Lab 6

Computer Communication Networks Lab

Intra AS Routing: RIP & OSPF

The lab's schedule

Published date:	19/05/22
Quiz date:	26,24/05/22
Deadline for the final report:	09/06/22

BEN-GURION UNIVERSITY OF THE NEGEV - COMMUNICATION SYSTEMS ENGINEERING

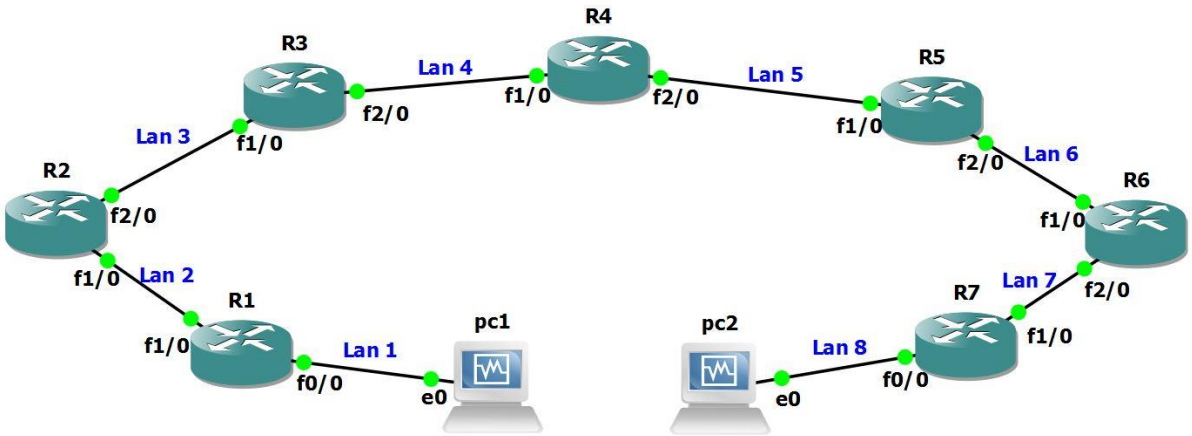
General instructions

- **The final report** will be based on your answers to the "*Attach to your report*" sections at this document.
- **Remember** to pay attention to the "Final report submission" in the Introduction Lab.
- Most of the laboratory experiments based on the book: "*Mastering Networks: An Internet Lab Manual*"
- It is recommended to **read the all exercise before you do it, to understand the main idea.**
- IP address is composed of four octets ("octet1.octet2.octet3.octet4").
  In our Labs all IP addresses will be according to the Network Figure, except *octet2* in network addresses between routers.
  ***Octet2* will be according to the pair's number ("10.*X*.0.1", *X = pair's number*).**
- **Write the number *X*** clearly **on the title page** of your final report.

Reading material

- Read about *RIP Version 2 /1* at the following link:
  http://docwiki.cisco.com/wiki/Routing_Information_Protocol
- The *basic RIP protocol* suffers from several problems.
  Read about its main problems:
  http://www.tcpipguide.com/free/t_RIPProtocolLimitationsandProblems.htm
  http://www.tcpipguide.com/free/t_RIPProtocolLimitationsandProblems-2.htm
  http://www.tcpipguide.com/free/t_RIPProtocolLimitationsandProblems-3.htm
- Be familiar with the term of "*Passive mode*" from the following link (Three first lines):
  https://web.archive.org/web/20170801132809/https://www.cisco.com/c/en/us/td/docs/security/asa/asa82/configuration/guide/config/route_rip.html
- Read about the *RIP's* mechanisms: "*Route poisoning*", "*Split Horizon*" and "*Triggered updates*", from the following links:
  http://study-ccna.com/rip-loop-prevention
  http://technet.microsoft.com/en-us/library/cc940478.aspx
- Read about *OSPF*, and about the differences between *RIP* (based distance vector) and *OSPF* (based link state).
  http://docwiki.cisco.com/wiki/Open_Shortest_Path_First

Preliminary questions



- This network is using *RIP* as its dynamic routing protocol.
  R1-f0/0 and *R7-f1/0* interfaces were configured to be in *passive mode*.
  -Explain what kind of protocol's massages are sent or received at the passive interfaces?
  -Does setting this options impact other kind of massages?
  -consider *R7's* routing table, will it be full?
  -Explain why a Ping from *pc1* to *pc2* would <u>fail</u>.
  -Which interface in this network would you set as *passive* to allow the ping from PC1 to PC2 to function correctly.

- This network is using *basic RIP* as its dynamic routing protocol (without *triggered updates*, *split horizon* and *route poisoning*).
  Now assume we shutdown(close) the interface *R6-f2/0*.
  -How much time approximately it takes until *R1* receive the update?
  -How much time approximately until *R7's* routing table converge?
  Base your answer on the update procedure of the protocol and its timers.

- The network uses *basic RIP* as its dynamic routing protocol.
  Try to describe a "*count to infinity*" situation in the network, when we shutdown *R6-f2/0*.

- What is the maximal network diameter in a *RIP* network? Why?

- Explain how the "*route poisoning*" mechanism of *RIP* protocol works.

- Explain how the "*split horizon*" mechanism of *RIP* protocol works.

- Explain how "*triggered updates*" mechanism helps *RIP* to converge.

- Describe the main differences of a *distance vector* routing scheme versus a *link state* scheme.
  Refer to the information distribution throughout the network and its content.
  Give an example of dynamic routing protocol for each scheme.

- What is the '*Hello' mechanism's* purposes in the *OSPF* protocol?

- What is the purposes of the *Database description* packets and the *Link-state* packets, in the *OSPF* protocol?

- What is the main idea of *Hierarchical OSPF* in the context of the network structure, and what are its main advantages?

The practical section

Before we start:

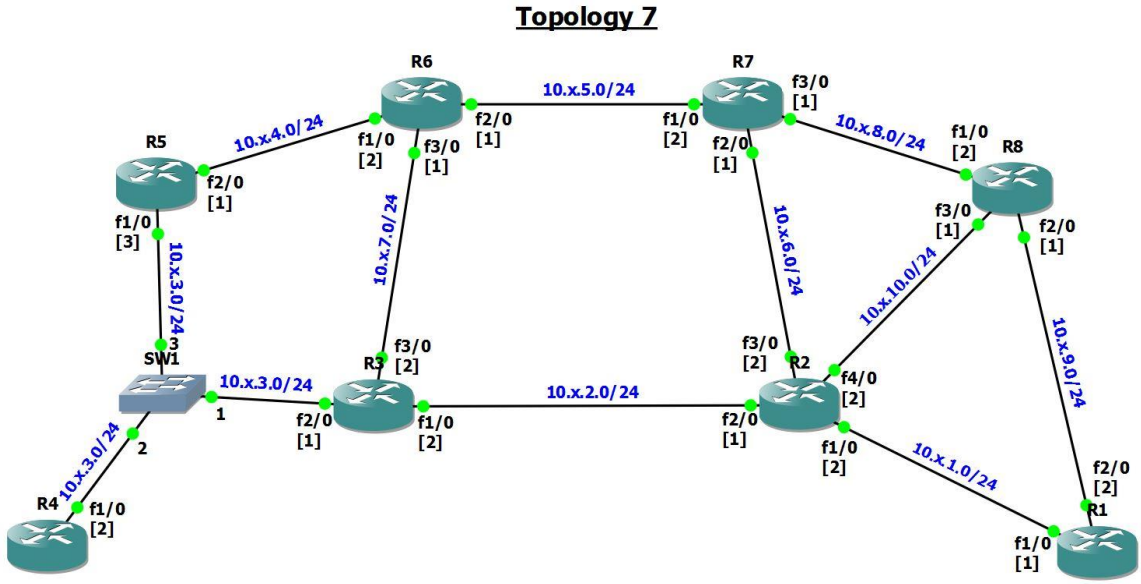
How to work effectively in this lab:

Note that the router interface has a copy paste function. Click *ctrl+c* on a text you wish to copy and right click inside the router terminal to copy commands into the terminal. You can copy several commands at once into the router.

You can use this function effectively if Before you enter commands into a router you write down the commands inside an ordered text file, and only then copy and paste them into the terminal. Ask the lab instructors if you aren't sure how.

Intra AS Routing : RIP

1. Topology 7 Configuration



Pre

1.1. The topology is based on *Topology 6*.
But first, before making any changes, save as a new project as *Topology 7*.

Do

1.2. Build the topology according to the figure.

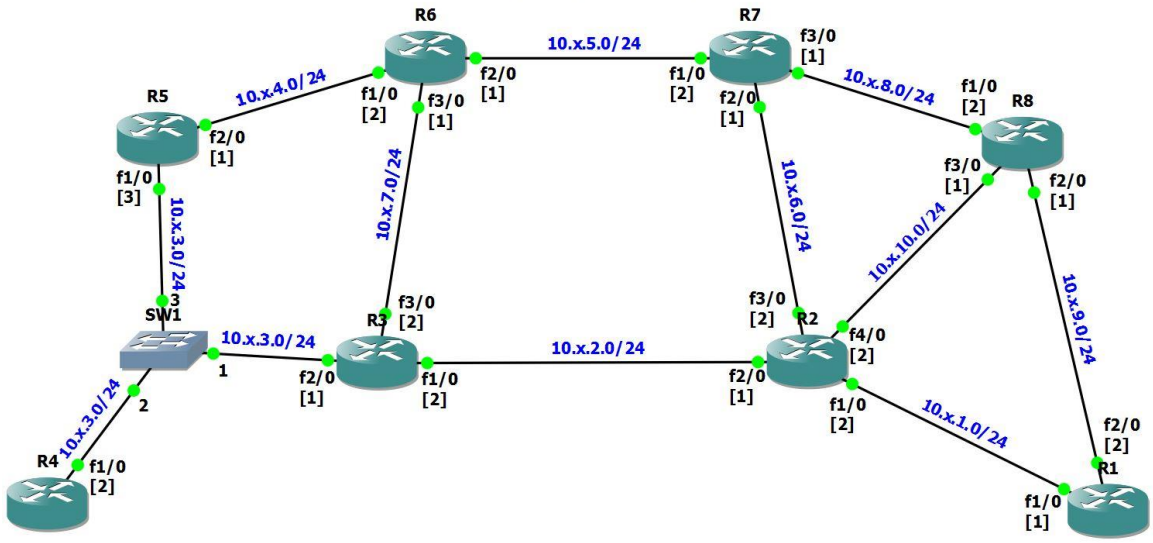
1.3. Configure the network interfaces on each router according to the figure.

1.4. Verify the connection between each pair of neighboring routers using ping.

1.5. Perform "`write`" on each router, save the topology and ZIP it.

2. Configuring RIP version 2 on Cisco Routers

Topology 8 (RIP)



Pre

2.1. The exercise is based on *Topology 7*.
But first, before making any changes, save as a new project as *Topology 8*.

Do

2.2. Display the content of *R1* and *R2 routing tables* and their *RIP databases*.
Use the commands:

```
R1#show ip route
R1#show ip rip database
```

Take a screenshot of the outputs.

2.3. Start Wireshark on *R1* interface *f1/0*.

2.4. Enable *RIP* on all routers, as the following example:

```
R1#configure terminal
R1(config)#router rip
R1(config-router)#version 2
R1(config-router)#no auto-summary
R1(config-router)#network 10.x.0.0
R1(config-router)#end
```

2.5. Wait for the network convergence.

2.6. Display again the content of *R1* and *R2 routing tables* and *RIP databases*, and take a screenshot of the outputs.

2.7. Use another way to display detailed information about RIP configuration.
On *R2*, perform the following command and save the output:

```
R2#show ip protocols
```

2.8. Stop capture and save the wireshark pcap to file.

2.9. You should now be able to ping any router from any other router.
Test your network by using *ping* from some router to all the others.

2.10. **Perform** "`write`" on each router, save the topology (as *Topology 8*) and ZIP it.

Attach to your report

2.11. Explain the meaning of each command from step 4.

2.12. How do you know that the network was convened?
List two ways to see it.

2.13. What is the destination IP address on the *RIP* packets?
Explain why they are not sent as unicast.

2.14. Do routers forward *RIP* packets? Explain how do you know that.

2.15. Which type of routing message do you observe (the type is indicated by value of the field command)?
For each packet that you observe, explain the role of that message in the *RIP protocol*.

2.16. A *RIP* message may contain multiple routing table entries.
How many bytes are consumed in a *RIP* message for each routing table entry?

2.17. What information transmitted by a *RIP* packet and its entries?

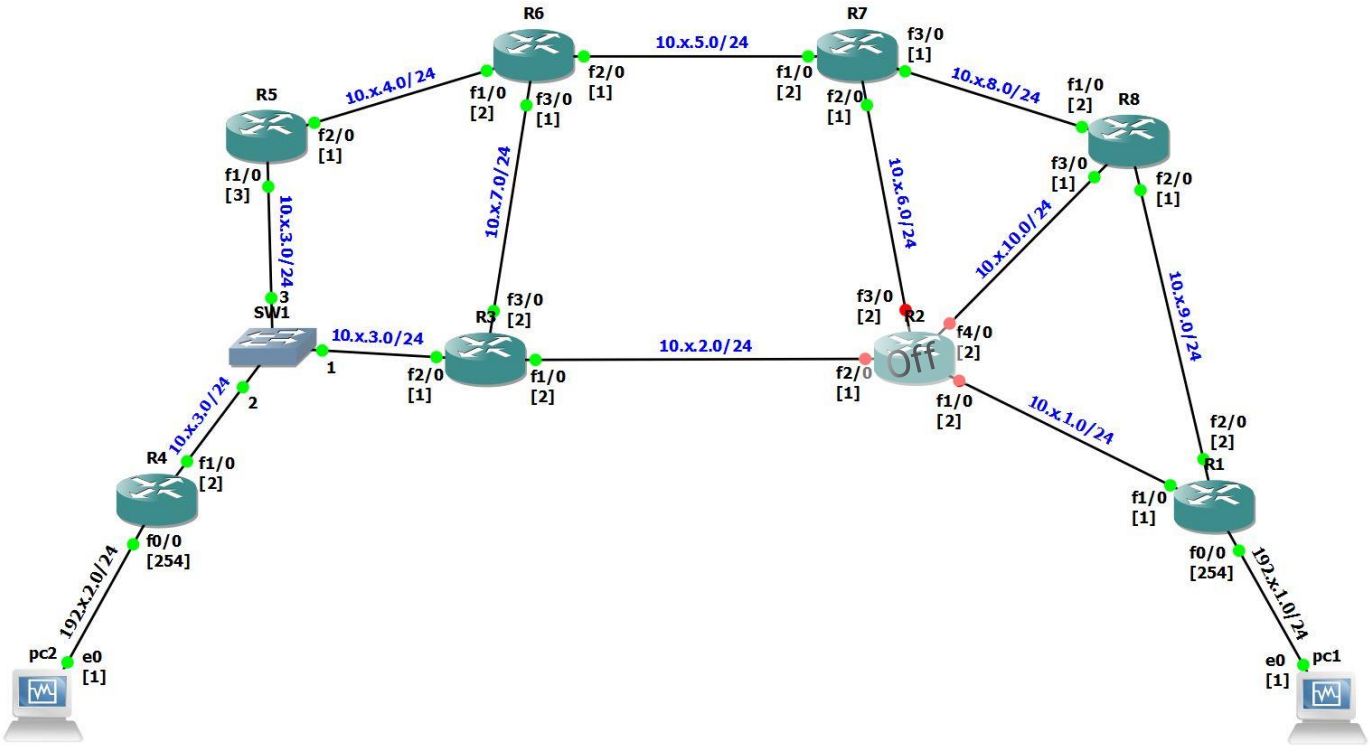
2.18. What information can we get about *RIP* configuration, from the output of the command "`show ip protocols`"?

2.19. Describe the differences between the *routing table* and the *RIP database*.
Base your answer on the outputs that you have saved.
In your answer, referred also to the way the tables are filled.

2.20. From the wireshark output, locate two *RIP* packets that sent after the network convergence (one from *R1* and one from *R2*).
Find whether there are identical entries (entries that advertise the same network) between the two packets.
Based on the routing tables of both routers, explain why it is happening.

2.21. Use the screenshot of the routing tables of R2, give a short explanation for each field in the routing table.

3. Measuring RIP version 2 Convergence Time



In this exercise, you measure the time it takes to *RIP version* 2 protocol to distribute a piece of information to the entire network. In other words, the protocol convergence time.

Pre

3.1. The exercise is based on *Topology 8*.
But first, save as a new project, before making any changes.

Do

3.2. Start all the network components, **except for R2**.

3.3. **Do not set** yet additional interfaces to R1 and R4.

3.4. Add and configure the PCs according to the figure.
Pay attention to the change in the network addresses.
Set appropriate default gateway for each PC.

3.5. Configure **only** the network interface *f0/0* of *R1* and set as *passive-interface*.
Add the new network to the RIP routing process.

```
R1#configure terminal
R1(config)#router rip
R1(config-router)#network 192.x.1.0
R1(config-router)#passive-interface FastEthernet 0/0
R1(config-router)#exit
R1(config)#interface FastEthernet 0/0
R1(config-if)#ip address 192.x.1.254 255.255.255.0
```

```
R1(config-if)#no shutdown
R1(config-if)#end
```

3.6. Use *traceroute* from *PC1* in order to verify that your network convened.
`pc1% traceroute 10.x.3.2`

3.7. Look at the *routing table* of *R1* and save the output.

3.8. Start Wireshark on *R1* interface *f2/0*, on *R8* interface *f1/0* and on *R4* interface *f1/0*.

3.9. Change the time display format of the Wireshark's time Column, to "Time of day".
Wireshark -> View -> Time Display Format -> Time Of Day.

3.10. Issue an infinite *ping* from *PC1* to *PC2*. You should see that the ping failed.

3.11. Similarly to *R1*, now configure the network interface *f0/0* of *R4* and set as *passive-interface*. Add the new network to the RIP routing process.

3.12. Wait for the network convergence. The ping should start working properly.

3.13. Stop the ping command using Ctrl+C.

3.14. Stop capture and save the wireshark pcaps to files.

3.15. Look again at the *routing table* of *R1* and save the output.

3.16. Perform "*write*" on routers *R1* and *R4*, and save the topology for later exercises.

Attach to your report

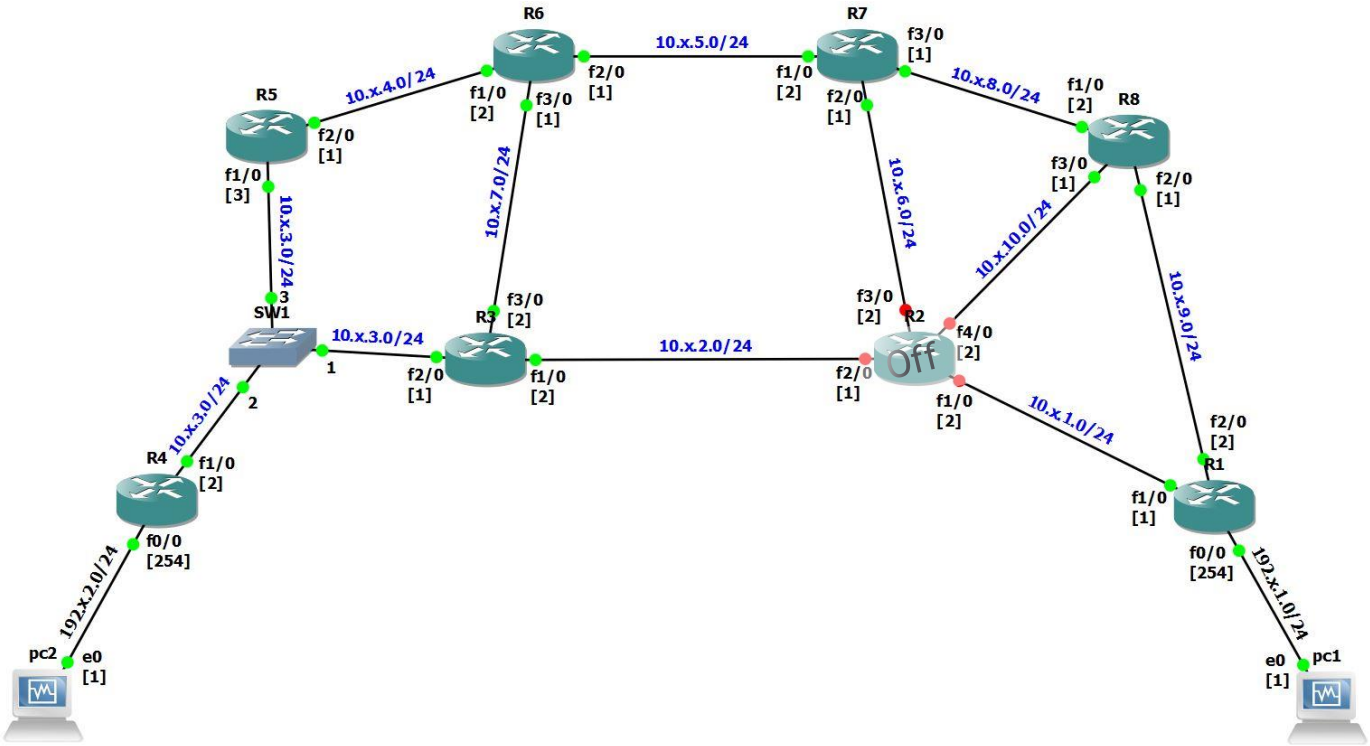
3.17. Explain why the ping failed at first (which router dropped the packets and why).

3.18. What is the reason to configure the interface as *passive-interface*?
What would happen if we did not configure it that way?
You can watch at link PC1-R1 and check it out.

3.19. At each link, locate the first packet that advertised the network 192.168.2.0/24.
Using the time of these packets, measure how long it takes the network to converge.
Meaning, how long it takes for *RIP* information to spread throughout the entire network, and how long it takes for the information to pass through a single hop (through *R8*).

3.20. Which mechanism responsible for the faster convergence, compared to the *basic RIP* (as you learned in the lecture)?
Explain how it works.

4. Good News Propagation (Adding a Router)



In this exercise, you are going to find out how fast "good news" propagates via *RIP version 2* protocol. You will add a new router to the network and watch how the network re-convenes.

Pre

4.1. Use the saved topology from the previous exercise "Measuring RIP version 2 Convergence Time".

Do

4.2. Start all the network components, **except for R2**.

4.3. Use *traceroute* from *PC2* to *PC1* in order to verify that your network convened, and save the output. (This probably will require several attempts until all the ARP tables will be filled.)

4.4. Start Wireshark on *R4* interface *f1/0*, on *R3* interface *f1/0* and on *R1* interface *f1/0*.

4.5. Issue an infinite *ping* from *PC2* to *PC1*.

4.6. Start *R2* router and wait for the network to converge.
Notice that when we start *R2*, we provide a better route between the two PCs.
Observe Wireshark on links *R3-R2* and *R2-R1* to see when the ICMP packets (the Request and the reply) start to pass through the new route.

4.7. Look at the routing table of *R1* and *R4* and verify that the network was convened.

4.8. Stop the ping command using Ctrl+C and save the ping statistics.

4.9. Stop capture and save the Wireshark pcaps to files.

4.10. Use *traceroute* again from *PC2* to *PC1* in order to see the change in the network, and save the output.

Attach to your report

4.11. Describe the packets route from *PC2* to *PC1* before and after starting *R2*.

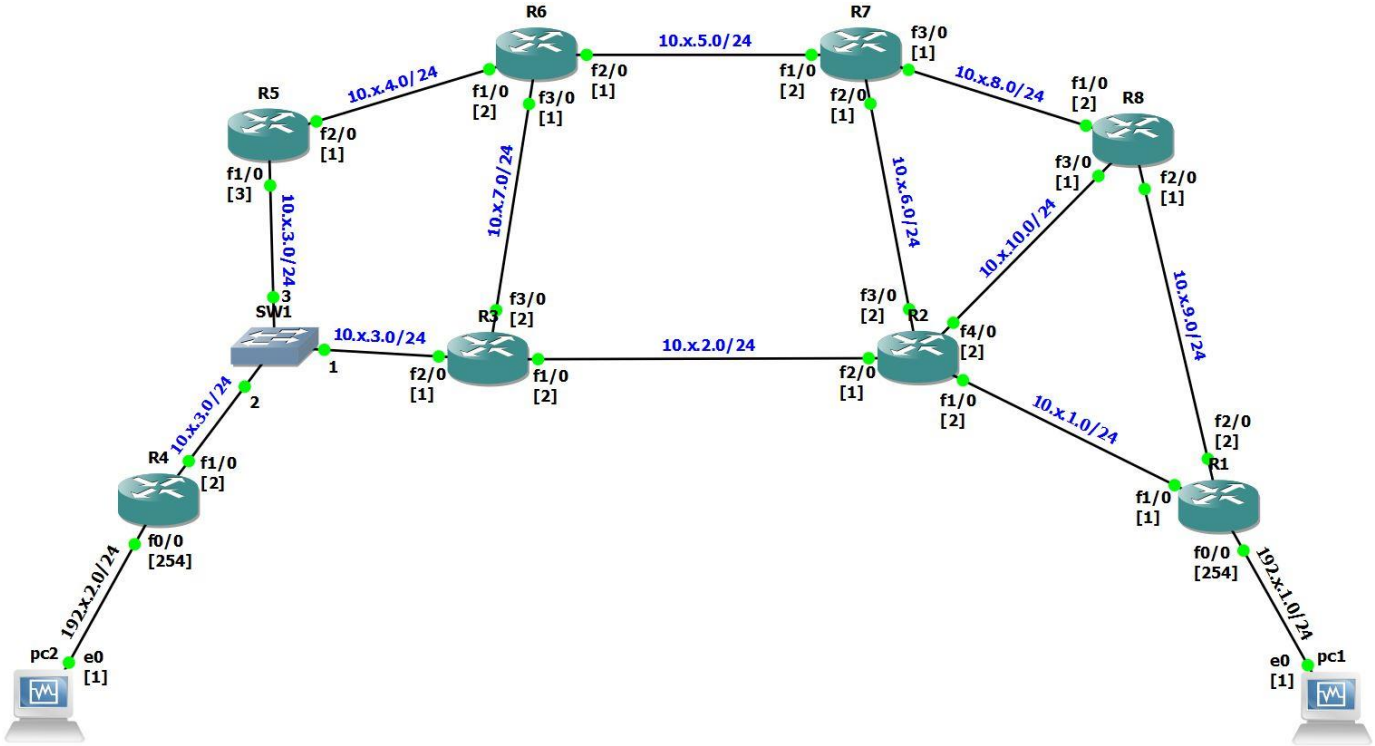
4.12. Is the network changes affected the ping application?
If so, how many packets were lost?

4.13. Using the Wireshark pcaps, measure how long it takes the network to converge.
Explain how you calculated it.

4.14. Locate the *RIP* packets, which directly affected the change in the ICMP packets route (those that the routing change is done right after them).
Attach screenshots to your report showing the situations.
For each screenshot, explain about the update that caused this change.

4.15. Based on the Wireshark Pcaps, describe the process that caused change in the route of the ICMP packets, after starting R2. Describe the updates process step by step.

5. Bad News Propagation (Router Down)



In this exercise, you are going to find out how fast (or slow) "bad news" propagates via *RIP version 2* protocol. You **turn off a router** and see how *RIP* reacts and how the network converges accordingly.

Pre

5.1. Use the saved topology from the previous exercise "Measuring RIP version 2 Convergence Time".

5.2. Read about the *RIP* protocol timers from the following link: http://www.cisco.com/c/en/us/td/docs/ios-xml/ios/iproute_rip/command/irr-cr-book/irr-cr-rip.html#wp2030003280

Do

5.3. Start all the network components.

5.4. Use *traceroute* from *PC2* to *PC1* in order to verify that your network convened, and save the output. (This probably will require several attempts until all the ARP tables will be filled.)

5.5. Start Wireshark on *R4* interface *f1/0*, on *R3* interface *f1/0* and on *R1* interface *f2/0*.

5.6. Issue an infinite *ping* from *PC2* to *PC1*.

5.7. Using your computer's clock (at the windows system), stop the router *R2* and write down the "stop time" (in resolution of seconds).

5.8. When the router stops, the ping fails. Wait for the network to converge until the ping returns to work properly. That will happen when an alternate path is found and may take several minutes.

5.9. Stop the ping command using Ctrl+C and save the ping statistics.

5.10. Stop capture and save the wireshark pcaps to files.

5.11. Use *traceroute* again from *PC2* to *PC1* in order to see the change in the network, and save the output.

Attach to your report

5.12. Describe the packets route from *PC2* to *PC1* before and after stopping *R2*.

5.13. Using the Wireshark pcaps and the "stop time" you wrote before, measure how long it takes the network to converge.
Explain how you calculated it.

5.14. At the link *R3-R2*, locate the packet indicating that *R3* realize that the connection with *R2* was lost.
Using this packet and the "stop time" you wrote before, measure how long it takes to *R3* realize that *R2* is down.

5.15. Using the ping statistics, calculate the time it took RIP to converge (Note that Ping packets are sent every one second).
Is this measure is consistent with the calculation you made before, about the convergence time?

5.16. Explain, what in the protocol supports the waiting time you calculated?

6. Bad News Propagation (Interface Down)

In this exercise, you will find out how the *RIP* protocol reacts for a different kind of "bad news" in the network. Now, you will **shut down an interface** and see how *RIP* reacts and how the network converges accordingly.

Pre

6.1. Use the saved topology from the previous exercise "Measuring RIP version 2 Convergence Time".

Do

6.2. Start all the network components.

6.3. Use *traceroute* from *PC2* to *PC1* in order to verify that your network convened, and save the output.

6.4. Issue an infinite *ping* from *PC2* to *PC1*.

6.5. Shut down the interface *f2/0* of router *R2*.

6.5.1. This is not done by shutting down the router or by removing the link in GNS3's topology!

6.5.2. Instead of the line "no shutdown" in the router configuration ,you should write " shutdown"

6.6. When the interface is shut down, the ping fails. Wait for the network to converge until the ping returns to work properly. That will happen when an alternate path is found and may take several minutes.

6.7. Stop the *ping* command using Ctrl+C and save the *ping* statistics.

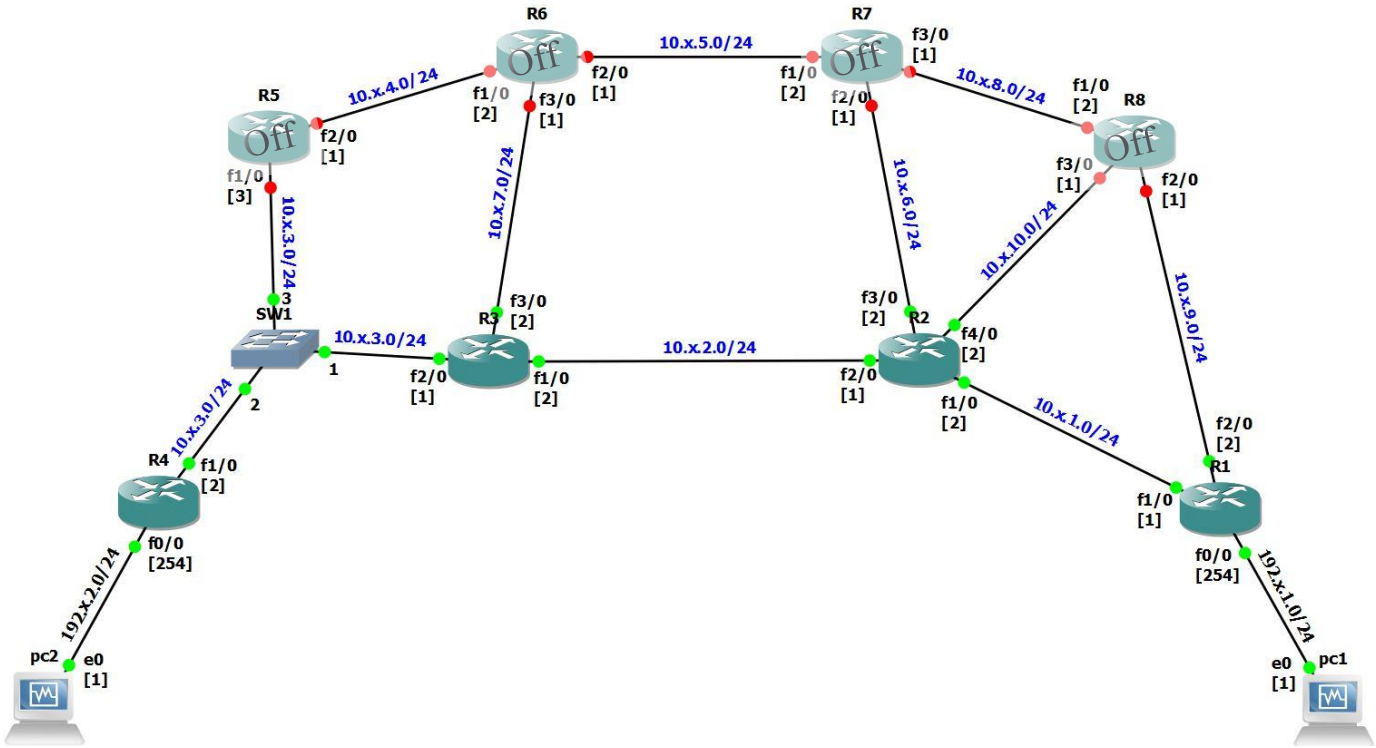
6.8. Use *traceroute* again from *PC2* to *PC1* in order to see the change in the network, and save the output.

Attach to your report

6.9. Using the *ping* statistics, calculate the time it took *RIP* to converge.

6.10. Describe the change in the route, before and after the experiment.

7. Mechanisms to prevent "Count to Infinity Problem"



In this exercise, you will see how the additional *RIP* protocol mechanisms (*triggered updates*, *split horizon* and *route poisoning*) are expressed. You will create a potential situation for *count-to-infinity problem* existence, and you will understand how the additional *RIP* mechanisms help to prevent this problem.

Pre

7.1. Use the saved topology from the previous exercise "Measuring RIP version 2 Convergence Time".

Do

7.2. **Start only** the following network components: PC1, PC2, R1, R2, R3, R4.

7.3. Use *traceroute* from *PC2* to *PC1* in order to verify that your network convened. (This probably will require several attempts until all the ARP tables will be filled.)

7.4. Start Wireshark on *R4* interface *f1/0*, on *R3* interface *f1/0* and on *R1* interface *f1/0*.

7.5. Issue an infinite *ping* from *PC2* to *PC1*.

7.6. Using your computer's clock (at the windows system), shut down the interface *f1/0* of *R2* (from the console) and write down the "Shutdown time" (in resolution of seconds).

7.7. The ping will fall. After several attempts that result in "Destination Host Unreachable", stop the ping.

7.8. Stop capture and save the wireshark pcaps to files.

Attach to your report

7.9. Assume, using the same network topology for the experiment you just did, but the *basic RIP* runs on the routers. '*Basic RIP*' refers to protocol without the mechanisms "*split horizon*", "*triggered updates*" and "*route poisoning*".
Describe a scenario that could lead to a situation of *count-to-infinity* on this network.

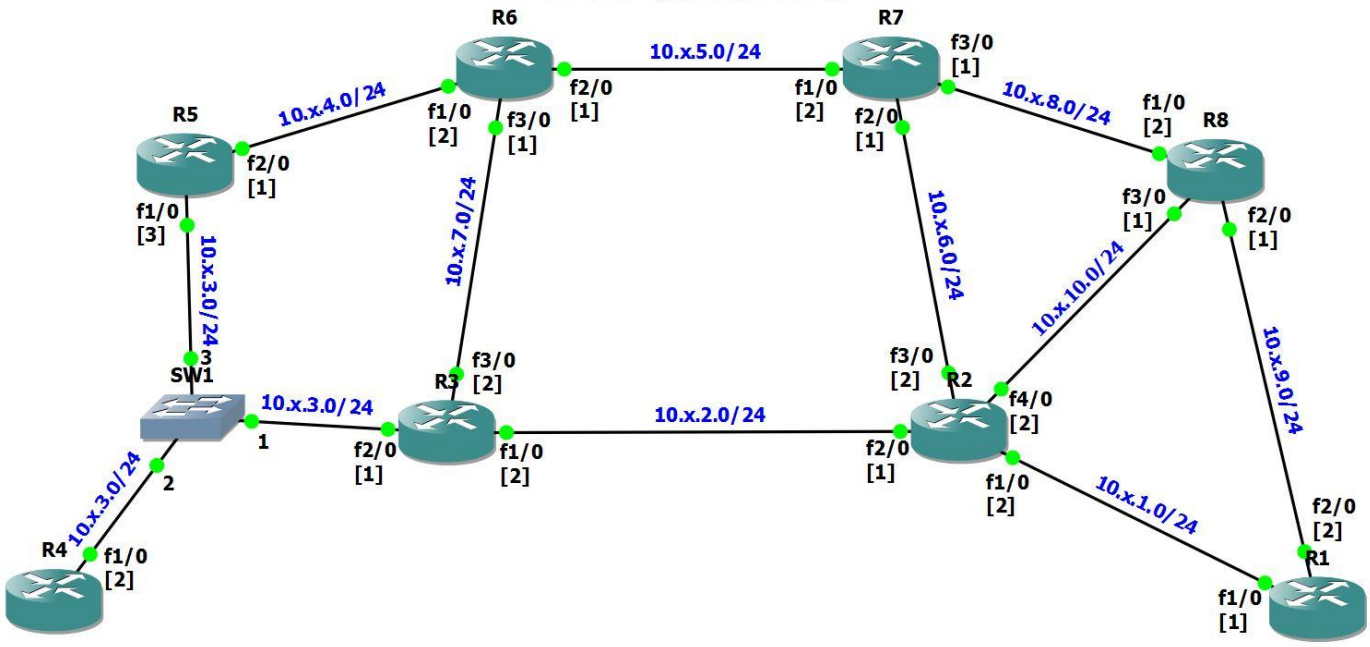
7.10. Locate on wireshark pcaps for the following situations:
Two situations caused by different reasons, in which can be seen the use of *triggered updates* mechanism.
A situation, in which can be seen the use of *split horizon* mechanism.
A situation, in which can be seen the use of *route poisoning* mechanism.
Describe each of these situations you have found and attach screenshots from Wireshark. Where it is relevant, specify the time intervals.

7.11. For each such situation, explain how the relevant mechanism contributes with prevention of *count-to-infinity* problem.

Intra AS Routing (Part B) : OSPF

8. Configuring OSPF on Cisco Routers

Topology 9 (OSPF)



Pre

8.1. The exercise is based on *Topology 7*.
But first, before making any changes, save as a new project as *Topology 9*.

8.2. Read about the concept of "*wildcard-mask*".
https://en.wikipedia.org/wiki/Wildcard_mask

Do

8.3. Start all the network components.

8.4. Start *Wireshark* on *R4* interface *f1/0*.

8.5. Enable *OSPF* on all routers, while each router *R?* is configured with the ***router-ID*** "*10.x.0.?*". Configure the routers starting from *R1* to *R8*.
The *OSPF* should advertise the network 10.x.0.0/16.
For example on *R1*:

```
R1#configure terminal
R1(config)#router ospf 1
R1(config-router)#router-id 10.x.0.1
R1(config-router)#network 10.x.0.0 0.0.255.255 area 1
R1(config-router)#end
```

8.6. Wait for the network convergence.

8.7. Stop capture and save the *Wireshark pcap* to file.

8.8. You should now be able to ping any router from any other router.
Test your network by using *ping* from some router to all the others.

8.9. **Perform** "*write*" on each router, save the topology (as *Topology 9*) and ZIP it.

8.10. Leave the network running, in order to answer the exercise questions.

Attach to your report

8.11. Explain the meaning of each command from step 5.

8.12. Which type of encapsulation is used for *OSPF* packets (*TCP*, *UDP* or other)?

8.13. Locate on the *Wireshark pcap* the largest *OSPF* packet from each type.
For each packet, explain what its purpose in the protocol.
For each packet, described the information that the protocol can learn from it (refer to the main fields).

8.14. The rationale behind this question is to understand how *OSPF* can see the entire network from the graph it built, according to the information it received from its neighbors.
Use all the following commands (and only those) to *trace* the route from *R1* to the network 10.x.3.0/24, using the information from *R1 OSPF database*.

```
show ip ospf database
show ip ospf database network network-ID
show ip ospf database router router-ID
```

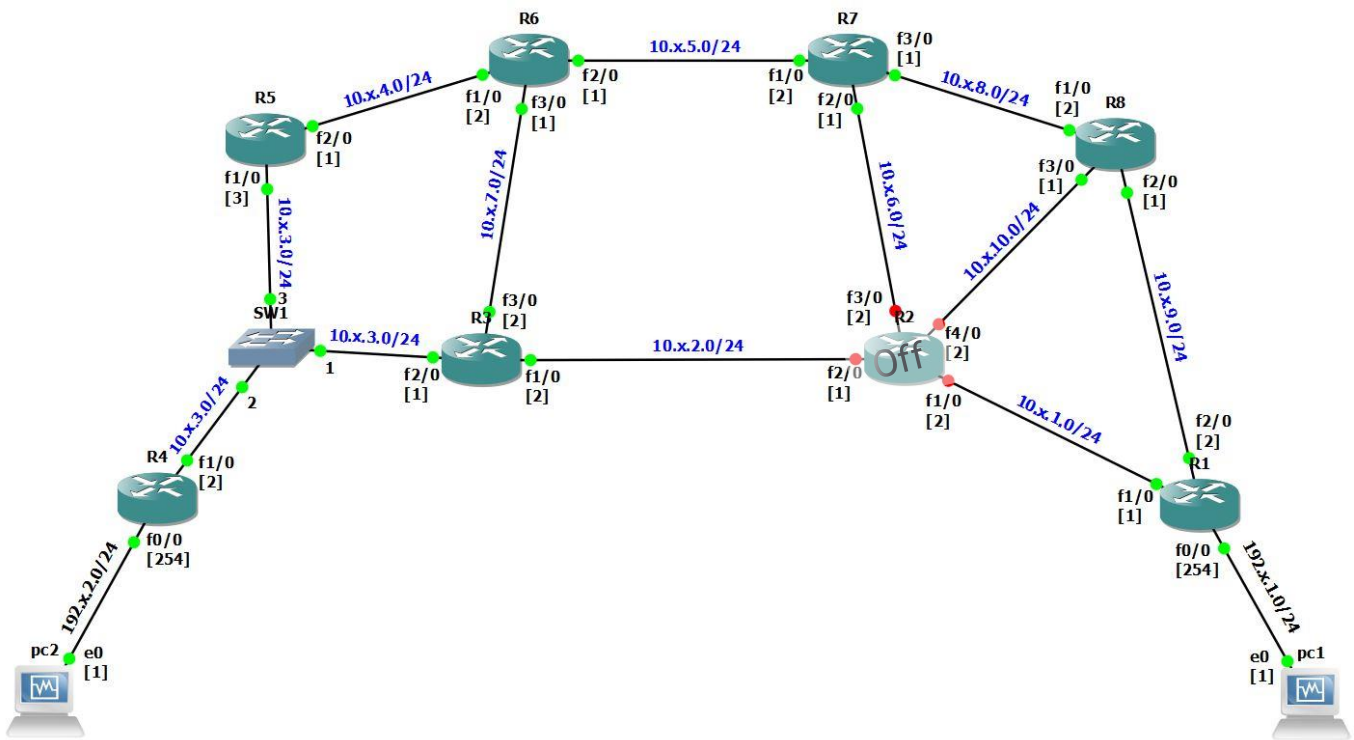
-What is the path?
-Explain how were you able to trace the path using the commands, add relevant screenshots.
- Explain what information is gained from each command and how did you use it in your trace and build the network's topology/graph?

(Hint: What are the `network-ID` and `router-ID` ?Use information gained from a command as input for the other two. You don't have to use the commands in any particular order.
you can use the information found in the following link to help you understand the output from each command: https://supportforums.cisco.com/t5/network-infrastructure-documents/reading-and-understanding-the-ospf-database/ta-p/3145995)

8.15. Is the *OSPF* database on all the routers identical? Why?

8.16. Attach to your report a screenshot of *R4 OSPF database* for a later exercise.
`R4#show ip ospf database`

9. Measuring OSPF Convergence Time



In this exercise, you measure the time it takes to *OSPF* protocol to distribute a piece of information to the entire network. In other words, the protocol convergence time.

Pre

9.1. The exercise is based on *Topology 9*.
But first, save as a new project, before making any changes.

Do

9.2. Start all the network components, **except for R2**.

9.3. **Do not set** yet additional interfaces to R1 and R4.

9.4. Add and configure the *PCs* according to the figure.
Set appropriate default gateway for each *PC*.

9.5. Configure only the network interface *f0/0* of *R1* and set as *passive-interface*.
Add the new network to the OSPF routing process.

```
R1#configure terminal
R1(config)#router ospf 1
R1(config-router)#network 192.x.1.0 0.0.0.255 area 1
R1(config-router)#passive-interface FastEthernet 0/0
R1(config-router)#interface FastEthernet 0/0
R1(config-if)#ip address 192.x.1.254 255.255.255.0
R1(config-if)#no shutdown
R1(config-if)#end
```

9.6. Use *traceroute* from *PC1* in order to verify that your network convened.

```
pc1% traceroute 10.x.3.2
```

9.7. Start Wireshark on *R1* interface *f2/0*, on *R8* interface *f1/0* and on *R4* interface *f1/0*.

9.8. Change the time display format of the *Wireshark's* time Column, to "Time of day".

9.9. Issue an infinite *ping* from *PC1* to *PC2*. You should see that the ping failed.

9.10. Similarly to *R1*, now configure the network interface *f0/0* of *R4* and set as *passive-interface*. Add the new network to the OSPF routing process.

9.11. Wait for the network convergence. The ping should start working properly.

9.12. Stop the ping command using Ctrl+C.

9.13. Stop capture and save the *wireshark pcaps* to files.

9.14. Perform "`write`" on routers *R1* and *R4*, and save the topology for later exercises.

9.14.1. Keep the project open for question 9.17 .

Attach to your report

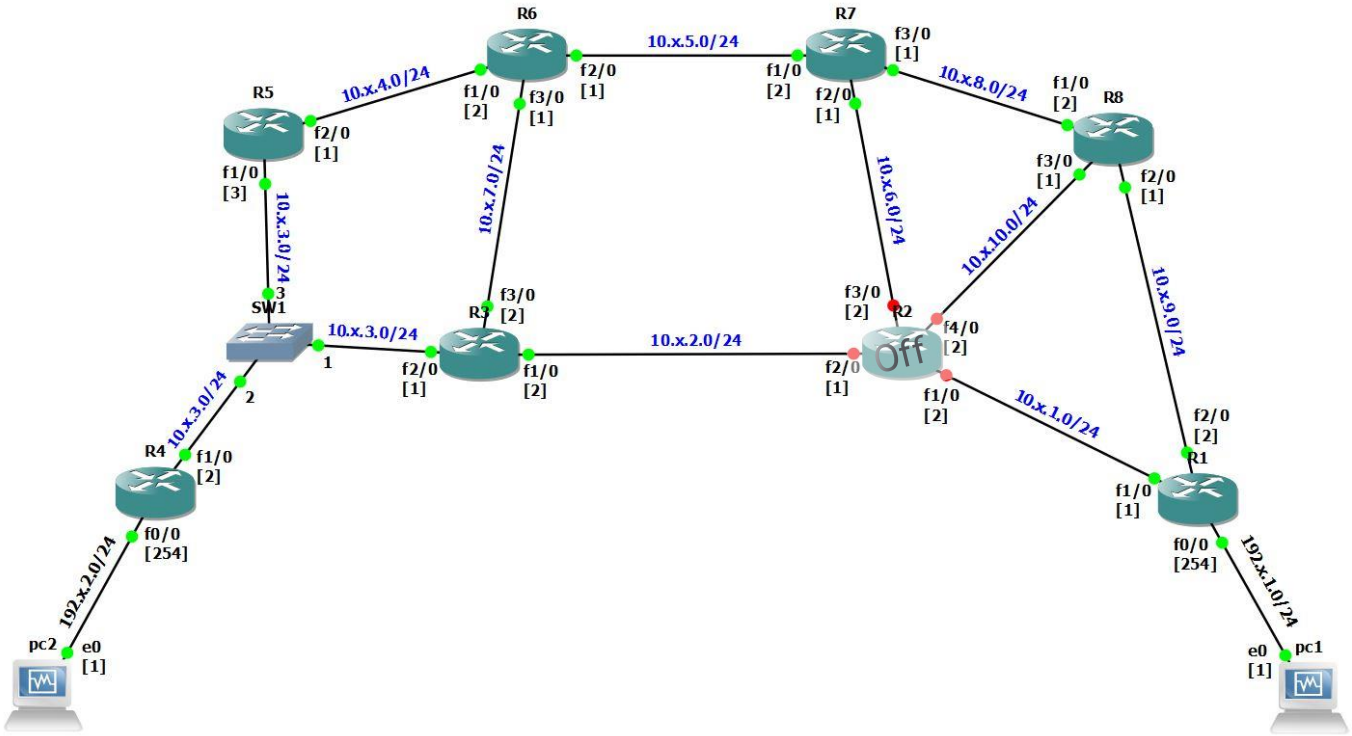
9.15. Describe the *OSPF* distribution process of the information about the new added subnet. Pay attention to times and to the chronological order of the packets. Refer to information that the packets contain.

9.16. Measure how long it takes for *OSPF* information to spread throughout the entire network, and how long it takes for the information to pass through a single hop.

9.17. Find out how *OSPF* calculates the *metric* for an interface.
Explain what causes the difference in *metric* between the advertised entries.
For that purpose, use the following command and explain how it can help to find the reason.

```
show interfaces FastEthernet interface-number
```

10. Good News Propagation (Adding a Router)



In this exercise, you are going to find out how fast "good news" propagates via *OSPF* protocol. You will add a new router to the network and watch how the network re-convenes.

Pre

10.1. Use the saved topology from the previous exercise "Measuring OSPF Convergence Time".

Do

10.2. Start all the network components, **except for R2**.

10.3. Use *traceroute* from *PC2* to *PC1* in order to verify that your network convened.

10.4. Start *Wireshark* on *R4* interface *f1/0*, on *R3* interface *f1/0* and on *R1* interface *f1/0*.

10.5. Issue an infinite *ping* from *PC2* to *PC1*.

10.6. Start router *R2* and wait for the network to converge.
Notice that when we start *R2*, we provide a better route between the two *PCs*.
Observe *Wireshark* on links *R3-R2* and *R2-R1* to see when the ICMP packets (the Request and the reply) start to pass through the new route.

10.7. Look at the routing table of *R1* and *R4* and verify that the network was convened.

10.8. Stop the ping command using Ctrl+C and save the ping statistics.

10.9. Stop capture and save the *wireshark pcaps* to files.

Attach to your report

10.10. Did the network changes affect the ping application?
If so, how many packets were lost?

10.11. Using the *Wireshark pcaps*, measure how long it takes the network to converge.
Explain, how did you calculate this?

10.12. At the pcap of subnet 10.x.3.0, follow the update packets that spread in the network because the router started.
Use the display filter: "`ospf.msg.lsupdate and ip.dst == 224.0.0.5`".
Describe the update packets which spread in the network, and the reason for each one.

11. Bad News Propagation (Router Down)

Now, you **turn off a router** and see how *OSPF* reacts and how the network converges accordingly.

Pre

11.1. Use the saved topology from the previous exercise "Measuring OSPF Convergence Time".

11.2. Read about some *OSPF* protocol timers (*hello-interval* and *dead-interval*).
From the following links:
http://www.cisco.com/c/en/us/td/docs/ios-xml/ios/iproute_ospf/command/iro-cr-book/ospf-i1.html#wp4134450560

http://www.cisco.com/c/en/us/td/docs/ios-xml/ios/iproute_ospf/command/iro-cr-book/ospf-a1.html#wp2917383021

Do

11.3. Start all the network components.

11.4. Use *traceroute* from *PC2* to *PC1* in order to verify that your network convened.

11.5. Issue an infinite *ping* from *PC2* to *PC1*.

11.6. Stop the router *R2*.

11.7. When the router stops, the ping fails. Wait for the network to converge until the ping returns to work properly. That will happen when an alternate path is found and may take several minutes.

11.8. Stop the ping command using Ctrl+C and save the ping statistics.

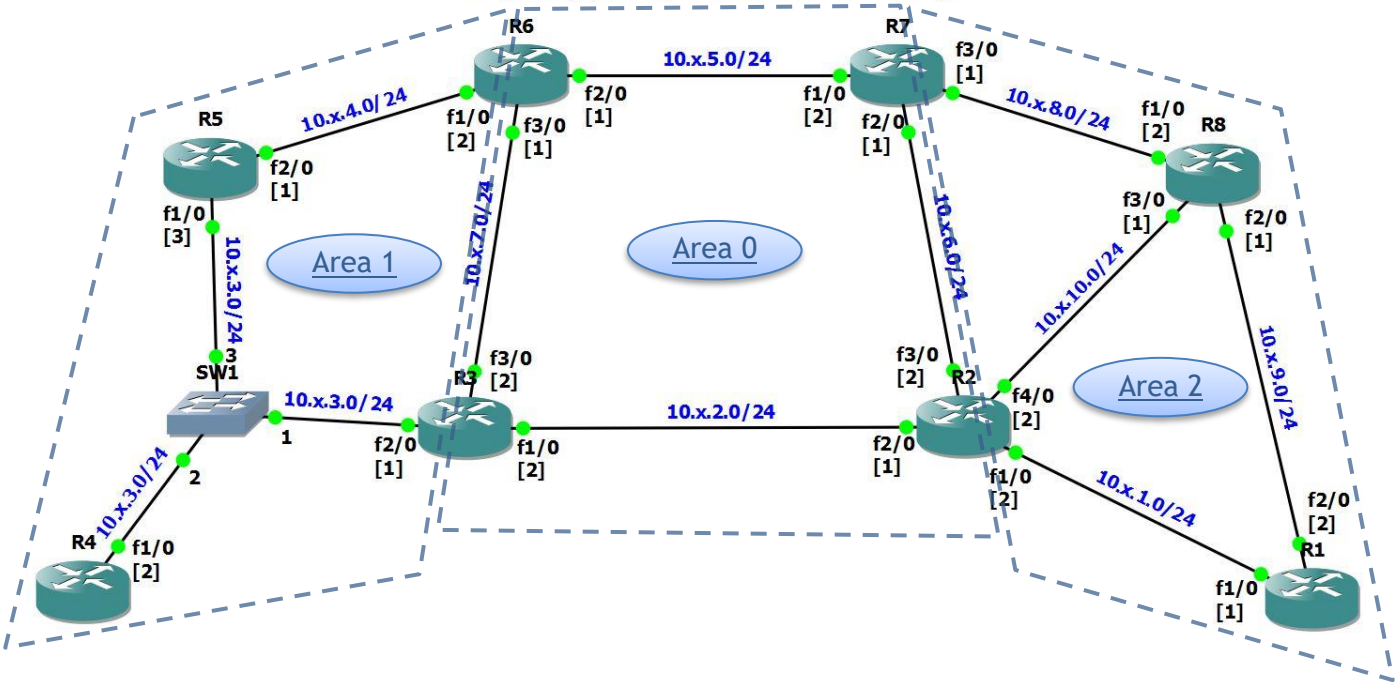
Attach to your report

11.9. Using the ping statistics, calculate the time it took *OSPF* to converge.

11.10. Explain, what in the protocol supports the waiting time you calculated?

12. Hierarchical Routing in OSPF

Topology 10 (Hierarchical OSPF)



In *OSPF*, networks can be divided into areas, to significantly reduce the amount of topological information routers have to learn. In *OSPF* all areas must be connected to *Area 0* which is known as the *backbone* area.

You will now define *Hierarchical OSPF* with two areas (*area 1* and *area 2*) connected through a *backbone* area (*area 0*). Routers that connect two areas are called *area border routers*.

Pre

12.1. The exercise is based on *Topology 7*.
But first, before making any changes, save as a new project as *Topology 10*.

Do

12.2. Start *Wireshark* on *R4* interface *f1/0*.

12.3. Enable *OSPF* on all routers, while each router *R?* configured with the `router-ID` "`10.x.0.?`". Configure the routers in this order: *R4, R5, R3, R6, R7, R2, R8, R1*.
Note that the configuration of the border routers is a little different.
Follow the examples.

12.4. Example for a router inside *area 1*:

```
R4#configure terminal
R4(config)#router ospf 1
R4(config-router)#router-id 10.x.0.4
R4(config-router)#network 10.x.0.0 0.0.255.255 area 1
R4(config-router)#end
```

12.5. Example for a border router:

```
R3#configure terminal
R3(config)#router ospf 1
R3(config-router)#router-id 10.x.0.3
R3(config-router)#network 10.x.3.0 0.0.0.255 area 1
R3(config-router)#network 10.x.2.0 0.0.0.255 area 0
R3(config-router)#network 10.x.7.0 0.0.0.255 area 0
R3(config-router)#end
```

12.6. Wait for the network convergence.

12.7. Stop capture and save the *wireshark pcap* to file.

12.8. You should now be able to ping any router from any other router.
Test your network by using *ping* from some router to all the others.

12.9. **Perform** "`write`" on each router, save the topology (as *Topology 10*) and ZIP it.

12.10. Leave the network running, in order to answer the exercise questions.

Attach to your report

Look at the *Wireshark pcap* using the display filter:
"`ospf.msg.lsupdate and ip.dst == 224.0.0.5`".

12.11. Using the *Wireshark pcap*, describe which information routers in area 1 received about area 2 and which information they received about the backbone area.

12.12. Locate on the *Wireshark pcap* the *Router-LSA* update that *R3* advertised.
Which interfaces it advertises? Why?

12.13. Display the *OSPF database* of *R4* and compare it to the one you saved earlier, at exercise "Configuring OSPF on Cisco Routers". Discuss the differences.

12.14. The idea behind this question is to understand how *OSPF* can 'see' the entire network from the graph it built, according to the information it received from its neighbors. Use <u>all</u> the following commands (and only those) to *trace* the from *R4* to the network 10.x.1.0/24, using the information from *R4 OSPF database*. The visible route for *R4*.

```
show ip ospf database
show ip ospf database network network-ID
show ip ospf database summary network-ID
show ip ospf database router router-ID
```

-What is the path?

-Explain how were you able to trace the path using the commands, add relevant screenshots.

- Explain what information is gained from each command and how did you use it in your trace?

12.15. Display the *OSPF database* of routers *R4* and *R7*, and discuss the differences in the *OSPF database* of regular routers and border routers.

12.16. What are the advantage and disadvantages of *Hierarchical OSPF* relative to *OSPF*?