LAB REPORT

LAB TITLE: Analyzing Data Link and Network Layer Traffic with Wireshark

Name: Gabriel Miracle Nwauche

PSU ID: GMN5192@psu.edu

Course: IST 220

Instructor: Nick Giacobe

Date: 2/16/2023

General Context

In today's lab, the prompt made us venture into new measures in the world of endless protocols and networking. Once again this lab took place in the PuTTY application, which would lead the user (network engineer) to log into the application Quagga for some command line interface configuration. I connected to the first router by initiating the PuTTY application, to do this you must type in the IP address of the specific router and make sure it is an SSH protocol used for the configuration. This instruction would be repeated to connect to the other routers as well.

Here comes the tricky part, actually having to connect the routers to one another. After a few tries, the routers were finally connected to each other. The reason it took more than one attempt is because the lab didn't specify how meticulous the user had to be with the command line input. However, this called for some resiliency on the users end to configure the routing tables and see the OSPF traffic for the packets as well. Once the user got familiar with the OSPF commands, it took some perseverance to modify the cost for the bandwidth on my own. This is what switches the OSPF route from the shorter to longer route.

Technical Context

As expected, this lab began to get extremely difficult once the instructions labeled the user as a "Senior Network Engineer." The acceleration of the lab was a difficult curve to cover which inflated the magnitude of the situation in each section. First, the user was prompted to use the dynamic routing protocol RIPv2 to start the configuration of the routing for the network. After the user got familiar with those commands in the command line interface, this escalated the user to use an OSPFv2 for the routers. To check if the IP addresses in the router connections were viable, I had to check using the IP routing table to confirm. The skillful portion in the lab came to the part where the network topology was displayed. Reading the topology chart carefully made this process a lot smoother than it would've been if the user didn't.

The shutdown for the specific interface hostname was the climax of this lab. This was the interface name (ens256) that had a direct connection to router1 and router3. Once the user shut those connections off, the ip routing table began to look a lot different. To insinuate this configuration, I had to first: configure the table. Next, I had to route the OSPF using the command *router ospf* . At this point the user would have to check the IP routes to see which interface and which IP address connected the networks together, once found the user would need to calculate the fastest route for bandwidth by adding the components from each network according to the route and the distance vector. This calculation was crucial for the convincing of the route to take. Using the command *ip ospf cost* I was able to change the bandwidth to make this route longer. After configuration of that OSPF route, the traceroute shown in the last screenshot took the route which was said to take longer (router1 > router3).

Section 1: Part 1: Configure RIPv2 on the routers

Make a screen capture showing the **currently running RIP configuration on router1**.

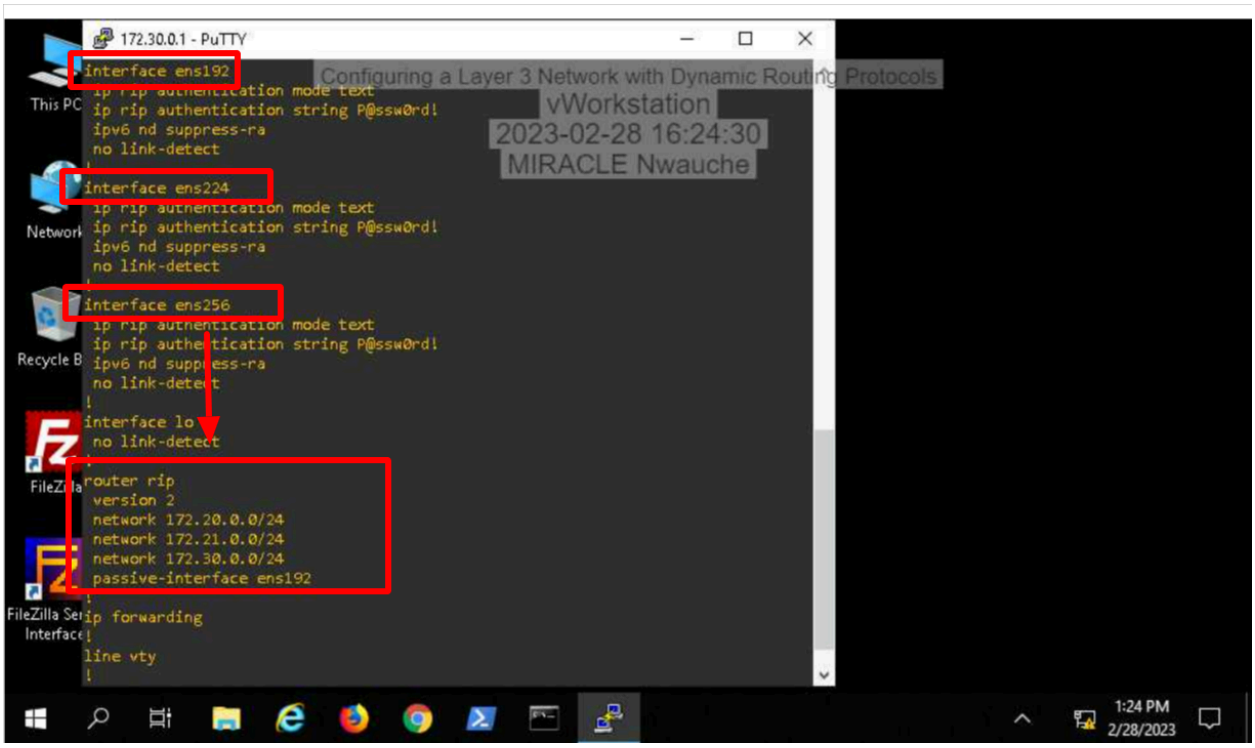


Figure 1, Section 1, Part 1: This is the result of the commands inputted into the command line, this shows that the RIPv2 protocol has been configured and the interfaces have been connected to the router. The last red highlighted box shows all the IP addresses of the device connected under router RIP.

24. **Make a screen capture** showing the **output of the show ip rip status command**.

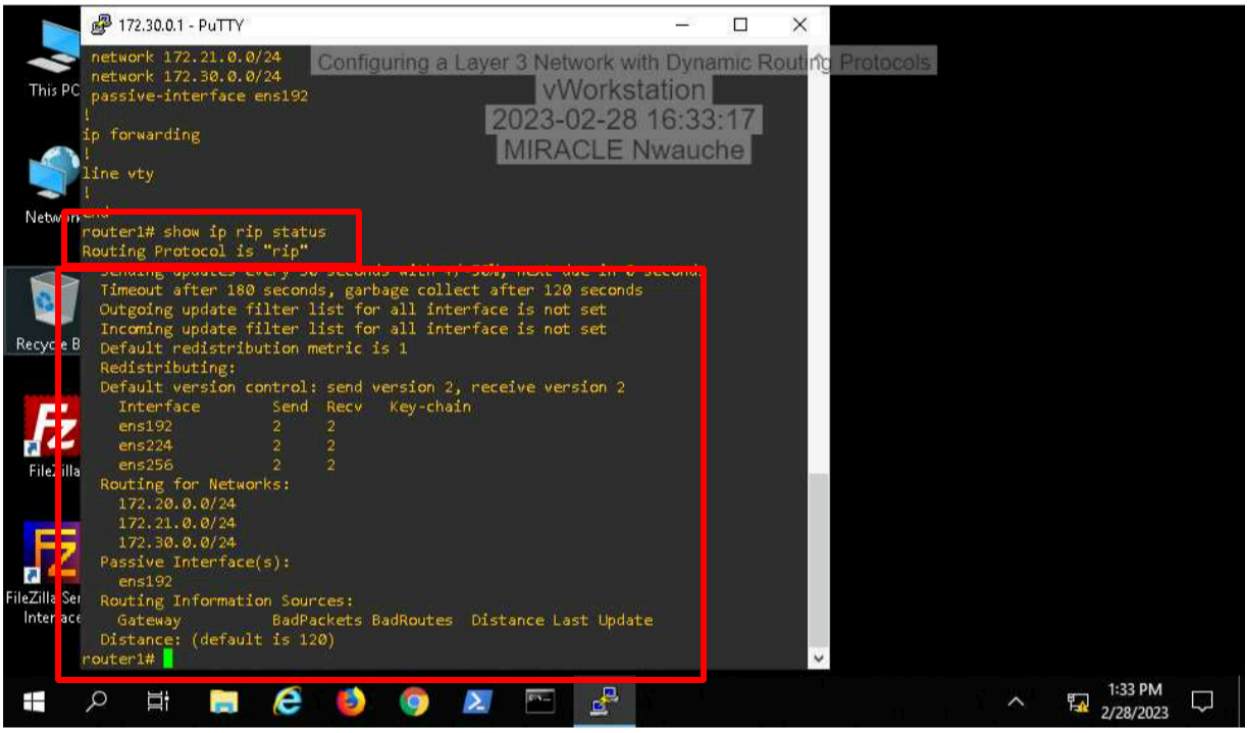


Figure 2, Section 1, Part 1: After the *show ip rip status command* is executed, the routing table appears on the screen. The routing table entails sending updates every -, timeout after(180), garbage collect after, and distance value. All of which have their own default times to do their functions.

30. Make a screen capture showing the currently running RIP configuration on router 2

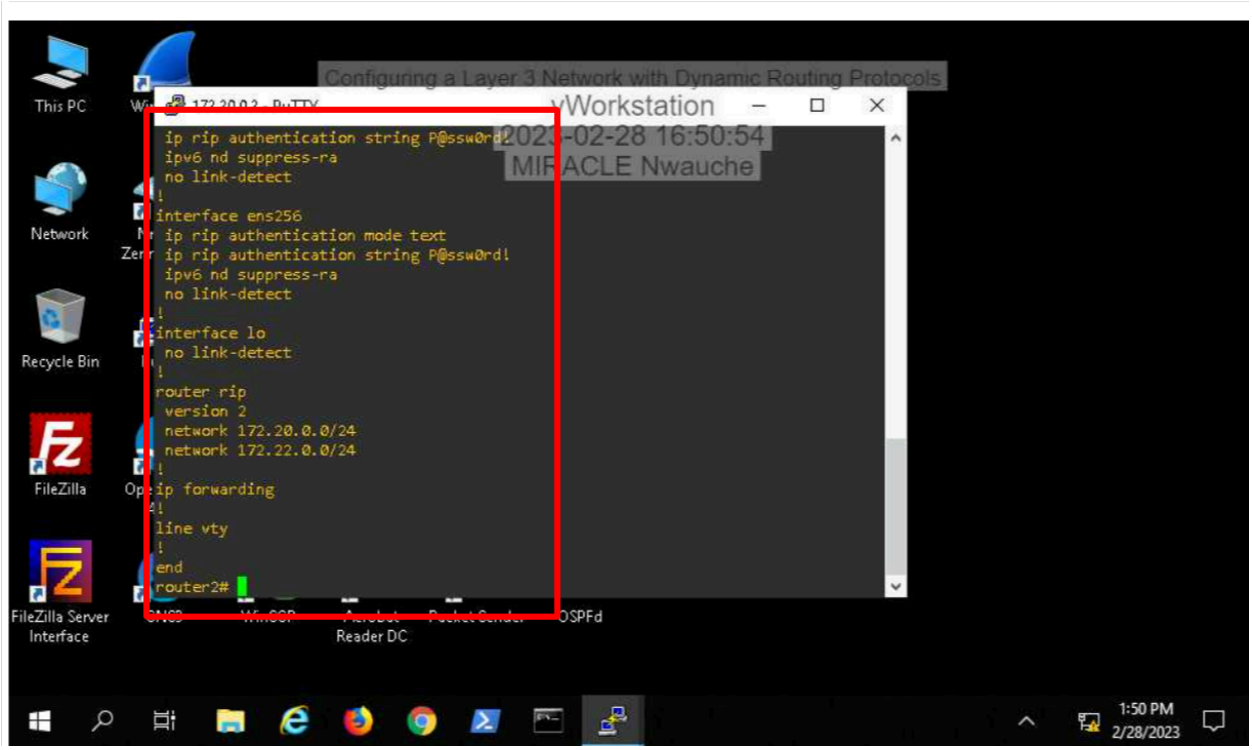


Figure 3, Section 1, Part 1: Pictured is the repeated process of configuring a RIP command for this router which is specifically router 2. The same commands were typed and executed for this router and router 1, the only difference was some of the interface names.

41. **Make a screen capture** showing the **routes known by router3**.

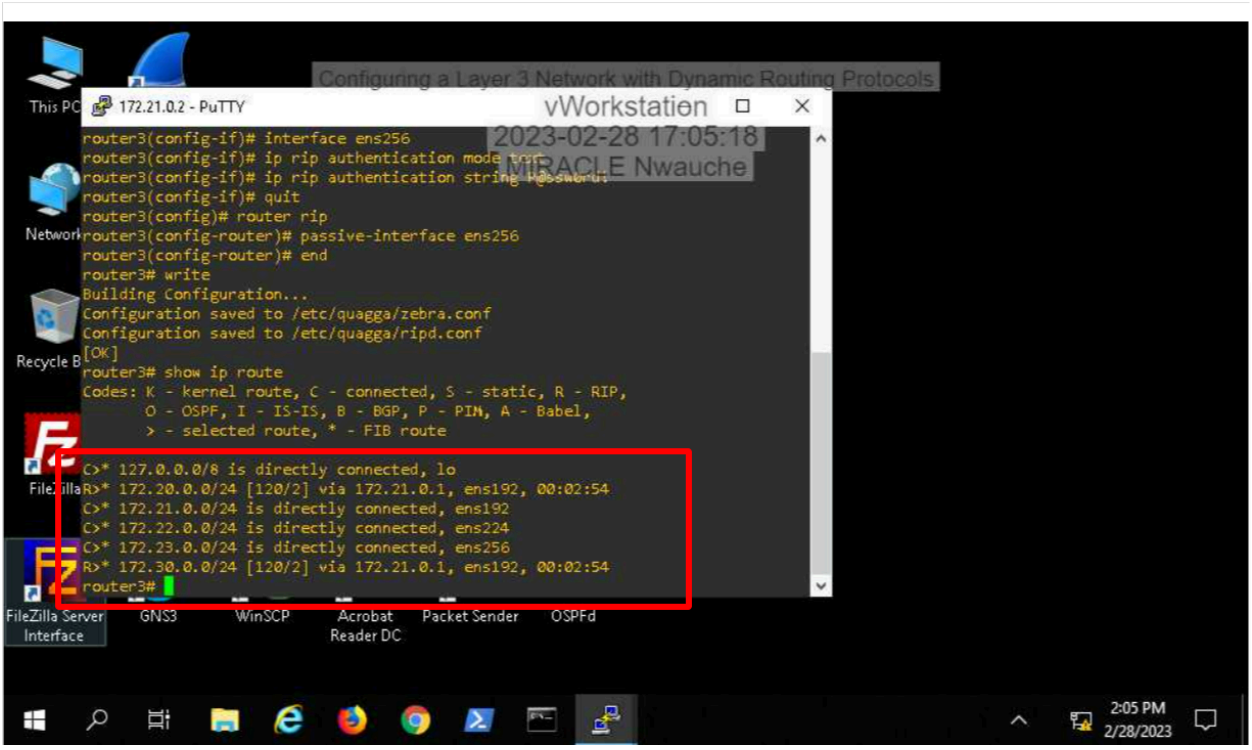


Figure 4, Section 1, Part 1: Here are the connected and alternative routes in the network that are connected from router to router. Some of the IP addresses state they are directly connected and others determine themselves as RIP, which means they could be classified as the alternative routes for the network.

Part 2: Test the RIPv2 Configuration

15. **Make a screen capture** showing the **"Destination net unreachable" messages, including the successful responses that preceded and succeeded them**.

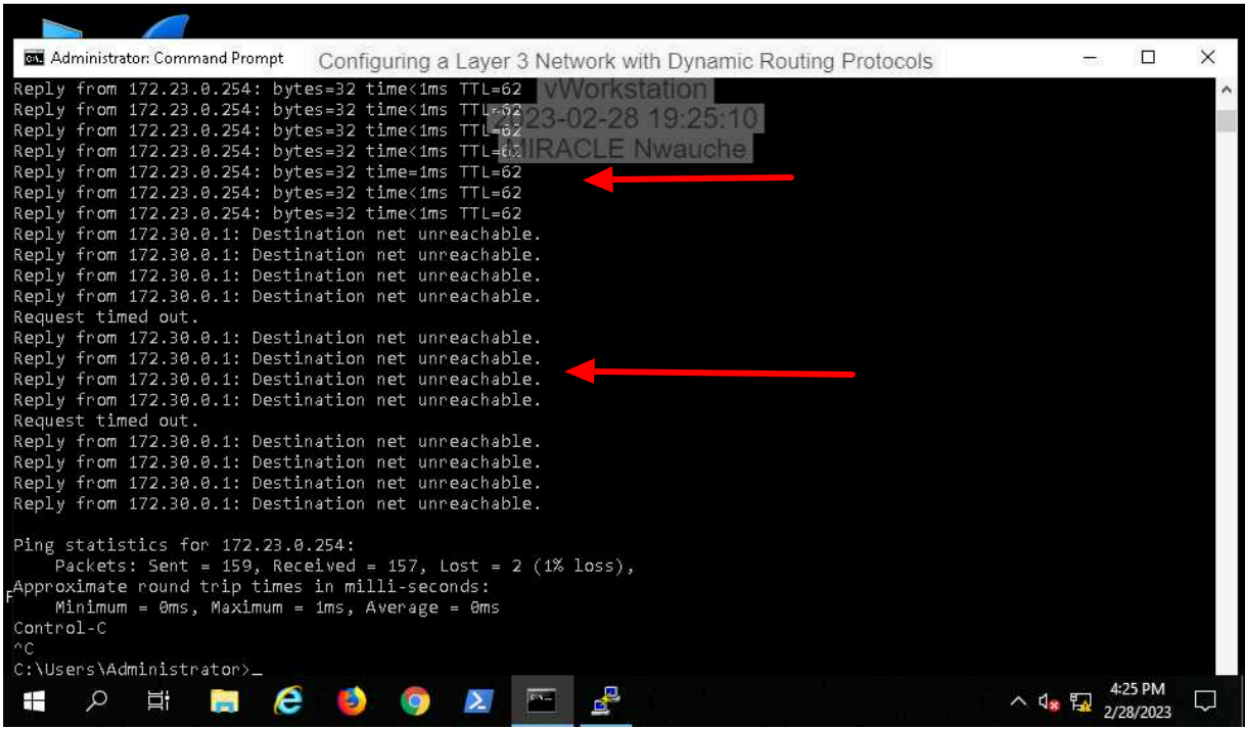


Figure 5, Section 1, Part 2: *Destination unreachable*, In this particular frame in the command prompt we see what happened to the router when the connection was shut off. Due to the command in the puTTY system, the configuration in the Command line shows the user that the destination is now unreachable. The top arrow of the picture is showing the successful attempts as well.

20. **Make a screen capture** showing the **new routing table on router1 that resulted from the ens256 link removal**.

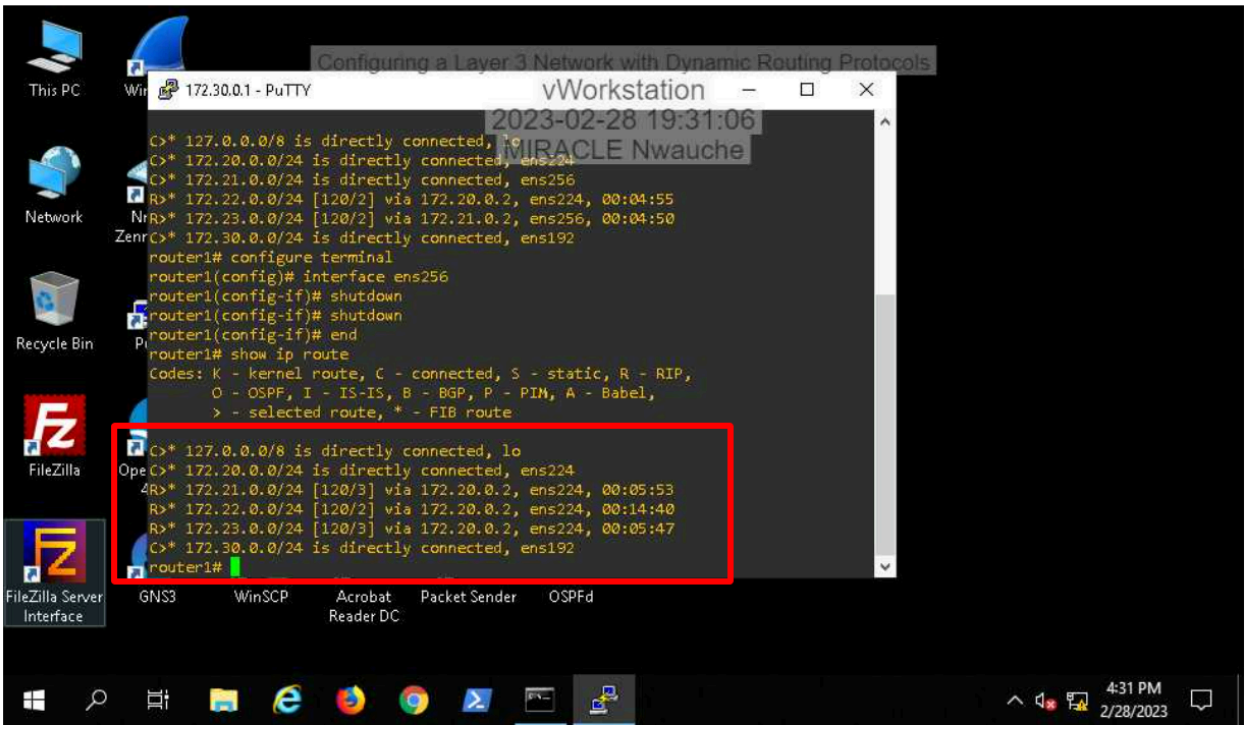


Figure 6, Section 1, Part 2: The results from the ens256 interface link removal are as shown. All the routes for the network seem to only be able to pass through the router2. You can see in all the connections, they all show the interface name ens224 (besides the last one). All routes also share the IP address destination 172.20.0.2

26. **Make a screen capture** showing the **updated routing table on router1**.

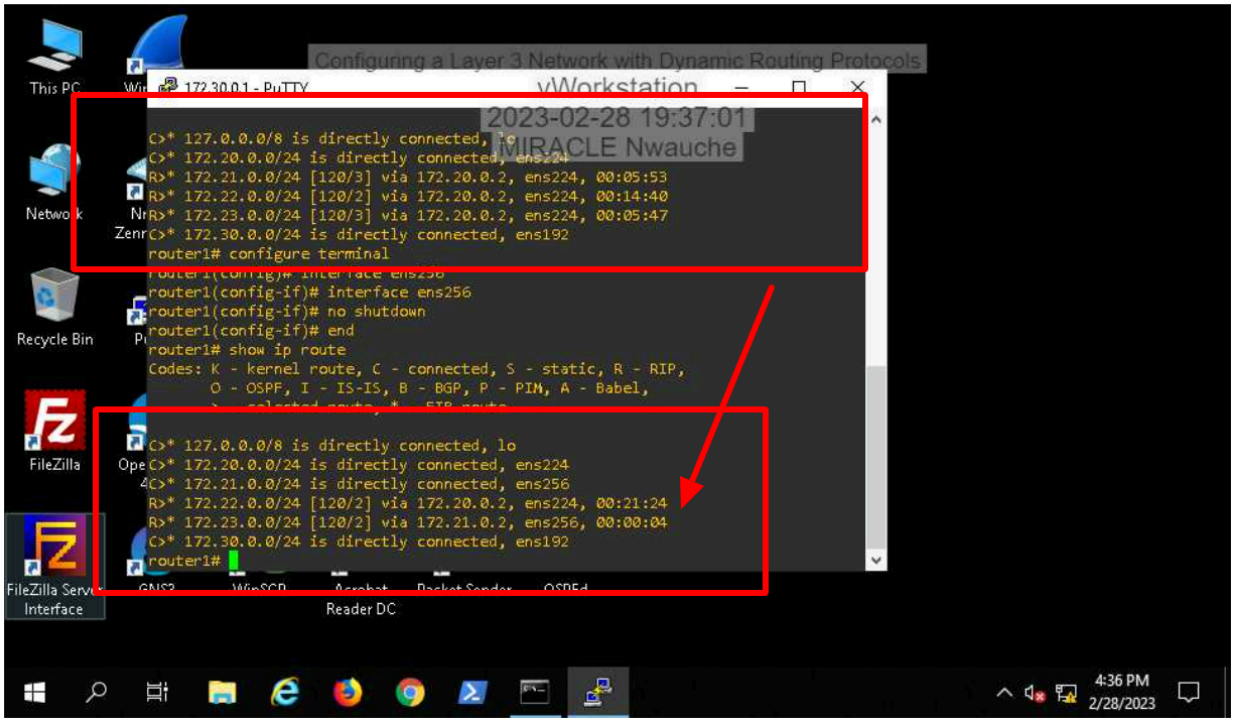


Figure 7, Section 1, Part 2: The reverse shutdown command on the puTTy device was executed. It involved re - configuring the interface ens256 command with the same commands following it as well. When you get to the last line of the command, the reverse was executed by *no shutdown*, the updated configuration table is shown at the bottom. Like the shutdown never even happened.

SECTION 2, Part 1: Configure OSPFv2 on the Routers

17. **Make a screen capture** showing the **running OSPF configuration on router1**.

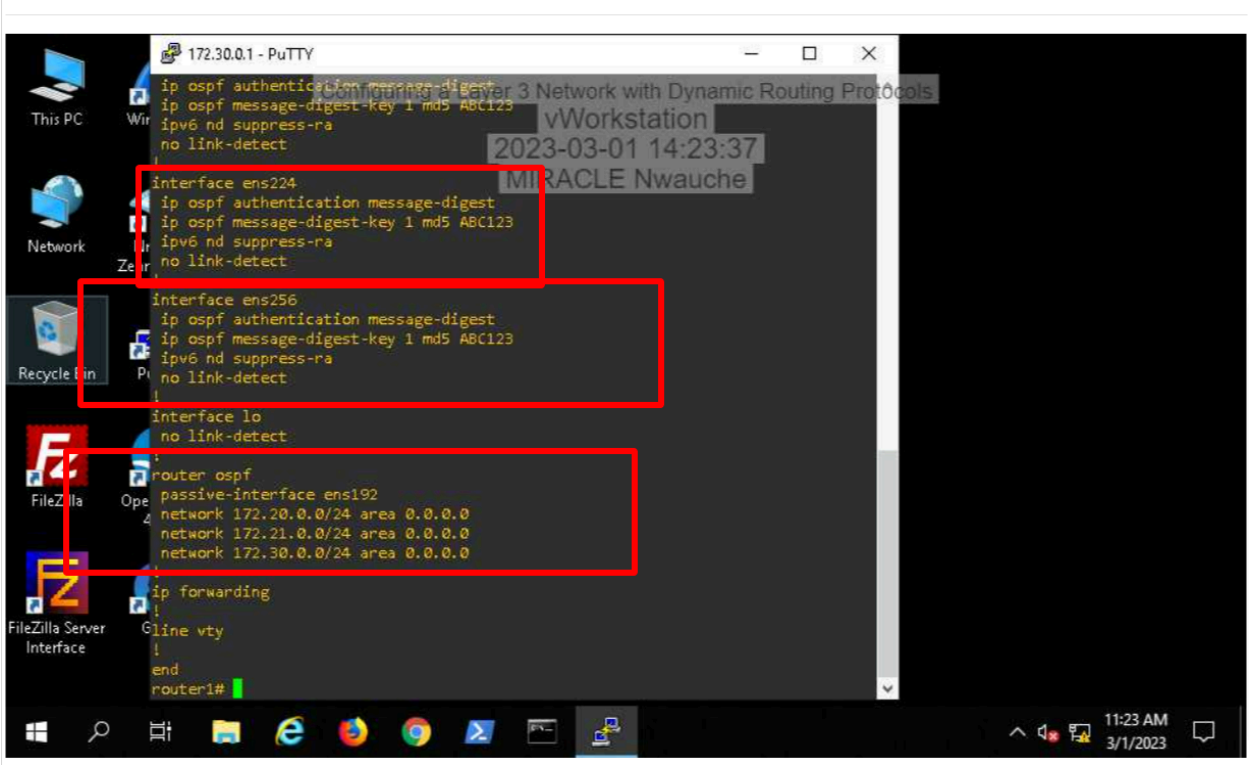


Figure 8, Section 2, Part 1: *ospf configuration*: Using many of the same commands to configure the router, I added more commands that were specific to the OSPF area itself, which in this case was area 0. The message digest keys are displayed in the routing table as well, one thing that sticks out is where the table says “no link detect.”

20. **Make a screen capture** showing the **current OSPF routing table on router1**.

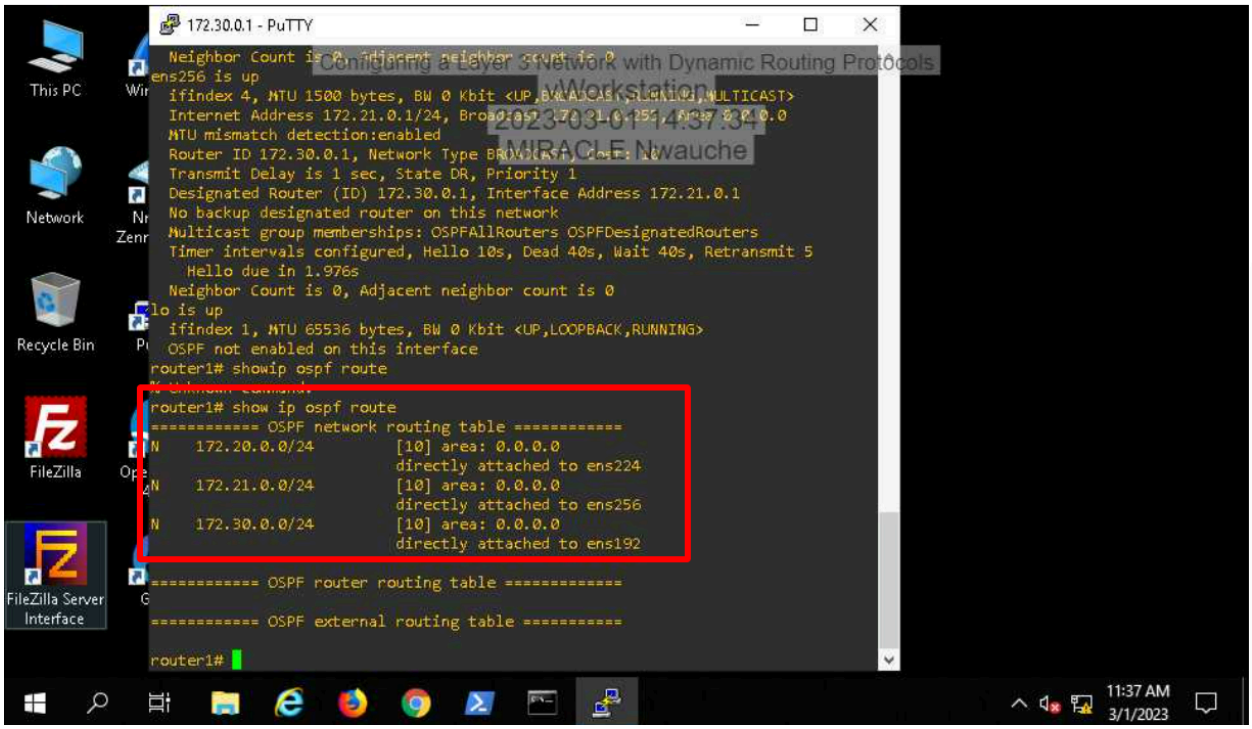


Figure 9, Section 2, Part 2: *ospf routing table:* In this capture, the user is shown the OSPF routing table which contains some critical information for the router that it was configured to. The [10] shows the cost that the OSPF has configured for the network based on which route it should take. This number is derived from the bandwidth value.

26. **Make a screen capture** showing the **updated OSPF routing table on router1**.

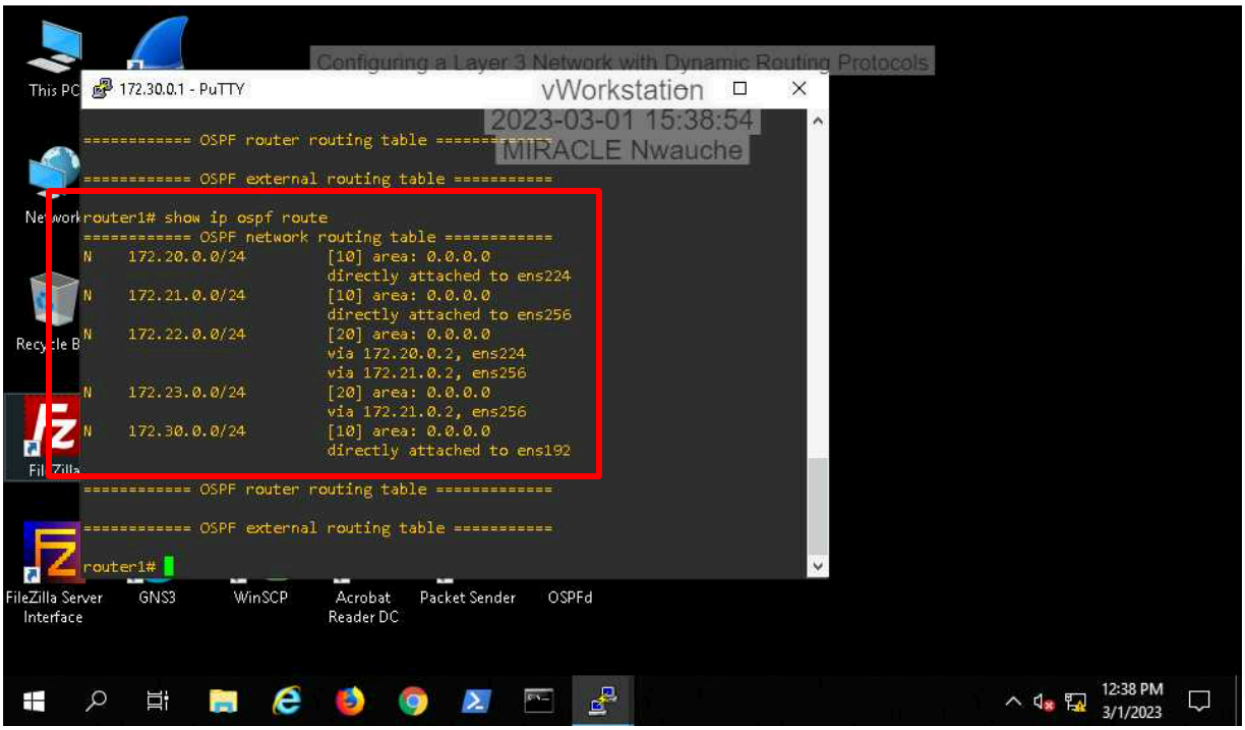


Figure 10, Section 2, Part 1: Here is the successful attachment of the routers together in the OSPF routing table. According to the network topology, this is how the network should look in the routing table, all the interface names and IP addresses are also stated in the table. Some of the bandwidth ([20]) lengths are now doubled which probably means they will not be considered the fastest route for the data.

Part 2: Test the OSPFv2 Configuration

13. **Make a screen capture** showing that **ens256 is down per the OSPF interface output**.

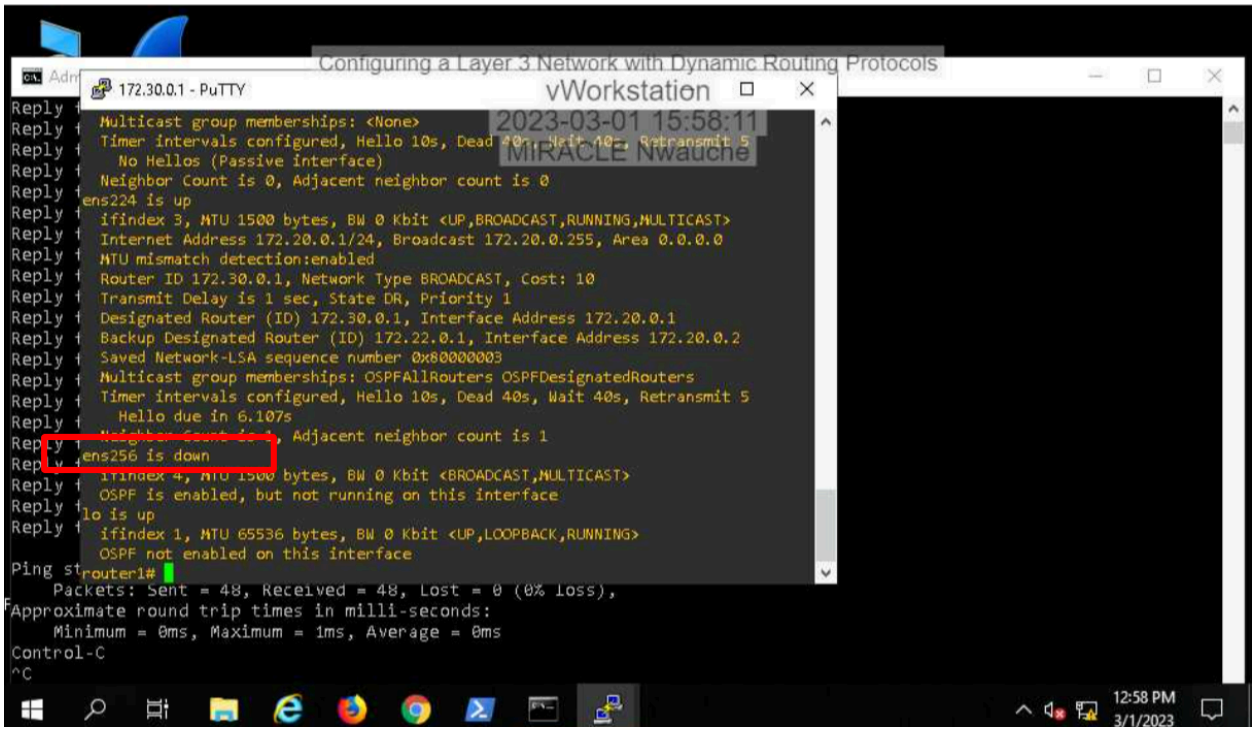


Figure 11, Section 2, Part 2: *ens256 is down* : Once prompted to log back into the router1, the user is supposed to terminate the connection between ens256 and the router in the OSPF interface. After shutting down the connection, you can see in the details after executing *show ip ospf interface* the table above shows “ens256 is down.”

16. **Make a screen capture** showing the **traceroute path through router2**.

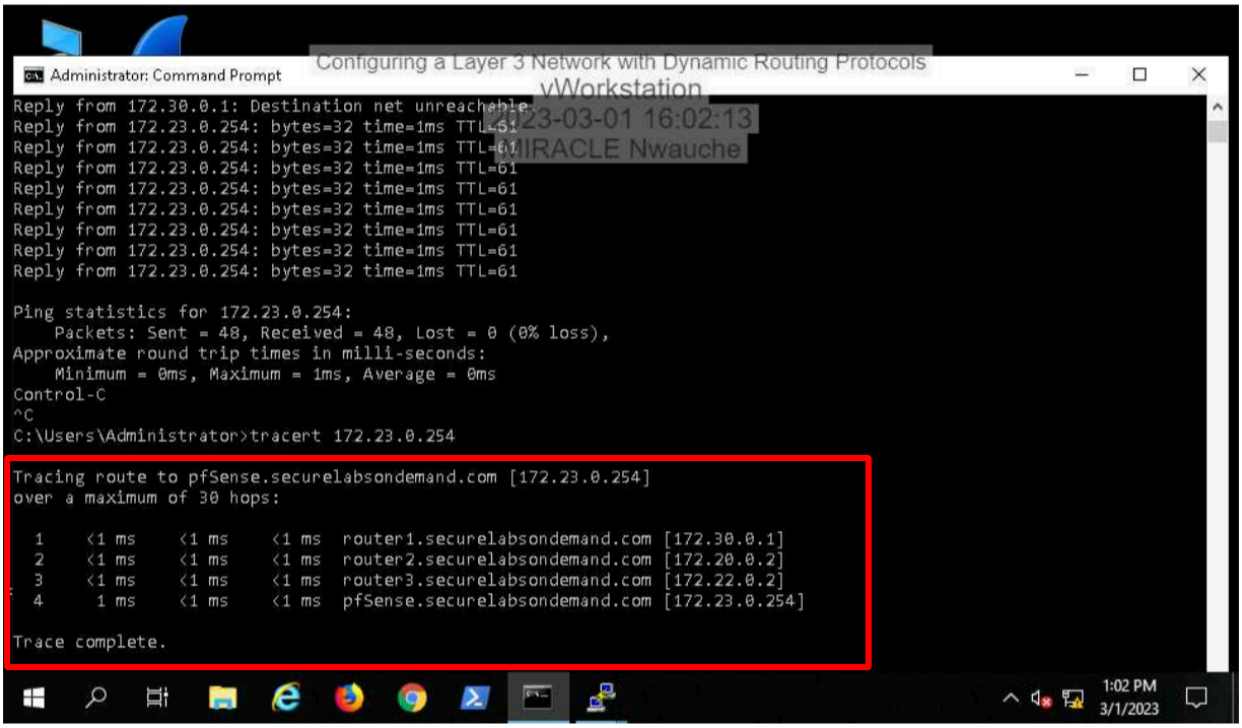


Figure 12, Section 2, Part 2: *traceroute:* In the command line prompt, I executed the command to configure the traceroute for the given network. The details shown in the highlighted area show the connection to the router2 device and router3 ensuing that device as well. It ranks them in order from the fastest route to the slowest, since the ens256 connection was terminated shortly before.

23. **Make a screen capture** showing the **full routing table on router1**.

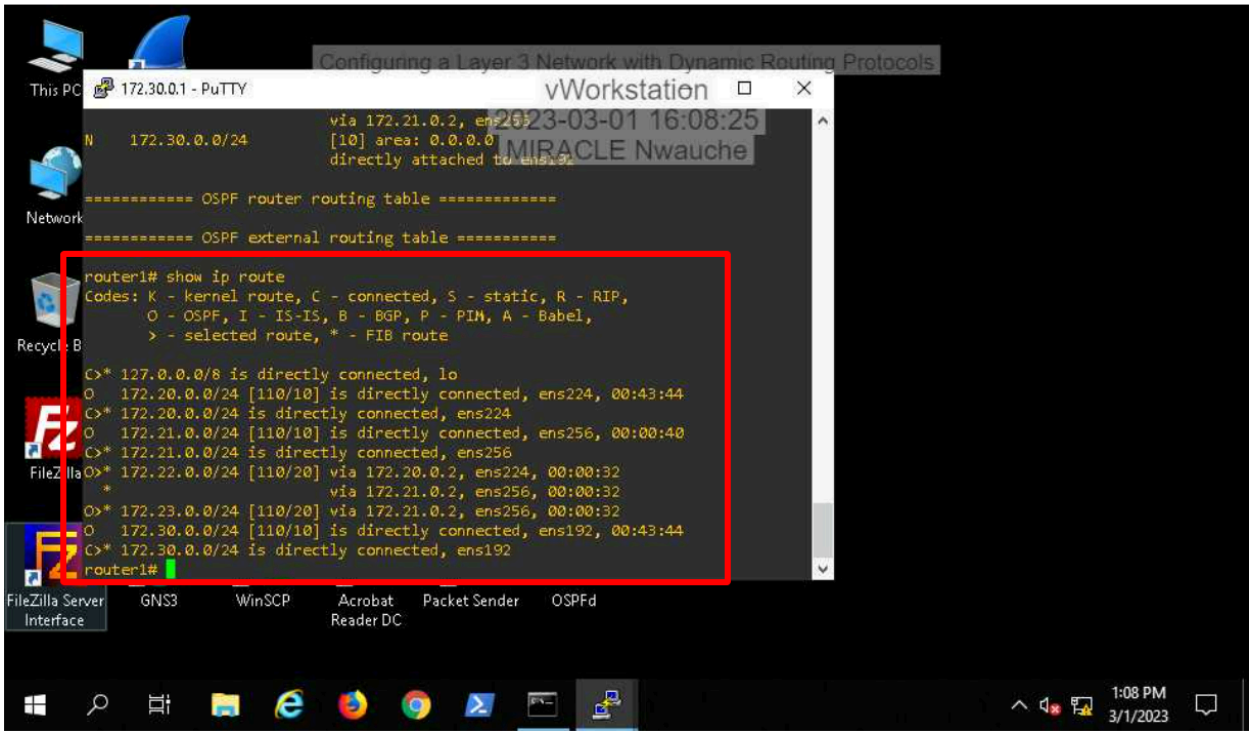
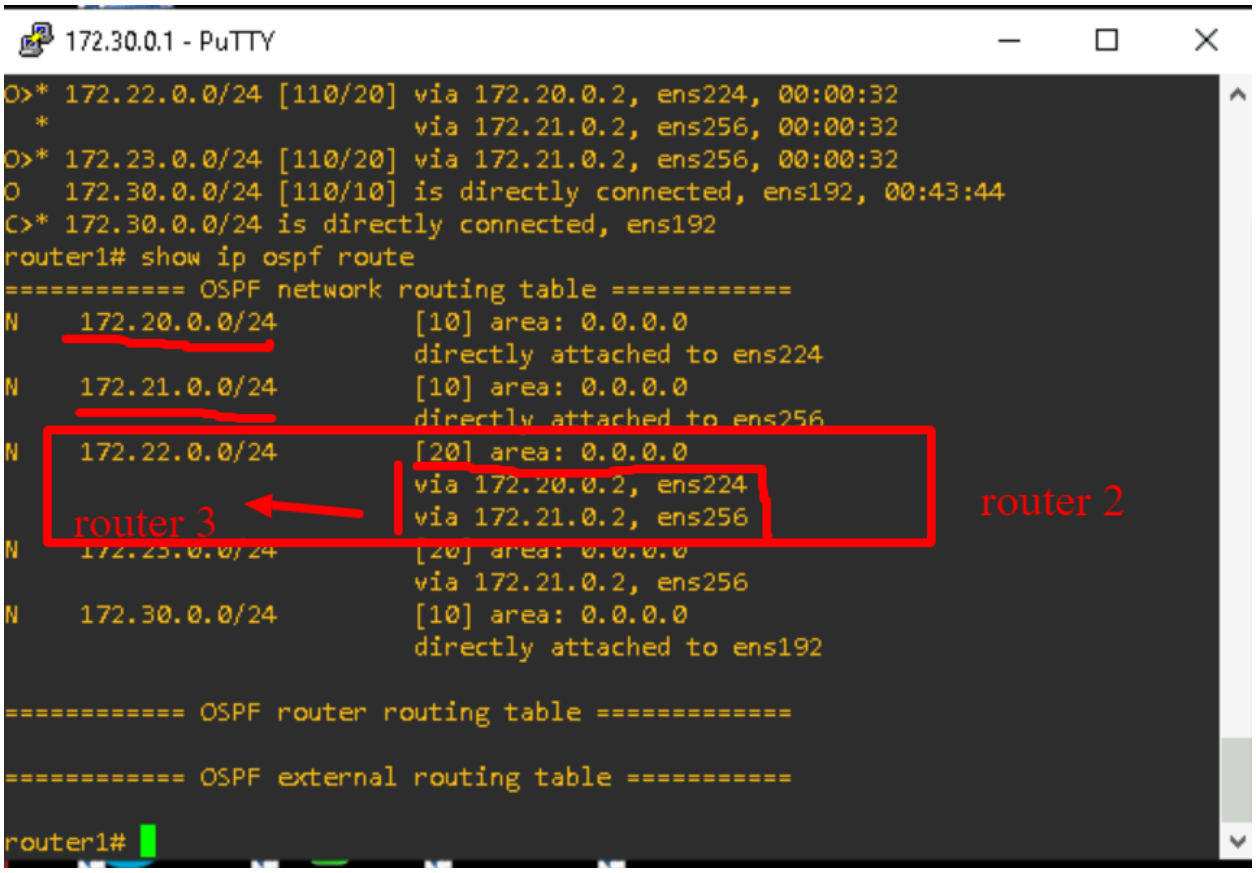


Figure 13, Section 2, Part 2: *show ip route:* I executed the command to reverse the shutdown of the device connected to ens256, after the route was restored I proceeded to open the routing table. From what it displayed, there were many connected ports and OSPF related ports on the table, including ens256.

Record the minimum OSPF cost needed for the router1 > router3 link to convince OSPF that this path is less efficient than the router1 > router2 > router3 path. **Explain** how you calculated this value.

The minimum cost for the router1 > router 3 link is [50] because if you add the bandwidth from the networks associated with router1 (172.20.0.0, 172.21.0.0, 172.30.0.0) those combine to [21] because each of their cost are [10] then you will have to add the [20] directly from router3 , where as if you connect it through router1 > router2 > router3, router 2 already includes all of those connections from router 3 in its OSPF path which is the same price and before the connection to the last port (172.30.0.0) in the table



Make a screen capture showing the **new cost assignments on router1's OSPF routes**.

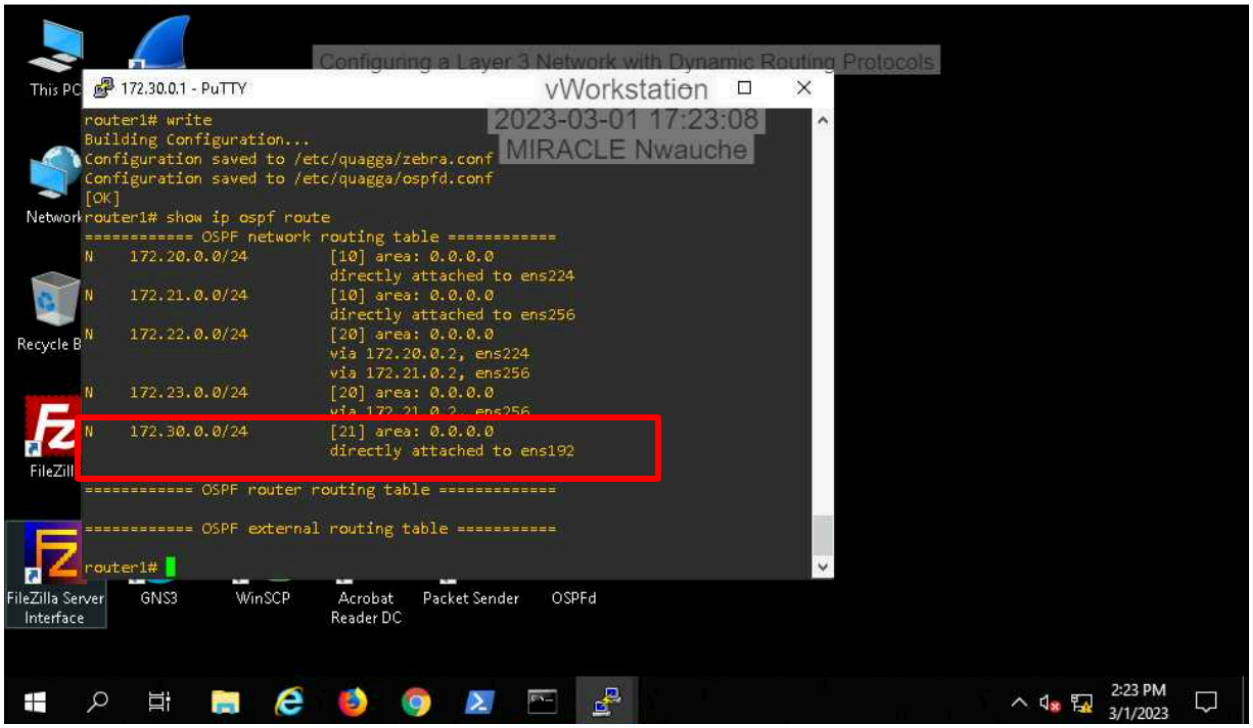


Figure 14, Section 3, Part 2: *ospf cost:* Here is the new OSPF cost I configured into the ip OSPF route in the table above. The command I used to change the cost was *ip opsf cost 21.* This changed the cost in the table, which would inevitably convince the user to use a different route for the packets to be transferred to.

Make a screen capture showing the **new path taken to reach the pfSense appliance**.

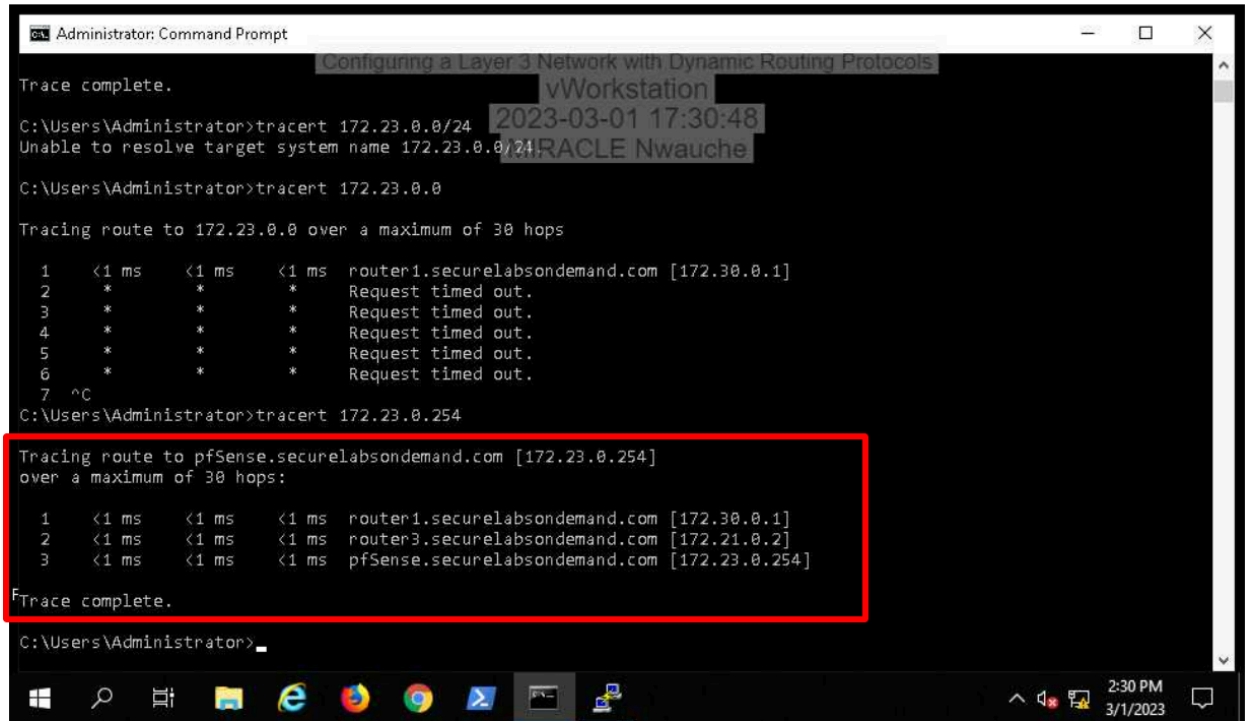


Figure 14, Section 3, Part 3: *traceroute update:* Here is when we configure the traceroute for the OSPF and change it to the longer route after changing the cost. As you can see, instead of the previous route it took (found in the earlier captures) it would take this route, making the packets take longer in the transfer. Then after router1 > router3 the pfsense is next in line.