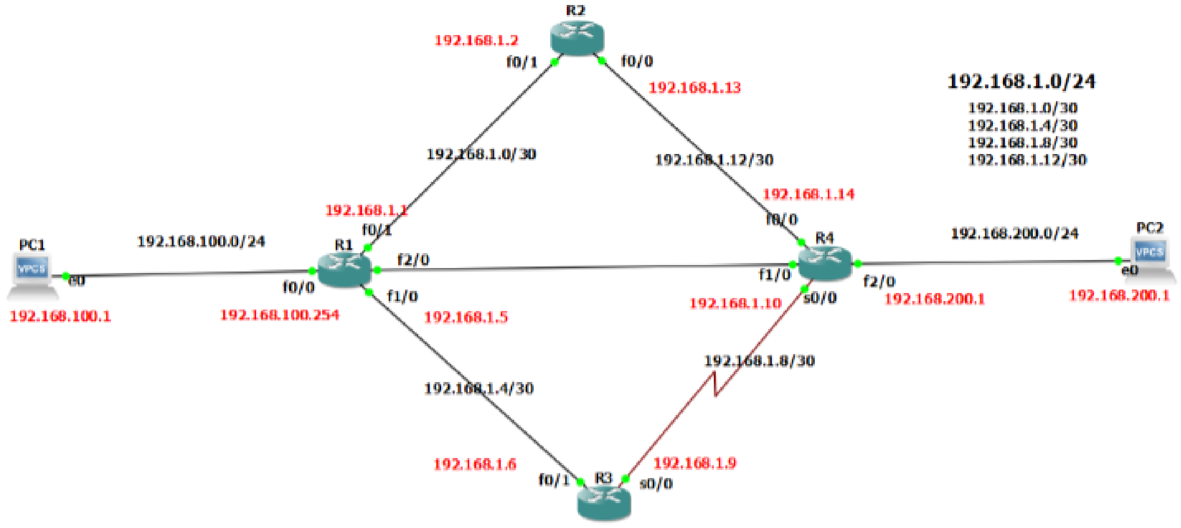
Computer Networks

Assignment 4

Melat Haile

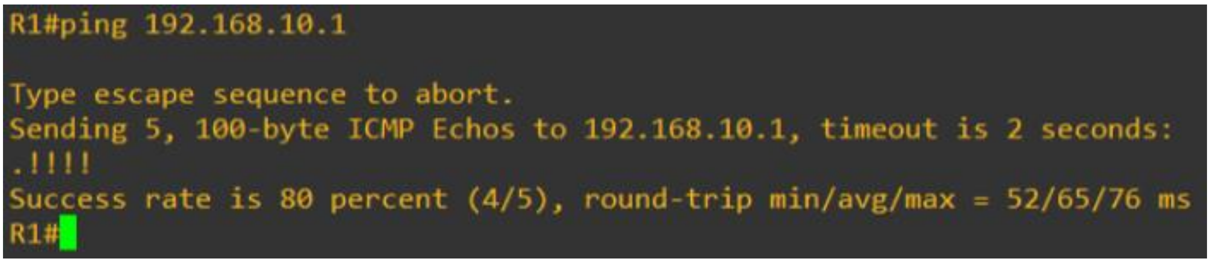
1

a)



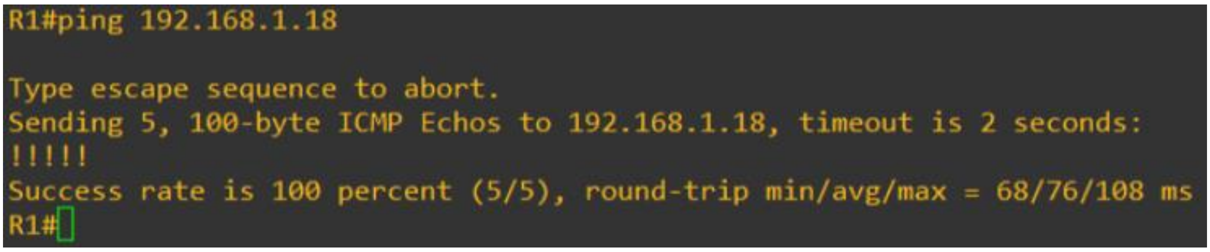
b)

R1 - PC1



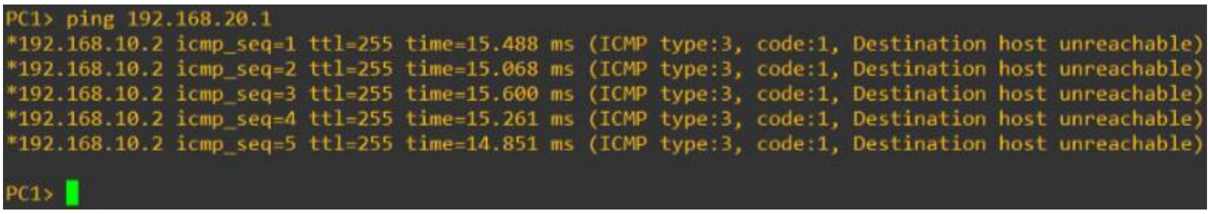
```
R1#ping 192.168.10.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.10.1, timeout is 2 seconds:
.!!!!
Success rate is 80 percent (4/5), round-trip min/avg/max = 52/65/76 ms
R1#
```

R1 – R4



```
R1#ping 192.168.1.18

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.1.18, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 68/76/108 ms
R1#
```

PC1 – PC2



```
PC1> ping 192.168.20.1
*192.168.10.2 icmp_seq=1 ttl=255 time=15.488 ms (ICMP type:3, code:1, Destination host unreachable)
*192.168.10.2 icmp_seq=2 ttl=255 time=15.068 ms (ICMP type:3, code:1, Destination host unreachable)
*192.168.10.2 icmp_seq=3 ttl=255 time=15.600 ms (ICMP type:3, code:1, Destination host unreachable)
*192.168.10.2 icmp_seq=4 ttl=255 time=15.261 ms (ICMP type:3, code:1, Destination host unreachable)
*192.168.10.2 icmp_seq=5 ttl=255 time=14.851 ms (ICMP type:3, code:1, Destination host unreachable)

PC1>
```

c)

The WIC-1T module is an interface module that enables serial communication in connected devices such as switches and routers. It is also sometimes referred to as a WAN Interface Card (WIC) and is commonly used for connecting devices to a Wide Area Network (WAN), such as a private network. The "1T" in the module's name stands for "1-port serial WAN interface card." It utilizes a standard RS-232 serial port to communicate with other nodes at speeds of up to 1.544 Mbps using serial networks. This feature makes it suitable for establishing a fixed serial link between two nodes or connecting to outdated networking hardware.

The WIC-1T module is often combined with other interaction modules to provide a range of networking options. It is typically inserted into a connector on a router or switch.

On the other hand, network equipment such as routers and switch devices require the NM-1FE-TX type of network Ethernet interface to establish a Fast Ethernet connection. This interface, also known as a Network Module (NM), is frequently used to expand the total number of Ethernet adapters available on a device.

The "TX" in the module's name signifies that the port uses a twisted pair cable for communication, and "1FE" stands for "1 Fast Ethernet port." The module is compatible with both Category 5 and Category 5e Ethernet cables and supports data transfer rates of up to 100 Mbps.

The NM-1FE-TX is commonly integrated with other interface modules to offer various connectivity options. It is typically inserted into a connector on a router or switch and serves as a cost-effective solution for adding more LAN ports to a computer in medium- to small-sized connections.

d)

A /24 subnet provides 254 accessible addresses (28^2-2). On the other hand, a /30 subnet allows for only 2 reachable domains (22-2). This subnet range is commonly used for establishing point-to-point links between two systems or networks. As a result, the number of usable IPs and the overall size of the network serve as the primary operational differences between a /24 and a /30 subnet. A /30 subnet is significantly smaller and often used for hop-to-hop connectivity, whereas a /24 subnet is considerably larger and can support multiple domains.

2

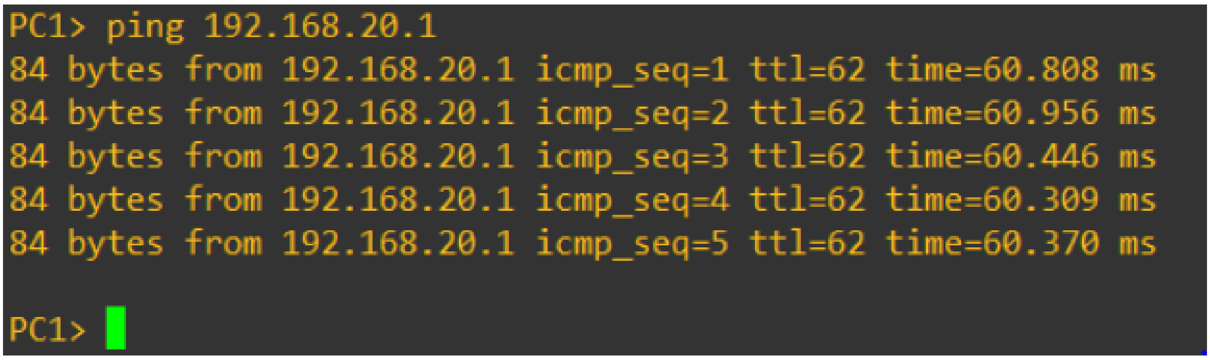
a)

In a /24 subnet, there is a total of 254 available addresses (28^2-2). Conversely, a /30 subnet allows for just 2 reachable domains (22-2). This specific subnet range is frequently employed to establish direct links between two systems or networks. Consequently, the quantity of usable IPs and the overall network size are the primary operational distinctions between a /24 and a /30 subnet. A /30 subnet is notably smaller and commonly utilized for hop-to-hop connectivity, whereas a /24 subnet is significantly larger and has the capacity to support multiple domains.

b)

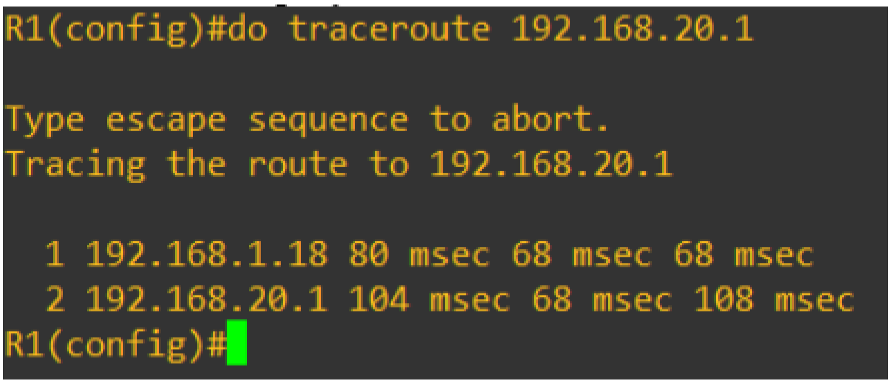
By configuring a static route, PC1 is capable of establishing reachability to PC2. One of the factors influencing my decision to choose this path is the number of routers involved in the process. Since this route requires fewer routers to reach the destination, it was selected.

To verify the connection, I will initiate ping requests from PC-1 to PC-2.



```
PC1> ping 192.168.20.1
84 bytes from 192.168.20.1 icmp_seq=1 ttl=62 time=60.808 ms
84 bytes from 192.168.20.1 icmp_seq=2 ttl=62 time=60.956 ms
84 bytes from 192.168.20.1 icmp_seq=3 ttl=62 time=60.446 ms
84 bytes from 192.168.20.1 icmp_seq=4 ttl=62 time=60.309 ms
84 bytes from 192.168.20.1 icmp_seq=5 ttl=62 time=60.370 ms

PC1>
```

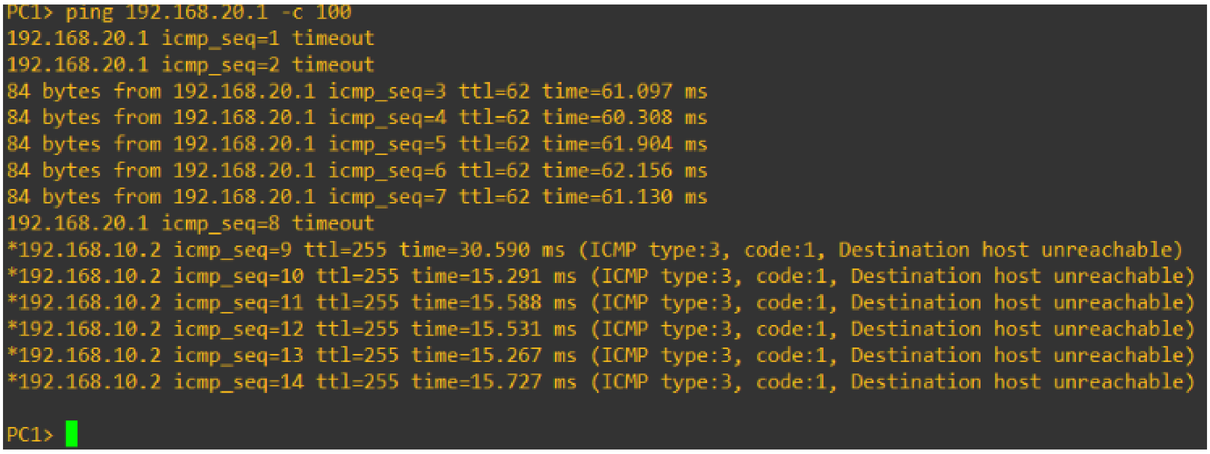
Traceroute to display the route



```
R1(config)#do traceroute 192.168.20.1

Type escape sequence to abort.
Tracing the route to 192.168.20.1

  1 192.168.1.18 80 msec 68 msec 68 msec
  2 192.168.20.1 104 msec 68 msec 108 msec
R1(config)#
```

c)

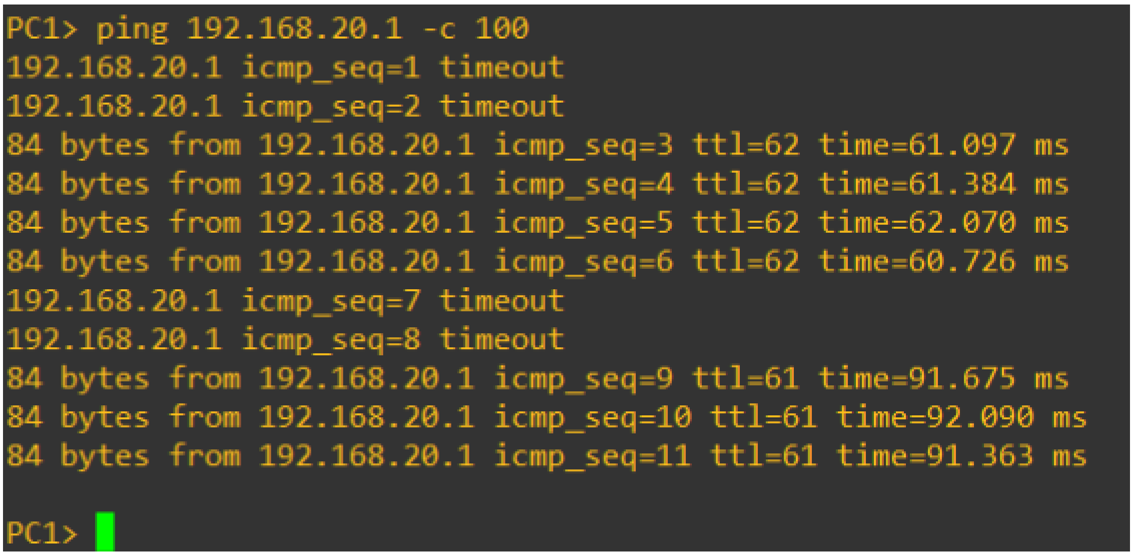
Based on the screenshot, it is evident that when both running routers were powered off, the ping test failed, and the transmission could not reach its intended destination.



```
PC1> ping 192.168.20.1 -c 100
192.168.20.1 icmp_seq=1 timeout
192.168.20.1 icmp_seq=2 timeout
84 bytes from 192.168.20.1 icmp_seq=3 ttl=62 time=61.097 ms
84 bytes from 192.168.20.1 icmp_seq=4 ttl=62 time=60.308 ms
84 bytes from 192.168.20.1 icmp_seq=5 ttl=62 time=61.904 ms
84 bytes from 192.168.20.1 icmp_seq=6 ttl=62 time=62.156 ms
84 bytes from 192.168.20.1 icmp_seq=7 ttl=62 time=61.130 ms
192.168.20.1 icmp_seq=8 timeout
*192.168.10.2 icmp_seq=9 ttl=255 time=30.590 ms (ICMP type:3, code:1, Destination host unreachable)
*192.168.10.2 icmp_seq=10 ttl=255 time=15.291 ms (ICMP type:3, code:1, Destination host unreachable)
*192.168.10.2 icmp_seq=11 ttl=255 time=15.588 ms (ICMP type:3, code:1, Destination host unreachable)
*192.168.10.2 icmp_seq=12 ttl=255 time=15.531 ms (ICMP type:3, code:1, Destination host unreachable)
*192.168.10.2 icmp_seq=13 ttl=255 time=15.267 ms (ICMP type:3, code:1, Destination host unreachable)
*192.168.10.2 icmp_seq=14 ttl=255 time=15.727 ms (ICMP type:3, code:1, Destination host unreachable)

PC1>
```

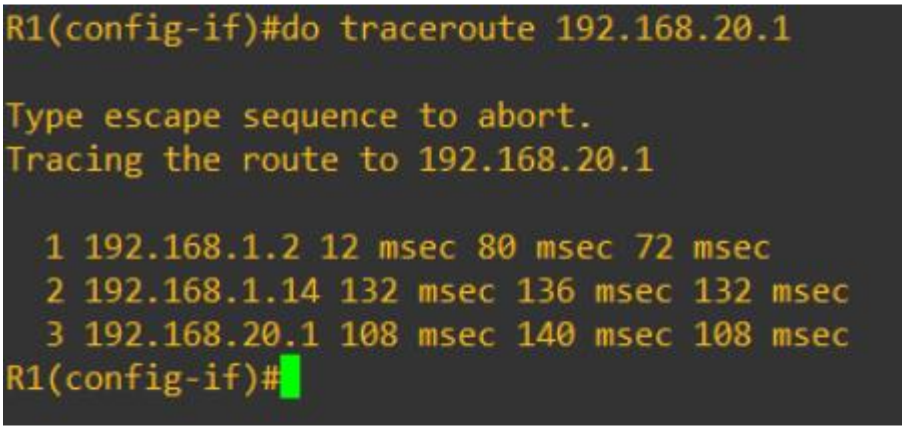
d)

In order to reach the destination, two extra static routes were established. In the event of a network connection failure, R1 utilizes a metric value of 2 to choose a backup route. The metric serves as a determining factor for selecting the most appropriate path to an endpoint. Moreover, once two data packets are lost, both PC1 and PC2 regain reachability.



```
PC1> ping 192.168.20.1 -c 100
192.168.20.1 icmp_seq=1 timeout
192.168.20.1 icmp_seq=2 timeout
84 bytes from 192.168.20.1 icmp_seq=3 ttl=62 time=61.097 ms
84 bytes from 192.168.20.1 icmp_seq=4 ttl=62 time=61.384 ms
84 bytes from 192.168.20.1 icmp_seq=5 ttl=62 time=62.070 ms
84 bytes from 192.168.20.1 icmp_seq=6 ttl=62 time=60.726 ms
192.168.20.1 icmp_seq=7 timeout
192.168.20.1 icmp_seq=8 timeout
84 bytes from 192.168.20.1 icmp_seq=9 ttl=61 time=91.675 ms
84 bytes from 192.168.20.1 icmp_seq=10 ttl=61 time=92.090 ms
84 bytes from 192.168.20.1 icmp_seq=11 ttl=61 time=91.363 ms

PC1>
```

After the router successfully utilized the primary path, the traceroute tool generated an updated output, which is visible in the screenshot below: Destination > Source.

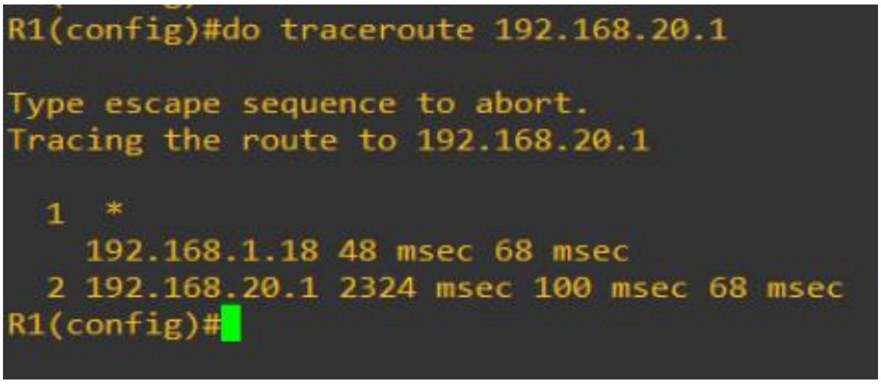


```
R1(config-if)#do traceroute 192.168.20.1

Type escape sequence to abort.
Tracing the route to 192.168.20.1

  1 192.168.1.2 12 msec 80 msec 72 msec
  2 192.168.1.14 132 msec 136 msec 132 msec
  3 192.168.20.1 108 msec 140 msec 108 msec
R1(config-if)#
```

3

a)

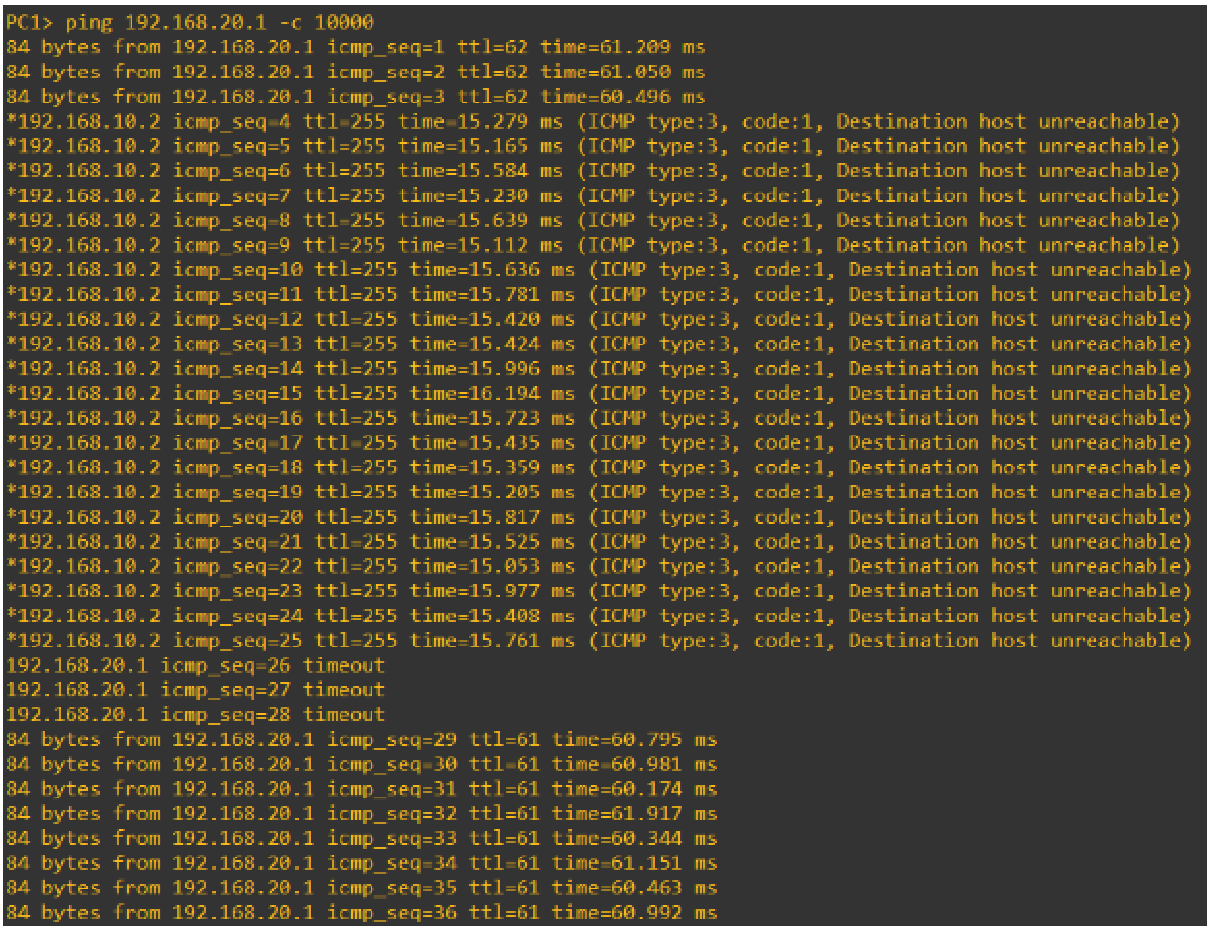
Rip, a distance vector dynamic routing protocol, utilizes the number of hops as a statistic to determine the preferred path among multiple options. By following Rip's method, the path is identified as: Target Network > R1 > R3 > Source Network. The output generated by the traceroute tool is displayed below.



```
R1(config)#do traceroute 192.168.20.1

Type escape sequence to abort.
Tracing the route to 192.168.20.1

  1  *
    192.168.1.18 48 msec 68 msec
  2 192.168.20.1 2324 msec 100 msec 68 msec
R1(config)#
```

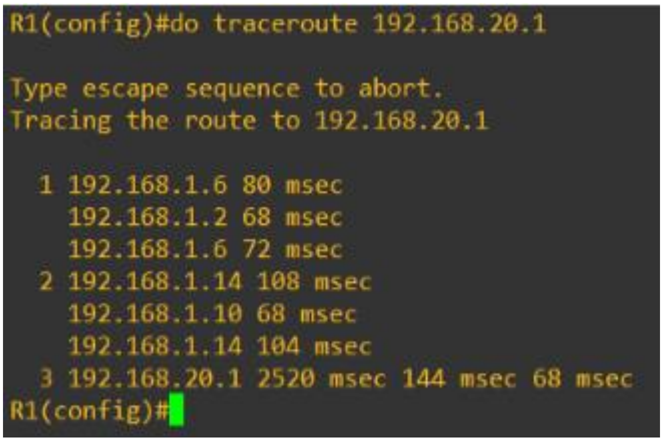
b)

The image below provides a visual representation of the packet loss that occurred prior to the establishment of a functional alternative network path. This situation arises when one of the operational routers is turned off. Despite having some enhancements such as triggered updates, specific notifications, and split horizons, the primary challenges with RIP primarily stem from its inadequate routing capabilities. RIP is considered a dated dynamic routing protocol with a relatively slower resolution quality. Additionally, RIP does not foster close neighbor relationships.



```
PC1> ping 192.168.20.1 -c 10000
84 bytes from 192.168.20.1 icmp_seq=1 ttl=62 time=61.209 ms
84 bytes from 192.168.20.1 icmp_seq=2 ttl=62 time=61.050 ms
84 bytes from 192.168.20.1 icmp_seq=3 ttl=62 time=60.496 ms
*192.168.10.2 icmp_seq=4 ttl=255 time=15.279 ms (ICMP type:3, code:1, Destination host unreachable)
*192.168.10.2 icmp_seq=5 ttl=255 time=15.165 ms (ICMP type:3, code:1, Destination host unreachable)
*192.168.10.2 icmp_seq=6 ttl=255 time=15.584 ms (ICMP type:3, code:1, Destination host unreachable)
*192.168.10.2 icmp_seq=7 ttl=255 time=15.230 ms (ICMP type:3, code:1, Destination host unreachable)
*192.168.10.2 icmp_seq=8 ttl=255 time=15.639 ms (ICMP type:3, code:1, Destination host unreachable)
*192.168.10.2 icmp_seq=9 ttl=255 time=15.112 ms (ICMP type:3, code:1, Destination host unreachable)
*192.168.10.2 icmp_seq=10 ttl=255 time=15.636 ms (ICMP type:3, code:1, Destination host unreachable)
*192.168.10.2 icmp_seq=11 ttl=255 time=15.781 ms (ICMP type:3, code:1, Destination host unreachable)
*192.168.10.2 icmp_seq=12 ttl=255 time=15.420 ms (ICMP type:3, code:1, Destination host unreachable)
*192.168.10.2 icmp_seq=13 ttl=255 time=15.424 ms (ICMP type:3, code:1, Destination host unreachable)
*192.168.10.2 icmp_seq=14 ttl=255 time=15.996 ms (ICMP type:3, code:1, Destination host unreachable)
*192.168.10.2 icmp_seq=15 ttl=255 time=16.194 ms (ICMP type:3, code:1, Destination host unreachable)
*192.168.10.2 icmp_seq=16 ttl=255 time=15.723 ms (ICMP type:3, code:1, Destination host unreachable)
*192.168.10.2 icmp_seq=17 ttl=255 time=15.435 ms (ICMP type:3, code:1, Destination host unreachable)
*192.168.10.2 icmp_seq=18 ttl=255 time=15.359 ms (ICMP type:3, code:1, Destination host unreachable)
*192.168.10.2 icmp_seq=19 ttl=255 time=15.205 ms (ICMP type:3, code:1, Destination host unreachable)
*192.168.10.2 icmp_seq=20 ttl=255 time=15.817 ms (ICMP type:3, code:1, Destination host unreachable)
*192.168.10.2 icmp_seq=21 ttl=255 time=15.525 ms (ICMP type:3, code:1, Destination host unreachable)
*192.168.10.2 icmp_seq=22 ttl=255 time=15.053 ms (ICMP type:3, code:1, Destination host unreachable)
*192.168.10.2 icmp_seq=23 ttl=255 time=15.977 ms (ICMP type:3, code:1, Destination host unreachable)
*192.168.10.2 icmp_seq=24 ttl=255 time=15.408 ms (ICMP type:3, code:1, Destination host unreachable)
*192.168.10.2 icmp_seq=25 ttl=255 time=15.761 ms (ICMP type:3, code:1, Destination host unreachable)
192.168.20.1 icmp_seq=26 timeout
192.168.20.1 icmp_seq=27 timeout
192.168.20.1 icmp_seq=28 timeout
84 bytes from 192.168.20.1 icmp_seq=29 ttl=61 time=60.795 ms
84 bytes from 192.168.20.1 icmp_seq=30 ttl=61 time=60.981 ms
84 bytes from 192.168.20.1 icmp_seq=31 ttl=61 time=60.174 ms
84 bytes from 192.168.20.1 icmp_seq=32 ttl=61 time=61.917 ms
84 bytes from 192.168.20.1 icmp_seq=33 ttl=61 time=60.344 ms
84 bytes from 192.168.20.1 icmp_seq=34 ttl=61 time=61.151 ms
84 bytes from 192.168.20.1 icmp_seq=35 ttl=61 time=60.463 ms
84 bytes from 192.168.20.1 icmp_seq=36 ttl=61 time=60.992 ms
```

As both possible routes have an equal number of hops, the RIP protocol evenly distributes the data load between them for load balancing.



```
R1(config)#do traceroute 192.168.20.1

Type escape sequence to abort.
Tracing the route to 192.168.20.1

  1 192.168.1.6 80 msec
    192.168.1.2 68 msec
    192.168.1.6 72 msec
  2 192.168.1.14 108 msec
    192.168.1.10 68 msec
    192.168.1.14 104 msec
  3 192.168.20.1 2520 msec 144 msec 68 msec
R1(config)#
```

4

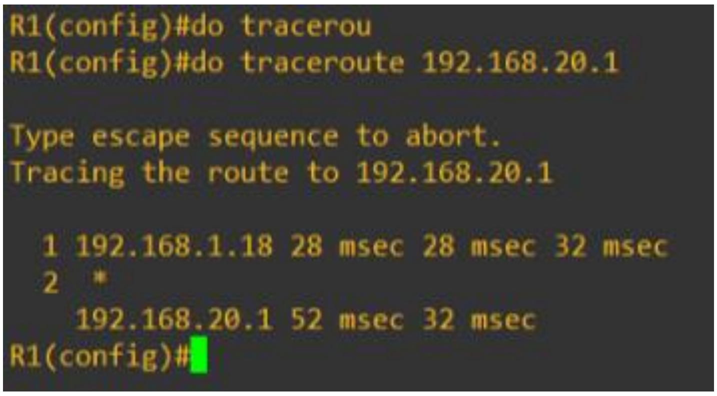
a)

While it is possible to fit each interface into a single space, the intention is to illustrate a hierarchical architecture within OSPF. The goal is to minimize the network's size by implementing a hierarchical system of areas. This approach enhances network scalability and optimizes network parameters. Routers in separate groups have limited knowledge of each other's network topologies, whereas routers within the same network possess comprehensive awareness of the overall topology. Consequently, there is reduced traffic and computational load required for a router to determine the optimal routing path for a packet.

OSPF selects the most suitable route for data transmission across routers by considering a parameter called "Cost." This cost is determined based on the bandwidth of the connection between two routers, where a lower cost signifies higher bandwidth. Therefore, OSPF prioritizes pathways with greater bandwidth and favors connections with higher capacity over those with lower capacity.

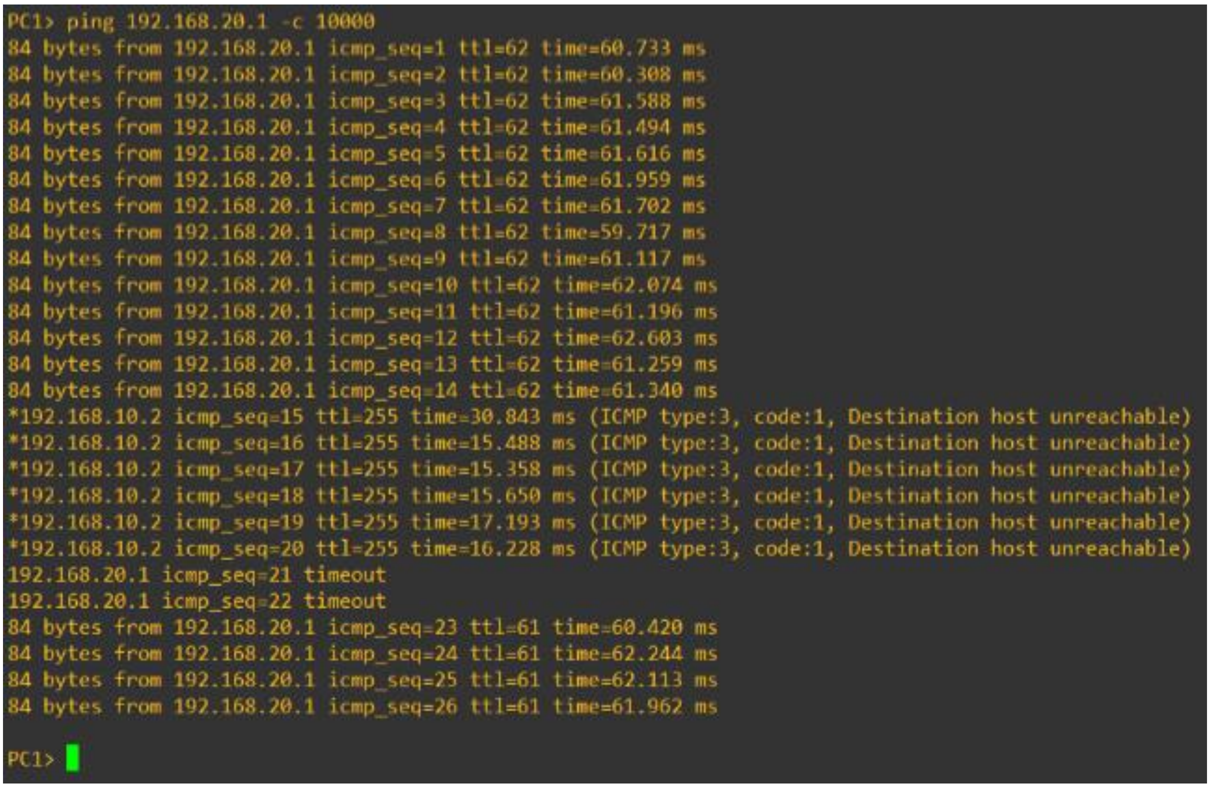
b)

A simulated scenario of link failures has been implemented, but due to the rapid iterations within the OSPF network, the destination can still be reached by utilizing an alternative path following a few missed packets. The results of a keep query are depicted in the image below.



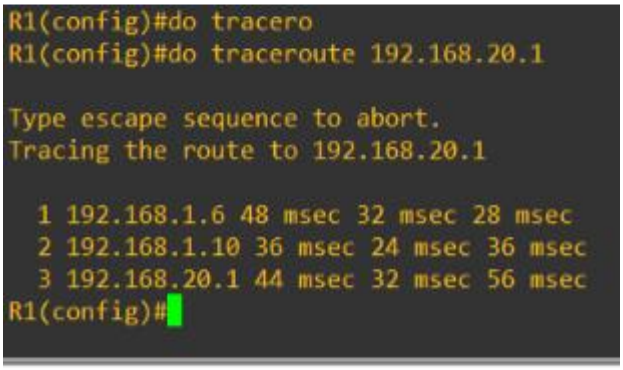
```
R1(config)#do tracerou
R1(config)#do traceroute 192.168.20.1

Type escape sequence to abort.
Tracing the route to 192.168.20.1

  1 192.168.1.18 28 msec 28 msec 32 msec
  2  *
    192.168.20.1 52 msec 32 msec
R1(config)#
```

The following illustration portrays the sequence of events that unfold after a deliberate network loss and how OSPF effectively recovers from it.



```
PC1> ping 192.168.20.1 -c 10000
84 bytes from 192.168.20.1 icmp_seq=1 ttl=62 time=60.733 ms
84 bytes from 192.168.20.1 icmp_seq=2 ttl=62 time=60.308 ms
84 bytes from 192.168.20.1 icmp_seq=3 ttl=62 time=61.588 ms
84 bytes from 192.168.20.1 icmp_seq=4 ttl=62 time=61.494 ms
84 bytes from 192.168.20.1 icmp_seq=5 ttl=62 time=61.616 ms
84 bytes from 192.168.20.1 icmp_seq=6 ttl=62 time=61.959 ms
84 bytes from 192.168.20.1 icmp_seq=7 ttl=62 time=61.702 ms
84 bytes from 192.168.20.1 icmp_seq=8 ttl=62 time=59.717 ms
84 bytes from 192.168.20.1 icmp_seq=9 ttl=62 time=61.117 ms
84 bytes from 192.168.20.1 icmp_seq=10 ttl=62 time=62.074 ms
84 bytes from 192.168.20.1 icmp_seq=11 ttl=62 time=61.196 ms
84 bytes from 192.168.20.1 icmp_seq=12 ttl=62 time=62.603 ms
84 bytes from 192.168.20.1 icmp_seq=13 ttl=62 time=61.259 ms
84 bytes from 192.168.20.1 icmp_seq=14 ttl=62 time=61.340 ms
*192.168.10.2 icmp_seq=15 ttl=255 time=30.843 ms (ICMP type:3, code:1, Destination host unreachable)
*192.168.10.2 icmp_seq=16 ttl=255 time=15.488 ms (ICMP type:3, code:1, Destination host unreachable)
*192.168.10.2 icmp_seq=17 ttl=255 time=15.358 ms (ICMP type:3, code:1, Destination host unreachable)
*192.168.10.2 icmp_seq=18 ttl=255 time=15.650 ms (ICMP type:3, code:1, Destination host unreachable)
*192.168.10.2 icmp_seq=19 ttl=255 time=17.193 ms (ICMP type:3, code:1, Destination host unreachable)
*192.168.10.2 icmp_seq=20 ttl=255 time=16.228 ms (ICMP type:3, code:1, Destination host unreachable)
192.168.20.1 icmp_seq=21 timeout
192.168.20.1 icmp_seq=22 timeout
84 bytes from 192.168.20.1 icmp_seq=23 ttl=61 time=60.420 ms
84 bytes from 192.168.20.1 icmp_seq=24 ttl=61 time=62.244 ms
84 bytes from 192.168.20.1 icmp_seq=25 ttl=61 time=62.113 ms
84 bytes from 192.168.20.1 icmp_seq=26 ttl=61 time=61.962 ms

PC1> 
```

The OSPF selecting a new route.



```
R1(config)#do tracero
R1(config)#do traceroute 192.168.20.1

Type escape sequence to abort.
Tracing the route to 192.168.20.1

  1 192.168.1.6 48 msec 32 msec 28 msec
  2 192.168.1.10 36 msec 24 msec 36 msec
  3 192.168.20.1 44 msec 32 msec 56 msec
R1(config)#
```

5

OSPF, a dynamic routing protocol employing a link-state algorithm, is utilized to determine the optimal route to a destination network. It is designed to cater to large and sophisticated networks, making it widely adopted in both enterprise and Internet environments. OSPF utilizes a cost metric based on connection bandwidth, with lower costs indicating better paths. Each router within OSPF creates a topology map by exchanging data with neighboring routers. This allows OSPF to support load balancing and route analysis while employing Dijkstra's method to find the shortest route to a destination node.

RIP, on the other hand, is an older dynamic routing protocol that implements a distance-vector approach to identify the best route to a destination node. Although RIP is less popular in modern networks, it measures the route using hop counts. RIP can be slow to adapt to changes in network structure and is prone to frequent routing circuit occurrences.

While static routing is straightforward to configure, it lacks adaptability and becomes challenging to manage in large networks. Manual reconfiguration is required for static routing when there are alterations in the network topology, as it does not respond automatically to such changes.

Consequently, RIP is considered an outdated and less commonly used protocol due to its inefficiencies, while OSPF stands as a reliable, scalable, and practical routing protocol suitable for large and complex networks. Static routing, while simple to set up, lacks scalability and fails to respond to modifications in the network architecture.