

Práctica 1.5. RIP y BGP

Objetivos

En esta práctica se afianzan los conceptos elementales del encaminamiento. En particular, se estudia un protocolo de encaminamiento interior y otro exterior: RIP (*Routing Information Protocol*) y BGP (*Border Gateway Protocol*).

Existen muchas implementaciones de los protocolos de encaminamiento. En esta práctica vamos a utilizar Quagga, que actualmente implementa RIP (versiones 1 y 2), RIPng, OSPF, OSPFv3, IS-IS y BGP. Quagga está estructurado en diferentes servicios (uno para cada protocolo) controlados por un servicio central (Zebra) que hace de interfaz entre la tabla de reenvío del *kernel* y las tabla de encaminamiento de cada protocolo.

Todos los ficheros de configuración han de almacenarse en el directorio `/etc/quagga`. La sintaxis de estos ficheros es sencilla y está disponible en <http://quagga.net>. Revisar especialmente la correspondiente a RIP y BGP en <https://www.quagga.net/docs/quagga.html>. Además, en `/usr/share/doc/quagga-0.99.22.4` hay ficheros de ejemplo.



Activar el **portapapeles bidireccional** (menú Dispositivos) en las máquinas virtuales.

Usar la opción de Virtualbox (menú Ver) para realizar **capturas de pantalla**.

La **contraseña** del usuario `cursoresdes` es `cursoresdes`.

Contenidos

Parte I. Protocolo interior: RIP

- Preparación del entorno

- Configuración del protocolo RIP

Parte II. Protocolo exterior: BGP

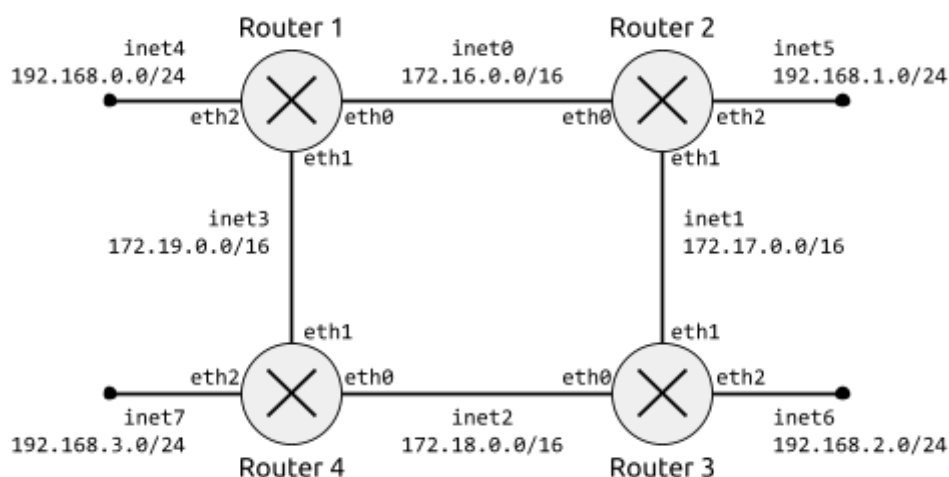
- Preparación del entorno

- Configuración del protocolo BGP

Parte I. Protocolo interior: RIP

Preparación del entorno

Configuraremos la topología de red que se muestra en la siguiente figura, donde cada encaminador (Router1...Router4) tiene tres interfaces, cada uno conectado a una red diferente.:



Al igual que en prácticas anteriores, usaremos la herramienta vtopo1 para construir automáticamente esta topología. A continuación se muestra el contenido del fichero de configuración de la topología:

```
netprefix inet
machine 1 0 0 1 3 2 4
machine 2 0 0 1 1 2 5
machine 3 0 2 1 1 2 6
machine 4 0 2 1 3 2 7
```

Para facilitar la configuración de las máquinas, la siguiente tabla muestra las direcciones de cada uno de los interfaces de los encaminadores:

Máquina virtual	Interfaz	Dirección de red	Dirección IP
Router1	eth0	172.16.0.0/16	172.16.0.1
	eth1	172.19.0.0/16	172.19.0.1
	eth2	192.168.0.0/24	192.168.0.1
Router2	eth0	172.16.0.0/16	172.16.0.2
	eth1	172.17.0.0/16	172.17.0.2
	eth2	192.168.1.0/24	192.168.1.2
Router3	eth0	172.18.0.0/16	172.18.0.3
	eth1	172.17.0.0/16	172.17.0.3
	eth2	192.168.2.0/24	192.168.2.3
Router4	eth0	172.18.0.0/16	172.18.0.4
	eth1	172.19.0.0/16	172.19.0.4
	eth2	192.168.3.0/24	192.168.3.4

Configurar todos los encaminadores según la figura y tabla anterior. Además, activar el reenvío de paquetes IPv4 igual que en la práctica 1.1. Después, comprobar:

- Que los encaminadores adyacentes son alcanzables, por ejemplo, Router1 puede hacer *ping* a Router2 y Router4.
- Que la tabla de reenvío de cada encaminador es la correcta e incluye una entrada para cada una de las tres redes a las que está conectado.

Configuración del protocolo RIP

Ejercicio 1. Configurar RIP en todos los encaminadores para que intercambien información:

- Crear un fichero `ripd.conf` en `/etc/quagga` con el contenido que se muestra a continuación.
- Iniciar el servicio RIP (y Zebra) con `service ripd start`.

Contenido del fichero `/etc/quagga/ripd.conf`:

```
# Activar el encaminamiento por RIP
router rip
# Definir la versión del protocolo que se usará
version 2
# Habilitar información de encaminamiento en redes asociadas a los interfaces
network eth0
network eth1
network eth2
```

Ejercicio 2. Consultar la tabla de encaminamiento de RIP y de Zebra en cada encaminador con el comando `vttysh` (`sudo vttysh -c "show ip rip"` y `sudo vttysh -c "show ip route"`). Comprobar también la tabla de reenvío de IPv4 con el comando `ip (ip route)`.

ROUTER 1:

[cursoredes@localhost ~]\$ sudo vttysh -c "show ip rip"

Codes: R - RIP, C - connected, S - Static, O - OSPF, B - BGP

Sub-codes:

(n) - normal, (s) - static, (d) - default, (r) - redistribute,
(i) - interface

Network	Next Hop	Metric From	Tag Time
C(i) 172.16.0.0/16	0.0.0.0	1 self	0
R(n) 172.17.0.0/16	172.16.0.2	2 172.16.0.2	0 02:50
R(n) 172.18.0.0/16	172.19.0.4	2 172.19.0.4	0 02:56
C(i) 172.19.0.0/16	0.0.0.0	1 self	0
C(i) 192.168.0.0/24	0.0.0.0	1 self	0
R(n) 192.168.1.0/24	172.16.0.2	2 172.16.0.2	0 02:50
R(n) 192.168.2.0/24	172.16.0.2	3 172.16.0.2	0 02:50
R(n) 192.168.3.0/24	172.19.0.4	2 172.19.0.4	0 02:56

[cursoredes@localhost ~]\$ sudo vttysh -c "show ip route"

Codes: K - kernel route, C - connected, S - static, R - RIP,

O - OSPF, I - IS-IS, B - BGP, A - Babel,

> - selected route, * - FIB route

```
C>* 127.0.0.0/8 is directly connected, lo
C>* 172.16.0.0/16 is directly connected, eth0
R>* 172.17.0.0/16 [120/2] via 172.16.0.2, eth0, 00:01:07
R>* 172.18.0.0/16 [120/2] via 172.19.0.4, eth1, 00:00:55
C>* 172.19.0.0/16 is directly connected, eth1
C>* 192.168.0.0/24 is directly connected, eth2
R>* 192.168.1.0/24 [120/2] via 172.16.0.2, eth0, 00:01:07
R>* 192.168.2.0/24 [120/3] via 172.16.0.2, eth0, 00:01:01
R>* 192.168.3.0/24 [120/2] via 172.19.0.4, eth1, 00:00:55
```

[cursoredes@localhost ~]\$ ip route

```
172.16.0.0/16 dev eth0 proto kernel scope link src 172.16.0.1
172.17.0.0/16 via 172.16.0.2 dev eth0 proto zebra metric 2
172.18.0.0/16 via 172.19.0.4 dev eth1 proto zebra metric 2
172.19.0.0/16 dev eth1 proto kernel scope link src 172.19.0.1
192.168.0.0/24 dev eth2 proto kernel scope link src 192.168.0.1
192.168.1.0/24 via 172.16.0.2 dev eth0 proto zebra metric 2
192.168.2.0/24 via 172.16.0.2 dev eth0 proto zebra metric 3
192.168.3.0/24 via 172.19.0.4 dev eth1 proto zebra metric 2
```

ROUTER 2:

[cursoredes@localhost ~]\$ sudo vtysh -c "show ip rip"

Codes: R - RIP, C - connected, S - Static, O - OSPF, B - BGP

Sub-codes:

(n) - normal, (s) - static, (d) - default, (r) - redistribute,
(i) - interface

Network	Next Hop	Metric From	Tag Time
C(i) 172.16.0.0/16	0.0.0.0	1 self	0
C(i) 172.17.0.0/16	0.0.0.0	1 self	0
R(n) 172.18.0.0/16	172.17.0.3	2 172.17.0.3	0 02:42
R(n) 172.19.0.0/16	172.16.0.1	2 172.16.0.1	0 02:34
R(n) 192.168.0.0/24	172.16.0.1	2 172.16.0.1	0 02:34
C(i) 192.168.1.0/24	0.0.0.0	1 self	0
R(n) 192.168.2.0/24	172.17.0.3	2 172.17.0.3	0 02:42
R(n) 192.168.3.0/24	172.16.0.1	3 172.16.0.1	0 02:34

[cursoredes@localhost ~]\$ sudo vtysh -c "show ip route"

Codes: K - kernel route, C - connected, S - static, R - RIP,

O - OSPF, I - IS-IS, B - BGP, A - Babel,

> - selected route, * - FIB route

```
C>* 127.0.0.0/8 is directly connected, lo
C>* 172.16.0.0/16 is directly connected, eth0
C>* 172.17.0.0/16 is directly connected, eth1
R>* 172.18.0.0/16 [120/2] via 172.17.0.3, eth1, 00:02:52
R>* 172.19.0.0/16 [120/2] via 172.16.0.1, eth0, 00:02:58
R>* 192.168.0.0/24 [120/2] via 172.16.0.1, eth0, 00:02:58
C>* 192.168.1.0/24 is directly connected, eth2
R>* 192.168.2.0/24 [120/2] via 172.17.0.3, eth1, 00:02:52
R>* 192.168.3.0/24 [120/3] via 172.16.0.1, eth0, 00:02:46
```

[cursoredes@localhost ~]\$ ip route

```
172.16.0.0/16 dev eth0 proto kernel scope link src 172.16.0.2
172.17.0.0/16 dev eth1 proto kernel scope link src 172.17.0.2
172.18.0.0/16 via 172.17.0.3 dev eth1 proto zebra metric 2
172.19.0.0/16 via 172.16.0.1 dev eth0 proto zebra metric 2
192.168.0.0/24 via 172.16.0.1 dev eth0 proto zebra metric 2
192.168.1.0/24 dev eth2 proto kernel scope link src 192.168.1.2
192.168.2.0/24 via 172.17.0.3 dev eth1 proto zebra metric 2
192.168.3.0/24 via 172.16.0.1 dev eth0 proto zebra metric 3
```

ROUTER 3:

[cursoredes@localhost ~]\$ sudo vtysh -c "show ip rip"

Codes: R - RIP, C - connected, S - Static, O - OSPF, B - BGP

Sub-codes:

(n) - normal, (s) - static, (d) - default, (r) - redistribute,
(i) - interface

Network	Next Hop	Metric From	Tag Time
R(n) 172.16.0.0/16	172.17.0.2	2 172.17.0.2	0 02:46
C(i) 172.17.0.0/16	0.0.0.0	1 self	0
C(i) 172.18.0.0/16	0.0.0.0	1 self	0
R(n) 172.19.0.0/16	172.18.0.4	2 172.18.0.4	0 02:34
R(n) 192.168.0.0/24	172.17.0.2	3 172.17.0.2	0 02:46
R(n) 192.168.1.0/24	172.17.0.2	2 172.17.0.2	0 02:46
C(i) 192.168.2.0/24	0.0.0.0	1 self	0
R(n) 192.168.3.0/24	172.18.0.4	2 172.18.0.4	0 02:34

[cursoredes@localhost ~]\$ sudo vtysh -c "show ip route"

Codes: K - kernel route, C - connected, S - static, R - RIP,

O - OSPF, I - IS-IS, B - BGP, A - Babel,

> - selected route, * - FIB route

C>* 127.0.0.0/8 is directly connected, lo
R>* 172.16.0.0/16 [120/2] via 172.17.0.2, eth1, 00:04:04
C>* 172.17.0.0/16 is directly connected, eth1
C>* 172.18.0.0/16 is directly connected, eth0
R>* 172.19.0.0/16 [120/2] via 172.18.0.4, eth0, 00:03:57
R>* 192.168.0.0/24 [120/3] via 172.17.0.2, eth1, 00:04:04
R>* 192.168.1.0/24 [120/2] via 172.17.0.2, eth1, 00:04:04
C>* 192.168.2.0/24 is directly connected, eth2
R>* 192.168.3.0/24 [120/2] via 172.18.0.4, eth0, 00:03:57

[cursoredes@localhost ~]\$ ip route

172.16.0.0/16 via 172.17.0.2 dev eth1 proto zebra metric 2
172.17.0.0/16 dev eth1 proto kernel scope link src 172.17.0.3
172.18.0.0/16 dev eth0 proto kernel scope link src 172.18.0.3
172.19.0.0/16 via 172.18.0.4 dev eth0 proto zebra metric 2
192.168.0.0/24 via 172.17.0.2 dev eth1 proto zebra metric 3
192.168.1.0/24 via 172.17.0.2 dev eth1 proto zebra metric 2
192.168.2.0/24 dev eth2 proto kernel scope link src 192.168.2.3
192.168.3.0/24 via 172.18.0.4 dev eth0 proto zebra metric 2

ROUTER 4:

[cursoredes@localhost ~]\$ sudo vtysh -c "show ip rip"

Codes: R - RIP, C - connected, S - Static, O - OSPF, B - BGP

Sub-codes:

(n) - normal, (s) - static, (d) - default, (r) - redistribute,
(i) - interface

Network	Next Hop	Metric From	Tag Time
R(n) 172.16.0.0/16	172.19.0.1	2 172.19.0.1	0 02:37
R(n) 172.17.0.0/16	172.18.0.3	2 172.18.0.3	0 02:52
C(i) 172.18.0.0/16	0.0.0.0	1 self	0
C(i) 172.19.0.0/16	0.0.0.0	1 self	0
R(n) 192.168.0.0/24	172.19.0.1	2 172.19.0.1	0 02:37
R(n) 192.168.1.0/24	172.18.0.3	3 172.18.0.3	0 02:52
R(n) 192.168.2.0/24	172.18.0.3	2 172.18.0.3	0 02:52
C(i) 192.168.3.0/24	0.0.0.0	1 self	0

```
[cursoredes@localhost ~]$ sudo vtysh -c "show ip route"
```

Codes: K - kernel route, C - connected, S - static, R - RIP,
O - OSPF, I - IS-IS, B - BGP, A - Babel,
> - selected route, * - FIB route

```
C>* 127.0.0.0/8 is directly connected, lo
R>* 172.16.0.0/16 [120/2] via 172.19.0.1, eth1, 00:05:35
R>* 172.17.0.0/16 [120/2] via 172.18.0.3, eth0, 00:05:35
C>* 172.18.0.0/16 is directly connected, eth0
C>* 172.19.0.0/16 is directly connected, eth1
R>* 192.168.0.0/24 [120/2] via 172.19.0.1, eth1, 00:05:35
R>* 192.168.1.0/24 [120/3] via 172.18.0.3, eth0, 00:05:35
R>* 192.168.2.0/24 [120/2] via 172.18.0.3, eth0, 00:05:35
C>* 192.168.3.0/24 is directly connected, eth2
```

```
[cursoredes@localhost ~]$ ip route
```

```
172.16.0.0/16 via 172.19.0.1 dev eth1 proto zebra metric 2
172.17.0.0/16 via 172.18.0.3 dev eth0 proto zebra metric 2
172.18.0.0/16 dev eth0 proto kernel scope link src 172.18.0.4
172.19.0.0/16 dev eth1 proto kernel scope link src 172.19.0.4
192.168.0.0/24 via 172.19.0.1 dev eth1 proto zebra metric 2
192.168.1.0/24 via 172.18.0.3 dev eth0 proto zebra metric 3
192.168.2.0/24 via 172.18.0.3 dev eth0 proto zebra metric 2
192.168.3.0/24 dev eth2 proto kernel scope link src 192.168.3.4
```

Ejercicio 3. Con la herramienta wireshark, estudiar los mensajes RIP intercambiados, en particular:

- Encapsulado.
- Direcciones origen y destino.
- Campo de versión.
- Información para cada ruta: dirección de red, máscara de red, siguiente salto y distancia.

CAPTURAS DE ROUTER 1 EN ETH0

Mensaje Request:

No.	Time	Source	Destination	Protoc	Lengt	Info
1	0.00000000	172.16.0.1	224.0.0.9	RIPv2	66	Request

▶ Frame 1: 66 bytes on wire (528 bits), 66 bytes captured (528 bits) on interface 0
▶ Ethernet II, Src: CadmusCo_df:be:4e (08:00:27:df:be:4e), Dst: IPv4mcast_00:00:09 (01:00:5e:00:00:09)
▶ Internet Protocol Version 4, Src: 172.16.0.1 (172.16.0.1), Dst: 224.0.0.9 (224.0.0.9)
▶ User Datagram Protocol, Src Port: router (520), Dst Port: router (520)
▼ Routing Information Protocol
Command: Request (1)
Version: RIPv2 (2)
▶ Address not specified, Metric: 16

Encapsulado UDP

Dir origen: 172.10.0.1 (dir. IP de router 1 en eth0)

Dir destino: 224.0.0.9 (dir. multicast)

Version: RIPv2

Mensaje Response:

No.	Time	Source	Destination	Protoc	Length	Info
3	0.00142562	172.16.0.2	172.16.0.1	RIPv2	126	Response
Ethernet II, Src: CadmusCo_59:04:8b (08:00:27:59:04:8b), Dst: CadmusCo_df:be:4e (08:00:27:df:be:4e)						
Internet Protocol Version 4, Src: 172.16.0.2 (172.16.0.2), Dst: 172.16.0.1 (172.16.0.1)						
User Datagram Protocol, Src Port: router (520), Dst Port: router (520)						
Routing Information Protocol						
Command: Response (2)						
Version: RIPv2 (2)						
IP Address: 172.17.0.0, Metric: 1						
Address Family: IP (2)						
Route Tag: 0						
IP Address: 172.17.0.0 (172.17.0.0)						
Netmask: 255.255.0.0 (255.255.0.0)						
Next Hop: 0.0.0.0 (0.0.0.0)						
Metric: 1						
IP Address: 172.18.0.0, Metric: 2						
Address Family: IP (2)						
Route Tag: 0						
IP Address: 172.18.0.0 (172.18.0.0)						
Netmask: 255.255.0.0 (255.255.0.0)						
Next Hop: 0.0.0.0 (0.0.0.0)						
Metric: 2						
IP Address: 192.168.1.0, Metric: 1						
Address Family: IP (2)						
Route Tag: 0						
IP Address: 192.168.1.0 (192.168.1.0)						
Netmask: 255.255.255.0 (255.255.255.0)						
Next Hop: 0.0.0.0 (0.0.0.0)						
Metric: 1						
IP Address: 192.168.2.0, Metric: 2						
Address Family: IP (2)						
Route Tag: 0						
IP Address: 192.168.2.0 (192.168.2.0)						
Netmask: 255.255.255.0 (255.255.255.0)						
Next Hop: 0.0.0.0 (0.0.0.0)						
Metric: 2						

Me salen solo 4 redes porque no me están saliendo los paquetes de longitud 146 y se deja una red (no se por qué). En router 3 sí me salían estos mensajes

Ejercicio 4. Eliminar el enlace entre Router1 y Router4 (por ejemplo, desactivando el interfaz eth1 en Router4). Comprobar que Router1 deja de recibir los anuncios de Router4 y que, pasados aproximadamente 3 minutos (valor de *timeout* por defecto para las rutas), ha reajustado su tabla.

En Router 4:

```
[cursoredes@localhost ~]$ ip link set dev eth1 down
```

Esperamos 3 minutos...

```
[cursoredes@localhost ~]$ sudo vtysh -c "show ip rip"
```

Codes: R - RIP, C - connected, S - Static, O - OSPF, B - BGP

Sub-codes:

(n) - normal, (s) - static, (d) - default, (r) - redistribute,
(i) - interface

Network	Next Hop	Metric From	Tag	Time
C(i) 172.16.0.0/16	0.0.0.0	1 self	0	
R(n) 172.17.0.0/16	172.16.0.2	2 172.16.0.2	0	02:35

R(n) 172.18.0.0/16	172.19.0.4	16 172.19.0.4	0 01:54
C(i) 172.19.0.0/16	0.0.0.0	1 self 0	
C(i) 192.168.0.0/24	0.0.0.0	1 self 0	
R(n) 192.168.1.0/24	172.16.0.2	2 172.16.0.2	0 02:35
R(n) 192.168.2.0/24	172.16.0.2	3 172.16.0.2	0 02:35
R(n) 192.168.3.0/24	172.16.0.2	4 172.16.0.2	0 02:57

Ejercicio 5 (Opcional). Los servicios de Quagga pueden configurarse de forma interactiva mediante un terminal (telnet), de forma similar a los encaminadores comerciales. Configurar ripd vía VTY:

- Añadir "password asor" al fichero ripd.conf, desactivar el protocolo (no router rip) y comentar el resto de entradas. Una vez cambiado el fichero, reiniciar el servicio.
- Conectar al VTY de ripd y configurarlo. En cada comando se puede usar ? para mostrar la ayuda asociada.

```
$ telnet localhost ripd
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.

Hello, this is Quagga (version 0.99.20.1)
Copyright © 1996-2005 Kunihiro Ishiguro, et al.
User Access Verification

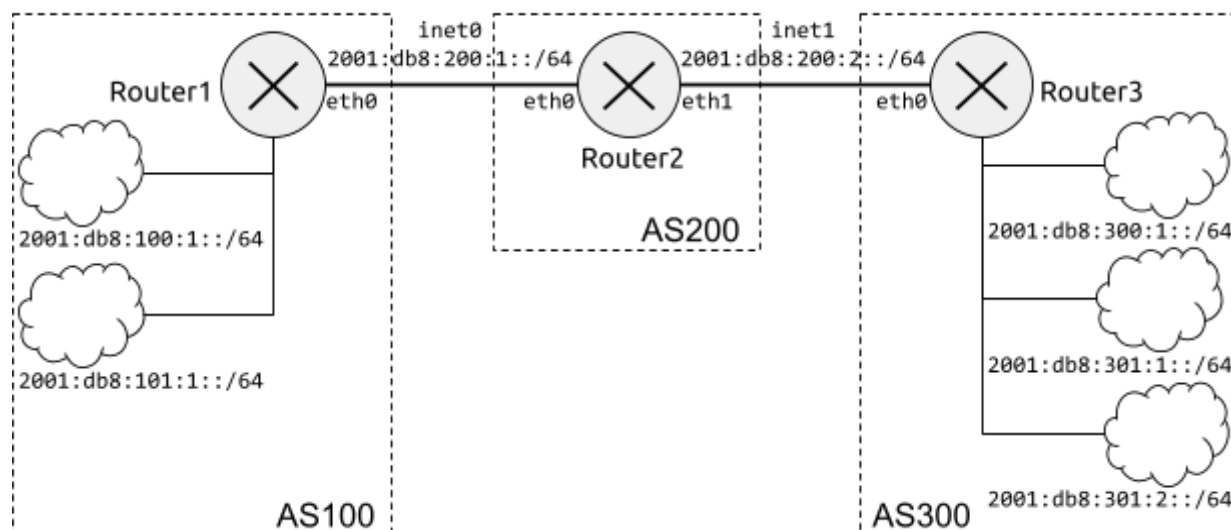
Password: asor
localhost.localdomain> enable
localhost.localdomain# configure terminal
localhost.localdomain(config)# router rip
localhost.localdomain(config-router)# version 2
localhost.localdomain(config-router)# network eth0
localhost.localdomain(config-router)# write
Configuration saved to /etc/quagga/ripd.conf
localhost.localdomain(config-router)# exit
localhost.localdomain(config)# exit
localhost.localdomain# show running-config
Current configuration:
!
password asor
!
router rip
version 2
network eth0
!
line vty
!
end
localhost.localdomain# write
Configuration saved to /etc/quagga/ripd.conf
localhost.localdomain# exit
```

Nota: Para poder escribir la configuración en ripd.conf, el usuario quagga debe tener los permisos adecuados sobre el fichero. Para cambiar el propietario del fichero, ejecutar el comando `chown quagga:quagga /etc/quagga/ripd.conf`.

Parte II. Protocolo exterior: BGP

Preparación del entorno

Configuraremos la topología de red con 3 AS, siendo uno de ellos el proveedor de los otros dos:



Nota: El prefijo 2001:db8::/32 está reservado para documentación y ejemplos (RFC 3849).

Crearemos esta topología (sin las redes internas de los AS) con la herramienta vtopo1 y el siguiente fichero:

```
netprefix inet
machine 1 0 0
machine 2 0 0 1 1
machine 3 0 1
```

Para facilitar la configuración de las máquinas, la siguiente tabla muestra las direcciones de cada uno de los interfaces de los encaminadores:

Máquina virtual	Interfaz	Dirección de red	Dirección IP
Router1	eth0	2001:db8:200:1::/64	2001:db8:200:1::1
Router2	eth0	2001:db8:200:1::/64	2001:db8:200:1::2
	eth1	2001:db8:200:2::/64	2001:db8:200:2::2
Router3	eth0	2001:db8:200:2::/64	2001:db8:200:2::3

Configurar los encaminadores según se muestra en la figura anterior. Debe comprobarse la conectividad entre máquinas adyacentes.

Configuración del protocolo BGP

Ejercicio 6. Consultar la documentación de las clases de teoría para determinar el tipo de AS (*stub*, *multihomed* o *transit*) y los prefijos de red que debe anunciar. Recordar que el prefijo global de encaminamiento es de 48 bits y que los prefijos anunciados deben agregarse al máximo.

Número de AS	Tipo	Prefijos agregados
100	STUB (subred)	2001:db8:100::/47 (prefijos de la subred)
200	TRANSIT	
300	STUB	2001:db8:300::/47

Ejercicio 7. Configurar BGP en los encaminadores para que intercambien información:

- Crear un fichero `bgpd.conf` en `/etc/quagga` usando como referencia el que se muestra a continuación.
- Iniciar el servicio BGP (y Zebra) con `service bgpd start`.

Por ejemplo, el contenido del fichero `/etc/quagga/bgpd.conf` de Router1 en el AS 100 sería:

```
Router 1:
# Activar el encaminamiento BGP en el AS 100
router bgp 100
# Establecer el identificador de encaminador BGP
bgp router-id 0.0.0.1
# Añadir el encaminador BGP vecino en el AS 200
neighbor 2001:db8:200:1::2 remote-as 200
# Empezar a trabajar con direcciones IPv6
address-family ipv6
# Anunciar un prefijo de red agregado
network 2001:db8:100::/47
# Activar IPv6 en el encaminador BGP vecino
neighbor 2001:db8:200:1::2 activate
# Dejar de trabajar con direcciones IPv6
exit-address-family
```

```
Router 2:
router bgp 200
bgp router-id 0.0.0.2
neighbor 2001:db8:200:1::1 remote-as 100
neighbor 2001:db8:200:2::3 remote-as 300
address-family ipv6
neighbor 2001:db8:200:1::1 activate 100
neighbor 2001:db8:200:2::3 activate 300
exit-address-family
```

Router 3:

```

router bgp 300
bgp router-id 0.0.0.3
neighbor 2001:db8:200:2::2 remote-as 200
address-family ipv6
network 2001:db8:300::/47
neighbor 2001:db8:200:2::2 activate
exit-address-family

```

Ejercicio 8. Consultar la tabla de encaminamiento de BGP y de Zebra en cada encaminador con el comando vtysh (sudo vtysh -c "show ipv6 bgp" y sudo vtysh -c "show ipv6 route"). Comprobar también la tabla de reenvío de IPv6 con el comando ip (ip -6 route).

ROUTER 1:

```
[cursoredes@localhost ~]$ sudo vtysh -c "show ipv6 bgp"
```

```
BGP table version is 0, local router ID is 0.0.0.1
```

```
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
               r RIB-failure, S Stale, R Removed
```

```
Origin codes: i - IGP, e - EGP, ? - incomplete
```

```

Network      Next Hop      Metric LocPrf Weight Path
*> 2001:db8:100::/47
      ::          0      32768 i
*> 2001:db8:300::/47
      2001:db8:200:1::2
                        0 200 300 i

```

```
Total number of prefixes 2
```

```
[cursoredes@localhost ~]$ sudo vtysh -c "show ipv6 route"
```

```
Codes: K - kernel route, C - connected, S - static, R - RIPng,
```

```
O - OSPFv6, I - IS-IS, B - BGP, A - Babel,
```

```
> - selected route, * - FIB route
```

```
C>* ::1/128 is directly connected, lo
```

```
C>* 2001:db8:200:1::/64 is directly connected, eth0
```

```
B>* 2001:db8:300::/47 [20/0] via fe80::a00:27ff:fe42:f605, eth0, 00:00:44
```

```
C>* fe80::/64 is directly connected, eth0
```

```
[cursoredes@localhost ~]$ ip -6 route
```

```
unreachable ::/96 dev lo metric 1024 error -113 pref medium
```

```
unreachable ::ffff:0.0.0.0/96 dev lo metric 1024 error -113 pref medium
```

```
2001:db8:200:1::/64 dev eth0 proto kernel metric 256 pref medium
```

```
2001:db8:300::/47 via fe80::a00:27ff:fe42:f605 dev eth0 proto zebra metric 1024 pref medium
```

```
unreachable 2002:a00::/24 dev lo metric 1024 error -113 pref medium
```

```
unreachable 2002:7f00::/24 dev lo metric 1024 error -113 pref medium
```

```
unreachable 2002:a9fe::/32 dev lo metric 1024 error -113 pref medium
```

```
unreachable 2002:ac10::/28 dev lo metric 1024 error -113 pref medium
```

```
unreachable 2002:c0a8::/32 dev lo metric 1024 error -113 pref medium
```

```
unreachable 2002:e000::/19 dev lo metric 1024 error -113 pref medium
```

```
unreachable 3ffe:ffff::/32 dev lo metric 1024 error -113 pref medium
```

```
fe80::/64 dev eth0 proto kernel metric 256 pref medium
```

ROUTER 2:

```
[cursoredes@localhost ~]$ sudo vtysh -c "show ipv6 bgp"
```

BGP table version is 0, local router ID is 0.0.0.2

Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
r RIB-failure, S Stale, R Removed

Origin codes: i - IGP, e - EGP, ? - incomplete

Network	Next Hop	Metric	LocPrf	Weight	Path
*> 2001:db8:100::/47	2001:db8:200:1::1	0	0	100	i
*> 2001:db8:300::/47	2001:db8:200:2::3	0	0	300	i

Total number of prefixes 2

```
[cursoredes@localhost ~]$ sudo vtysh -c "show ipv6 route"
```

Codes: K - kernel route, C - connected, S - static, R - RIPng,

O - OSPFv6, I - IS-IS, B - BGP, A - Babel,

> - selected route, * - FIB route

```
C>* ::1/128 is directly connected, lo
B>* 2001:db8:100::/47 [20/0] via fe80::a00:27ff:fe6:dc92, eth0, 00:06:01
C>* 2001:db8:200:1::/64 is directly connected, eth0
C>* 2001:db8:200:2::/64 is directly connected, eth1
B>* 2001:db8:300::/47 [20/0] via fe80::a00:27ff:feb3:7e54, eth1, 00:01:55
C * fe80::/64 is directly connected, eth1
C>* fe80::/64 is directly connected, eth0
```

```
[cursoredes@localhost ~]$ ip -6 route
```

```
unreachable ::/96 dev lo metric 1024 error -113 pref medium
unreachable ::ffff:0.0.0.0/96 dev lo metric 1024 error -113 pref medium
2001:db8:100::/47 via fe80::a00:27ff:fe6:dc92 dev eth0 proto zebra metric 1024 pref medium
2001:db8:200:1::/64 dev eth0 proto kernel metric 256 pref medium
2001:db8:200:2::/64 dev eth1 proto kernel metric 256 pref medium
2001:db8:300::/47 via fe80::a00:27ff:feb3:7e54 dev eth1 proto zebra metric 1024 pref medium
unreachable 2002:a00::/24 dev lo metric 1024 error -113 pref medium
unreachable 2002:7f00::/24 dev lo metric 1024 error -113 pref medium
unreachable 2002:a9fe::/32 dev lo metric 1024 error -113 pref medium
unreachable 2002:ac10::/28 dev lo metric 1024 error -113 pref medium
unreachable 2002:c0a8::/32 dev lo metric 1024 error -113 pref medium
unreachable 2002:e000::/19 dev lo metric 1024 error -113 pref medium
unreachable 3ffe:ffff::/32 dev lo metric 1024 error -113 pref medium
fe80::/64 dev eth0 proto kernel metric 256 pref medium
fe80::/64 dev eth1 proto kernel metric 256 pref medium
```

ROUTER 3:

```
[cursoredes@localhost ~]$ sudo vtysh -c "show ipv6 bgp"
```

BGP table version is 0, local router ID is 0.0.0.3

Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
r RIB-failure, S Stale, R Removed

Origin codes: i - IGP, e - EGP, ? - incomplete

Network	Next Hop	Metric	LocPrf	Weight	Path
*> 2001:db8:100::/47	2001:db8:200:2::2	0	200	100	i
*> 2001:db8:300::/47	::	0	32768	i	

Total number of prefixes 2

```
[cursoredes@localhost ~]$ sudo vtysh -c "show ipv6 route"
```

Codes: K - kernel route, C - connected, S - static, R - RIPng,

O - OSPFv6, I - IS-IS, B - BGP, A - Babel,

> - selected route, * - FIB route

C>* ::1/128 is directly connected, lo

B>* 2001:db8:100::/47 [20/0] via fe80::a00:27ff:feb0:b620, eth0, 00:03:28

C>* 2001:db8:200:2::/64 is directly connected, eth0

C>* fe80::/64 is directly connected, eth0

```
[cursoredes@localhost ~]$ ip -6 route
```

unreachable ::/96 dev lo metric 1024 error -113 pref medium

unreachable ::ffff:0.0.0.0/96 dev lo metric 1024 error -113 pref medium

2001:db8:100::/47 via fe80::a00:27ff:feb0:b620 dev eth0 proto zebra metric 1024 pref medium

2001:db8:200:2::/64 dev eth0 proto kernel metric 256 pref medium

unreachable 2002:a00::/24 dev lo metric 1024 error -113 pref medium

unreachable 2002:7f00::/24 dev lo metric 1024 error -113 pref medium

unreachable 2002:a9fe::/32 dev lo metric 1024 error -113 pref medium

unreachable 2002:ac10::/28 dev lo metric 1024 error -113 pref medium

unreachable 2002:c0a8::/32 dev lo metric 1024 error -113 pref medium

unreachable 2002:e000::/19 dev lo metric 1024 error -113 pref medium

unreachable 3ffe:ffff::/32 dev lo metric 1024 error -113 pref medium

fe80::/64 dev eth0 proto kernel metric 256 pref medium

Ejercicio 9. Con ayuda de la herramienta wireshark, estudiar los mensajes BGP intercambiados (OPEN, KEEPALIVE y UPDATE).

En ROUTER 1 y hago sudo service bgpd restart

Mensaje OPEN:

No.	Time	Source	Destination	Protoc	Length	Info
10	6.03564703	2001:db8:200:1::1	2001:db8:200:1::2	TCP	86	41632 > bgp [ACK] Seq=1 Ack=1 Win=28800 Len=0 TSval=3121662 TSecr=3096202
11	6.03602783	2001:db8:200:1::1	2001:db8:200:1::2	BGP	147	OPEN Message
12	6.03677901	2001:db8:200:1::2	2001:db8:200:1::1	TCP	86	bgp > 41632 [ACK] Seq=1 Ack=62 Win=28672 Len=0 TSval=3096203 TSecr=3121662
13	6.03680233	2001:db8:200:1::2	2001:db8:200:1::1	TCP	86	bgp > 41632 [RST, ACK] Seq=1 Ack=62 Win=28672 Len=0 TSval=0 TSecr=3121662
14	8.00765511	2001:db8:200:1::2	2001:db8:200:1::1	TCP	94	57890 > bgp [SYN] Seq=0 Win=28800 Len=0 MSS=1440 SACK_PERM=1 TSval=3098173 TSecr=0 WS=
15	8.00770971	2001:db8:200:1::1	2001:db8:200:1::2	TCP	94	bgp > 57890 [SYN, ACK] Seq=0 Ack=1 Win=28560 Len=0 MSS=1440 SACK_PERM=1 TSval=3123634
16	8.00854733	2001:db8:200:1::2	2001:db8:200:1::1	TCP	86	57890 > bgp [ACK] Seq=1 Ack=1 Win=28800 Len=0 TSval=3098174 TSecr=3123634
▼ Transmission Control Protocol, Src Port: 41632 (41632), Dst Port: bgp (179), Seq: 1, Ack: 1, Len: 61						
Source port: 41632 (41632)						
Destination port: bgp (179)						
[Stream index: 1]						
Sequence number: 1 (relative sequence number)						
[Next sequence number: 62 (relative sequence number)]						
Acknowledgment number: 1 (relative ack number)						
Header length: 32 bytes						
► Flags: 0x018 (PSH, ACK)						
Window size value: 225						
[Calculated window size: 28800]						
[Window size scaling factor: 128]						
► Checksum: 0x5fda [validation disabled]						
► Options: (12 bytes), No-Operation (NOP), No-Operation (NOP), Timestamps						
► [SEQ/ACK analysis]						
▼ Border Gateway Protocol - OPEN Message						
Marker: ffffffffffffffffffffffffffffffff						
Length: 61						
Type: OPEN Message (1)						
Version: 4						
My AS: 100						
Hold Time: 180						
BGP Identifier: 0.0.0.1 (0.0.0.1)						
Optional Parameters Length: 32						
► Optional Parameters						

Mensaje OPEN, mensaje KEEPALIVE:

No.	Time	Source	Destination	Protoc	Length	Info
17	8.00892605	2001:db8:200:1::2	2001:db8:200:1::1	BGP	147	OPEN Message
18	8.00895485	2001:db8:200:1::1	2001:db8:200:1::2	TCP	86	bgp > 57890 [ACK] Seq=1 Ack=62 Win=28672 Len=0 TSval=3123635 TSecr=3098174
19	8.01028164	2001:db8:200:1::1	2001:db8:200:1::2	BGP	166	OPEN Message, KEEPALIVE Message
20	8.01109680	2001:db8:200:1::2	2001:db8:200:1::1	TCP	86	57890 > bgp [ACK] Seq=62 Ack=81 Win=28800 Len=0 TSval=3098177 TSecr=3123636
21	8.01154786	2001:db8:200:1::2	2001:db8:200:1::1	BGP	124	KEEPALIVE Message, KEEPALIVE Message
22	8.01184532	2001:db8:200:1::1	2001:db8:200:1::2	BGP	105	KEEPALIVE Message
23	8.05256907	2001:db8:200:1::2	2001:db8:200:1::1	TCP	86	57890 > bgp [ACK] Seq=100 Ack=100 Win=28800 Len=0 TSval=3098218 TSecr=3123638
▼ Transmission Control Protocol, Src Port: bgp (179), Dst Port: 57890 (57890), Seq: 1, Ack: 62, Len: 80						
Source port: bgp (179)						
Destination port: 57890 (57890)						
[Stream index: 2]						
Sequence number: 1 (relative sequence number)						
[Next sequence number: 81 (relative sequence number)]						
Acknowledgment number: 62 (relative ack number)						
Header length: 32 bytes						
► Flags: 0x018 (PSH, ACK)						
Window size value: 224						
[Calculated window size: 28672]						
[Window size scaling factor: 128]						
► Checksum: 0x5fed [validation disabled]						
► Options: (12 bytes), No-Operation (NOP), No-Operation (NOP), Timestamps						
► [SEQ/ACK analysis]						
▼ Border Gateway Protocol - OPEN Message						
Marker: ffffffffffffffffffffffffffffffff						
Length: 61						
Type: OPEN Message (1)						
Version: 4						
My AS: 100						
Hold Time: 180						
BGP Identifier: 0.0.0.1 (0.0.0.1)						
Optional Parameters Length: 32						
► Optional Parameters						
▼ Border Gateway Protocol - KEEPALIVE Message						
Marker: ffffffffffffffffffffffffffffffff						
Length: 19						
Type: KEEPALIVE Message (4)						

Mensaje KEEPALIVE, mensaje KEEPALIVE:

No.	Time	Source	Destination	Protoc	Length	Info
19	8.01028164	2001:db8:200:1::1	2001:db8:200:1::2	BGP	166	OPEN Message, KEEPALIVE Message
20	8.01109680	2001:db8:200:1::2	2001:db8:200:1::1	TCP	86	57890 > bgp [ACK] Seq=62 Ack=81 Win=28800 Len=0 TSval=3098177 TSecr=3123636
21	8.01154786	2001:db8:200:1::2	2001:db8:200:1::1	BGP	124	KEEPALIVE Message, KEEPALIVE Message
22	8.01184532	2001:db8:200:1::1	2001:db8:200:1::2	BGP	105	KEEPALIVE Message
23	8.05256907	2001:db8:200:1::2	2001:db8:200:1::1	TCP	86	57890 > bgp [ACK] Seq=100 Ack=100 Win=28800 Len=0 TSval=3098218 TSecr=3123638
24	9.01338182	2001:db8:200:1::1	2001:db8:200:1::2	BGP	177	UPDATE Message
25	9.01371883	2001:db8:200:1::2	2001:db8:200:1::1	BGP	174	UPDATE Message
Transmission Control Protocol, Src Port: 57890 (57890), Dst Port: bgp (179), Seq: 62, Ack: 81, Len: 38						
Source port: 57890 (57890)						
Destination port: bgp (179)						
[Stream index: 2]						
Sequence number: 62 (relative sequence number)						
[Next sequence number: 100 (relative sequence number)]						
Acknowledgment number: 81 (relative ack number)						
Header length: 32 bytes						
Flags: 0x018 (PSH, ACK)						
Window size value: 225						
[Calculated window size: 28800]						
[Window size scaling factor: 128]						
Checksum: 0x0c99 [validation disabled]						
Options: (12 bytes), No-Operation (NOP), No-Operation (NOP), Timestamps						
[SEQ/ACK analysis]						
Border Gateway Protocol - KEEPALIVE Message						
Marker: ffffffffffffffffffffffffffffffff						
Length: 19						
Type: KEEPALIVE Message (4)						
Border Gateway Protocol - KEEPALIVE Message						
Marker: ffffffffffffffffffffffffffffffff						
Length: 19						
Type: KEEPALIVE Message (4)						

Mensaje KEEPALIVE:

No.	Time	Source	Destination	Protoc	Length	Info
19	8.01028164	2001:db8:200:1::1	2001:db8:200:1::2	BGP	166	OPEN Message, KEEPALIVE Message
20	8.01109680	2001:db8:200:1::2	2001:db8:200:1::1	TCP	86	57890 > bgp [ACK] Seq=62 Ack=81 Win=28800 Len=0 TSval=3098177 TSecr=3123636
21	8.01154786	2001:db8:200:1::2	2001:db8:200:1::1	BGP	124	KEEPALIVE Message, KEEPALIVE Message
22	8.01184532	2001:db8:200:1::1	2001:db8:200:1::2	BGP	105	KEEPALIVE Message
23	8.05256907	2001:db8:200:1::2	2001:db8:200:1::1	TCP	86	57890 > bgp [ACK] Seq=100 Ack=100 Win=28800 Len=0 TSval=3098218 TSecr=3123638
24	9.01338182	2001:db8:200:1::1	2001:db8:200:1::2	BGP	177	UPDATE Message
Transmission Control Protocol, Src Port: bgp (179), Dst Port: 57890 (57890), Seq: 81, Ack: 100, Len: 19						
Source port: bgp (179)						
Destination port: 57890 (57890)						
[Stream index: 2]						
Sequence number: 81 (relative sequence number)						
[Next sequence number: 100 (relative sequence number)]						
Acknowledgment number: 100 (relative ack number)						
Header length: 32 bytes						
Flags: 0x018 (PSH, ACK)						
Window size value: 224						
[Calculated window size: 28672]						
[Window size scaling factor: 128]						
Checksum: 0x5fb0 [validation disabled]						
Options: (12 bytes), No-Operation (NOP), No-Operation (NOP), Timestamps						
[SEQ/ACK analysis]						
Border Gateway Protocol - KEEPALIVE Message						
Marker: ffffffffffffffffffffffffffffffff						
Length: 19						
Type: KEEPALIVE Message (4)						

Mensaje UPDATE:

No.	Time	Source	Destination	Protoc	Length	Info
22	8.01184532	2001:db8:200:1::1	2001:db8:200:1::2	BGP	105	KEEPALIVE Message
23	8.05256907	2001:db8:200:1::2	2001:db8:200:1::1	TCP	86	57890 > bgp [ACK] Seq=100 Ack=100 Win=28800 Len=0 TSval=3098218 TSecr=3123638
24	9.01338182	2001:db8:200:1::1	2001:db8:200:1::2	BGP	177	UPDATE Message
25	9.01371883	2001:db8:200:1::2	2001:db8:200:1::1	BGP	174	UPDATE Message
26	9.01393400	2001:db8:200:1::2	2001:db8:200:1::1	TCP	86	57890 > bgp [ACK] Seq=188 Ack=191 Win=28800 Len=0 TSval=3099179 TSecr=3124639
27	9.05335516	2001:db8:200:1::1	2001:db8:200:1::2	TCP	86	bgp > 57890 [ACK] Seq=191 Ack=188 Win=28672 Len=0 TSval=3124680 TSecr=3099179
Internet Protocol Version 6, Src: 2001:db8:200:1::1 (2001:db8:200:1::1), Dst: 2001:db8:200:1::2 (2001:db8:200:1::2)						
Transmission Control Protocol, Src Port: bgp (179), Dst Port: 57890 (57890), Seq: 100, Ack: 100, Len: 91						
<ul style="list-style-type: none"> Source port: bgp (179) Destination port: 57890 (57890) [Stream index: 2] Sequence number: 100 (relative sequence number) [Next sequence number: 191 (relative sequence number)] Acknowledgment number: 100 (relative ack number) Header length: 32 bytes Flags: 0x018 (PSH, ACK) Window size value: 224 [Calculated window size: 28672] [Window size scaling factor: 128] Checksum: 0x5ff8 [validation disabled] Options: (12 bytes), No-Operation (NOP), No-Operation (NOP), Timestamps [SEQ/ACK analysis] 						
Border Gateway Protocol - UPDATE Message						
<ul style="list-style-type: none"> Marker: ffffffffffffffffffffffffffffffff Length: 91 Type: UPDATE Message (2) Unfeasible routes length: 0 bytes Total path attribute length: 68 bytes Path attributes <ul style="list-style-type: none"> ORIGIN: IGP (4 bytes) AS_PATH: 100 (10 bytes) MULTI_EXIT_DISC: 0 (7 bytes) MP_REACH_NLRI (47 bytes) 						