

Internetworking II - Ruteo

- Clase práctica -

Teoría de las Comunicaciones



Departamento de Computación
Facultad de Ciencias Exactas y Naturales
Universidad de Buenos Aires

30.09.2025

Agenda

1 Algunas consideraciones

- Routing vs Forwarding
- Ruteo como grafo

2 Protocolos de Ruteo Interno

- Vector de distancia - RIP
- Conteo al infinito - Problema
- Vector de distancia - Ejercicio

3 Protocolos de Ruteo Interno

- Estado de enlace - OSPF
- Estado de enlace - Ejercicio

Agenda

1 Algunas consideraciones

- Routing vs Forwarding
- Ruteo como grafo

2 Protocolos de Ruteo Interno

- Vector de distancia - RIP
- Conteo al infinito - Problema
- Vector de distancia - Ejercicio

3 Protocolos de Ruteo Interno

- Estado de enlace - OSPF
- Estado de enlace - Ejercicio

Tabla de Routing vs Tabla de Forwarding

Se suelen usar indistintamente, pero hay diferencias.

Tabla de Routing vs Tabla de Forwarding

Se suelen usar indistintamente, pero hay diferencias.

Tabla de Routing: Contiene **<Red, Costo, Próximo Salto>**.

- Es un proceso **basado en software**: el **resultado** de la ejecución de un **algoritmo**.
- Es exclusivo del **nivel de Red**.
- Es la precursora de la Tabla de Forwarding.
- Puede tener información sobre cómo fue construida y entradas repetidas.

Tabla de Routing vs Tabla de Forwarding

Se suelen usar indistintamente, pero hay diferencias.

Tabla de Routing: Contiene <**Red, Costo, Próximo Salto**>.

- Es un proceso **basado en software**: el **resultado** de la ejecución de un **algoritmo**.
- Es exclusivo del **nivel de Red**.
- Es la precursora de la Tabla de Forwarding.
- Puede tener información sobre cómo fue construida y entradas repetidas.

Tabla de Forwarding: Contiene <**Red, Próximo Salto**>.

- Proceso mecánico. Optimizado para velocidad y eficiencia.
- Usada directamente para la toma de decisiones sobre el reenvío de paquetes.
- Lo hemos visto en **nivel de Enlace** y **nivel de red**.
- No tiene entradas repetidas.

Notación para los ejercicios

Tabla de Routing

Network	Cost	Next hop
172.16.5.0/24	0	IF 0/0
10.4.2.0/27	0	IF 0/1
192.168.2.0/26	1	10.4.2.25
Default	2	10.4.2.25

Tabla de Forwarding

Network	Next hop
172.16.5.0/24	IF 0/0
10.4.2.0/27	IF 0/1
192.168.2.0/26	10.4.2.25
Default	10.4.2.25

Network (Red)	Next Hop (Próximo salto)
Red destino	- Interface de salida. (si la red esta directamente conectada) - Dirección IP del próximo salto. (Si la red destino es una red remota)

Notación para los ejercicios

Tabla de Routing

Network	Cost	Next hop
172.16.5.0/24	0	IF 0/0
10.4.2.0/27	0	IF 0/1
192.168.2.0/26	1	10.4.2.25
Default	2	10.4.2.25

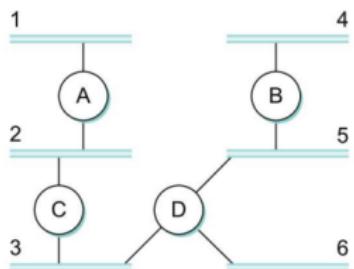
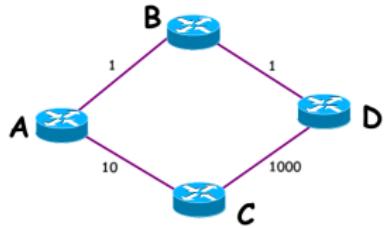
Tabla de Forwarding

Network	Next hop
172.16.5.0/24	IF 0/0
10.4.2.0/27	IF 0/1
192.168.2.0/26	10.4.2.25
Default	10.4.2.25

Network (Red)	Next Hop (Próximo salto)
Red destino	- Interface de salida. (si la red esta directamente conectada) - Dirección IP del próximo salto. (Si la red destino es una red remota)

¿Y cómo hace **Forwarding** el **switch**?

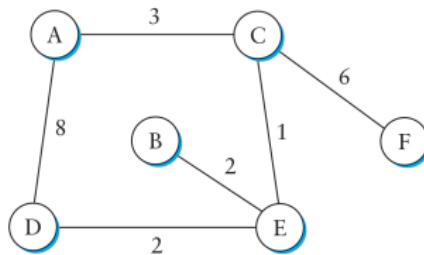
Ruteo como grafo



- **Ruteo** es, en esencia, resolver un problema de optimización de grafos (camino de menor costo) mediante la solución de algoritmos.
- Vemos que la red de destino son los nodos (routers) y los enlaces son los arcos pesados (donde el peso representa el costo, distancia).
- En Internet Ruteo difiere del modelo de grafos en que su objetivo es alcanzar redes en vez de alcanzar routers

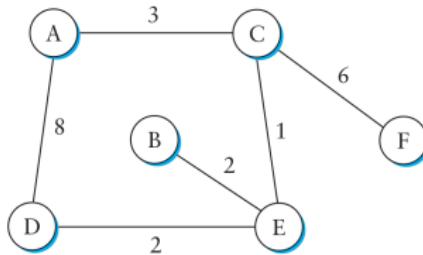
Ejercicio 1

En la red de la figura los enlaces están etiquetados con los costos relativos.



- Mostrar la **tabla de forwarding** para cada nodo. Cada tabla en cada nodo debe reflejar la ruta de menor costo para el envío de un paquete a un determinado destino.
- ¿De qué maneras se pueden llenar esas tablas? Mencione las diferencias más significativas.

Ejercicio 1: Tablas de Forwarding

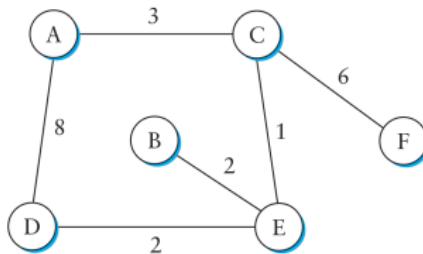


Router A

Nodo	Próximo salto
A	
B	
C	
D	
E	
F	

Cada tabla en cada nodo debe reflejar la ruta de menor costo para el envío de un paquete a un determinado destino.

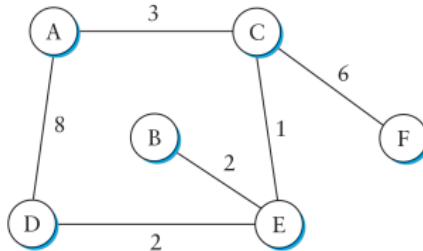
Ejercicio 1: Tablas de Forwarding



Router A

Nodo	Próximo salto
A	-
B	C
C	C
D	C
E	C
F	C

Ejercicio 1: Tablas de Forwarding



Router A	
Nodo	Próximo salto
A	-
B	C
C	C
D	C
E	C
F	C

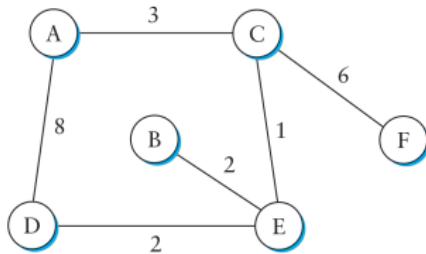
Router B	
Nodo	Próximo salto
A	
B	
C	
D	
E	
F	

Router D	
Nodo	Próximo salto
A	
B	
C	
D	
E	
F	

Router F	
Nodo	Próximo salto
A	
B	
C	
D	
E	
F	

Hay tablas que son fáciles de llenar

Ejercicio 1: Tablas de Forwarding



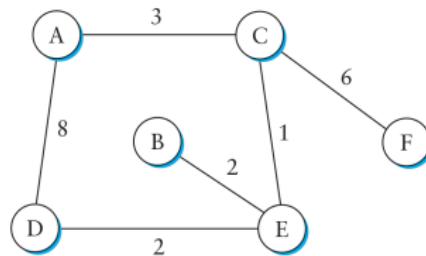
Router A	
Nodo	Próximo salto
A	-
B	C
C	C
D	C
E	C
F	C

Router B	
Nodo	Próximo salto
A	E
B	-
C	E
D	E
E	E
F	E

Router D	
Nodo	Próximo salto
A	E
B	E
C	E
D	-
E	E
F	E

Router F	
Nodo	Próximo salto
A	C
B	C
C	C
D	C
E	C
F	-

Ejercicio 1: Tablas de Forwarding



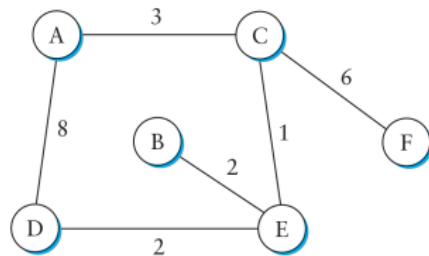
Router C

Nodo	Próximo salto
A	
B	
C	
D	
E	
F	

Router E

Nodo	Próximo salto
A	
B	
C	
D	
E	
F	

Ejercicio 1: Tablas de Forwarding



Router C

Nodo	Próximo salto
A	A
B	E
C	-
D	E
E	E
F	F

Router E

Nodo	Próximo salto
A	C
B	B
C	C
D	D
E	-
F	C

Ejercicio 1

- b.** ¿De qué maneras se pueden llenar esas tablas? Mencione las diferencias más significativas.

Ejercicio 1: Formas de enrutamiento

Estático

Dinámico

Ejercicio 1: Formas de enrutamiento

Estático

- Genera carga y tiempo de administración de red en redes grandes, debe configurarse **manualmente** el enrutamiento en cada router de la red.

Dinámico

- No genera mucha carga administrativa porque los routers **aprenden** a enrutarse de los demás routers de la red.

Ejercicio 1: Formas de enrutamiento

Estático

- Genera carga y tiempo de administración de red en redes grandes, debe configurarse **manualmente** el enrutamiento en cada router de la red.
- Los routers:
 - ▶ No comparten su tabla de enrutamiento con los routers vecinos.

Dinámico

- No genera mucha carga administrativa porque los routers **aprenden** a enrutarse de los demás routers de la red.
- Los routers:
 - ▶ Comparten su tabla de enrutamiento con los routers vecinos.

Ejercicio 1: Formas de enrutamiento

Estático

- Genera carga y tiempo de administración de red en redes grandes, debe configurarse **manualmente** el enrutamiento en cada router de la red.
- Los routers:
 - ▶ No comparten su tabla de enrutamiento con los routers vecinos.
 - ▶ No tienen capacidad de reacción ante un **fallo/cambio** en la red.

Dinámico

- No genera mucha carga administrativa porque los routers **aprenden** a enrutarse de los demás routers de la red.
- Los routers:
 - ▶ Comparten su tabla de enrutamiento con los routers vecinos.
 - ▶ Tienen capacidad de reacción ante un **fallo/cambio** en la red.

Ejercicio 1: Formas de enrutamiento

Estático

- Genera carga y tiempo de administración de red en redes grandes, debe configurarse **manualmente** el enrutamiento en cada router de la red.
- Los routers:
 - ▶ No comparten su tabla de enrutamiento con los routers vecinos.
 - ▶ No tienen capacidad de reacción ante un **fallo/cambio** en la red.

Dinámico

- No genera mucha carga administrativa porque los routers **aprenden** a enrutarse de los demás routers de la red.
- Los routers:
 - ▶ Comparten su tabla de enrutamiento con los routers vecinos.
 - ▶ Tienen capacidad de reacción ante un **fallo/cambio** en la red.
 - ▶ **Pero... no todo son ventajas**

Agenda

1 Algunas consideraciones

- Routing vs Forwarding
- Ruteo como grafo

2 Protocolos de Ruteo Interno

- Vector de distancia - RIP
- Conteo al infinito - Problema
- Vector de distancia - Ejercicio

3 Protocolos de Ruteo Interno

- Estado de enlace - OSPF
- Estado de enlace - Ejercicio

Vector de Distancia - RIP

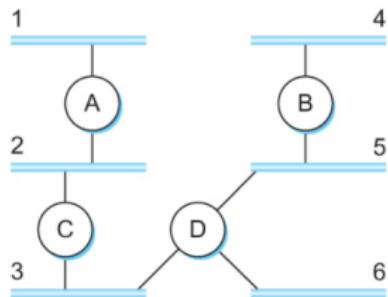
¿Qué lo caracteriza?

Envía información que tiene de TODA la red SOLO a sus vecinos directos.

Vector de Distancia - RIP

Paquete RIP

- Cada nodo arma un vector contenido <Red, Costo> a todos los demás nodos y lo envía a sus vecinos.



Red con RIP

0	8	16	31
Command	Version	Must be zero	
Family of net 1		Route Tags	
Address prefix of net 1			
Mask of net 1			
Distance to net 1			
Family of net 2		Route Tags	
Address prefix of net 2			
Mask of net 2			
Distance to net 2			

Formato del paquete RIPv2

Las distancias válidas son de 1 a 15.

Distancia 16 representa infinito: limitación de RIP a redes de tamaño pequeño.

Vector de Distancia - RIP

¿Cómo funciona?

- ① Cada nodo construye un vector contenido la **distancia** a todos los demás nodos.

Vector de Distancia - RIP

¿Cómo funciona?

- ① Cada nodo construye un vector contenido la **distancia** a todos los demás nodos.
- ② Al iniciar se asume que cada nodo conoce a sus vecinos inmediatos, cuya **distancia es 1**. Al resto se asigna ∞ .

Vector de Distancia - RIP

¿Cómo funciona?

- ① Cada nodo construye un vector contenido la **distancia** a todos los demás nodos.
- ② Al iniciar se asume que cada nodo conoce a sus vecinos inmediatos, cuya **distancia es 1**. Al resto se asigna ∞ .
- ③ Distribuye el vector a sus vecinos inmediatos.

Vector de Distancia - RIP

¿Cómo funciona?

- ① Cada nodo construye un vector contenido la **distancia** a todos los demás nodos.
- ② Al iniciar se asume que cada nodo conoce a sus vecinos inmediatos, cuya **distancia es 1**. Al resto se asigna ∞ .
- ③ Distribuye el vector a sus vecinos inmediatos.
- ④ Por cada mensaje recibido se suma 1 a la distancia de los nodos alcanzados por el vecino. Si esa distancia resultante es menor que la conocida, **se actualiza** (se aprende). En caso contrario se descarta.

Vector de Distancia - RIP

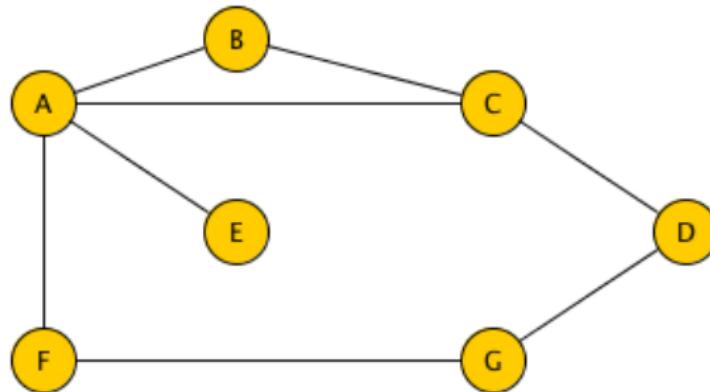
¿Cómo funciona?

- ① Cada nodo construye un vector contenido la **distancia** a todos los demás nodos.
- ② Al iniciar se asume que cada nodo conoce a sus vecinos inmediatos, cuya **distancia es 1**. Al resto se asigna ∞ .
- ③ Distribuye el vector a sus vecinos inmediatos.
- ④ Por cada mensaje recibido se suma 1 a la distancia de los nodos alcanzados por el vecino. Si esa distancia resultante es menor que la conocida, **se actualiza** (se aprende). En caso contrario se descarta.
- ⑤ **Decimos que RIP es una implementación bastante sencilla de Vector de Distancia**

Ejercicio 2

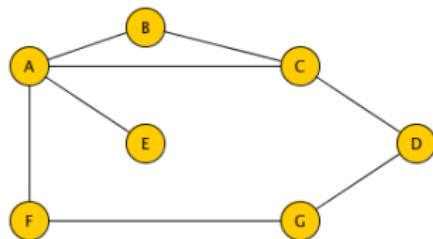
Mostrar los mensajes RIP generados por el nodo A y la matriz global de Vector de Distancia en los siguientes escenarios:

- a. Los nodos recién bootean y solo conocen las distancias de sus vecinos inmediatos.
- b. Los nodos ya propagaron la información del inciso anterior.
- c. La red ya convergió.



Ejercicio 2: RIP

- a. Los nodos recién bootan y solo conocen las distancias de sus vecinos inmediatos.



A	
Nodo	Costo

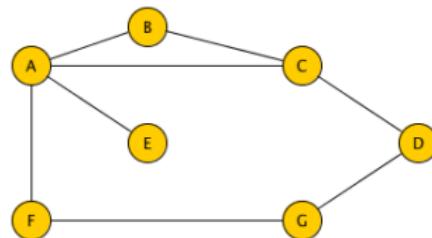
Formato mensaje RIP

Nodo	Distancia al Nodo						
	A	B	C	D	E	F	G
A							
B							
C							
D							
E							
F							
G							

Matriz global de Vector de distancia

Ejercicio 2: RIP

- a. Los nodos recién bootean y solo conocen las distancias de sus vecinos inmediatos.



A	
Nodo	Costo
A	0
B	1
C	1
E	1
F	1

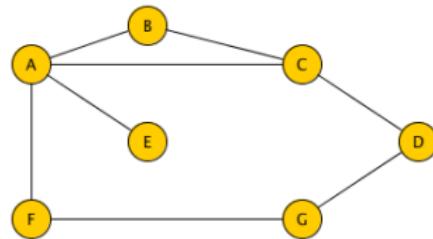
Mensaje RIP inicial

Nodo	Distancia al Nodo						
	A	B	C	D	E	F	G
A	0	1	1	∞	1	1	∞
B	1	0	1	∞	∞	∞	∞
C	1	1	0	1	∞	∞	∞
D	∞	∞	1	0	∞	∞	1
E	1	∞	∞	∞	0	∞	∞
F	1	∞	∞	∞	∞	0	1
G	∞	∞	∞	1	∞	1	0

Matriz global de Vector de distancia

Ejercicio 2: RIP

- a. ¿Qué pasa cuando los nodos propagan la información? Analicemos el caso de B cuando recibe el mensaje de A



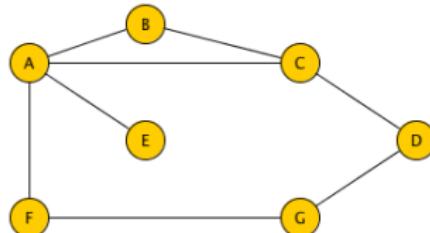
A	
Nodo	Costo
A	0
B	1
C	1
E	1
F	1

Mensaje RIP inicial

Nodo	Distancia al Nodo						
	A	B	C	D	E	F	G
B	1	0	1	∞	∞	∞	∞

Ejercicio 2: RIP

- b. Los nodos ya propagaron la información del inciso anterior.



A	
Nodo	Costo
A	0
B	1
C	1
D	2
E	1
F	1
G	2

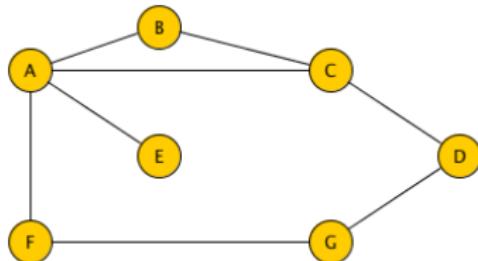
Mensaje RIP

Nodo	Distancia al Nodo						
	A	B	C	D	E	F	G
A	0	1	1	2	1	1	2
B	1	0	1	2	2	2	∞
C	1	1	0	1	2	2	2
D	2	2	1	0	∞	2	1
E	1	2	2	∞	0	2	∞
F	1	2	2	2	2	0	1
G	2	∞	2	1	∞	1	0

Matriz global de Vector de distancia

Ejercicio 2: RIP

- c. La red ya convergió.



A	
Nodo	Costo
A	0
B	1
C	1
D	2
E	1
F	1
G	2

Mensaje RIP

Nodo	Distancia al Nodo						
	A	B	C	D	E	F	G
A	0	1	1	2	1	1	2
B	1	0	1	2	2	2	3
C	1	1	0	1	2	2	2
D	2	2	1	0	3	2	1
E	1	2	2	3	0	2	3
F	1	2	2	2	2	0	1
G	2	3	2	1	3	1	0

Matriz global de Vector de distancia

Vector de Distancia - RIP. Propiedades.

- ① En ausencia de cambios en la topología, solo toma unos pocos intercambios de mensajes para que cada nodo logre completar su tabla.

Vector de Distancia - RIP. Propiedades.

- ① En ausencia de cambios en la topología, solo toma unos pocos intercambios de mensajes para que cada nodo logre completar su tabla.
- ② La métrica a utilizar para la distancia o costo es **la cantidad de saltos**

Vector de Distancia - RIP. Propiedades.

- ① En ausencia de cambios en la topología, solo toma unos pocos intercambios de mensajes para que cada nodo logre completar su tabla.
- ② La métrica a utilizar para la distancia o costo es **la cantidad de saltos**
- ③ El algoritmo es **distribuido**, los nodos trabajan independientemente con la información que disponen, pero colaboran para resolver un problema común.

Vector de Distancia - RIP. Propiedades.

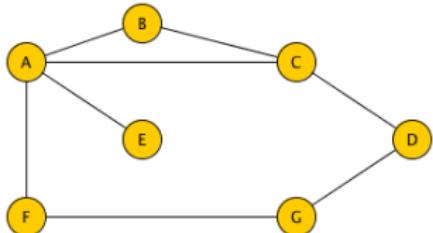
1. En ausencia de cambios en la topología, solo toma unos pocos intercambios de mensajes para que cada nodo logre completar su tabla.
2. La métrica a utilizar para la distancia o costo es **la cantidad de saltos**
3. El algoritmo es **distribuido**, los nodos trabajan independientemente con la información que disponen, pero colaboran para resolver un problema común.
4. Actualizaciones:

Vector de Distancia - RIP. Propiedades.

1. En ausencia de cambios en la topología, solo toma unos pocos intercambios de mensajes para que cada nodo logre completar su tabla.
2. La métrica a utilizar para la distancia o costo es **la cantidad de saltos**
3. El algoritmo es **distribuido**, los nodos trabajan independientemente con la información que disponen, pero colaboran para resolver un problema común.
4. Actualizaciones:
 - ▶ **Periódicas**, envían un update automáticamente en un intervalo de tiempo, aún si nada ha cambiado.
 - ▶ **Disparados**, ante la **caída** de un nodo o por cualquier mensaje que llegue que le genere un **cambio** en la tabla. Entonces un nuevo mensaje es transmitido.

Conteo al infinito - Problema

¿Qué sucede si el link A-E falla?



- Enlace de A a E falla
- A comunica a B y C distancia **infinita** a E (timing?)
- B y C comunican distancia 2 a E (timing?)
- Como el *distance vector* no utiliza información sobre el **next hop** B decide, gracias a C, que puede llegar a E en 3 hops
- B comunica esto a A
- A decide que puede llegar a E en 4 hops; comunica esto a C
- C decide que puede llegar a E en 5 hops; comunica esto a B
- Se repite indefinidamente.

Conteo al infinito - Solución

NO-INF No utilizar infinito, sino un número lo suficientemente alto (ej. 16). Esto no previene conteo al infinito, pero lo frena antes.

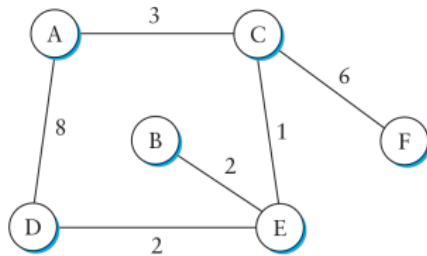
Split Horizon No mando actualizaciones de nuevas rutas, a los vecinos que me las enseñaron (requiere guardar quién me enseño cada ruta)

Split Horizon with poison reverse En vez de no enviar información al nodo que me enseño la ruta, mando información negativa (por ejemplo, con infinito).

El problema de estas soluciones es que funcionan para un conteo al infinito entre dos nodos.

Volviendo al Ejercicio 1...

En la red de la figura los enlaces están etiquetados con los costos relativos.



- a. ¿Qué podríamos decir ahora, si asumiéramos que se está ejecutando un protocolo de enrutamiento dinámico?

Ejercicio 3

En una red un router tiene solo dos interfaces configuradas con las direcciones IP y máscaras 10.0.2.1/24 y 10.0.3.1/30. A su vez, tiene directamente conectado un solo router vecino, del cuál recibe periódicamente un paquete RIP con la siguiente información de ruteo:

10.0.2.0	10.0.3.0	10.0.4.0	10.0.5.0
255.255.255.0	255.255.255.252	255.255.255.0	255.255.255.0
1	0	0	1

- Muestre una posible topología de red que se pueda deducir de la configuración del router y la información de ruteo que recibe.
- Muestre un posible paquete RIP que envía el router que recibe la información de ruteo.
- Mostrar la tabla de forwarding del router que recibe la información de ruteo.

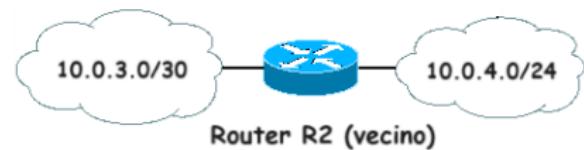
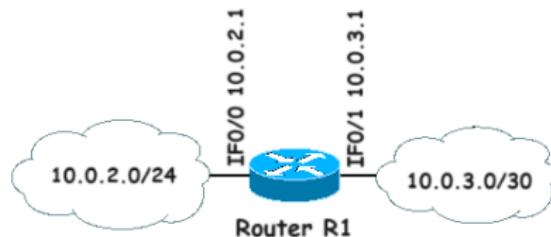
Ejercicio 3: Topología de red

Del enunciado del ejercicio podemos deducir:

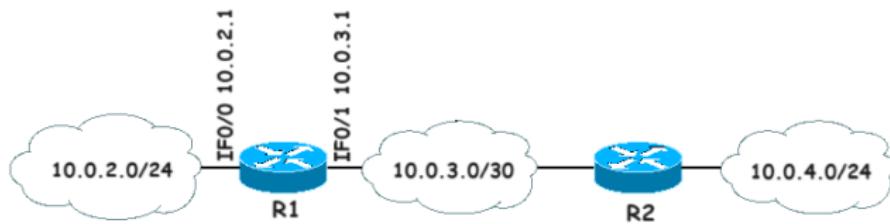


Ejercicio 3: Topología de red

Del enunciado del ejercicio podemos deducir:

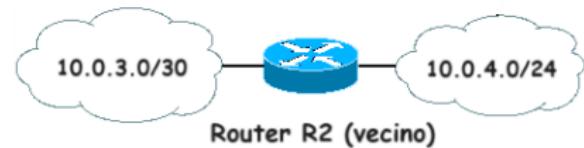
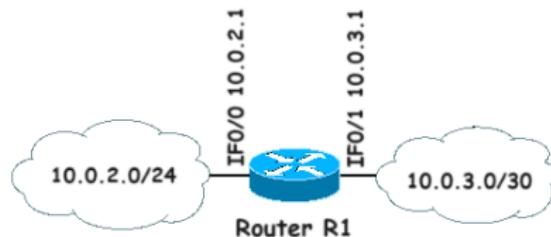


Los routers R1 y R2 tienen una red en común.

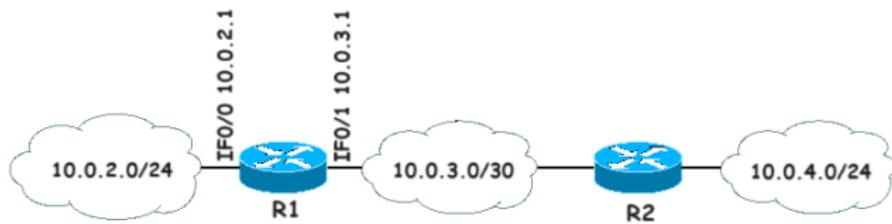


Ejercicio 3: Topología de red

Del enunciado del ejercicio podemos deducir:



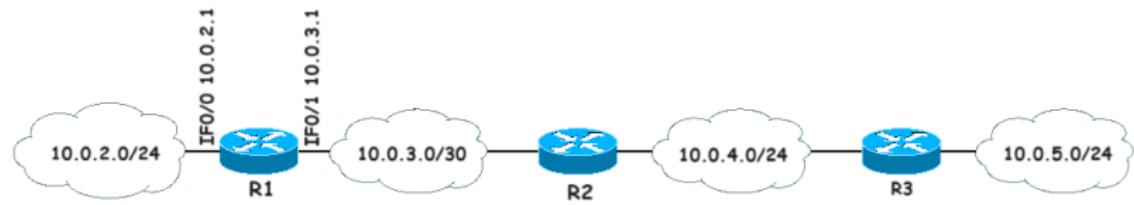
Los routers R1 y R2 tienen una red en común.



Pero debemos ubicar una cuarta red, la 10.0.5.0/24.

Ejercicio 3: Topología de red

Possible topología que podemos deducir:



Ejercicio 3: Mensaje RIP

- ⑥ Muestre un posible paquete RIP que envía el router que recibe la información de ruteo.

Ejercicio 3: Mensaje RIP

- b) Muestre un posible paquete RIP que envía el router que recibe la información de ruteo.

R1	
RED	Costo
10.0.2.0/24	0
10.0.3.0/30	0
10.0.4.0/24	1
10.0.5.0/24	2

Ejercicio 3: Tabla de forwarding

- Mostrar la tabla de forwarding del router que recibe la información de ruteo.

Ejercicio 3: Tabla de forwarding

- Mostrar la tabla de forwarding del router que recibe la información de ruteo.

R1	
RED	Próximo Salto
10.0.2.0/24	IF0/0
10.0.3.0/30	IF0/1
10.0.4.0/24	10.0.3.2
10.0.5.0/24	10.0.3.2

Agenda

1 Algunas consideraciones

- Routing vs Forwarding
- Ruteo como grafo

2 Protocolos de Ruteo Interno

- Vector de distancia - RIP
- Conteo al infinito - Problema
- Vector de distancia - Ejercicio

3 Protocolos de Ruteo Interno

- Estado de enlace - OSPF
- Estado de enlace - Ejercicio

Estado de enlace - OSPF

¿Qué lo caracteriza?

Envía Información **SOLO** de sus vecinos directos a **TODA** la red

Estado de enlace - OSPF

¿Qué lo caracteriza?

Envía Información **SOLO** de sus vecinos directos a **TODA** la red

Paquete OSPF

- Cada nodo arma un vector conteniendo
 $< ID\ nodo, SeqNum, TTL, Lista\ vecinos >$ a todos los demás nodos y lo envía a sus vecinos.

ID	nodo	SeqNum	TTL	Red	Costo
----	------	--------	-----	-----	-------

Estado de enlace - OSPF

¿Qué lo caracteriza?

Envía Información **SOLO** de sus vecinos directos a **TODA** la red

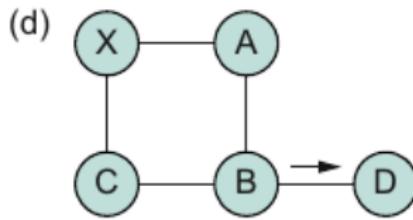
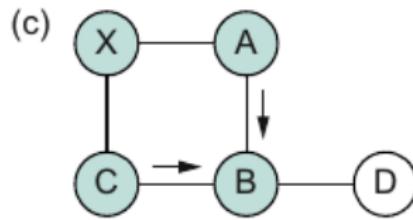
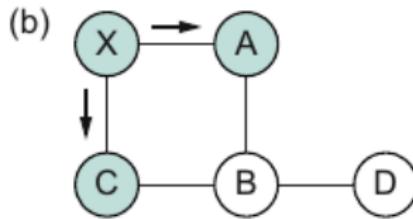
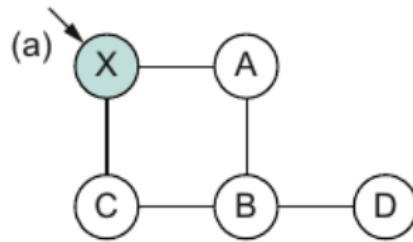
Paquete OSPF

- Cada nodo arma un vector contenido
 $< ID\ nodo, SeqNum, TTL, Lista\ vecinos >$ a todos los demás nodos y lo envía a sus vecinos.

ID	nodo	SeqNum	TTL	Red	Costo
----	------	--------	-----	-----	-------

¿Por qué OSPF usa **Nro. de Secuencia** y **TTL**?

Estado de enlace - Inundación confiable - Definición gráfica



Fuente: Larry L. Peterson and Bruce S. Davie. 2011.
Computer Networks, Fifth edition: A Systems Approach.

Estado de enlace - OSPF

¿Cómo funciona?

- 1 Descubrir a sus vecinos y conocer sus direcciones de red.

Estado de enlace - OSPF

¿Cómo funciona?

1. Descubrir a sus vecinos y conocer sus direcciones de red.
2. Establecer la métrica de distancia o de costo para cada uno de sus vecinos.

Estado de enlace - OSPF

¿Cómo funciona?

1. Descubrir a sus vecinos y conocer sus direcciones de red.
2. Establecer la métrica de distancia o de costo para cada uno de sus vecinos.
3. Construir un paquete que indique todo lo que acaba de aprender

Estado de enlace - OSPF

¿Cómo funciona?

1. Descubrir a sus vecinos y conocer sus direcciones de red.
2. Establecer la métrica de distancia o de costo para cada uno de sus vecinos.
3. Construir un paquete que indique todo lo que acaba de aprender
4. Enviar este paquete a todos los demás enrutadores y recibir paquetes de ellos.

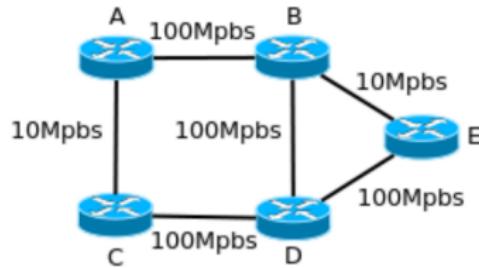
Estado de enlace - OSPF

¿Cómo funciona?

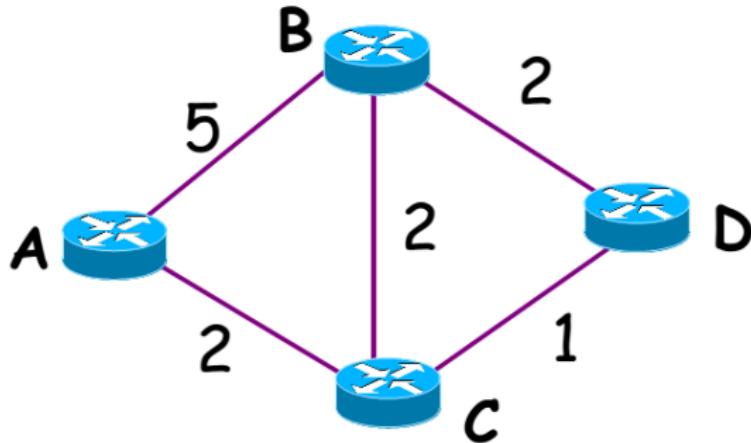
1. Descubrir a sus vecinos y conocer sus direcciones de red.
2. Establecer la métrica de distancia o de costo para cada uno de sus vecinos.
3. Construir un paquete que indique todo lo que acaba de aprender
4. Enviar este paquete a todos los demás enrutadores y recibir paquetes de ellos.
5. Calcular la ruta más corta a todos los demás enrutadores.

Métricas

- Métricas de RIP = 1
- Métricas de OSPF = 10^{10} /Ancho de banda



Ejercicio 4:



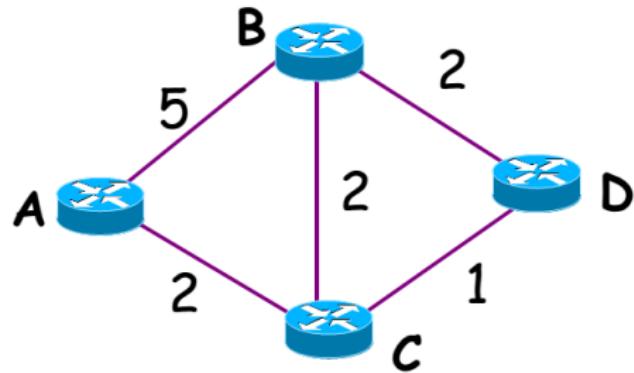
Dada la red de la figura, suponer que el protocolo de ruteo es utilizado es OSPF. Se pide:

- Mostrar todos los mensajes (el contenido de los campos relevantes) que recibe A hasta que la red converge.
- Explicar cómo A construye su tabla de ruteo a partir de los mensajes recibidos.

Ejercicio 4: OSPF

- a. Mostrar todos los mensajes (el contenido de los campos relevantes) que recibe A hasta que la red converge.

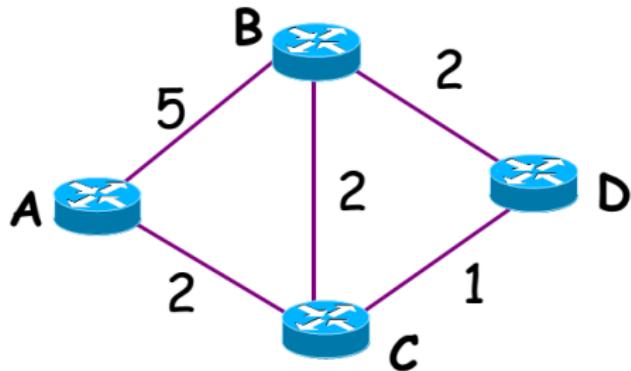
De B recibe:



Ejercicio 4: OSPF

- a. Mostrar todos los mensajes (el contenido de los campos relevantes) que recibe A hasta que la red converge.

De B recibe:

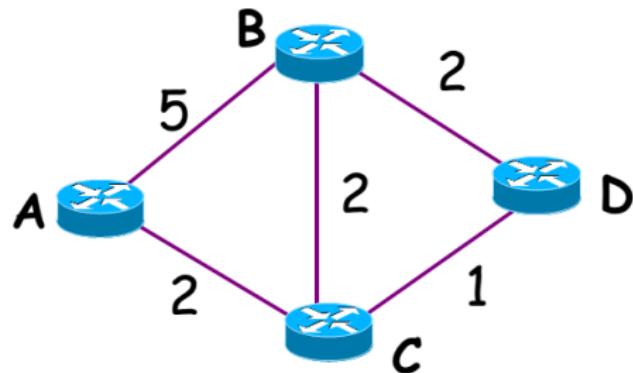


ID:	B
TTL:	X
#Seq:	Z
Nodo	Costo
A	5
C	2
D	2

Ejercicio 4: OSPF

- a. Mostrar todos los mensajes (el contenido de los campos relevantes) que recibe A hasta que la red converge.

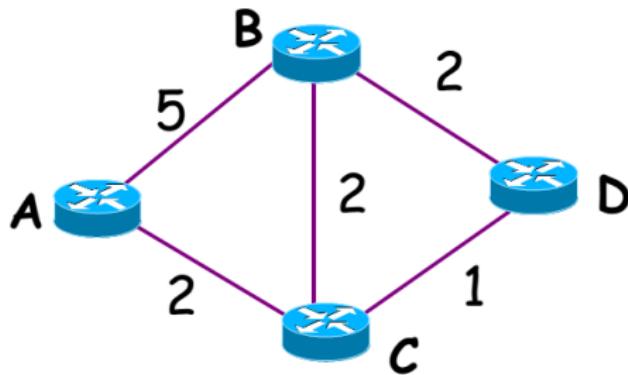
De C recibe:



Ejercicio 4: OSPF

- a. Mostrar todos los mensajes (el contenido de los campos relevantes) que recibe A hasta que la red converge.

De C recibe:

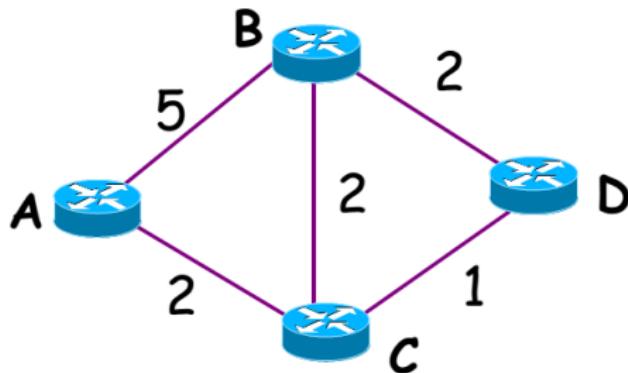


ID:	C
TTL:	X
#Seq:	Y
Nodo	Costo
A	2
B	2
D	1

Ejercicio 4: OSPF

- a. Mostrar todos los mensajes (el contenido de los campos relevantes) que recibe A hasta que la red converge.

De C recibe:



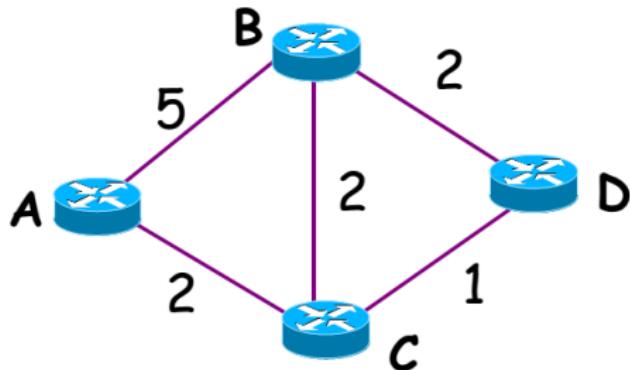
ID:	C
TTL:	X
#Seq:	Y
Nodo	Costo
A	2
B	2
D	1

Aclaración importante: el algoritmo aclara que en **todo momento** se manda la información del costo **directo** a sus vecinos. Entonces salvo número de secuencia y TTL, el resto de la información no varía en las iteraciones. Lo que sí se modifica es el estado de la tabla de ruteo de cada nodo

Ejercicio 4: OSPF

- a. Mostrar todos los mensajes (el contenido de los campos relevantes) que recibe A hasta que la red converge.

De D (gracias a la inundación) recibe:



ID:	D
TTL:	X-1
#Seq:	W
Nodo	Costo
B	2
C	1

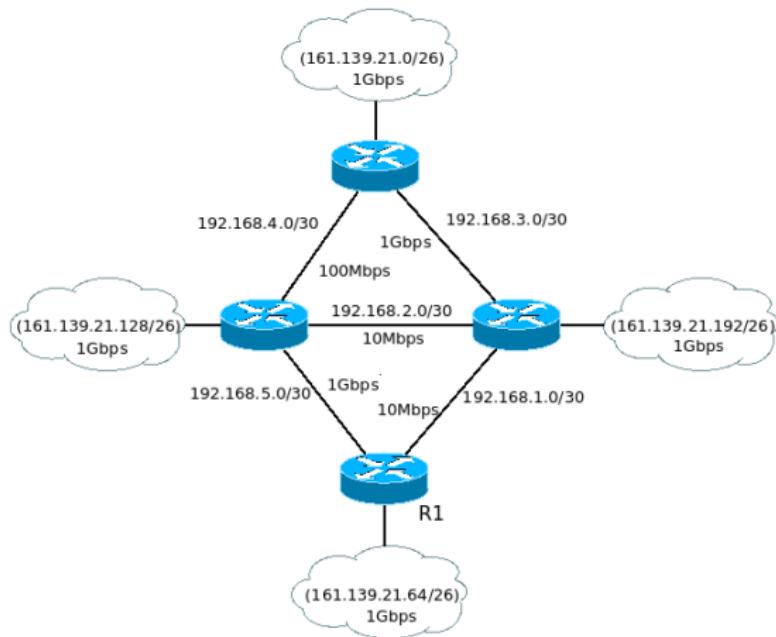
De D recibe un TTL menos ya que entre los nodos A y D hay otro router.

Ejercicio 4: OSPF

- b. Explicar cómo A construye su tabla de ruteo a partir de los mensajes recibidos.

(confirmado)	tentativo	
(A,0,-)		inicializo conmigo mismo
(A,0,-)	(B,5,B) (C,2,C)	traigo a tentativo los vecinos de A
(A,0,-) (C,2,C)	(B,5,B)	muevo a confirmado el de menor costo (C)
(A,0,-) (C,2,C)	(B,4,C) (D,3,C)	traigo los vecinos de C con pesos actualizados y reemplazo
(A,0,-) (C,2,C) (D,3,C)	(B,4,C)	traigo los vecinos de C con muevo el de menor costo (D)
A,0,-) (C,2,C) (D,3,C) (B,4,C)		ninguna vecino de C agrega o mejora lo que hay en tentativo, así que muevo B a confirmado y termino

Ejercicio 5:



En la red de la figura se interconectan 4 subredes mediante 4 routers, que corren el protocolo OSPF.

- Muestre los mensajes OSPF que envía el router R1 al resto.
- Mostrar la tabla de forwarding para el router R1. Asignar direcciones IPs a todas las interfaces que faltan.

Ejercicio 5: Mensaje OSPF

- a. Muestre el mensajes OSPF que envía el router R1 al resto.

Ejercicio 5: Mensaje OSPF

- a. Muestre el mensajes OSPF que envía el router R1 al resto.

ID:	R1
TTL:	X
#Seq:	W
Red	Costo
161.139.21.64/26	10
192.168.5.0/30	10
192.168.1.0/30	1000

Ejercicio 5: Tabla de forwarding

- b. Mostrar la tabla de forwarding para el router R1. Asignar direcciones IPs a todas las interfaces que haga falta.

Ejercicio 5: Tabla de forwarding

- b. Mostrar la tabla de forwarding para el router R1. Asignar direcciones IPs a todas las interfaces que haga falta.

R1	
Red	Próximo Salto
161.139.21.64/26	IF0/0
192.168.5.0/30	IF0/1
192.168.1.0/30	IF0/2
161.139.21.128/26	192.168.5.1
161.139.21.0/26	192.168.5.1
161.139.21.192/26	192.168.5.1
192.168.2.0/30	192.168.5.1
192.168.3.0/30	192.168.5.1
192.168.4.0/30	192.168.5.1

Referencias

- Computer Networks, (Fifth Edition): A Systems Approach. Larry L. Peterson, Bruce S. Davie 2011. Epígrafe 3.4
- Redes de computadoras (5ta edición) Andrew S. Tanenbaum, David J. Wetherall 2013. Capítulo 5.

¿Preguntas?

