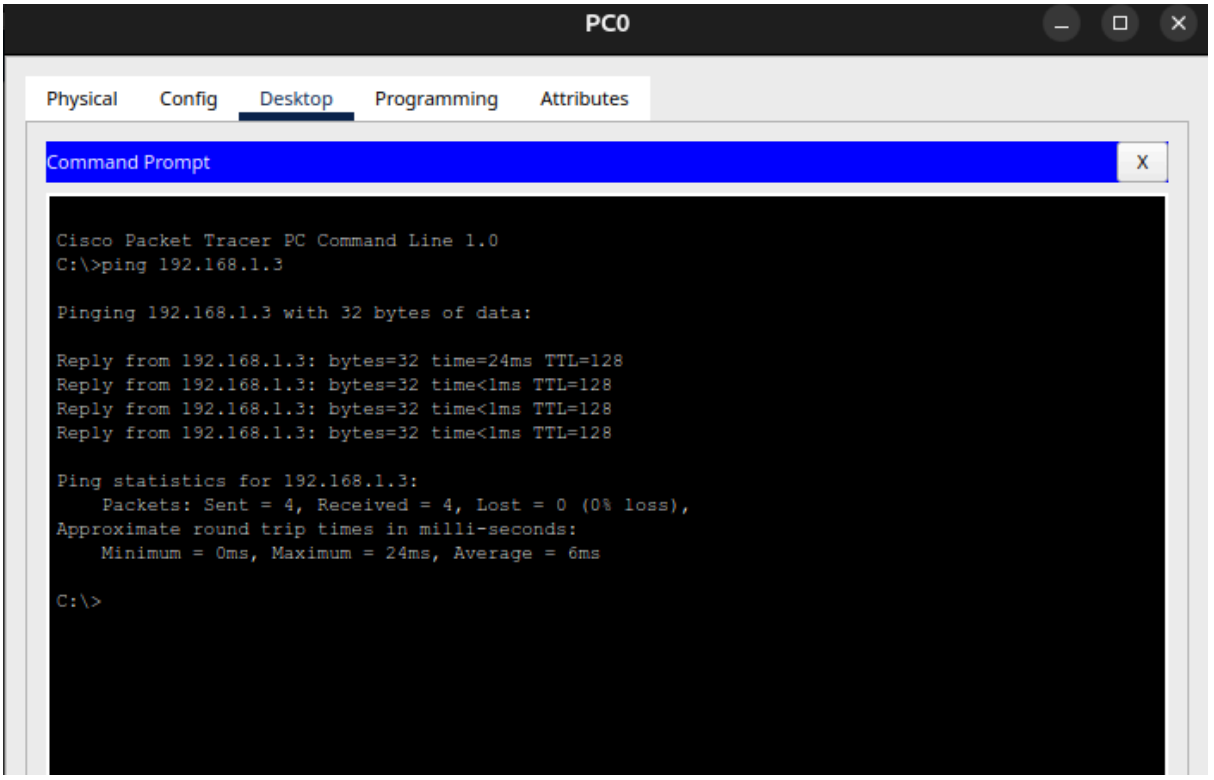
Switch Configuration:

```
Switch>enable
Switch#config t
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)#interface vlan 1
Switch(config-if)#ip address 192.168.1.4 255.255.255.0
Switch(config-if)#no shutdown
Switch(config-if)#
%LINK-5-CHANGED: Interface Vlan1, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface Vlan1,
changed state to up

Switch(config-if)#
```

Output:



LAB 5: Static Routing Implementation

Objective(s)

To understand the Static Routing, its Advantages and Drawbacks

Background

Static Routing

Static routing is useful in small network where numbers of routes are limited. In static routing we need to add route manually with IP route command. Like other routing methods static routing also has its pros and cons.

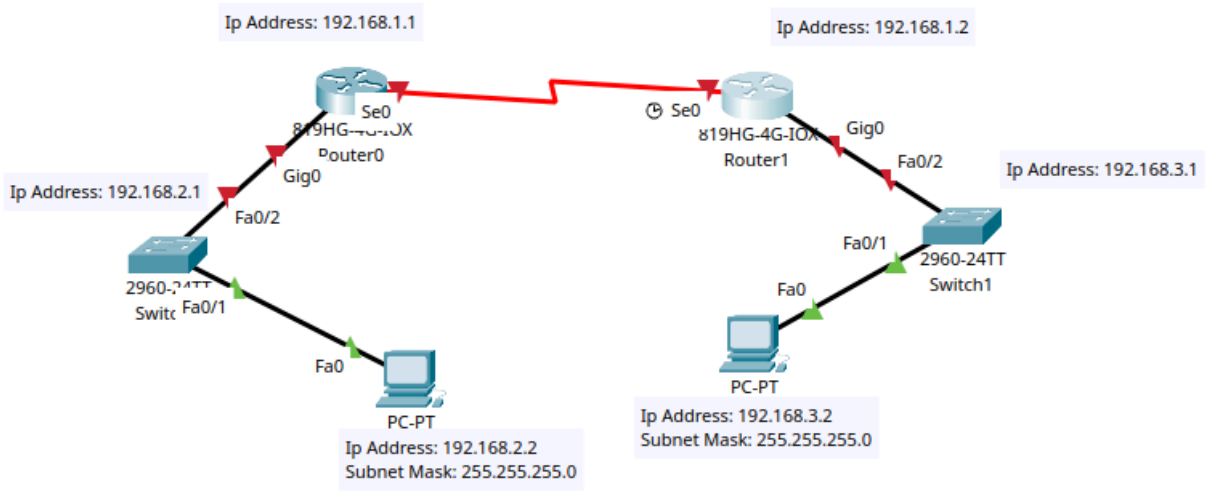
Advantage of static routing

- It is easy to implement.
- It is most secure way of routing, since no information is shared with other routers.
- It puts no overhead on resources such as CPU or memory.

Disadvantage of static routing

- It is suitable only for small network.
- If a link fails static route cannot reroute the traffic.

Configuration



1. Router Basic Configuration

Router 0

```
Router>enable
Router#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#hostname Suyog
Suyog(config)#enable password cisco
Suyog(config)#enable secret class
Suyog(config)#line console 0
Suyog(config-line)#password cisco
Suyog(config-line)#login
Suyog(config-line)#
Suyog(config-line)#line vty 0 15
Suyog(config-line)#password cisco
```

```
Suyog(config-line)#login
Suyog(config-line)#
Suyog(config-line)#line aux 0
Suyog(config-line)#password cisco
Suyog(config-line)#login
Suyog(config-line)#
Suyog(config-line)#exit
Suyog(config)#service password-encryption
```

Router 1

```
Router#
Router#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#hostname Suyog
Suyog(config)#enable password cisco
Suyog(config)#enable secret class
Suyog(config)#line console 0
Suyog(config-line)#password cisco
Suyog(config-line)#login
Suyog(config-line)#
Suyog(config-line)#line vty 0 15
Suyog(config-line)#password cisco
Suyog(config-line)#login
Suyog(config-line)#
Suyog(config-line)#line aux 0
Suyog(config-line)#password cisco
Suyog(config-line)#login
Suyog(config-line)#
Suyog(config-line)#exit
Suyog(config)#service password-encryption
```

2. Router Interface Configuration

Router 0

Serial Link

```
Suyog(config)#interface serial 0/0/0
Suyog(config-if)#description Link to Suyog
Suyog(config-if)#ip address 192.168.1.1 255.255.255.0
Suyog(config-if)#clock rate 64000
Suyog(config-if)#no shutdown
Suyog(config-if)#
```

Fast Ethernet

```
Suyog(config-if)#
Suyog(config)#interface fastethernet 0/0
Suyog(config-if)#ip address 192.168.2.1 255.255.255.0
```

```
Suyog(config-if)#no shutdown
Suyog(config-if)#exit
```

Router 1

Serial Link

```
Suyog(config)#interface serial 0/0/0
Suyog(config-if)#des link from LAN to internet
Suyog(config-if)#ip address 192.168.1.2 255.255.255.0
Suyog(config-if)#no shutdown
Suyog(config-if)#exit
Fast Ethernet
Suyog(config-if)#interface fastethernet 0/0
Suyog(config-if)#ip address 192.168.3.1 255.255.255.0
Suyog(config-if)#no shutdown
Suyog(config-if)#exit
```

3. Routes Configuration

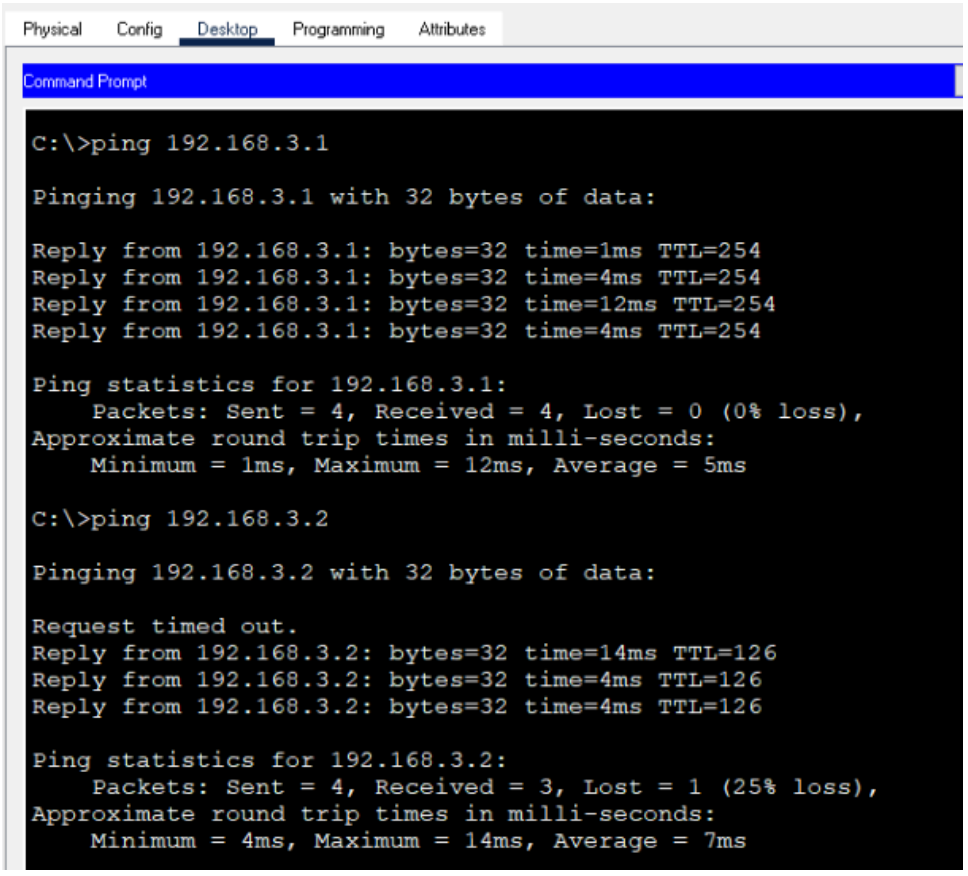
Router 0

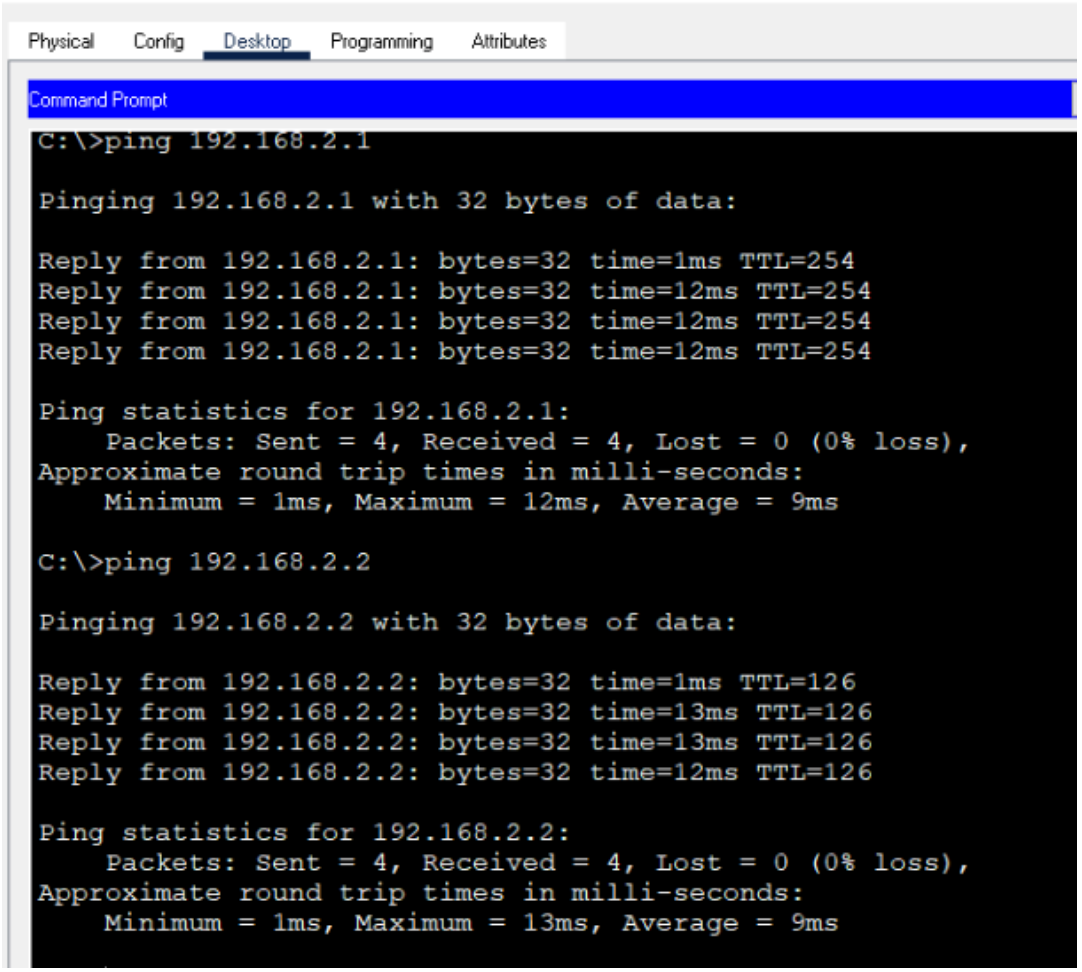
```
Suyog(config)#ip route 192.168.3.0 255.255.255.0 192.168.1.2
```

Router 1

```
Suyog(config)#ip route 192.168.2.0 255.255.255.0 192.168.1.1
```

Output:

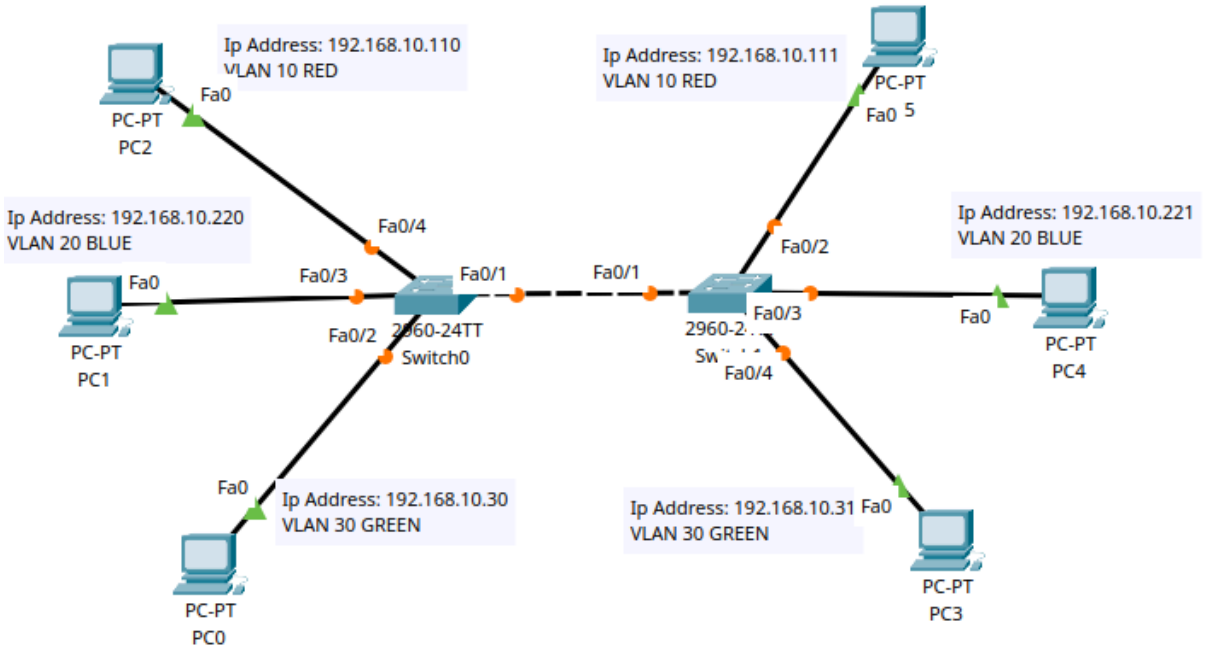



Physical
Config
Desktop
Programming
Attributes
Command Prompt
C:\>ping 192.168.2.1
Pinging 192.168.2.1 with 32 bytes of data:
Reply from 192.168.2.1: bytes=32 time=1ms TTL=254
Reply from 192.168.2.1: bytes=32 time=12ms TTL=254
Reply from 192.168.2.1: bytes=32 time=12ms TTL=254
Reply from 192.168.2.1: bytes=32 time=12ms TTL=254
Ping statistics for 192.168.2.1:
Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
Minimum = 1ms, Maximum = 12ms, Average = 9ms
C:\>ping 192.168.2.2
Pinging 192.168.2.2 with 32 bytes of data:
Reply from 192.168.2.2: bytes=32 time=1ms TTL=126
Reply from 192.168.2.2: bytes=32 time=13ms TTL=126
Reply from 192.168.2.2: bytes=32 time=13ms TTL=126
Reply from 192.168.2.2: bytes=32 time=12ms TTL=126
Ping statistics for 192.168.2.2:
Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
Minimum = 1ms, Maximum = 13ms, Average = 9ms

LAB 6: VLAN and VLAN Trunking.

Objective(s):
To understand LAN networking, creation of VLAN, IP addressing in the VLAN and VLAN Trunk.

VLAN Trunking Protocol (VTP) is a Cisco proprietary protocol that propagates the definition of Virtual Local Area Networks (VLAN) on the whole local area network. To do this, VTP carries VLAN information to all the switches in a VTP domain.
Trunk links are required to pass VLAN information between switches. A port on a Cisco switch is either an access port or a trunk port. Access ports belong to a single VLAN and do not provide any identifying marks on the frames that are passed between switches. Access ports also carry traffic that comes from only the VLAN assigned to the port. A trunk port is by default a member of all the VLANs that exist on the switch and carry traffic for all those VLANs between the switches. To distinguish between the traffic flows, a trunk port must mark the frames with special tags as they pass between the switches. Trunking is a function that must be enabled on both sides of a link.



1. Configuration VLAN on Both Switches

Switch 0

```
Switch>enable
Switch#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)#hostname Srm1
Srm1(config)#vlan 10
Srm1(config-vlan)#name RED
Srm1(config-vlan)#exit
Srm1(config)#vlan 20
Srm1(config-vlan)#name BLUE
```

```
Srm1(config-vlan)#exit
Srm1(config)#vlan 30
Srm1(config-vlan)#name GREEN
Srm1(config-vlan)#exit
Srm1(config)#exit
Srm1#
%SYS-5-CONFIG_I: Configured from console by console
show vlan brief
```

```
VLAN Name                                Status    Ports
---- ----------------------------------- --------- -------------------------------
1    default                             active    Fa0/1, Fa0/2,
Fa0/3, Fa0/4
                                                   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   RED                                 active
20   BLUE                                active
30   GREEN                               active
1002 fddi-default                        active
1003 token-ring-default                  active
1004 fddinet-default                     active
1005 trnet-default                       active
```

Switch 1

```
Switch>enable
Switch#configure t
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)#hostname srm2
srm2(config)#vlan 10
srm2(config-vlan)#name RED
srm2(config-vlan)#exit
srm2(config)#vlan 20
srm2(config-vlan)#name BLUE
srm2(config-vlan)#exit
srm2(config)#vlan 30
srm2(config-vlan)#name GREEN
srm2(config-vlan)#exit
srm2(config)#exit
```

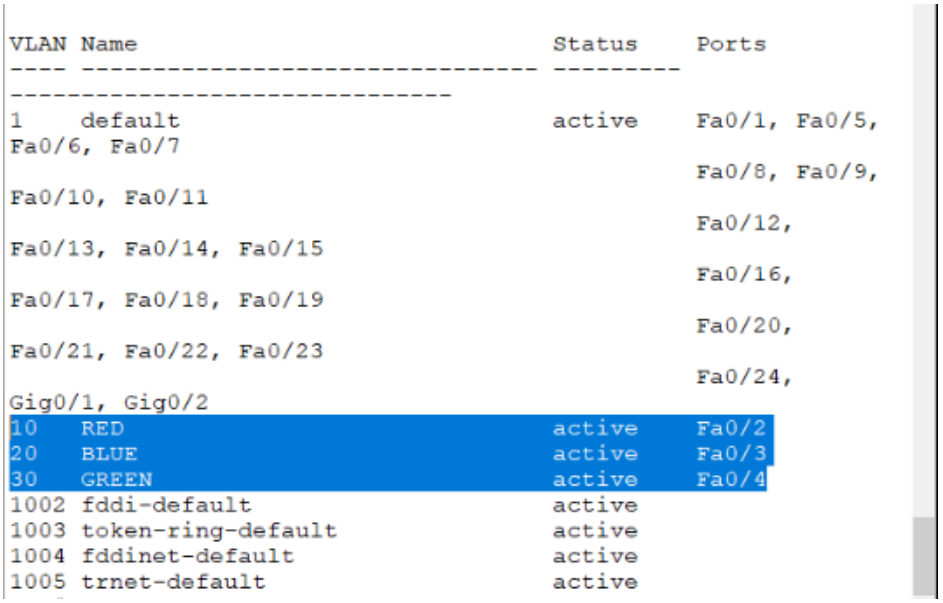
```
srm2#
%SYS-5-CONFIG_I: Configured from console by console
show vlan brief
```

```
VLAN Name                                Status    Ports
---- ----------------------------------- ---------
------------------------------
1    default                             active    Fa0/1, Fa0/2,
Fa0/3, Fa0/4
                                                   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   RED                                 active
20   BLUE                                active
30   GREEN                               active
1002 fddi-default                        active
1003 token-ring-default                  active
1004 fddinet-default                     active
1005 trnet-default                       active
```

2. Configure Access Mode on both the Switches

Switch 0

```
srm1(config)#interface fastethernet 0/2
srm1(config-if)#switchport mode access
srm1(config-if)#switchport access vlan 10
srm1(config-if)#interface fastethernet 0/3
srm1(config-if)#switchport mode access
srm1(config-if)#switchport access vlan 20
srm1(config-if)#interface fastethernet 0/4
srm1(config-if)#switchport mode access
srm1(config-if)#switchport access vlan 30
srm1(config-if)#exit
srm1(config)#exit
srm1#show vlan brief
```



```
VLAN Name                              Status    Ports
---- --------------------------------- --------- -------------------------------
1    default                           active    Fa0/1, 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   RED                               active    Fa0/2
20   BLUE                              active    Fa0/3
30   GREEN                             active    Fa0/4
1002 fddi-default                      active
1003 token-ring-default                active
1004 fddinet-default                   active
1005 trnet-default                     active
```

3. Configure the Trunk Mode – Configure the mode trunk to all interface of the switches that connects to another switches

Switch 0

```
srm1(config)#interface fastethernet 0/1
srm1(config-if)#switchport mode trunk
srm1(config-if)#switchport nonegotiate
srm1(config-if)#exit
%LINEPROTO-5-UPDOWN: Line protocol on Interface
FastEthernet0/1, changed state to down
%LINEPROTO-5-UPDOWN: Line protocol on Interface
FastEthernet0/1, changed state to up
srm1(config)#exit
```

Switch 1

```
srm1(config)#interface fastethernet 0/1
srm1(config-if)#switchport mode trunk
srm1(config-if)#switchport nonegotiate
srm1(config-if)#exit
%LINEPROTO-5-UPDOWN: Line protocol on Interface
FastEthernet0/1, changed state to down
%LINEPROTO-5-UPDOWN: Line protocol on Interface
FastEthernet0/1, changed state to up
srm1(config)#exit
```

4. Configure Trunk on Native VLAN 1 on both switch

Switch 0

```
srm1(config)#interface fastethernet 0/24
srm1(config-if)#switchport mode trunk
srm1(config-if)#switchport trunk native vlan 1
srm1(config-if)#exit
srm1(config)#exit
srm1#
```

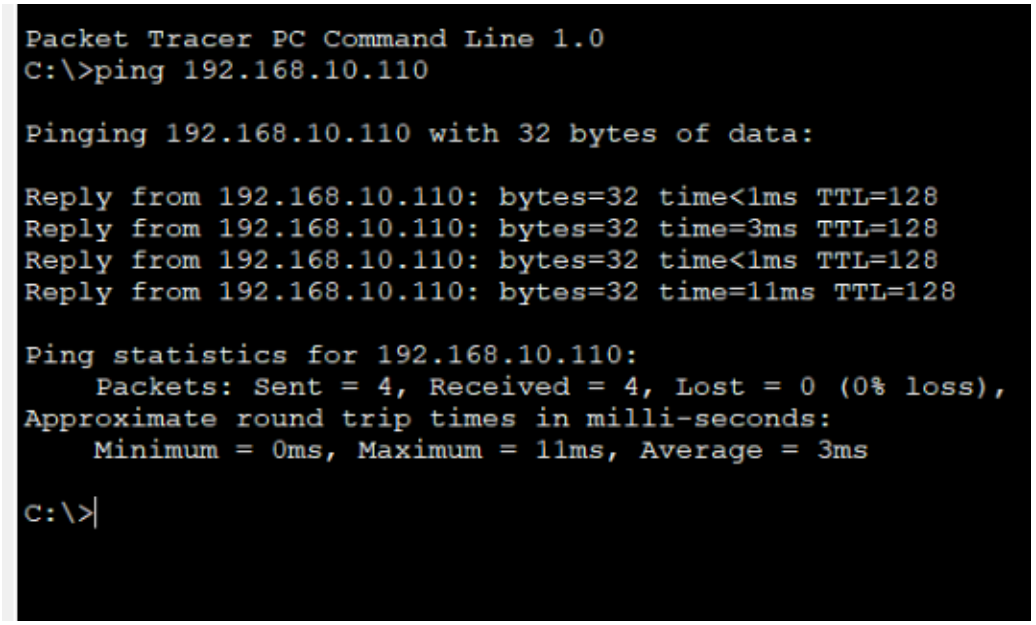
Switch 2

```
srm1(config)#interface fastethernet 0/24
srm1(config-if)#switchport mode trunk
srm1(config-if)#switchport trunk native vlan 1
srm1(config-if)#exit
srm1(config)#
```

5. Configure the IP address and subnet mask on the PCs as follows. There is no layer three devices on the network, so the default gateway will not be configured.

```
VLAN 10: 192.168.10.0/24
VLAN 20 : 192.168.20.0/24
VLAN 30: 192.168.30.0/24
PC2: 192.168.10.110 255.255.255.0
PC0 : 192.168.20.220 255.255.255.0
PC1: 192.168.30.330 255.255.255.0
PC3: 192.168.10.111 255.255.255.0
PC4: 192.168.20.221 255.255.255.0
PC5: 192.168.30.331 255.255.255.0
```

6. Verify the Connections



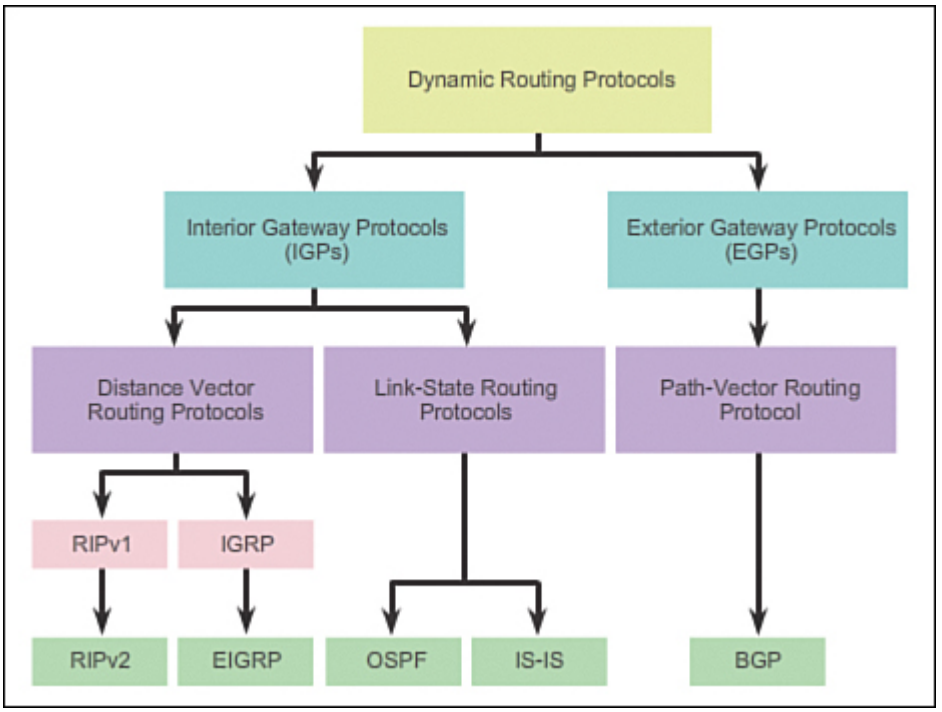
LAB 7: Implementation of RIP, OSPF and BGP

Objective(s):

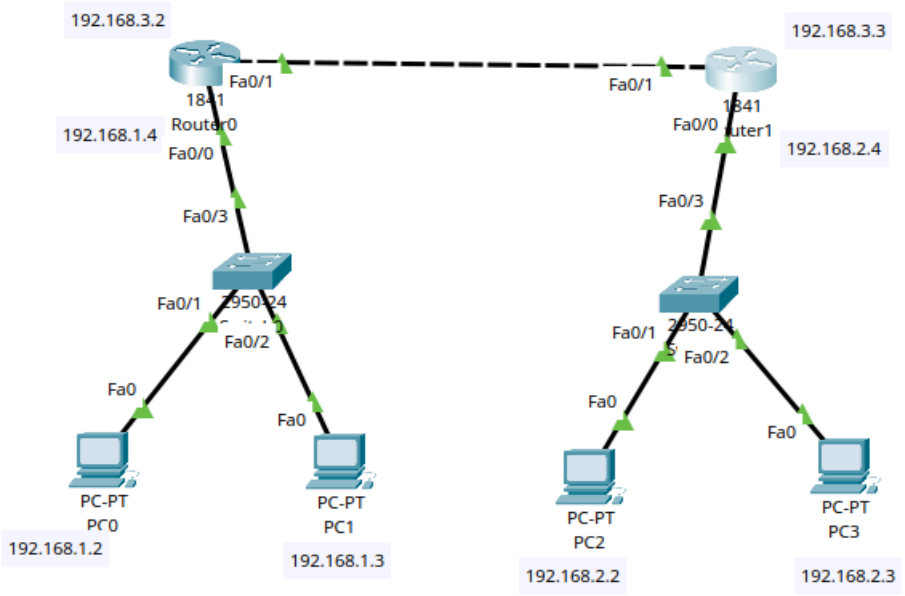
To Understand the Basic Operation(s) of RIP, OSPF and BGP

Background:

Routing Protocols are the set of defined rules used by the routers to communicate between source & destination. They do not move the information to the source to a destination, but only update the routing table that contains the information.

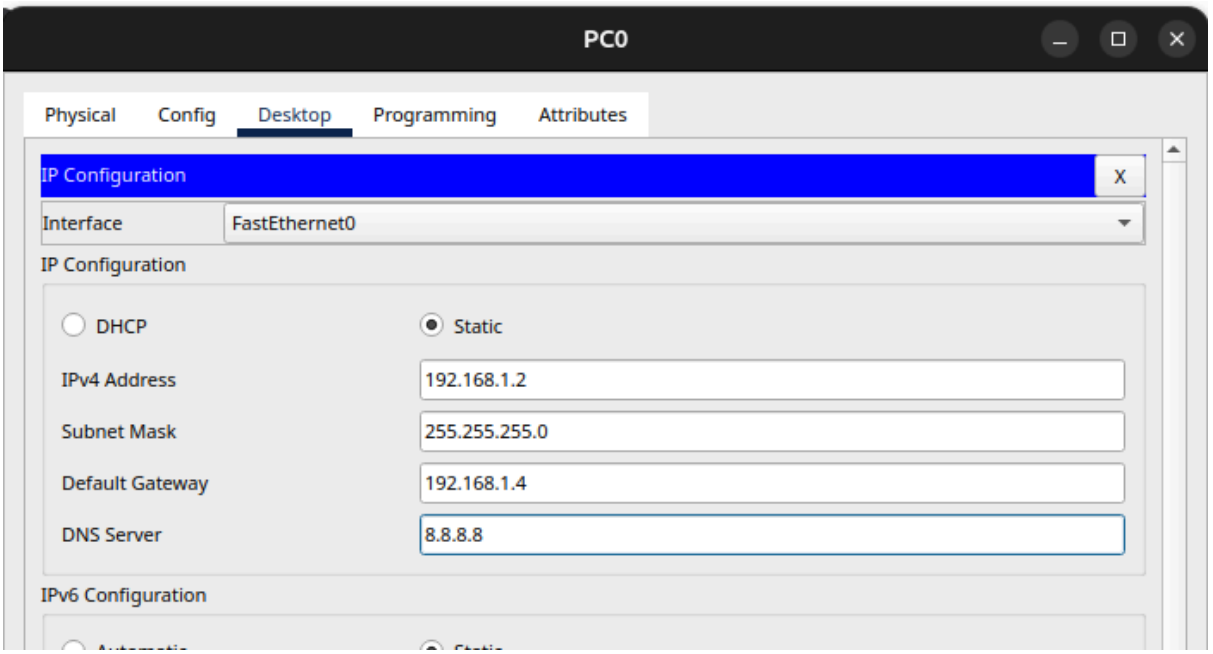


RIP Configuration:



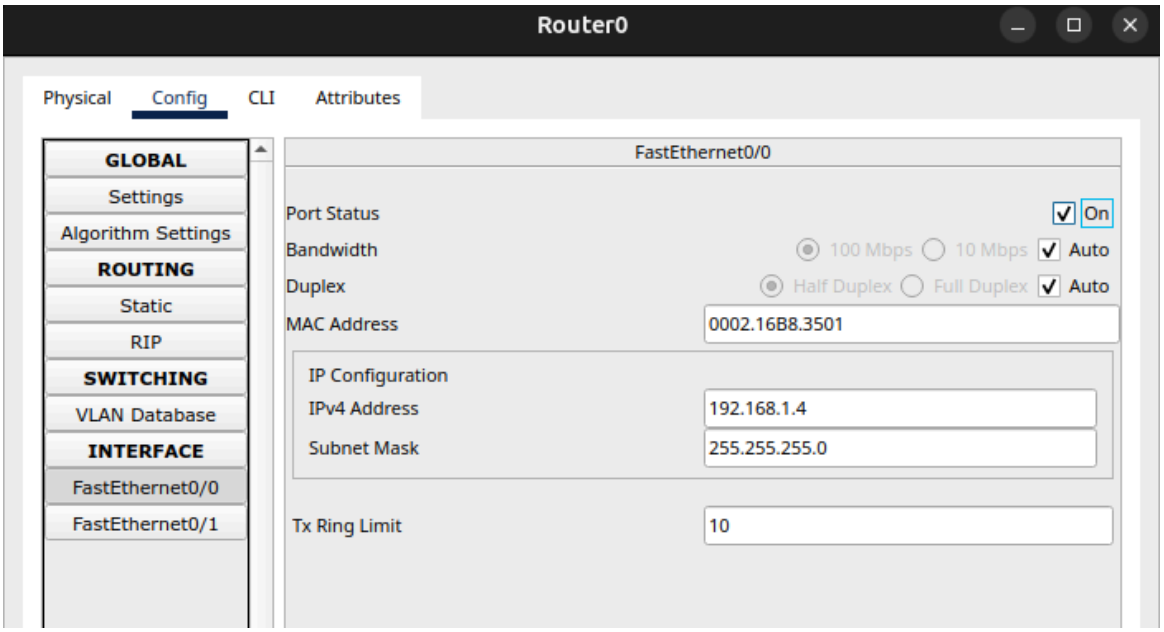
1. PC Configuration

- Go to Desktop > IP configuration and input the ip address subnet mask etc.
- Repeat this process for all remaining pc's



2. Router Configuration

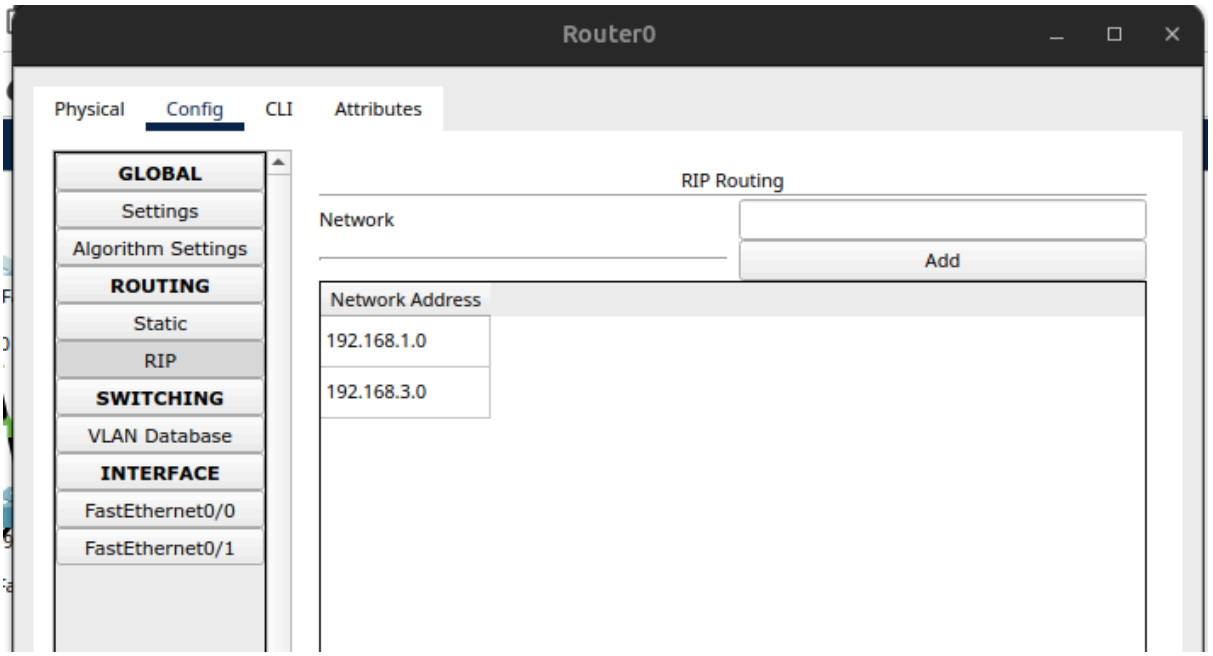
- Open router and click configuration
- Click fastethernet 0/0 and fill enter the IP address and click the on button in the top right corner



Now click fastethernet 0/1 and repeat the same process but with different ip(192.168.3.2)
After that do the same to the other router(Router 1) too

3. RIP config

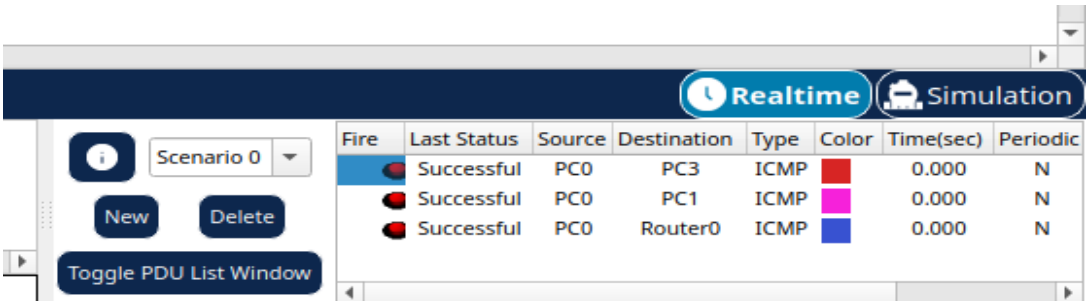
- Goto config > RIP and add the corresponding IP address 192.168.1.0 and 192.168.3.0 and click the add button
- Repeat this same process for Router1



Testing

The configuration can be tested in two ways.

A. By sending packets from one PC to another



Configuration of OSPF

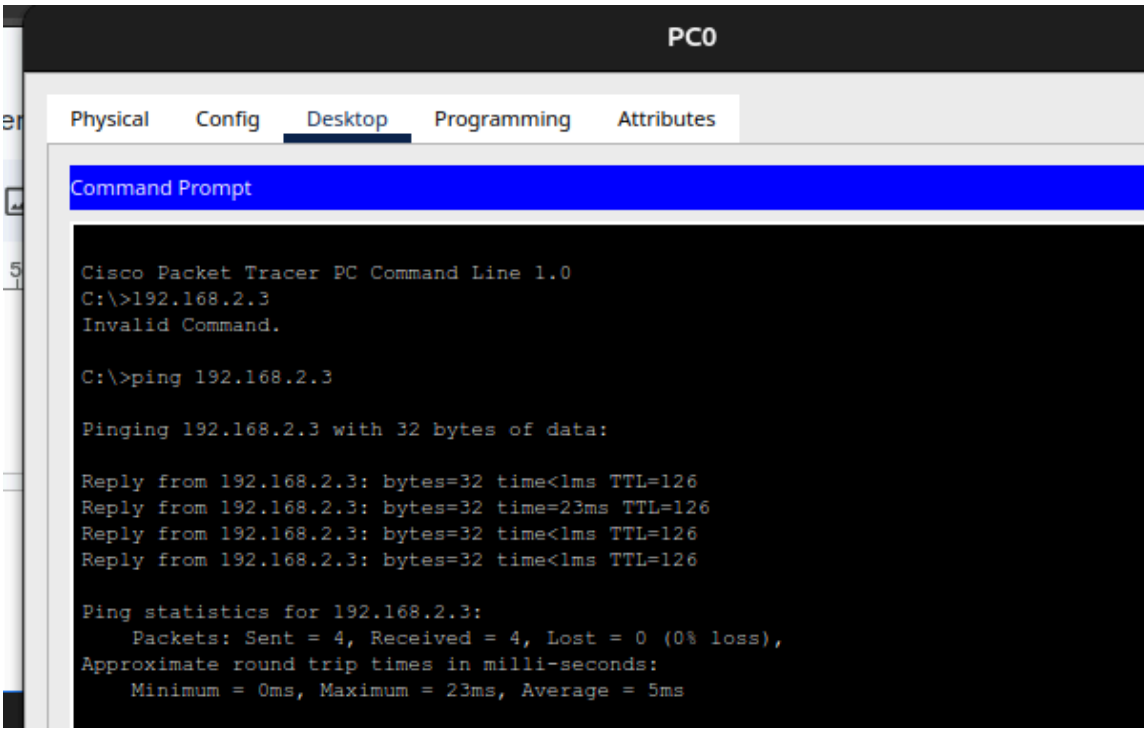
1. Router0

```
Router>enable
Router#configure terminal
Router(config)#router ospf 1
Router(config-router)#network 192.168.1.0 0.0.0.255 area 0
Router(config-router)#network 192.168.3.0 0.0.0.255 area 0
Router(config-router)#exit
Router(config)#
```

2.Router1

```
Router>enable
Router#configure terminal
Router(config)#router ospf 1
Router(config-router)#network 192.168.2.0 0.0.0.255 area 0
Router(config-router)#network 192.168.3.0 0.0.0.255 area 0
Router(config-router)#exit
```

Ping Test:



BGP Configuration:

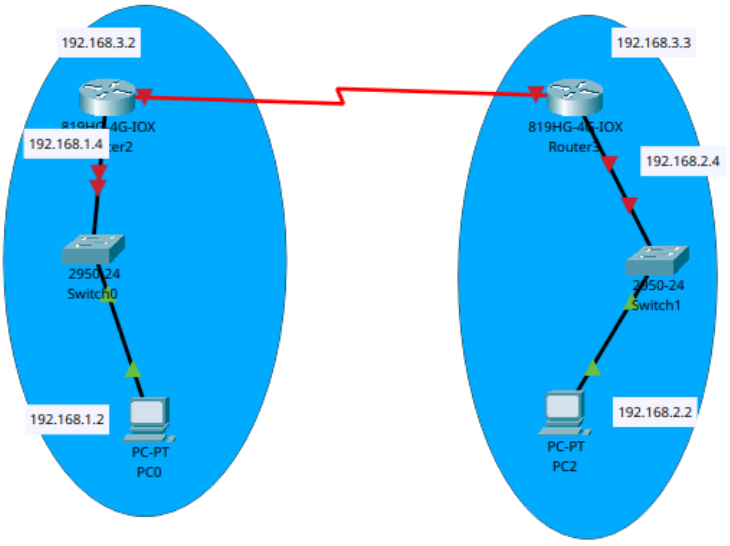
1.Router 0

```
Router>enable
Router#configure terminal
Router(config)#router bgp 65001
Router(config-router)#network 192.168.1.0 mask 255.255.255.0
Router(config-router)#neighbor 192.168.3.2 remote-as 65002
Router(config-router)#exit
Router(config)#exit
Router#
```

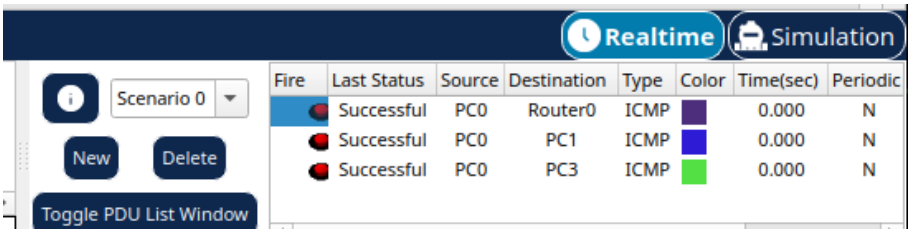
2.Router 1

```
Router>enable
Router#configure terminal
Router(config)#router bgp 65002
Router(config-router)#network 192.168.2.0 mask 255.255.255.0
Router(config-router)#neighbor 192.168.3.1 remote-as 65001
Router(config-router)#exit
Router(config)#exit
Router#
```

Layout:



Test:



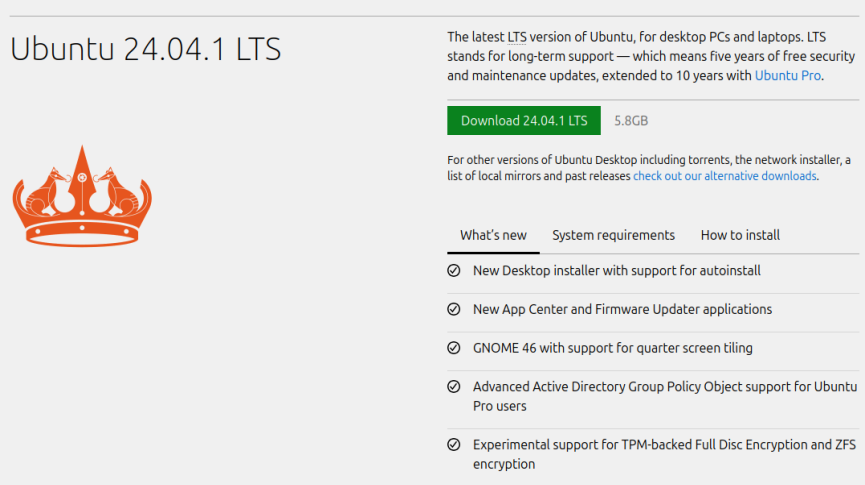
LAB 8: VMWare Workstation (Creating Virtual Machine and Installation process of Linux)

Objective(s):
To create a virtual machine in VMWare workstation and install Linux(Ubuntu)

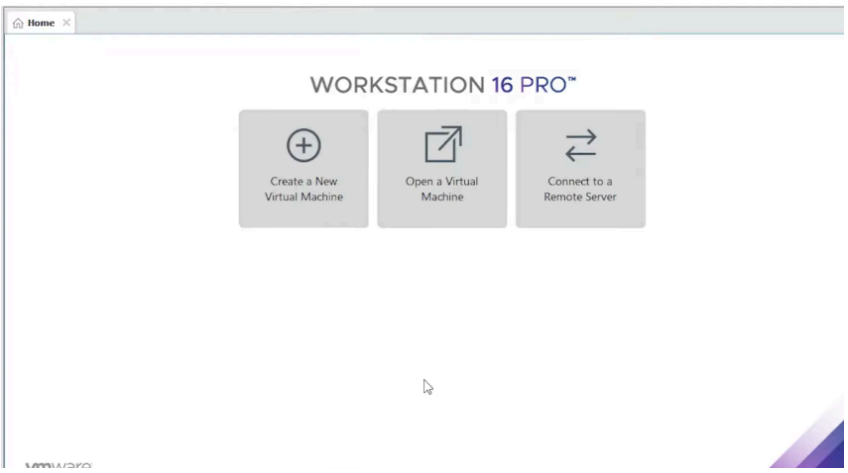
Step 1: Download VMware Workstation
Visit the official VMware website and download the latest version of VMware Workstation. You may need to create a VMware account and log in to access the download.

Step 2: Install VMware Workstation
Run the installer you downloaded in Step 1. Follow the on-screen instructions to install VMware Workstation on your computer. You may need to restart your system after the installation is complete.

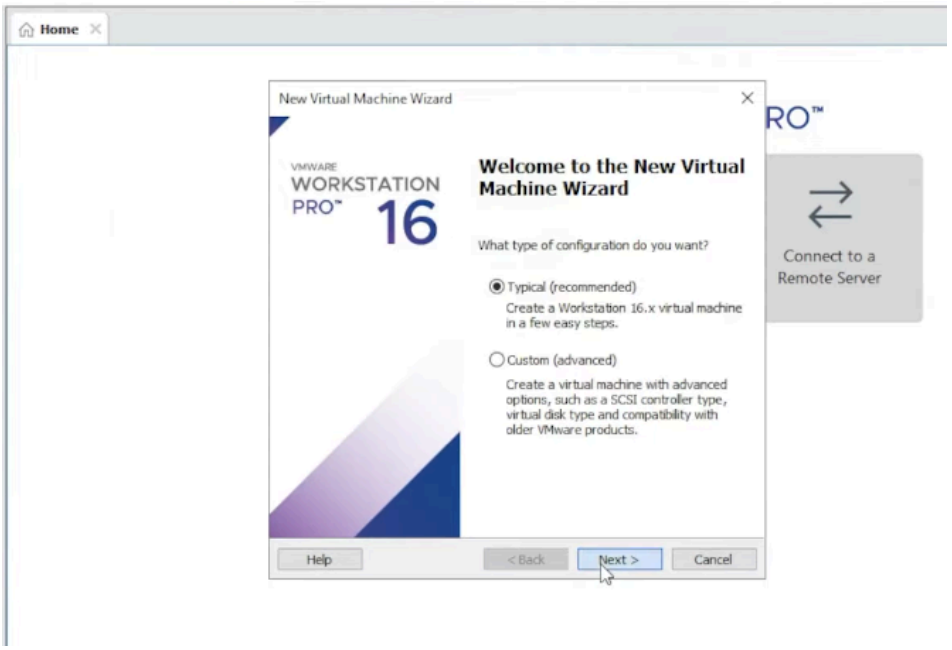
Step 3: Download Ubuntu ISO
Visit the official Ubuntu website **https://ubuntu.com/download/desktop** and download the latest version of the Ubuntu Desktop ISO. Choose the appropriate architecture **(32-bit or 64-bit)** based on your system.



Step 4: Create a New Virtual Machine in VMware Workstation

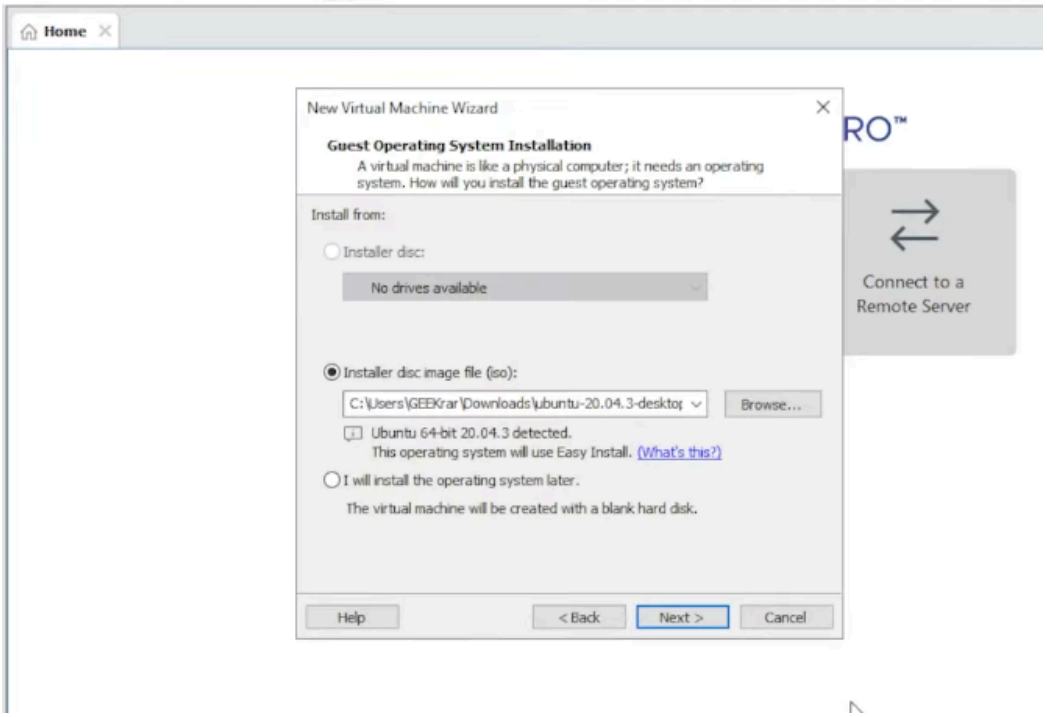


Open VMware Workstation:
Click on “File” in the menu and select “New Virtual Machine.”
The New Virtual Machine Wizard will open. Choose “Typical” and click “Next.”



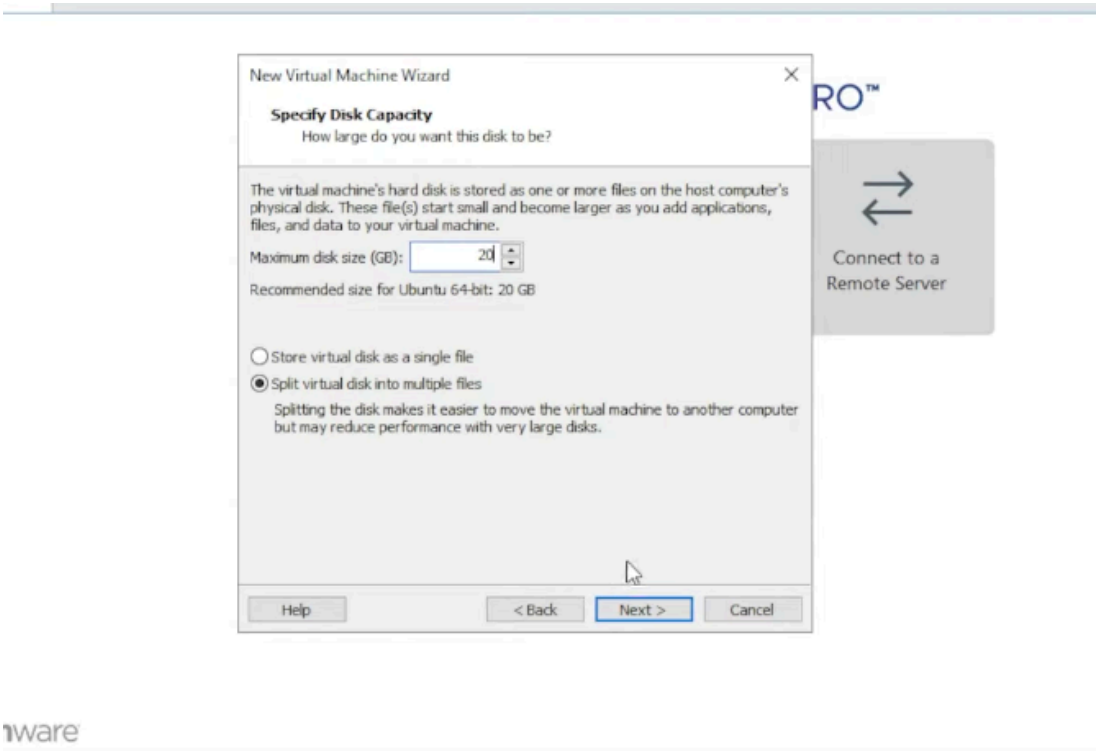
Step 5: Specify the Ubuntu ISO

Select “Installer disc image file (iso)” and click “Browse.”
Navigate to the location where you saved the Ubuntu ISO file and select it.
Click “Next” to proceed.



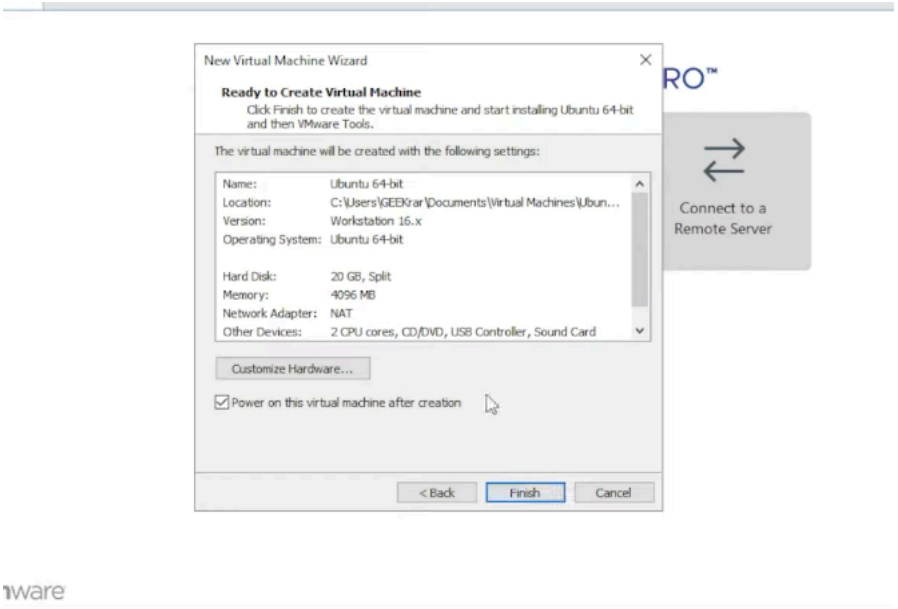
Step 6: Name and Specify Storage

Enter a name for your virtual machine.
Choose a location to store your virtual machine files.
Specify the disk capacity.
Select “Store virtual disk as a single file.”
Click ***“Next”*** to continue.



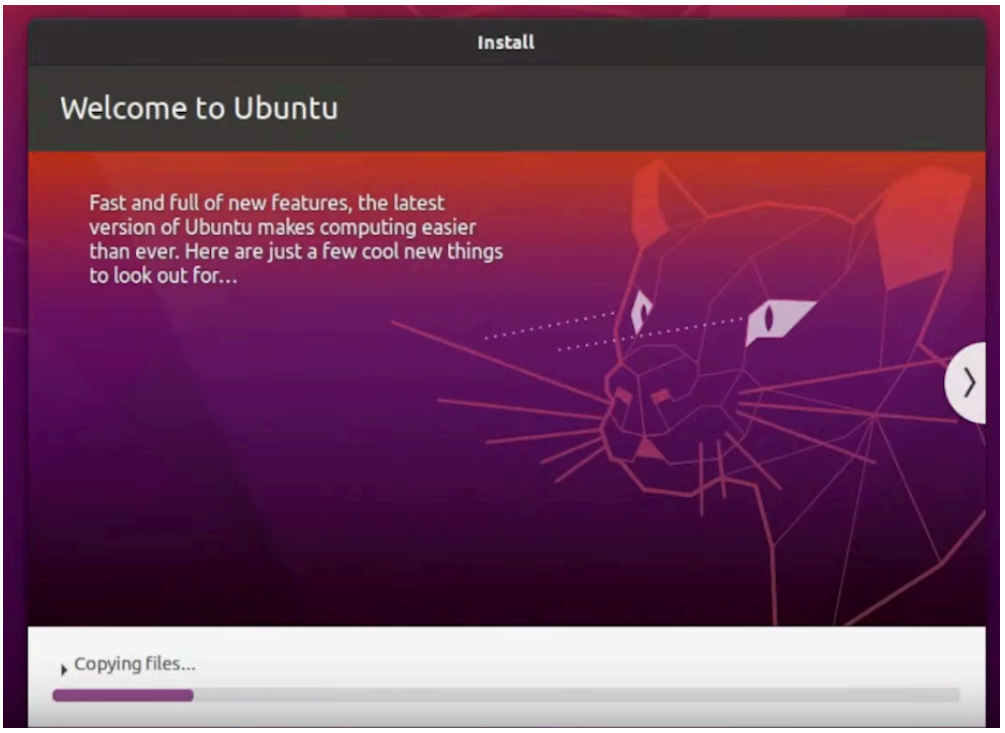
Step 7: Finish and Install

Review your settings and click “Finish.”



Step 8: Complete the installation

Ubuntu will now install the necessary files. This process may take some time depending on your machine and installation options.



Step 9: Start the Virtual Machine

The virtual machine will boot from the Ubuntu ISO.
Follow the on-screen instructions to install Ubuntu and login using your previously set password

Step 10: Complete

That’s it! You should now have Ubuntu installed and running in VMware Workstation on your system.