

Redes de Computadoras 2021

TP2: Ruteo Interno Dinamico

Estudiantes:

Germán Lamberti (IComp - 38279109 - german.lamberti@unc.edu.ar)

Martin Lujan (IComp - 39448179 - martin.lujan@mi.unc.edu.ar)

Jose Piro (IComp - xxxxxxxx - josemanuelpiro@mi.unc.edu.ar)

Esteban Perez (IComp - 39026980 - esteban.perez@mi.unc.edu.ar)

21 de Abril, 2021



Objetivos

Ruteo estático, tablas de ruteo internas de los hosts. Ruteo dinámico, revisión de los algoritmos, revisión de los protocolos. Implementación de caso de uso con Packet Tracer. Implementación de caso de uso para IPv4 con containers.

Requisitos

- Computadora por cada 2 personas

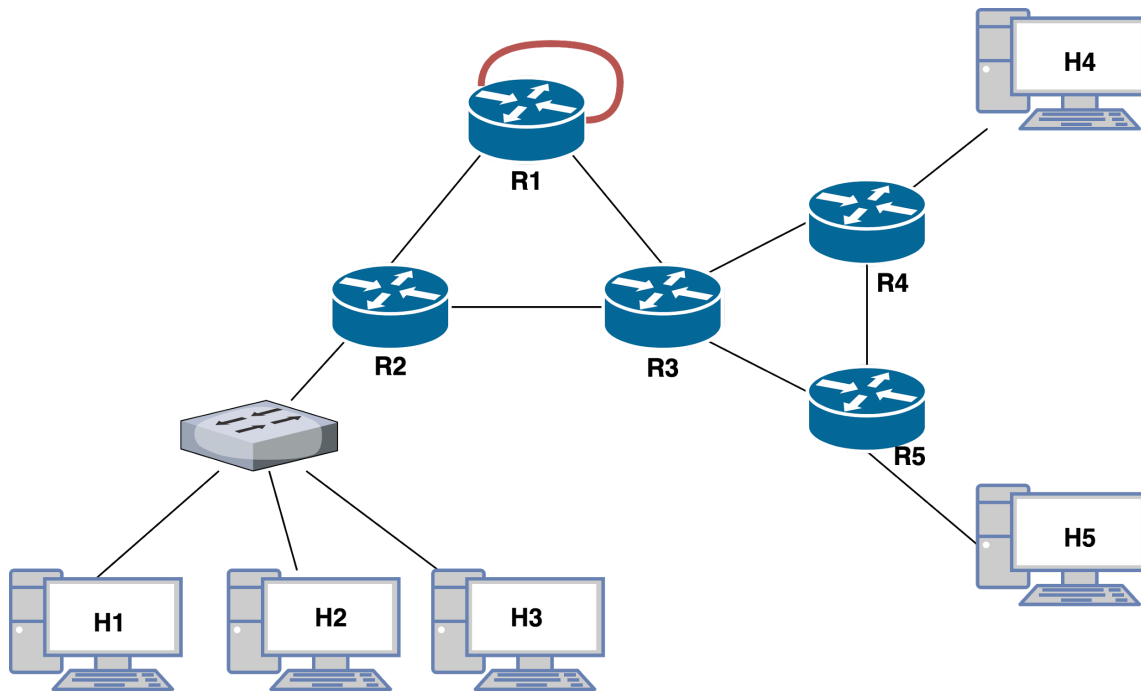
Consignas

Ruteo dinamico OSPF con docker-compose

Recomendaciones

- Lea con cuidado las consignas
- Tenga certeza de los comandos que ejecuta
- Realizar todas las configuraciones para IPv4

Diagrama de red



Consignas

1. Modificar el archivo docker-compose para replicar la topología definida en el diagrama.
2. Crear y modificar los archivos de configuración para cada router.
3. Configurar cada router para que funcione OSPF.
 - a. Probar interconexión entre los distintos puntos y verificar que las tablas de ruteo de los routers muestran las rutas OSPF.
4. Identificar y Analizar los mensajes de OSPF.
5. Controlar la elección del router designado. Describir el método utilizado.
6. Configurar OSPF en los routers.
 - a. Configurar el router para notificar las redes que están conectadas directamente.
 - b. Leer las entradas de las LSDB en cada uno de los routers.
7. Definir las áreas. R1 y R2 están en el área A. El resto en el área B.
 - a. Leer las entradas de las LSDB en cada uno de los routers
8. Verificar el funcionamiento de OSPF
 - a. En el router R2 consultar la información acerca de los vecinos R1 y R3 de OSPF.
 - b. En el router R2 ver información sobre las operaciones del protocolo de enrutamiento.
9. Configurar el costo de OSPF. Modificar los costos de las rutas de manera tal que el funcionamiento se modifique.

10. Redistribuir una ruta OSPF predeterminada
 - a. Configurar una dirección de loopback en R1 para simular un enlace a un ISP.
 - b. Configurar una ruta estática predeterminada en el router R1
 - c. Incluir la ruta estática en las actualizaciones de OSPF que se envían desde el router R1
11. Explicar que sucede en toda la red si se cae una interfaz del router R2.
12. Es lo mismo la tabla RIB (Routing Information Base) que la tabla FIB(Forwarding Information Base)? Justificar con capturas del práctico.

Links de ayuda

Configuración de OSPF

- <http://docs.frrouting.org/en/latest/>
- http://www.cisco.com/c/en/us/td/docs/ios-xml/ios/iproute_ospf/configuration/12-4t/iro-12-4t-book/iro-cfg.html
- [Configuración de OSPF en CISCO en ipv4 e ipv6](#)

Instalación de Docker CE:

- <https://docs.google.com/document/d/1TRYoo9j6BrvZqy7tFOMqfFEEIkYwXAI-sj3hJVFwIPQ/edit#>

Quagga y otros software OpenSource para ruteo:

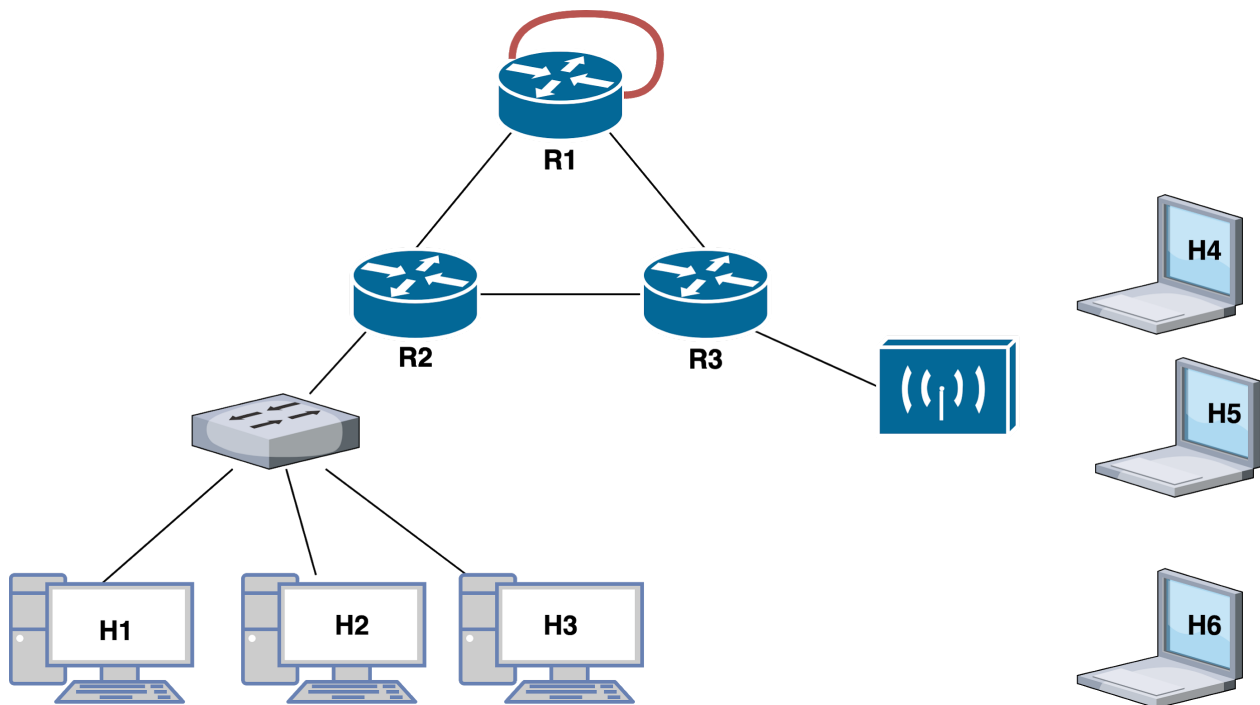
- <https://keepingitclassless.net/2015/05/open-source-routing-comparison/>

Ruteo dinámico OSPF con Cisco Packet Tracer

Recomendaciones


- Lea con cuidado las consignas
- Tenga certeza de los comandos que ejecuta
- Realizar todas las configuraciones para IPv6

Diagrama de red



Consignas

1. Configurar interfaces de los routers
2. Configurar OSPF en el router R1, R2 y R3 para IPv6
 - a. Configurar el router para notificar las redes que están conectadas directamente.
 - b. Definir las áreas. R1 y R2 están en área A. R3 está en área B.
3. Configurar el costo de OSPF
 - a. Modificar los costos de las rutas de manera tal que el funcionamiento se modifique

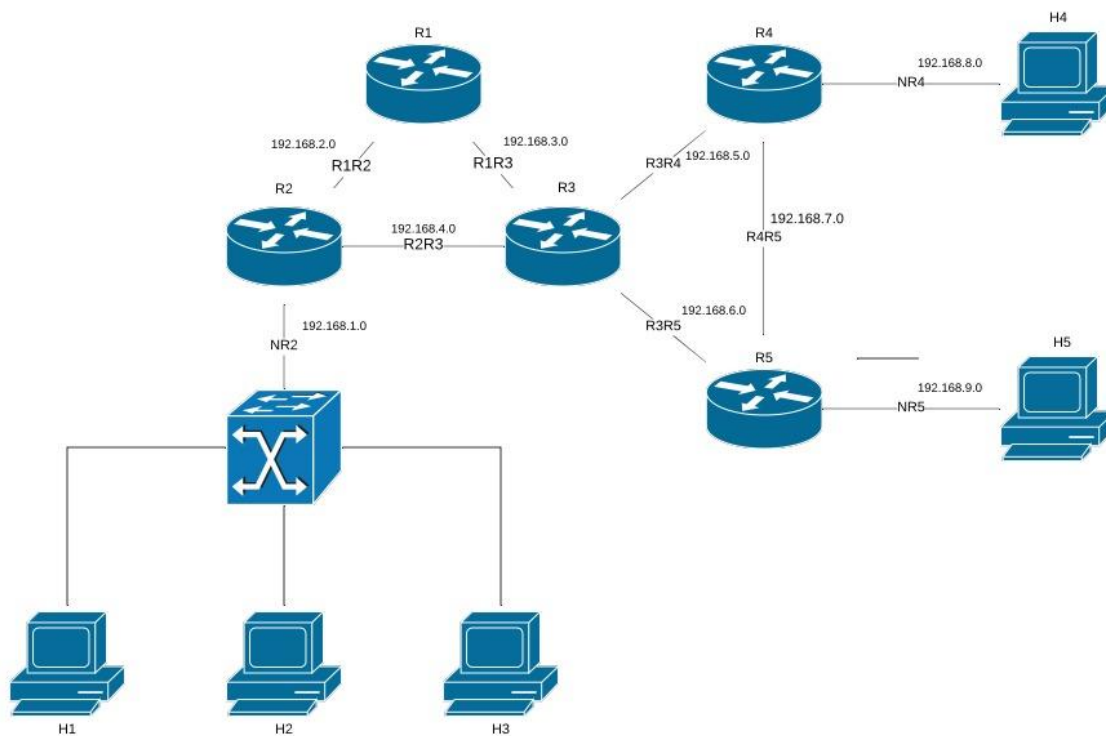
- 
- b. Realizar pruebas entre los clientes de los distintos routers verificando el funcionamiento con traceroute antes y después de la modificación.
 - 4. Redistribuir una ruta OSPF predeterminada
 - a. Configurar una dirección de loopback en R1 para simular un enlace a un ISP.
 - b. Configurar una ruta estática predeterminada en el router R1
 - c. Incluir la ruta estática en las actualizaciones de OSPF que se envían desde el router R1
 - 5. Enumerar diferencias entre OSPF y OSPFv3

Parte 1:

Introducción

Desarrollo

1. Compartimos la topología de la red solicitada, en un archivo docker-compose.yml



2. Detallamos las redes y las direcciones ip de las interfaces:

Redes:

- oam1: 192.168.0.0
- oam2: 192.168.1.0
- oam3: 192.168.2.0
- oam4: 192.168.3.0

- oam5: 192.168.4.0
- oam6: 192.168.5.0

Interfaces:

- R1:
 - eth0: 192.168.2.10
 - eth1: 192.168.3.10
- R2:
 - eth0: 192.168.1.10
 - eth1: 192.168.4.10
 - eth2: 192.168.2.11
- R3:
 - eth0: 192.168.4.11
 - eth1: 192.168.3.11
 - eth2: 192.168.5.10
 - eth3: 192.168.6.10
- R4:
 - eth0: 192.168.5.11
 - eth1: 192.168.7.10
 - eth2: 192.168.8.10
- R5:
 - eth0: 192.168.9.10
 - eth1: 192.168.7.11
 - eth2: 192.168.6.11

3. Configuramos los 5 Routers a través de OSPF para garantizar la conectividad entre los distintos puntos de la red. A continuación, compartimos la tabla de ruteo ospf en el presente informe, puntualmente el router 3.

Tabla de ruteo ospf del Router 3

```
r3# show ip ospf route
===== OSPF network routing table =====
N    192.168.1.0/24      [20] area: 0.0.0.1
      via 192.168.4.10, eth1
N    192.168.2.0/24      [20] area: 0.0.0.1
      via 192.168.3.10, eth0
      via 192.168.4.10, eth1
N    192.168.3.0/24      [10] area: 0.0.0.1
      directly attached to eth0
N    192.168.4.0/24      [10] area: 0.0.0.1
      directly attached to eth1
N    192.168.5.0/24      [10] area: 0.0.0.1
      directly attached to eth2
N    192.168.6.0/24      [10] area: 0.0.0.1
      directly attached to eth3
N    192.168.7.0/24      [20] area: 0.0.0.1
      via 192.168.6.11, eth3
      via 192.168.5.11, eth2
N    192.168.8.0/24      [20] area: 0.0.0.1
      via 192.168.5.11, eth2
N    192.168.9.0/24      [20] area: 0.0.0.1
      via 192.168.6.11, eth3
```

Para lograr esto, nos conectamos al router 3 vía comando telnet localhost 10018 por su puerta correspondiente a OSPF, y ejecutamos *sh ip ospf route* obteniendo tabla de ruteo OSPF.

Al comprobar la conectividad entre los distintos puntos de la red, realizamos el envío de paquetes ICMPv4 por la red.

A. Traceroute con origen host 1 y destino Host 4, Host 5.

```
docker exec -ti protocoloospf h1 1 traceroute 192.168.8.11
traceroute to 192.168.8.11 (192.168.8.11), 30 hops max, 46 byte packets
 1 192.168.1.1 (192.168.1.1)  0.005 ms  protocoloospf_r2 1.protocoloospf_nr2 (192.168.1.10)  0.004 ms  192.168.1.1 (192.168.1.1)  0.003 ms
 2 192.168.4.11 (192.168.4.11)  0.005 ms  *  0.005 ms
 3 192.168.5.11 (192.168.5.11)  0.002 ms  0.005 ms  0.002 ms
 4 192.168.8.11 (192.168.8.11)  0.002 ms  *  0.004 ms

docker exec -ti protocoloospf h1 1 traceroute 192.168.9.11
traceroute to 192.168.9.11 (192.168.9.11), 30 hops max, 46 byte packets
 1 192.168.1.1 (192.168.1.1)  0.003 ms  protocoloospf_r2 1.protocoloospf_nr2 (192.168.1.10)  0.005 ms  192.168.1.1 (192.168.1.1)  0.003 ms
 2 192.168.4.11 (192.168.4.11)  0.004 ms  *  0.007 ms
 3 192.168.6.11 (192.168.6.11)  0.002 ms  0.007 ms  0.002 ms
 4 192.168.9.11 (192.168.9.11)  0.002 ms  *  0.206 ms
```

4. Al momento de configurar las redes, interfaces y routers, a través de wireshark pudimos ver los **mensajes del protocolo OSPF** que ocurrían en la red.

69087	55.570779037	192.168.4.10	224.0.0.5	OSPF	80 LS Acknowledge
69088	55.570773896	192.168.4.10	224.0.0.5	OSPF	80 LS Acknowledge
69292	57.796528882	192.168.8.10	224.0.0.5	OSPF	80 Hello Packet
69293	57.796543876	192.168.8.10	224.0.0.5	OSPF	80 Hello Packet
69294	57.796528882	192.168.8.10	224.0.0.5	OSPF	80 Hello Packet
69295	57.796565547	192.168.5.11	224.0.0.5	OSPF	84 Hello Packet
69296	57.796570827	192.168.5.11	224.0.0.5	OSPF	84 Hello Packet
69297	57.796565547	192.168.5.11	224.0.0.5	OSPF	84 Hello Packet
69298	57.796591468	192.168.7.10	224.0.0.5	OSPF	84 Hello Packet
69299	57.796595668	192.168.7.10	224.0.0.5	OSPF	84 Hello Packet
69300	57.796591468	192.168.7.10	224.0.0.5	OSPF	84 Hello Packet
71978	59.589904305	192.168.4.11	192.168.4.10	OSPF	112 LS Update
71979	59.589925226	192.168.4.11	192.168.4.10	OSPF	112 LS Update
71980	59.589979022	192.168.4.11	192.168.4.10	OSPF	72 LS Request
71981	59.589985540	192.168.4.11	192.168.4.10	OSPF	72 LS Request
71982	59.590021460	192.168.4.10	192.168.4.11	OSPF	112 LS Update
71983	59.590029751	192.168.4.10	192.168.4.11	OSPF	112 LS Update
71984	59.590078589	192.168.4.10	224.0.0.5	OSPF	124 LS Update

65575	49.630358708	192.168.3.10	224.0.0.5	OSPF	84 Hello Packet
65862	51.564208925	192.168.2.11	192.168.2.10	OSPF	68 DB Description
65863	51.564224502	192.168.2.11	192.168.2.10	OSPF	68 DB Description
65864	51.564257878	192.168.4.10	192.168.4.11	OSPF	68 DB Description
65865	51.564262351	192.168.4.10	192.168.4.11	OSPF	68 DB Description
65866	51.564325835	192.168.2.10	192.168.2.11	OSPF	88 DB Description

Como podemos apreciar se pueden ver distintos tipos de paquetes como por ejemplo paquetes Hello, Database Description (DBD), Link-State Request (LSR), Link-State Update (LSU) y Link-State Acknowledge (LSAck), los cuales se detallaran a continuación.

- Los **paquetes Hello** son los primeros mensajes enviados por cada router que trabaja con OSPF, el cual es utilizado para formar adyacencias; el protocolo Hello indica que cuando un router recibe un mensaje Hello de su vecino, y este contiene el nombre del router local en la sección Neighbor, se posee una comunicación bidireccional entre ambos (comunicación o estado two-way).
- Los **paquetes DBD** son intercambiados una vez se tenga una comunicación bidireccional entre routers OSPF, donde estos mensajes transportan las cabeceras de los LSA. Los routers en este momento definen quien será el router designado.
- Los **paquetes LSR** son usados para consultar la información completa de un LSA que no se conozca o que se encuentre desactualizado. Los LSAs a consultar son los aprendidos en el intercambio de los paquetes DBD.
- Los **paquetes LSU** inundan con los LSAs consultados por el vecino OSPF, donde un solo LSU puede contener múltiples LSAs. Las redes que soportan multicast, utilizan dicha dirección para enviar los LSUs.

- Los **paquetes LSAck** son respuestas a los LSUs; es necesario que la inundación de LSAs sea confiable, por esto, se necesita un mecanismo de acuse de recibo para saber si los LSAs llegaron al destino.

Como podemos ver el verdadero propósito del protocolo OSPF es informar a los routers de la composición de la red y mantenerlos actualizados en todo momento.

5. En la transmisión de los paquetes DBD mencionamos la designación de un **Router Designado (DR)**, el cual se trata del router que representa en la red el punto de recolección y distribución de las LSA (Link State Advertisement) enviadas y recibidas, para de esa forma evitar la saturación de LSA's en toda la red. Normalmente se usa como router designado al de mayor número ID que en nuestro caso es el Router 4.

```
r4# show ip ospf neighbor
```

Neighbor ID	Pri	State	Dead Time	Address	Interface	RXmtL	RqstL	DBsmL
3.3.3.3	1	Full/Backup	32.999s	192.168.5.10	eth1:192.168.5.11	0	0	0
5.5.5.5	1	Full/DR	32.471s	192.168.7.11	eth2:192.168.7.10	0	0	0

También se elige una especie de router secundario, el cual se conoce como Router de Respaldo o Backup (BDR), el cual supervisa al DR y lo reemplaza en caso de falla de este. Todos los demás routers se conocerán como “otros” , los cuales sólo forman adyacencias completas con el DR y el BDR en la red.

6. Configurado OSPF en los routers, analizamos las tablas de la base de datos de cada router, aun sin agregar las áreas 1 y 2, es decir todos los routers se encuentran en el área 0.

Tabla database de router 2

```
OSPF Router with ID (2.2.2.2)
```

Router Link States (Area 0.0.0.1)					
Link ID	ADV Router	Age	Seq#	CkSum	Link count
1.1.1.1	1.1.1.1	171	0x80000007	0x9aae	2
2.2.2.2	2.2.2.2	169	0x80000009	0x9522	3
3.3.3.3	3.3.3.3	170	0x8000000e	0x70a3	4
4.4.4.4	4.4.4.4	171	0x8000000a	0x97fb	3
5.5.5.5	5.5.5.5	175	0x80000009	0xaadd	3

Net Link States (Area 0.0.0.1)					
Link ID	ADV Router	Age	Seq#	CkSum	
192.168.2.11	2.2.2.2	169	0x80000001	0x953a	
192.168.3.11	3.3.3.3	175	0x80000001	0x9e28	
192.168.4.11	3.3.3.3	175	0x80000001	0xb50c	
192.168.5.11	4.4.4.4	177	0x80000001	0xe0d3	
192.168.6.11	5.5.5.5	176	0x80000001	0xe9c1	
192.168.7.11	5.5.5.5	181	0x80000001	0x01a5	

Tabla database de router 5

OSPF Router with ID (5.5.5.5)						
Router Link States (Area 0.0.0.1)						
Link ID	ADV Router	Age	Seq#	CkSum	Link count	
1.1.1.1	1.1.1.1	234	0x80000007	0x9aae	2	
2.2.2.2	2.2.2.2	234	0x80000009	0x9522	3	
3.3.3.3	3.3.3.3	233	0x8000000e	0x70a3	4	
4.4.4.4	4.4.4.4	233	0x8000000a	0x97fb	3	
5.5.5.5	5.5.5.5	237	0x80000009	0xaadd	3	
Net Link States (Area 0.0.0.1)						
Link ID	ADV Router	Age	Seq#	CkSum		
192.168.2.11	2.2.2.2	234	0x80000001	0x953a		
192.168.3.11	3.3.3.3	238	0x80000001	0x9e28		
192.168.4.11	3.3.3.3	238	0x80000001	0xb50c		
192.168.5.11	4.4.4.4	239	0x80000001	0xe0d3		
192.168.6.11	5.5.5.5	237	0x80000001	0xe9c1		
192.168.7.11	5.5.5.5	242	0x80000001	0x01a5		

Observando las base de datos de los routers nos muestran que todos los routers se han comunicado entre sí, mediante OSPF, por lo que sus tablas de ruteo contienen el camino para llegar directamente a todos los demás routers.

7. Luego una vez configuradas las áreas 1 y 2, mostramos las siguientes tablas de base de datos:

Tabla database del router 2

```

OSPF Router with ID (2.2.2.2)

Router Link States (Area 0.0.0.1)

Link ID          ADV Router      Age  Seq#           CkSum  Link count
1.1.1.1          1.1.1.1         129  0x80000006     0xfe51 1
2.2.2.2          2.2.2.2         126  0x80000009     0x9522 3
3.3.3.3          3.3.3.3         127  0x80000005     0xa299 1

Net Link States (Area 0.0.0.1)

Link ID          ADV Router      Age  Seq#           CkSum
192.168.2.11     2.2.2.2         129  0x80000001     0x953a
192.168.4.11     3.3.3.3         127  0x80000001     0xb50c

Summary Link States (Area 0.0.0.1)

Link ID          ADV Router      Age  Seq#           CkSum  Route
192.168.3.0     1.1.1.1         131  0x80000003     0xda06 192.168.3.0/24
192.168.3.0     3.3.3.3         171  0x80000001     0xa238 192.168.3.0/24
192.168.5.0     1.1.1.1         122  0x80000001     0x2da9 192.168.5.0/24
192.168.5.0     3.3.3.3         130  0x80000003     0x884e 192.168.5.0/24
192.168.6.0     1.1.1.1         117  0x80000002     0x20b4 192.168.6.0/24
192.168.6.0     3.3.3.3         124  0x80000004     0x7b59 192.168.6.0/24
192.168.7.0     1.1.1.1         127  0x80000001     0x7b4f 192.168.7.0/24

```

Tabla database del router 5

```
OSPF Router with ID (5.5.5.5)

  Router Link States (Area 0.0.0.2)

Link ID      ADV Router    Age Seq#       CkSum Link count
3.3.3.3      3.3.3.3        171 0x80000009 0xcd5c 2
4.4.4.4      4.4.4.4        172 0x8000000a 0x97fb 3
5.5.5.5      5.5.5.5        166 0x80000009 0xaadd 3

  Net Link States (Area 0.0.0.2)

Link ID      ADV Router    Age Seq#       CkSum
192.168.5.11 4.4.4.4        172 0x80000001 0xe0d3
192.168.6.11 5.5.5.5        166 0x80000001 0xe9c1
192.168.7.11 5.5.5.5        171 0x80000001 0x01a5

  Summary Link States (Area 0.0.0.2)

Link ID      ADV Router    Age Seq#       CkSum Route
192.168.1.0  3.3.3.3        159 0x80000002 0x1bb6 192.168.1.0/24
192.168.2.0  3.3.3.3        164 0x80000001 0x12bf 192.168.2.0/24
192.168.3.0  3.3.3.3        214 0x80000001 0xa238 192.168.3.0/24
192.168.4.0  3.3.3.3         4 0x80000002 0x9543 192.168.4.0/24
```

En las tablas database de ambos routers no podemos observar todos los routers y las redes, esto se debe a que cada router tiene noción de las redes y routers que están dentro de su misma área, es por esto que el Router Designado conoce cual es el Router Borde de área (router que hace de intercomunicador ambas áreas), el DR comunica a todos los miembros de su área por cual router se van a comunicar a la otra área.

Contenido de las tablas de base de datos

- LINK ID: ID del router.
- ADV Router: ID del router que anuncia la ruta.
- Age: Edad del estado del enlace.
- Seq#: Número de secuencia del LSA, para detectar LSAs antiguos.
- Checksum: Suma de comprobación del LSA.
- Link count: Número de interfaces detectados por router.

8.

a. Para comprender de mejor manera el funcionamiento de OSPF, nos ubicamos en el router 2 vía telnet con el comando *telnet localhost 10014*, y ejecutando luego el comando *show ip ospf neighbor* podemos ver información de los vecinos en este caso Router 1 y Router 3.

Información que podemos ver:

- ID del enlace del router
- Estado de los vecinos

- El estado de vecino en este caso es Full que quiere decir que el router 2 le comunica absolutamente todo a sus vecinos.
- En el estado también se puede ver que el router 3 es el router designado en el área y el router 1 el backup o designado de respaldo.
- La dirección del router que puede no coincidir con el ID
- Interfaz con la cual se comunica con el Router 2.

```
r2> show ip ospf neighbor
```

Neighbor ID	Pri	State	Dead Time	Address	Interface	RXmtL	RqstL	DBsml
1.1.1.1	1	Full/Backup	34.938s	192.168.2.10	eth1:192.168.2.11	0	0	0
3.3.3.3	1	Full/DR	36.686s	192.168.4.11	eth2:192.168.4.10	0	0	0

b. En la siguiente imagen se puede ver la tabla de enrutamiento de OSPF del router 2. En ella se puede ver las distintas redes aprendidas mediante ospf ya sea porque están directamente conectadas como por ejemplo la red 192.162.0.0 y 192.162.2.0 o son conocidas vía otros routers por ejemplo la red 192.168.1.0 via la puerta del router 1 - 192.168.0.11

```
r2# show ip route
Codes: K - kernel route, C - connected, S - static, R - RIP,
       0 - OSPF, I - IS-IS, B - BGP, P - PIM, A - Babel, N - NHRP,
       > - selected route, * - FIB route

K>* 0.0.0.0/0 via 192.168.1.1, eth0
C>* 127.0.0.0/8 is directly connected, lo
0 192.168.1.0/24 [110/10] is directly connected, eth0, 00:09:19
C>* 192.168.1.0/24 is directly connected, eth0
0 192.168.2.0/24 [110/10] is directly connected, eth1, 00:09:19
C>* 192.168.2.0/24 is directly connected, eth1
0>* 192.168.3.0/24 [110/20] via 192.168.2.10, eth1, 00:08:29
   *                      via 192.168.4.11, eth2, 00:08:29
0 192.168.4.0/24 [110/10] is directly connected, eth2, 00:09:19
C>* 192.168.4.0/24 is directly connected, eth2
0>* 192.168.5.0/24 [110/20] via 192.168.4.11, eth2, 00:08:31
0>* 192.168.6.0/24 [110/20] via 192.168.4.11, eth2, 00:08:26
0>* 192.168.7.0/24 [110/30] via 192.168.4.11, eth2, 00:08:31
0>* 192.168.8.0/24 [110/30] via 192.168.4.11, eth2, 00:08:31
0>* 192.168.9.0/24 [110/30] via 192.168.4.11, eth2, 00:08:28
```

9. Se trazó el camino que seguía un paquete ICMPv4 desde el Host 1, hacia el Host 4, sin modificar el costo de los enlaces.

```
traceroute to 192.168.8.11 (192.168.8.11), 30 hops max, 46 byte packets
 1 protocoloospf r2 1.protocoloospf nr2 (192.168.1.10) 0.005 ms 192.168.1.1
 1 protocoloospf nr2- (192.168.1.10) 0.005 ms
 2 * 192.168.4.11 (192.168.4.11) 0.006 ms 0.005 ms
 3 192.168.5.11 (192.168.5.11) 0.004 ms 0.006 ms 0.002 ms
 4 * 192.168.8.11 (192.168.8.11) 0.004 ms
```

Luego se modificó el costo del enlace entre host 1 y host 4, precisamente en eth1 del R2

```
hostname r2
password admin
!
interface eth0
| !ip ospf cost 1
interface eth1
| !ip ospf cost 1000
```

Y podemos ver en el traceroute que el paquete va por otro camino con menor costo.

```
docker exec -ti protocoloospf h1 1 traceroute 192.168.8.11
traceroute to 192.168.8.11 (192.168.8.11), 30 hops max, 46 byte packets
 1 protocoloospf_r2 1.protocolospf_nr2 (192.168.1.10)  0.004 ms  0.004 ms  0.002 ms
 2 192.168.2.10 (192.168.2.10)  0.002 ms  0.004 ms  0.002 ms
 3 192.168.3.11 (192.168.3.11)  0.002 ms  0.005 ms  0.003 ms
 4 192.168.5.11 (192.168.5.11)  0.002 ms  0.005 ms  0.002 ms
 5 192.168.8.11 (192.168.8.11)  0.002 ms  0.005 ms  0.003 ms
```

10. Se procedió a configurar la dirección loopback en *zebra.conf* del Router1.

```
interface lo
| ip address 127.0.0.1/24
ip route 0.0.0.0 0.0.0.0 lo
```

Luego por enrutamiento estático se designa que los paquetes con dirección ip de destino 0.0.0.0/0 se deben redirigir hacia la interfaz loopback del Router1.

```
r1# show ip route
Codes: K - kernel route, C - connected, S - static, R - RIP,
       O - OSPF, I - IS-IS, B - BGP, P - PIM, A - Babel, N - NHRP,
       > - selected route, * - FIB route

S    0.0.0.0/0 [1/0] is directly connected, lo
K>*  0.0.0.0/0 via 192.168.2.1, eth0
C>*  127.0.0.0/8 is directly connected, lo
C>*  127.0.0.0/24 is directly connected, lo
O    192.168.2.0/24 [110/10] is directly connected, eth0, 00:00:13
C>*  192.168.2.0/24 is directly connected, eth0
O    192.168.3.0/24 [110/10] is directly connected, eth1, 00:00:13
C>*  192.168.3.0/24 is directly connected, eth1
```

11. Comprobación del funcionamiento de la red en el caso de caerse un enlace del Router 2.


```

docker exec -ti protocoloospf h1 1 traceroute 192.168.8.11
traceroute to 192.168.8.11 (192.168.8.11), 30 hops max, 46 byte packets
 1 protocoloospf_r2 1.protocoloospf_nr2 (192.168.1.10)  0.004 ms  0.004 ms  0.002 ms
 2 192.168.2.10 (192.168.2.10)  0.002 ms  0.004 ms  0.002 ms
 3 192.168.3.11 (192.168.3.11)  0.002 ms  0.005 ms  0.003 ms
 4 192.168.5.11 (192.168.5.11)  0.002 ms  0.005 ms  0.002 ms
 5 192.168.8.11 (192.168.8.11)  0.002 ms  0.005 ms  0.003 ms

```

Enlace entre R2 y R3 (192.168.4.0) caído. Se puede ver en la imagen que la comunicación se realiza sobre 192.168.3.0 (enlace entre R3 y R1).

12. La **Base de información de enrutamiento (RIB)** es una selección de información de enrutamiento aprendida a través de una definición estática o un protocolo de enrutamiento dinámico(todas las aprendidas).

La **Forwarding Information Base (FIB)** es conceptualmente similar a una tabla de enrutamiento o base de información. La forwarding information base (FIB) es la información real que utiliza un dispositivo de enrutamiento / Switcheo para elegir la interfaz que utilizará un paquete dado para la salida o sea en estas tablas están las rutas elegidas(solamente).

RIB (Base de información de enrutamiento) está siendo construida por varios procesos de enrutamiento en el nodo, con información de protocolos de enrutamiento, como OSPF, IS-IS, BGP o incluso las entradas estáticas.

Después de seleccionar todas las rutas en RIB, las mejores se seleccionan y se copian en **FIB** (Base de información de reenvío).

Parte 2:

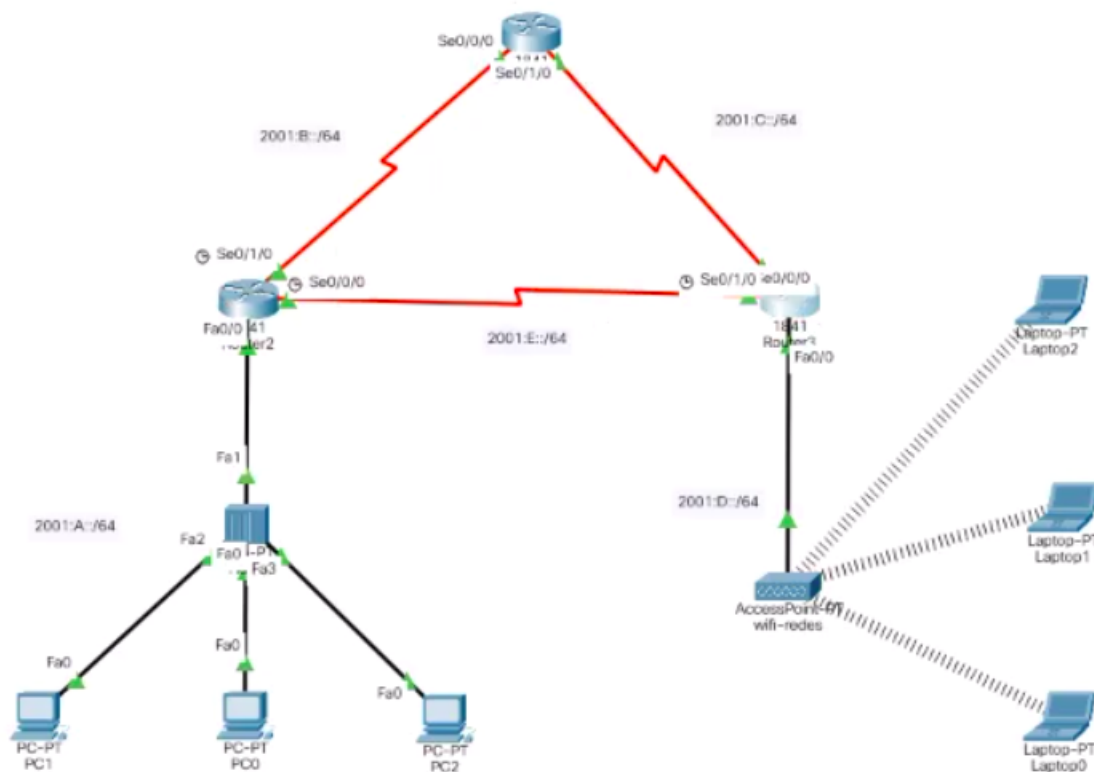
Introducción

Se confeccionará la siguiente topología basándose en el ruteo dinámico OSPF para IPv6, utilizando Packet Tracer de Cisco. Configuraremos las direcciones IPv6 para permitir la comunicación entre todos los dispositivos de la red.

Desarrollo

1. Configurar interfaces de los routers.

Decidimos utilizar routers 1843 y se debió agregar el módulo *wpc300* en las Laptop's para que estas fueran capaces de conectarse a la red wireless definida por el AccesPoint conectado al Router 3.



2. Configuración de las interfaces

Comandos utilizados para la configuración de OSPF:

En cada uno de los routers en su CLI se ejecuta los siguientes comandos:

```
ipv6 unicast-routing
ipv6 router ospf 1
router-id 1.1.1.1
exit
```

Continuamos con:

```
interface serial 0/0/0
ipv6 ospf 1 area 0
exit
interface serial 0/1/0
ipv6 ospf 1 area 1
exit
interface fa 0/0
ipv6 ospf 1 area 0
exit
```

De este modo configuramos las interfaces con sus respectivas áreas. Este proceso se repite para los restantes routers. Se le asigna otro **id** a cada router y sus interfaces a las áreas correspondientes.

Dentro del Router 1, escribimos el comando:

```
show ipv6 route
```

```
Router(config)#do sh ipv6 route
IPv6 Routing Table - 8 entries
Codes: C - Connected, L - Local, S - Static, R - RIP, B - BGP
       U - Per-user Static route, M - MIPv6
       I1 - ISIS L1, I2 - ISIS L2, IA - ISIS interarea, IS - ISIS summary
       ND - ND Default, NDp - ND Prefix, DCE - Destination, NDr - Redirect
       O - OSPF intra, OI - OSPF inter, OE1 - OSPF ext 1, OE2 - OSPF ext 2
       ON1 - OSPF NSSA ext 1, ON2 - OSPF NSSA ext 2
       D - EIGRP, EX - EIGRP external
O 2001:A::/64 [110/65]
  via FE80::201:96FF:FE28:E001, Serial0/0/0
C 2001:B::/64 [0/0]
  via ::, Serial0/0/0
L 2001:B::1/128 [0/0]
  via ::, Serial0/0/0
C 2001:C::/64 [0/0]
  via ::, Serial0/1/0
L 2001:C::1/128 [0/0]
  via ::, Serial0/1/0
O 2001:D::/64 [110/65]
  via FE80::201:C7FF:FEBE:E001, Serial0/1/0
OI 2001:E::/64 [110/128]
  via FE80::201:96FF:FE28:E001, Serial0/0/0
--More--
```

Ctrl+F6 to exit CLI focus

Copy

Paste

Se pueden observar las rutas aprendidas por OSPF, como por ejemplo:

- 2001:A::/64
- 2001:D::/64
- 2001:E::/64

3. Configurar el costo OSPF

Primero realizamos el ping entre PC0 y Laptop0, sin modificación del costo de enlace:

```
C:\>ipconfig

FastEthernet0 Connection:(default port)

    Connection-specific DNS Suffix...:
    Link-local IPv6 Address . . . . .: FE80::260:3EFF:FE47:770B
    IPv6 Address . . . . .: 2001:A::4
    IPv4 Address. . . . .: 0.0.0.0
    Subnet Mask . . . . .: 0.0.0.0
    Default Gateway . . . . .: 2001:A::2
                                0.0.0.0

Bluetooth Connection:

    Connection-specific DNS Suffix...:
    Link-local IPv6 Address . . . . .: ::
    IPv6 Address . . . . .: ::
    IPv4 Address. . . . .: 0.0.0.0
    Subnet Mask . . . . .: 0.0.0.0
    Default Gateway . . . . .: ::
                                0.0.0.0

C:\>ping 2001:D::6

Pinging 2001:D::6 with 32 bytes of data:

Reply from 2001:D::6: bytes=32 time=46ms TTL=126
Reply from 2001:D::6: bytes=32 time=36ms TTL=126
Reply from 2001:D::6: bytes=32 time=32ms TTL=126
Reply from 2001:D::6: bytes=32 time=41ms TTL=126

Ping statistics for 2001:D::6:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 32ms, Maximum = 46ms, Average = 38ms

C:\>
```

Realizamos el Traceroute entre PC0 y Laptop0:

imagen

Modificamos el costo de enlace para hacer trabajar a OSPF

- **Router 2:**

```
configure terminal
interface serial 0/1/0
Ipv6 ospf cost 400
exit
interface serial 0/0/0
Ipv6 ospf cost 100
exit
```

- **Router 3:**

```
configure terminal
interface serial 0/0/0
Ipv6 ospf cost 200
exit
```

Comparamos con los resultados anteriores:

The image shows a comparison between router configurations and their corresponding traceroute results. On the left, a terminal window displays the configuration for Router 2 (top, red box) and Router 3 (bottom, green box). On the right, two traceroute outputs are shown, corresponding to the configurations. Red and green arrows link the configurations to their respective traceroute results.

Router 2 Configuration (Red Box):

```
Router(config)#
Router(config)#interface serial 0/1/0
Router(config-if)#
Router(config-if)#ipv6 ospf cost 400
Router(config-if)#
Router(config-if)#interface serial 0/0/0
Router(config-if)#
Router(config-if)#ipv6 ospf cost 100
Router(config-if)#
Router(config-if)#exit
```

Router 2 Traceroute (Red Box):

```
C:\>tracert 2001:D::6

Tracing route to 2001:D::6 over a maximum of 30 hops:

  1  0 ms   0 ms   0 ms   2001:A::2
  2  13 ms  23 ms  21 ms  2001:C::4
  3  12 ms  47 ms  17 ms  2001:D::6

Trace complete.
```

Router 3 Configuration (Green Box):

```
Router(config)#interface serial 0/1/0
Router(config-if)#
Router(config-if)#ipv6 ospf cost 100
Router(config-if)#
Router(config-if)#interface serial 0/0/0
Router(config-if)#
Router(config-if)#ipv6 ospf cost 400
Router(config-if)#
```

Router 3 Traceroute (Green Box):

```
C:\>tracert 2001:D::6

Tracing route to 2001:D::6 over a maximum of 30 hops:

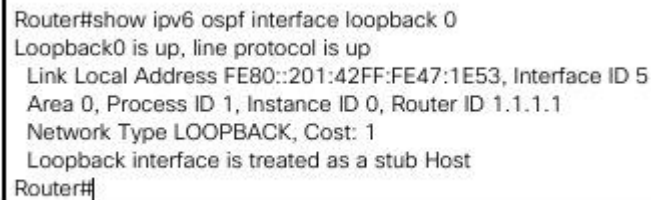
  1  0 ms   0 ms   0 ms   2001:A::2
  2  0 ms   0 ms   7 ms   2001:B::1
  3  18 ms  22 ms   6 ms   2001:C::4
  4  36 ms  34 ms  29 ms   2001:D::6

Trace complete.
```

4. Redistribuir una ruta OSPF predeterminada

a) Configurar una dirección de loopback en R1 para simular un enlace a ISP

```
configure terminal
Interface loopback 0
Ipv6 address 1:1:1:1::1/64
Ipv6 ospf 1 area 0
exit
```



```
Router#show ipv6 ospf interface loopback 0
Loopback0 is up, line protocol is up
  Link Local Address FE80::201:42FF:FE47:1E53, Interface ID 5
  Area 0, Process ID 1, Instance ID 0, Router ID 1.1.1.1
  Network Type LOOPBACK, Cost: 1
  Loopback interface is treated as a stub Host
Router#
```

Ctrl+F6 to exit CLI focus

b) Configurar una ruta estática predeterminada en el router R1

Se introdujo en la CLI del router el comando

```
ipv6 route::/0 loopback 0
```

Esto nos permite indicarle al router que cuando no encuentre la dirección de destino del paquete, se la envía a *loopback0*.

c) Incluir la ruta estática en las actualizaciones de OSPF que se envían desde el router R1.

Dentro de la CLI del router tipeamos lo siguiente:

```
ipv6 loopback 0
ipv6 route ospf 1
default-information originate
```

Ahora si ejecutamos:

```
show ipv6 route
```

Obtendremos la siguiente salida:

```
Router#show ipv6 route
IPv6 Routing Table - 11 entries
Codes: C - Connected, L - Local, S - Static, R - RIP, B - BGP
        U - Per-user Static route, M - MIPv6
        I1 - ISIS L1, I2 - ISIS L2, IA - ISIS interarea, IS - ISIS summary
        ND - ND Default, NDp - ND Prefix, DCE - Destination, NDr - Redirect
        O - OSPF intra, OI - OSPF inter, OE1 - OSPF ext 1, OE2 - OSPF ext 2
        ON1 - OSPF NSSA ext 1, ON2 - OSPF NSSA ext 2
        D - EIGRP, EX - EIGRP external
S ::0 [1/0]
  via ::, Loopback0
C 1:1:1:1::/64 [0/0]
  via ::, Loopback0
L 1:1:1:1::1/128 [0/0]
  via ::, Loopback0
O 2001:A::/64 [110/65]
  via FE80::201:96FF:FE28:E001, Serial0/0/0
C 2001:B::/64 [0/0]
  via ::, Serial0/0/0
L 2001:B::1/128 [0/0]
  via ::, Serial0/0/0
C 2001:C::/64 [0/0]
  via ::, Serial0/1/0
```

Como podemos ver, cuando se envíe un paquete con una dirección de destino desconocida, el Router 1 la va a rutear por *loopback 0*.

5. Enumerar diferencias entre OSPF y OSPFv3

- **Anuncios:** OSPF anuncia rutas IPv4, mientras que OSPFv3 anuncia rutas para IPv6.
- **Dirección de origen:** los mensajes OSPF se originan en la dirección IPv4 de la interfaz de salida. En OSPFv3, los mensajes OSPF se originan con la dirección link-local de la interfaz de salida.
- **Dirección de multidifusión de todos los routers OSPF:** OSPF utiliza la dirección 224.0.0.5, mientras que OSPFv3 utiliza la dirección FF02::5.
- **Dirección de multidifusión de DR/BDR:** OSPFv2 utiliza la dirección 224.0.0.6, mientras que OSPFv3 utiliza la dirección FF02::6.
- **Anuncio de redes:** OSPF anuncia las redes mediante el comando de configuración del router **network**, mientras que OSPFv3 utiliza el comando de configuración de interfaz **ipv6 ospf id-proceso area id-área**.

- **Routing de unidifusión IP:** habilitado de manera predeterminada en IPv4; en cambio, el comando de configuración global *ipv6 unicast-routing* se debe configurar.
- **Autenticación:** OSPF utiliza autenticación de texto no cifrado o autenticación MD5. OSPFv3 utiliza autenticación IPv6.

Anexo I