

Implementación y análisis de RIPv2 en GNS3 con Wireshark

Tarea 1

Bastían Rubio Moya

COM4102 – Redes

Escuela de Ingeniería, Universidad de O'Higgins

03, Octubre, 2023

bastian.rubio@pregrado.uoh.cl

Abstract—Este proyecto de redes se centra en la implementación y análisis del protocolo de enrutamiento RIP versión 2, con el propósito de establecer una comunicación entre las interfaces de los routers R0 y R2, a través del router R1. La configuración precisa de RIP versión 2 en ambos dispositivos sirve como punto de partida, abarcando la definición de redes, subredes y la activación del protocolo en las interfaces pertinentes. Esto garantiza un intercambio fluido de información y el mantenimiento de tablas de enrutamiento coherentes entre los routers.

Además, se llevará a cabo un análisis exhaustivo de la operación de RIP versión 2 utilizando las herramientas GNS3 y Wireshark. Se capturarán y analizarán los paquetes de enrutamiento intercambiados entre R0 y R2, brindando información esencial sobre cómo el protocolo actualiza la información de enrutamiento y toma decisiones de enrutamiento. Este estudio tiene como objetivo comprender el rendimiento del protocolo y su capacidad de adaptación a los cambios en la topología de la red. En otras palabras, este proyecto se enfoca en implementar y evaluar RIP versión 2 para habilitar una comunicación efectiva entre las interfaces de los routers R0 y R2, haciendo uso de GNS3 y Wireshark como herramientas clave de análisis.

I. INTRODUCCIÓN

En el mundo de redes, la habilidad para enrutar datos de manera efectiva y eficiente es esencial para el funcionamiento óptimo de cualquier infraestructura de comunicación. Uno de los protocolos de enrutamiento ampliamente utilizados es el RIP (Routing Information Protocol) en su versión 2, el cual ha demostrado ser valioso en la gestión de rutas en redes IP.

Este proyecto se enfoca en: establecer una comunicación fluida entre las interfaces de dos routers, R0 y R2 (los cuales están conectados a otro router R1), mediante la implementación de RIP versión 2. Para abordar esto de manera completa, primero se realizará una configuración inicial de los routers sin la activación de RIP versión 2. Posteriormente, se llevará a cabo la configuración y activación de RIP versión 2 en todos dispositivos.

Cabe destacar que la red tiene conexiones específicas: R0 se conecta a R1 a través de una interfaz serial, mientras que la conexión entre R1 y R2 se realiza mediante interfaces Fast Ethernet. Esta distinción de interfaces añade una dimensión importante a la configuración y operación de RIP versión 2

en el proyecto, lo que nos permitirá observar la diferencia de velocidades entre una conexión serial y una Fast Ethernet.

Además, para llevar a cabo este proyecto se utilizará GNS3 y se realizará un análisis de la operación de RIP versión 2 utilizando Wireshark. A través de la captura y el análisis de los paquetes de enrutamiento intercambiados entre R0 y R2, se buscará comprender cómo este protocolo gestiona la información de enrutamiento.

Resumiendo, este proyecto se centra en implementar y analizar el protocolo RIP versión 2 para la comunicación efectiva entre las interfaces de los routers R0 y R2, considerando las conexiones específicas entre R0, R1 y R2 a través de interfaces serial y Fast Ethernet. La configuración inicial sin RIP versión 2 y la posterior comparación de resultados, junto con el uso de herramientas como GNS3 y Wireshark, permitirá un análisis detallado de su funcionamiento y su capacidad para gestionar la información de enrutamiento en un entorno de red específico, así como la observación de la diferencia de velocidades entre las interfaces serial y Fast Ethernet.

II. MARCO TEÓRICO

A continuación definiremos todos los conceptos necesarios para poder entender esta tarea:

A. Máscara de subred

Una máscara de subred es un número binario que se utiliza en redes para dividir una red IP en subredes más pequeñas. Su función es determinar qué parte de una dirección IP pertenece a la red y qué parte pertenece a los hosts dentro de esa red. Por ejemplo, si tenemos una dirección IP 192.168.1.10 y una máscara de subred 255.255.255.0, los primeros 24 bits de la dirección IP son la parte de red, y los últimos 8 bits son del host dentro de esa red.

B. Router

Un router es un dispositivo de red que se utiliza para conectar diferentes redes informáticas entre sí y dirigir el tráfico de datos entre ellas. Posee dos grandes funciones, la primera es **Forwarding** la cual es una función propia del switch pero se apoya en las tablas de ruteo para hacer

conmutación, dicho de otra forma, es redireccionar paquetes e ir moviéndolos de origen a destino. Y la segunda es **Routing** que es comunicarse con los otros routers para mantener las tablas de ruteo.

C. Loopback

Es una interfaz virtual de red que se utiliza para probar la conectividad y la funcionalidad de red en una computadora o dispositivo sin necesidad de una conexión física a una red externa. La dirección IP de loopback siempre se refiere a la propia máquina.

D. Protocolos

1) **RIP**: El Protocolo de Información de Enrutamiento (Routing Information Protocol) es un protocolo de enrutamiento utilizado en redes de computadoras para intercambiar información de enrutamiento y tomar decisiones sobre la mejor ruta para enviar paquetes de datos dentro de una red IP. RIP es uno de los protocolos de enrutamiento más antiguos y simples, diseñado para redes pequeñas y medianas. Algunas características son:

- **Métrica de Conteo de Saltos**: Métrica basada en el "conteo de saltos" para determinar la mejor ruta. Cada salto entre routers se cuenta como una unidad, y RIP elige la ruta con el menor número de saltos como la mejor. (Con un máximo de 15 saltos)
- **Actualizaciones Periódicas**: Cada 30 segundos envía actualizaciones de enrutamiento, las cuales contienen información sobre las rutas conocidas por el router.
- **Divulgación de Cambios**: Cuando un router detecta un cambio en la red, divulga esa información a otros routers.
- **Loop Prevention**: Técnicas como la cuenta regresiva (TTL, Time to Live) en los paquetes de enrutamiento y el "split horizon" para prevenir bucles de enrutamiento.

2) **RIPv2**: El Protocolo de Información de Enrutamiento versión 2 (RIPv2 o RIP versión 2) es una evolución del Protocolo RIP original. A diferencia de RIPv1, RIPv2 ofrece varias mejoras, como lo son:

- **Soporte para Subredes**: Permite la división de redes en subredes más pequeñas
- **Autenticación**: Los routers pueden verificar la autenticidad de las actualizaciones de enrutamiento que reciben de otros routers.
- **Soporte para IPv6**
- **Multicasting**: Utiliza direcciones multicast para enviar actualizaciones de enrutamiento en lugar de difusión, reduciendo la carga en la red
- **Eliminación de Broadcasts**: Elimina la necesidad de utilizar direcciones de difusión para transmitir actualizaciones.
- **Soporte para VLSM**: (Variable Length Subnet Masking) Puede manejar subredes con máscaras de longitud variable, permitiendo un uso más eficiente de las direcciones IP en una red.

3) **IP**: El Protocolo de Internet (Internet Protocol) es un conjunto de reglas y estándares que rigen la forma en que los datos se envían, reciben y direccionan a través de Internet y otras redes de computadoras. Es un componente fundamental de la arquitectura de Internet y se utiliza para la identificación y el enrutamiento de paquetes de datos en la red. Algunas de sus características son:

- **Direcciones IP**: Cada dispositivo conectado a una red, ya sea en Internet o en una red local, se le asigna una dirección IP única que lo identifica en la red.
- **Enrutamiento**: Encargada de determinar la mejor ruta para enviar paquetes de datos desde su origen hasta su destino a través de una serie de routers intermedios.
- **Fragmentación y Reensamblaje**: Permite dividir paquetes de datos en fragmentos (frames) más pequeños cuando es necesario para su transporte. Los fragmentos se reensamblan en el destino para reconstruir el mensaje original.
- **IPv4 e IPv6**: Existen dos versiones principales de IP. IPv4 ha sido la versión dominante durante décadas, pero debido a la escasez de direcciones IPv4 y otras limitaciones, IPv6 fue desarrollado para solucionar dicho problema.

E. Conexiones

- **Fast Ethernet**: Estándar de red que se refiere a una velocidad de transferencia de datos en una red Ethernet de 100 megabits por segundo (Mbps). Se denota como "Fast Ethernet". Esta velocidad es aproximadamente diez veces más rápida que la Ethernet original, que opera a 10 Mbps.
- **Serial**: Refiere a una conexión de datos que transfiere información secuencialmente, un bit a la vez, a través de un solo conductor o un par de conductores (uno para transmitir y otro para recibir).

F. Códigos en la terminal

Existen diversos códigos en la terminal, pero mencionaremos los que se ocuparan para este proyecto

- **C**: El "C" en la salida del comando "show ip route" indica una ruta directamente conectada.
- **R**: El "R" en la salida del comando "show ip route" indica una ruta estática.

G. Comandos utilizados

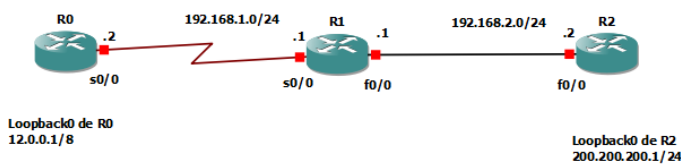
- **interface**: Configurar interfaces
- **ping**: Envía paquetes de solicitud de eco ICMP (Internet Control Message Protocol) a un host de destino y espera recibir respuestas de eco ICMP del host.
- **sh ip int br**: Se utiliza para mostrar una lista resumida de las interfaces de red en el dispositivo y proporciona información clave sobre el estado de cada interfaz (abreviación de show ip interface brief).
- **config t**: Para ingresar al modo de configuración global.
- **ip address**: Configura la dirección IP.
- **show ip route**: Mostrar la tabla de enrutamiento IP actual

- **no shutdown:** Activar una interfaz de red que previamente había sido desactivada
- **network:** Para especificar la red que se debe anunciar a través del protocolo RIP

III. METODOLOGÍA

A. Preparación

Para llevar a cabo este proyecto, debemos tener instalado GNS3 y una imagen de router. Lo primero que debemos hacer es crear un nuevo proyecto, y agregarle los 3 routers para llevar a cabo este experimento. Una vez agregados los routers (Para efectos prácticos los routers se resumirán con una R y el número)debemos conectarlos de la siguiente forma R0 a R1 con una conexión serial, luego R1 a R2 con una conexión Fast Ethernet, asignarle las labels (es decir, los labels de las IP's que tendrán) correspondientes y debería quedar de la siguiente forma:



Si aplicamos el comando "sh ip int br" a cada uno de los routers nos debería aparecer lo siguiente:

```
R0#sh ip int br
```

Interface	IP-Address	OK?	Method	Status	Protocol
FastEthernet0/0	unassigned	YES	unset	administratively down	down
Serial0/0	unassigned	YES	unset	administratively down	down
FastEthernet0/1	unassigned	YES	unset	administratively down	down

Fig. 1. Router 0 Inicializado

```
R1#sh ip int br
```

Interface	IP-Address	OK?	Method	Status	Protocol
FastEthernet0/0	unassigned	YES	unset	administratively down	down
Serial0/0	unassigned	YES	unset	administratively down	down
FastEthernet0/1	unassigned	YES	unset	administratively down	down

Fig. 2. Router 1 Inicializado

```
R2#sh ip int br
```

Interface	IP-Address	OK?	Method	Status	Protocol
FastEthernet0/0	unassigned	YES	unset	administratively down	down
Serial0/0	unassigned	YES	unset	administratively down	down
FastEthernet0/1	unassigned	YES	unset	administratively down	down

Fig. 3. Router 2 Inicializado

Esto los que nos indica es que ninguno de los tres routers se ha configurado o se le han asignado las direcciones IP

B. Codificación Routers

Una vez inicializados todos los routers, lo que debemos hacer es configurar las interfaces para cada uno de los routers (asignándoles las IP's correspondientes) como se muestra a continuación.

```
R0#config t
Enter configuration commands, one per line. End with CNTL/Z.
R0(config)#interface s0/0
R0(config-if)#ip address 192.168.1.2 255.255.255.0
R0(config-if)#no shutdown
R0(config-if)#exit
R0(config)#
*Mar 1 00:02:51.767: %LINK-3-UPDOWN: Interface Serial0/0, changed state to up
R0(config)#
*Mar 1 00:02:52.771: %LINEPROTO-5-UPDOWN: Line protocol on Interface Serial0/0, changed state to up
R0(config)#interface lo0
R0(config-if)#ip address
*Mar 1 00:03:04.843: %LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback0, changed state to up
R0(config-if)#ip address 12.0.0.1 255.0.0.0
R0(config-if)#
*Mar 1 00:03:22.335: %LINEPROTO-5-UPDOWN: Line protocol on Interface Serial0/0, changed state to down
R0(config-if)#no shutdown
R0(config-if)#exit
R0(config)#exit
R0#
*Mar 1 00:04:05.599: %SYS-5-CONFIG_I: Configured from console by console
R0#sh ip int br
```

Interface	IP-Address	OK?	Method	Status	Protocol
FastEthernet0/0	unassigned	YES	unset	administratively down	down
Serial0/0	192.168.1.2	YES	manual	up	down
FastEthernet0/1	unassigned	YES	unset	administratively down	down
Loopback0	12.0.0.1	YES	manual	up	up

Fig. 4. Router 0

```
R1#config t
Enter configuration commands, one per line. End with CNTL/Z.
R1(config)#interface s0/0
R1(config-if)#ip address 192.168.1.1 255.255.255.0
R1(config-if)#no shutdown
R1(config-if)#exit
*Mar 1 00:05:09.343: %LINK-3-UPDOWN: Interface Serial0/0, changed state to up
R1(config-if)#exit
*Mar 1 00:05:10.347: %LINEPROTO-5-UPDOWN: Line protocol on Interface Serial0/0, changed state to up
R1(config-if)#exit
R1(config)#interface f0/0
R1(config-if)#ip address 192.168.2.1 255.255.255.0
R1(config-if)#no shutdown
R1(config-if)#
*Mar 1 00:05:48.863: %LINK-3-UPDOWN: Interface FastEthernet0/0, changed state to up
*Mar 1 00:05:49.863: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/0, changed state to up
R1(config-if)#exit
R1(config)#exit
R1#
*Mar 1 00:05:54.095: %SYS-5-CONFIG_I: Configured from console by console
R1#sh ip int br
```

Interface	IP-Address	OK?	Method	Status	Protocol
FastEthernet0/0	192.168.2.1	YES	manual	up	up
Serial0/0	192.168.1.1	YES	manual	up	up
FastEthernet0/1	unassigned	YES	unset	administratively down	down

Fig. 5. Router 1

```
R2#config t
Enter configuration commands, one per line. End with CNTL/Z.
R2(config)#interface f0/0
R2(config-if)#ip address 192.168.2.2 255.255.255.0
R2(config-if)#no shutdown
R2(config-if)#exit
R2(config)#
*Mar 1 00:10:17.119: %LINK-3-UPDOWN: Interface FastEthernet0/0, changed state to up
*Mar 1 00:10:18.119: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/0, changed state to up
R2(config)#interface lo0
R2(config-if)#ip a
*Mar 1 00:10:36.943: %LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback0, changed state to up
R2(config-if)#ip address 200.200.200.1 255.255.255.0
R2(config-if)#no shutdown
R2(config-if)#exit
R2(config)#exit
R2#
*Mar 1 00:11:01.159: %SYS-5-CONFIG_I: Configured from console by console
R2#sh ip int br
```

Interface	IP-Address	OK?	Method	Status	Protocol
FastEthernet0/0	192.168.2.2	YES	manual	up	up
Serial0/0	unassigned	YES	unset	administratively down	down
FastEthernet0/1	unassigned	YES	unset	administratively down	down
Loopback0	200.200.200.1	YES	manual	up	up

Fig. 6. Router 2

Como se puede apreciar cada router ya posee una asignación de dirección IP correspondiente, además de cada router se debe configurar de manera distinta, ya que, algunos están conectados mediante serial mientras que otros con Fast Ethernet, hasta este punto deberían estar conectados los tres routers.

C. Tabla de enrutamiento

Para comprobar que todos están conectados, mostraremos la tabla de enrutamiento de cada router para ver las conexiones que poseen (otra forma de verlo es con `sh ip int br`)

```
R0#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
E1 - OSPF external type 1, E2 - OSPF external type 2
i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
ia - IS-IS inter area, * - candidate default, U - per-user static route
o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

C    12.0.0.0/8 is directly connected, Loopback0
C    192.168.1.0/24 is directly connected, Serial0/0
```

Fig. 7. Tabla enrutamiento Router 0

```
R1#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
E1 - OSPF external type 1, E2 - OSPF external type 2
i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
ia - IS-IS inter area, * - candidate default, U - per-user static route
o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

C    192.168.1.0/24 is directly connected, Serial0/0
C    192.168.2.0/24 is directly connected, FastEthernet0/0
```

Fig. 8. Tabla enrutamiento Router 1

```
R2#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
E1 - OSPF external type 1, E2 - OSPF external type 2
i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
ia - IS-IS inter area, * - candidate default, U - per-user static route
o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

C    200.200.200.0/24 is directly connected, Loopback0
C    192.168.2.0/24 is directly connected, FastEthernet0/0
```

Fig. 9. Tabla enrutamiento Router 2

Con esto lo que podemos observar que la "C", en la terminal significa que están conectados a la dirección IP siguiente. Por ejemplo en el router 0, está conectado a 12.0.0.0 por un serial. Y así con los demás routers.

D. Activación de RIPv2

Ahora lo que haremos será intentar hacer ese mismo ping pero previamente configuraremos al protocolo RIPv2, entonces configuraremos todos los routers de la siguiente forma.

```
R0#config t
Enter configuration commands, one per line. End with CNTL/Z.
R0(config)#router rip
R0(config-router)#version 2
R0(config-router)#network 12.0.0.0
R0(config-router)#network 192.168.1.0
R0(config-router)#exit
R0(config)#exit
```

Fig. 10. Configuración RIPv2 R0

```
R1#config t
Enter configuration commands, one per line. End with CNTL/Z.
R1(config)#router rip
R1(config-router)#version 2
R1(config-router)#network 192.168.1.0
R1(config-router)#network 192.168.2.0
R1(config-router)#exit
R1(config)#exit
```

Fig. 11. Configuración RIPv2 R1

```
R2#config t
Enter configuration commands, one per line. End with CNTL/Z.
R2(config)#router rip
R2(config-router)#version 2
R2(config-router)#network 200.200.200.0
R2(config-router)#network 192.168.2.0
R2(config-router)#exit
R2(config)#exit
```

Fig. 12. Configuración RIPv2 R2

E. Tablas de enrutamiento post RIPv2

Una vez configurados procedemos a revisar de nuevo las nuevas tablas de ruteo de cada router.

```
R0#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
E1 - OSPF external type 1, E2 - OSPF external type 2
i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
ia - IS-IS inter area, * - candidate default, U - per-user static route
o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

C    12.0.0.0/8 is directly connected, Loopback0
C    192.168.1.0/24 is directly connected, Serial0/0
R    192.168.2.0/24 [120/1] via 192.168.1.1, 00:00:25, Serial0/0
```

Fig. 13. Tabla enrutamiento Router 0

```
R1#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
E1 - OSPF external type 1, E2 - OSPF external type 2
i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
ia - IS-IS inter area, * - candidate default, U - per-user static route
o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

R    200.200.200.0/24 [120/1] via 192.168.2.2, 00:00:18, FastEthernet0/0
R    12.0.0.0/8 [120/1] via 192.168.1.2, 00:00:26, Serial0/0
C    192.168.1.0/24 is directly connected, Serial0/0
C    192.168.2.0/24 is directly connected, FastEthernet0/0
```

Fig. 14. Tabla enrutamiento Router 1

```
R2#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
E1 - OSPF external type 1, E2 - OSPF external type 2
i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
ia - IS-IS inter area, * - candidate default, U - per-user static route
o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

C    200.200.200.0/24 is directly connected, Loopback0
R    12.0.0.0/8 [120/2] via 192.168.2.1, 00:00:29, FastEthernet0/0
R    192.168.1.0/24 [120/1] via 192.168.2.1, 00:00:29, FastEthernet0/0
C    192.168.2.0/24 is directly connected, FastEthernet0/0
```

Fig. 15. Tabla enrutamiento Router 2

"R" indica que se trata de una ruta estática. Luego, se muestra la dirección de red o subred de destino (en este

caso haremos como ejemplo a R0, 12.0.0.0/8), seguida de información adicional sobre la métrica ([120/2]) y la puerta de enlace predeterminada a través de la cual se enruta el tráfico (via 192.168.2.1).

F. Wireshark

Por último lo que haremos será capturar los paquetes mientras hacemos un ping desde R0 a R2 a través de Wireshark.

IV. RESULTADOS

A. Resultados de ping antes de protocolo RIPv2

Como se mostró en pasos anteriores cada router debería estar conectado a otro, es decir, R0 debería estar conectado a R1 y R1 a R2, entonces R0 podría mandarle paquetes a R2, para comprobar esto lo que haremos será realizar ping desde R0 a las interfaces de R2, y posteriormente de R2 a las interfaces de R0. Entonces lo que debería ocurrir, es que el paquete se mande de R0 a R1, luego pase por R1 hacia R2.

```
R0#ping 192.168.2.2

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.2.2, timeout is 2 seconds:
.....
Success rate is 0 percent (0/5)
```

Fig. 16. Ping R0 a 192.168.2.2

```
R0#ping 200.200.200.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 200.200.200.1, timeout is 2 seconds:
.....
Success rate is 0 percent (0/5)
```

Fig. 17. Ping R0 a 200.200.200.1

```
R2#ping 192.168.1.2

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.1.2, timeout is 2 seconds:
.....
Success rate is 0 percent (0/5)
```

Fig. 18. Ping R2 a 192.168.1.2

```
R2#ping 12.0.0.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 12.0.0.1, timeout is 2 seconds:
.....
Success rate is 0 percent (0/5)
```

Fig. 19. Ping R2 a 12.0.0.1

B. Resultados de ping después de protocolo RIPv2

En esta parte ya está aplicado el protocolo RIPv2 en cada uno de los routers. (Colocaremos solamente dos imágenes una de cada lado para demostrar que hay conexiones de los dos lados, tanto de R0 a R2 como de R2 a R0)

```
R0#ping 192.168.2.2

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.2.2, timeout is 2 seconds:
!!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 36/38/44 ms
```

Fig. 20. Ping R0 a 192.168.2.2

```
R2#ping 192.168.1.2

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.1.2, timeout is 2 seconds:
!!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 68/72/76 ms
```

Fig. 21. Ping Ping R2 a 192.168.1.2

C. Resultados pings capturados en wireshark

A continuación se mostrarán algunos extractos de los paquetes que fueron capturados (El archivo completo se envió junto al código y todo lo demás). Se puede observar en la Figura 22 el ping que se capturó en wireshark y en la Figura 23 el protocolo RIPv2.

No.	Time	Source	Destination	Protocol	Length	Info
97	0.488978	192.168.2.2	12.0.0.1	ICMP	114	Echo (ping) request seq=27512, ttl=255 (no response found)
98	0.489978	192.168.2.2	12.0.0.1	ICMP	114	Echo (ping) request seq=27512, ttl=254 (reply in 99)
99	0.490978	12.0.0.1	192.168.2.2	ICMP	114	Echo (ping) reply seq=27512, ttl=255 (request in 98)
100	0.501438	12.0.0.1	192.168.2.2	ICMP	114	Echo (ping) reply seq=27512, ttl=254
101	0.517488	192.168.2.2	12.0.0.1	ICMP	114	Echo (ping) request seq=37768, ttl=255 (no response found)
102	0.517488	192.168.2.2	12.0.0.1	ICMP	114	Echo (ping) request seq=37768, ttl=254 (reply in 103)
103	0.518048	12.0.0.1	192.168.2.2	ICMP	114	Echo (ping) reply seq=37768, ttl=255 (request in 102)
104	0.532974	12.0.0.1	192.168.2.2	ICMP	114	Echo (ping) reply seq=37768, ttl=254
105	0.547024	192.168.2.2	12.0.0.1	ICMP	114	Echo (ping) request seq=47824, ttl=255 (no response found)
106	0.547024	192.168.2.2	12.0.0.1	ICMP	114	Echo (ping) request seq=47824, ttl=254 (reply in 107)
107	0.548024	12.0.0.1	192.168.2.2	ICMP	114	Echo (ping) reply seq=47824, ttl=255 (request in 106)
108	0.553932	12.0.0.1	192.168.2.2	ICMP	114	Echo (ping) reply seq=47824, ttl=254
109	0.569044	192.168.2.2	192.168.2.2	ICMP	114	Echo (ping) request seq=0/0, ttl=255 (no response found)
110	0.569044	192.168.2.2	192.168.2.2	ICMP	114	Echo (ping) request seq=0/0, ttl=254 (reply in 111)
111	0.569044	192.168.2.2	192.168.2.2	ICMP	114	Echo (ping) request seq=0/0, ttl=253 (request in 117)
112	0.569044	192.168.2.2	192.168.2.2	ICMP	114	Echo (ping) request seq=0/0, ttl=252 (request in 117)
113	0.569044	192.168.2.2	192.168.2.2	ICMP	114	Echo (ping) request seq=0/0, ttl=251 (no response found)
114	0.569044	192.168.2.2	192.168.2.2	ICMP	114	Echo (ping) request seq=0/0, ttl=250 (reply in 123)
115	0.569044	192.168.2.2	192.168.2.2	ICMP	114	Echo (ping) request seq=0/0, ttl=249 (request in 122)
116	0.569044	192.168.2.2	192.168.2.2	ICMP	114	Echo (ping) request seq=0/0, ttl=248 (reply in 127)
117	0.569044	192.168.2.2	192.168.2.2	ICMP	114	Echo (ping) request seq=0/0, ttl=247 (no response found)
118	0.569044	192.168.2.2	192.168.2.2	ICMP	114	Echo (ping) request seq=0/0, ttl=246 (reply in 126)
119	0.569044	192.168.2.2	192.168.2.2	ICMP	114	Echo (ping) request seq=0/0, ttl=245 (no response found)
120	0.569044	192.168.2.2	192.168.2.2	ICMP	114	Echo (ping) request seq=0/0, ttl=244 (reply in 134)
121	0.569044	192.168.2.2	192.168.2.2	ICMP	114	Echo (ping) request seq=0/0, ttl=243 (no response found)
122	0.569044	192.168.2.2	192.168.2.2	ICMP	114	Echo (ping) request seq=0/0, ttl=242 (reply in 134)
123	0.569044	192.168.2.2	192.168.2.2	ICMP	114	Echo (ping) request seq=0/0, ttl=241 (no response found)
124	0.569044	192.168.2.2	192.168.2.2	ICMP	114	Echo (ping) request seq=0/0, ttl=240 (reply in 134)
125	0.569044	192.168.2.2	192.168.2.2	ICMP	114	Echo (ping) request seq=0/0, ttl=239 (no response found)
126	0.569044	192.168.2.2	192.168.2.2	ICMP	114	Echo (ping) request seq=0/0, ttl=238 (reply in 134)
127	0.569044	192.168.2.2	192.168.2.2	ICMP	114	Echo (ping) request seq=0/0, ttl=237 (no response found)
128	0.569044	192.168.2.2	192.168.2.2	ICMP	114	Echo (ping) request seq=0/0, ttl=236 (reply in 134)
129	0.569044	192.168.2.2	192.168.2.2	ICMP	114	Echo (ping) request seq=0/0, ttl=235 (no response found)
130	0.569044	192.168.2.2	192.168.2.2	ICMP	114	Echo (ping) request seq=0/0, ttl=234 (reply in 132)
131	0.569044	192.168.2.2	192.168.2.2	ICMP	114	Echo (ping) request seq=0/0, ttl=233 (no response found)
132	0.569044	192.168.2.2	192.168.2.2	ICMP	114	Echo (ping) request seq=0/0, ttl=232 (reply in 130)
133	0.569044	192.168.2.2	192.168.2.2	ICMP	114	Echo (ping) request seq=0/0, ttl=231 (no response found)
134	0.569044	192.168.2.2	192.168.2.2	ICMP	114	Echo (ping) request seq=0/0, ttl=230 (reply in 130)
135	0.569044	192.168.2.2	192.168.2.2	ICMP	114	Echo (ping) request seq=0/0, ttl=229 (no response found)
136	0.569044	192.168.2.2	192.168.2.2	ICMP	114	Echo (ping) request seq=0/0, ttl=228 (reply in 134)
137	0.569044	192.168.2.2	192.168.2.2	ICMP	114	Echo (ping) request seq=0/0, ttl=227 (no response found)
138	0.569044	192.168.2.2	192.168.2.2	ICMP	114	Echo (ping) request seq=0/0, ttl=226 (reply in 134)

Fig. 22. Ping Wireshark

No.	Time	Source	Destination	Protocol	Length	Info
5	13.003522	192.168.1.1	224.0.0.9	RIPv2	76	Response
18	22.318972	192.168.1.2	224.0.0.9	RIPv2	56	Response
19	22.472509	192.168.1.2	224.0.0.9	RIPv2	56	Response
12	24.152564	192.168.2.1	224.0.0.9	RIPv2	86	Response
21	41.458483	192.168.1.1	224.0.0.9	RIPv2	76	Response
25	50.557782	192.168.1.2	224.0.0.9	RIPv2	56	Response
27	51.397993	192.168.2.2	224.0.0.9	RIPv2	66	Response
28	53.572952	192.168.2.1	224.0.0.9	RIPv2	86	Response
68	72.061377	192.168.1.1	224.0.0.9	RIPv2	76	Response
82	76.536548	192.168.1.2	224.0.0.9	RIPv2	56	Response
86	81.288048	192.168.2.2	224.0.0.9	RIPv2	66	Response
87	81.813344	192.168.2.1	224.0.0.9	RIPv2	86	Response
135	101.977495	192.168.1.1	224.0.0.9	RIPv2	76	Response
136	104.233766	192.168.1.2	224.0.0.9	RIPv2	56	Response
138	109.964479	192.168.2.2	224.0.0.9	RIPv2	66	Response
144	112.467487	192.168.2.1	224.0.0.9	RIPv2	86	Response

Fig. 23. RIPv2

V. ANÁLISIS

A. Análisis de ping antes y después del protocolo RIPv2

Podemos observar que la conectividad entre R0 y R2 funciona correctamente cuando se utiliza RIPv2 pero no cuando intentamos hacer un ping directo entre ellos (antes de agregar RIPv2), esto nos sugiere que el protocolo de enrutamiento RIPv2 está funcionando correctamente y anuncia las rutas entre los routers R0, R1 y R2. Ahora, la posible explicación de porque con RIPv2 funciona y antes no, es la siguiente:

Cuando se utiliza RIPv2, el protocolo de enrutamiento se encarga de anunciar y mantener las rutas automáticamente

entre los routers. Entonces RIPv2 anuncia con éxito las rutas entre R0, R1 y R2, como consecuencia, las rutas necesarias para la comunicación deberían estar presentes y configuradas automáticamente en las tablas de enrutamiento de cada router.

Ahora, con RIPv2 habilitado, RIPv2 está anunciando y manteniendo las rutas automáticamente. Esto significa que R0 aprende la ruta a la red de R2 a través de R1 (ya que R1 está anunciando esa ruta a R0 a través de RIPv2). Por lo tanto, cuando hacemos ping desde R0 a R2, el tráfico va a través de R1 como router intermedio, siguiendo la ruta anunciada por R1.

Por otro lado, sin RIPv2 (ping directo), antes de configurar RIPv2 y cuando hicimos un ping directo desde R0 a R2, R0 puede que no conozca la ruta correcta para llegar a R2, ya que no hay anuncios de ruta dinámica a través de RIPv2. En este caso, R0 podría intentar enrutar el tráfico directamente a R2 a través de su tabla de enrutamiento estática o predeterminada, pero es posible que esa ruta no esté configurada correctamente.

Ahora, algo curioso que podemos observar entre los pings realizado de R0 a R2, al momento de ser efectivos (en otras palabras, cuando si hubo comunicación entre routers) podemos notar que al realizar el ping desde R0 a R2, tiene mejores metricas que desde cuando realizamos el mismo ping de R2 a R0, esto se debe deber a las interfaces seriales generalmente tienen velocidades más bajas en comparación con las interfaces Fast Ethernet. Entonces se pudo haber generado algun tipo de cuello de botella.

B. Análisis de Wireshark

Partiendo por la Figura 22 podemos notar que tiene distintos campos (o labels), los cuales nos indican el Número, el tiempo que demoró, el Origen (router de origen) y el destino, podemos ver su protocolo ICMP (Internet Control Message Protocol) el cuál es un protocolo de comunicación fundamental en las redes de Internet, el largo que tiene que varía de 104 a 114 y por último la información, si es un request o un reply, la secuencia, el id, el ttl (time to live) que se mencionó anteriormente y entre paréntesis si se respondió o no. Cabe destacar que contiene un color de fondo distinto a los demás, entre otras cosas. Ahora pasando a la Figura 23 podemos notar de entrada que tiene un fondo de color rojo, podemos notar que al igual que en el ping, posee un número, el tiempo, la fuente y algo que llama la atención es que todos los paquetes van dirigidos a la dirección IP 224.0.0.9, la cual se utiliza para la transmisión de actualizaciones de enrutamiento entre routers que participan en la misma red multicast RIP, en el caso de RIPv1 la dir IP es 224.0.0.2 mientras que en la RIPv2 es 224.0.0.9, lo que facilita la comunicación y el intercambio de información de enrutamiento en una red. También se muestra el protocolo, el largo y por último la información, que en este caso son todos respuestas.

VI. CONCLUSIONES

Para concluir, en este proyecto de redes, se logró implementar y analizar el protocolo de enrutamiento RIPv2 para establecer una comunicación efectiva entre las interfaces de

los routers R0 y R2 a través del router R1. A continuación, se resumen los principales hallazgos y conclusiones:

- 1) **Implementación de RIPv2:** Se configuraron los routers R0, R1 y R2 para utilizar el protocolo de enrutamiento RIPv2. Lo que permitió que los routers compartieran información de enrutamiento de manera automática y eficiente. Pudiendo comparar las conexiones con y sin el uso de RIPv2.
- 2) **Conectividad:** Después de la implementación de RIPv2, se realizó una prueba de conectividad la cual resulto exitosa, es decir, que los paquetes de datos pudieron viajar de R0 a R2 a través de R1, lo que demuestra que el protocolo de enrutamiento estaba funcionando correctamente.
- 3) **Análisis mediante el uso de Wireshark:** Se utilizó Wireshark para capturar y analizar paquetes que estaban siendo intercambiados entre los routers. Proporcionando información sobre cómo RIPv2 actualiza la información de enrutamiento y toma decisiones de enrutamiento en tiempo real. Además de poder ver las direcciones de entrada y salida, junto a otros campos.
- 4) **Diferencias de velocidad entre Fast Ethernet y Serial:** Pudimos observar que la interfaz serial generalmente tienen velocidades más bajas en comparación con las interfaces Fast Ethernet.
- 5) **Multicast:** Pudimos observar que RIPv2 utiliza la dirección IP multicast 224.0.0.9 para transmitir actualizaciones de enrutamiento. Simplificando la comunicación y el intercambio de información en una red.

Resumiendo los anteriores puntos, este proyecto logró su objetivo de implementar y evaluar el protocolo de enrutamiento RIPv2 para habilitar una comunicación efectiva entre los routers R0 y R2 (a través del router R1). La configuración y análisis detallado de RIPv2, junto con el uso de herramientas como GNS3 y Wireshark, proporcionaron una comprensión de su funcionamiento y la capacidad para gestionar la información de enrutamiento en una red. Este proyecto contribuye al conocimiento en el campo de las redes y el enrutamiento, brindando una experiencia práctica y valiosa en la implementación de protocolos de enrutamiento en un entorno de laboratorio.

REFERENCES

- [1] Cisco Topology. Disponible en: docs.gns3.com
- [2] Captura de tráfico. Disponible en: [Video Youtube](#)
- [3] Mascara de subred. Disponible en: [IBM/docs](#)
- [4] Protocolo RIPv2. Disponible en: [RFC 2453](#)
- [5] Protocolo IP. Disponible en: [RFC 791](#)
- [6] Fast Ethernet. Disponible en: [Fast Ethernet](#)
- [7] Dir 224.0.0.9. Disponible en: [networkengineering](#)