Práctico FCEFyN Redes de computadoras

Trabajo Práctico 5

Docente: Matías R. Cuenca del Rey

Mail: [mcuenca@unc.edu.ar](mailto:mcuenca@unc.edu.ar)

Ayudantes alumnos: Elisabeth Leonhardt - Andrés Serjoy

Redes de computadoras

Facultad de Ciencias Exactas, Físicas y Naturales

Universidad Nacional de Córdoba

Práctico 5: Ruteo Internet.

Presentación de consignas.

### Ejercicio 1: Ruteo internet.

#### Recomendaciones

* Lea con cuidado las consignas.
* Tenga certeza de los comandos que ejecuta.
* Para contenerización utilizar Docker CE.

#### Esquema

* Se realizará bajo IPv6. No se usará IPv4.
* Se borrarán las rutas por defecto que son asignadas por Docker CE.
* Se realizará un shell script que pueda replicar el trabajo práctico.
* Se usará una máquina física por grupo.
* Se usará la máquina virtual Desktop.
* Se usarán containers dentro de la máquina virtual para simular routers.

#### Diagrama



#### Tabla de asignación de direcciones IPv6

Crear la tabla de asignación de direcciones IP

#### 

|  |  |  |
| --- | --- | --- |
| **Computadora** | **Interfaz de red** | **Dirección IP** |

#### 

|  |  |  |  |
| --- | --- | --- | --- |
| **Subnets** | **Adress** | **Devices** | **Address** |
| **redb12** | 2001:b:12::/64 | **b1** | 2001:a:2::10 |
| **redb13** | 2001:b:13::/64 | 2001:a:3::10 |
| **redb32** | 2001:b:32::/64 | 2001:a:4::10 |
| **red2** | 2001:a:2::/64 | 2001:b:12::10 |
| **red3** | 2001:a:3::/64 | 2001:b:13::10 |
| **red4** | 2001:a:4::/64 | **b2** | 2001:a:12::10 |
| **red5** | 2001:a:5::/64 | 2001:a:13::10 |
| **red6** | 2001:a:6::/64 | 2001:a:14::10 |
| **red7** | 2001:a:7::/64 | 2001:b:12::20 |
| **red9** | 2001:a:9::/64 | 2001:b:32::10 |
| **red12** | 2001:a:12::/64 | **b3** | 2001:a:9::10 |
| **red13** | 2001:a:13::/64 | 2001:b:13::20 |
| **red14** | 2001:a:14::/64 | 2001:b:32::20 |
| **red15** | 2001:a:15::/64 | **r3** | 2001:a:2::20 |
|  |  | **r4** | 2001:a:12::20 |
|  |  | **r5** | 2001:a:3::20 |
|  |  | 2001:a:5::10 |
|  |  | 2001:a:7::10 |
|  |  | **r6** | 2001:a:13::20 |
|  |  | 2001:a:15::10 |
|  |  | 2001:a:17::10 |
|  |  | **r7** | 2001:a:4::20 |
|  |  | 2001:a:6::10 |
|  |  | 2001:a:7::20 |
|  |  | **r8** | 2001:a:14::20 |
|  |  | 2001:a:16::10 |
|  |  | 2001:a:17::20 |
|  |  | **r9** | 2001:a:9::20 |
|  |  | **h11** | 2001:a:5::20 |
|  |  | **h12** | 2001:a:15::20 |
|  |  | **h13** | 2001:a:6::20 |
|  |  | **h14** | 2001:a:16::20 |

#### 

#### Links de ayuda

Instalación de Docker CE: <https://docs.docker.com/install/linux/docker-ce/ubuntu/>

Instalación de Docker compose: <https://www.digitalocean.com/community/tutorials/how-to-install-docker-compose-on-ubuntu-16-04>

Quagga y otros software OpenSource para ruteo: <https://keepingitclassless.net/2015/05/open-source-routing-comparison/>

#### Consignas

##### Preparación de entorno

1.- Sobre Desktop instalar Docker CE, docker-compose y git

2.- Clonar el siguiente repositorio: <https://github.com/maticue/docker_quagga>

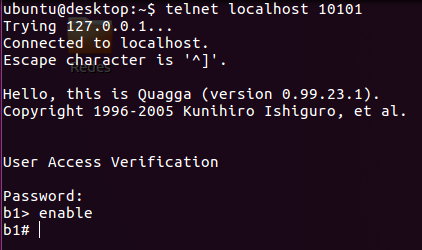
3.- Siguiendo las instrucciones del repositorio, configurar Docker CE con soporte para IPv6

4.- Siguiendo las instrucciones del repositorio, probar de iniciar el entorno de pruebas para bgp.

5.- Leer el archivo docker-compose.yml e identificar cada sección.

5.1.- ¿En qué puerto escucha el servicio de administración de BGP? Conectarse usando telnet

El servicio de administración de BGP escucha en el puerto 2601 que está mapeado con un puerto correspondiente a cada dispositivo. Todos los containers levantan servicios en el mismo puerto. En este caso se muestra el acceso al servicio desde el Router b1, puerto 10101:



6.- Analizar los archivos de configuración de los servicios.

6.1.- Identificar el password de los servicios y utilizarlo para autenticarse en la conexión telnet creada en el punto anterior.

7.- Eliminar el default gateway asignado por Docker en cada container

Se creó el siguiente script para eliminar los default gateway de los routers y hosts:

Direccionamiento.sh:

|  |
| --- |
| #!/bin/bash echo Borrando defaults gateway... docker exec -it bgp\_b1\_1 ip -6 r del default docker exec -it bgp\_b2\_1 ip -6 r del default docker exec -it bgp\_b3\_1 ip -6 r del default docker exec -it bgp\_r3\_1 ip -6 r del default docker exec -it bgp\_r4\_1 ip -6 r del default docker exec -it bgp\_r5\_1 ip -6 r del default docker exec -it bgp\_r6\_1 ip -6 r del default docker exec -it bgp\_r7\_1 ip -6 r del default docker exec -it bgp\_r8\_1 ip -6 r del default docker exec -it bgp\_r9\_1 ip -6 r del default docker exec -it bgp\_h11\_1 ip -6 r del default docker exec -it bgp\_h12\_1 ip -6 r del default docker exec -it bgp\_h13\_1 ip -6 r del default docker exec -it bgp\_h14\_1 ip -6 r del default echo Asignando defaults gateway... docker exec -it bgp\_r3\_1 ip -6 r add default via 2001:a:2::10 docker exec -it bgp\_r4\_1 ip -6 r add default via 2001:a:12::10 docker exec -it bgp\_r5\_1 ip -6 r add default via 2001:a:3::10 docker exec -it bgp\_r6\_1 ip -6 r add default via 2001:a:13::10 docker exec -it bgp\_r7\_1 ip -6 r add default via 2001:a:4::10 docker exec -it bgp\_r8\_1 ip -6 r add default via 2001:a:14::10 docker exec -it bgp\_r9\_1 ip -6 r add default via 2001:a:9::10 docker exec -it bgp\_h11\_1 ip -6 r add default via 2001:a:5::10 docker exec -it bgp\_h12\_1 ip -6 r add default via 2001:a:15::10 docker exec -it bgp\_h13\_1 ip -6 r add default via 2001:a:6::10 docker exec -it bgp\_h14\_1 ip -6 r add default via 2001:a:16::10 echo HECHO! |

##### Creación de entorno

7.- Modificar el archivo docker-compose para replicar la topología definida en el diagrama.

[docker-compose.yml](https://drive.google.com/open?id=1Yz-Ql0e0qZ07xUf5ycxAJpD7vY8BrHIX)

8.- Crear y modificar los archivos de configuración para cada router.

8.1.- Configurar que cada sistema autónomo funcione con OSPFv3.

Router b1 - Se configuran 3 de sus 5 interfaces como OSPF.

|  |
| --- |
| ! -\*- ospfv3 -\*- ! ! OSPF6d configuration file ! ! hostname b1 password admin ! interface eth0  ipv6 ospf6 cost 1  ipv6 ospf6 hello-interval 10  ipv6 ospf6 dead-interval 40  ipv6 ospf6 retransmit-interval 5  ipv6 ospf6 priority 1  ipv6 ospf6 transmit-delay 1 interface eth1  ipv6 ospf6 cost 1  ipv6 ospf6 hello-interval 10  ipv6 ospf6 dead-interval 40  ipv6 ospf6 retransmit-interval 5  ipv6 ospf6 priority 1  ipv6 ospf6 transmit-delay 1 interface eth4  ipv6 ospf6 cost 1  ipv6 ospf6 hello-interval 10  ipv6 ospf6 dead-interval 40  ipv6 ospf6 retransmit-interval 5  ipv6 ospf6 priority 1  ipv6 ospf6 transmit-delay 1  ! router ospf6  area 0.0.0.1 range 2001:a:2::/64  area 0.0.0.1 range 2001:a:3::/64  area 0.0.0.1 range 2001:a:4::/64  interface eth0 area 0.0.0.1  interface eth1 area 0.0.0.1  interface eth4 area 0.0.0.1 ! log file /var/log/quagga/ospf6d.log |

8.2.- Configurar para que cada router de borde, en su proceso de OSPFv3 se anuncie como default gateway de su sistema autónomo.

Este paso se hace en el punto 7, (dentro del script Direccionamiento.sh) cuando a cada router se le asigna como default gateway su router de borde.

8.3.- Configurar para que cada router de borde de cada sistema autónomo funcione con BGP.

Router b1 - Se configuran 2 de sus 5 interfaces como BGP.

|  |
| --- |
| ! -\*- bgp -\*- ! ! BGPd configuration file ! ! hostname b1 password admin ! router bgp 101  bgp router-id 192.168.1.10 !  no auto-summary  no synchronization !  neighbor 2001:b:12::20 remote-as 202  neighbor 2001:b:12::20 description B ! neighbor 2001:b:13::20 remote-as 303 ! neighbor 2001:b:13::20 description B !  address-family ipv6  network 2001:a:2::/64  network 2001:a:3::/64  network 2001:a:4::/64  network 2001:a:5::/64  network 2001:a:6::/64  network 2001:a:7::/64  neighbor 2001:b:12::20 activate  neighbor 2001:b:13::20 activate  neighbor 2001:b:12::20 weight 500 ! ! ! neighbor upstream peer-group ! neighbor upstream remote-as 64515 ! neighbor upstream capability dynamic ! neighbor upstream prefix-list pl-allowed-adv out ! neighbor 10.1.1.1 peer-group upstream ! ! ip prefix-list pl-allowed-adv seq 5 permit 82.195.133.0/25 ! ip prefix-list pl-allowed-adv seq 10 deny any ! log file /var/log/quagga/bgpd.log |

9.- Probar interconexión entre los distintos puntos y verificar que que las tablas de ruteo de los routers muestran las rutas OSPF y BGP

ping de h11(AS101) a r9(AS303) y a h14(AS202)

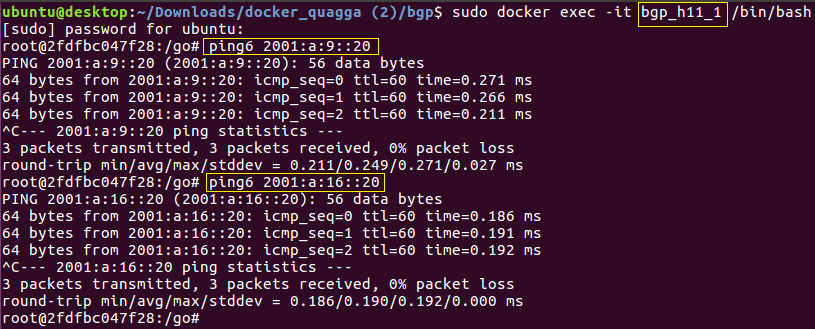


Tabla de ruteo de b1

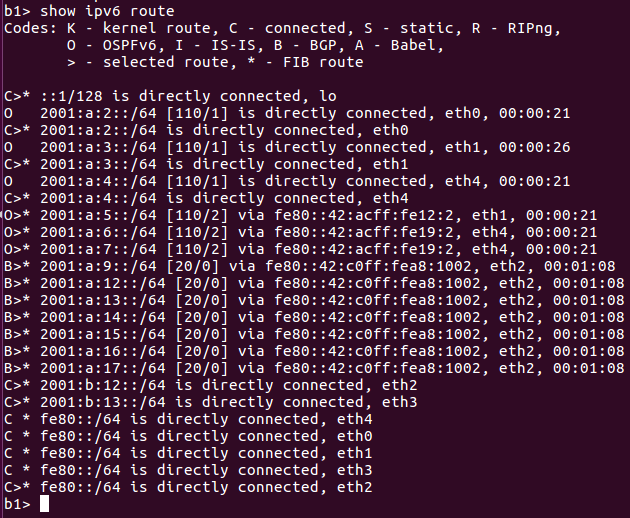


Tabla de ruteo de b2

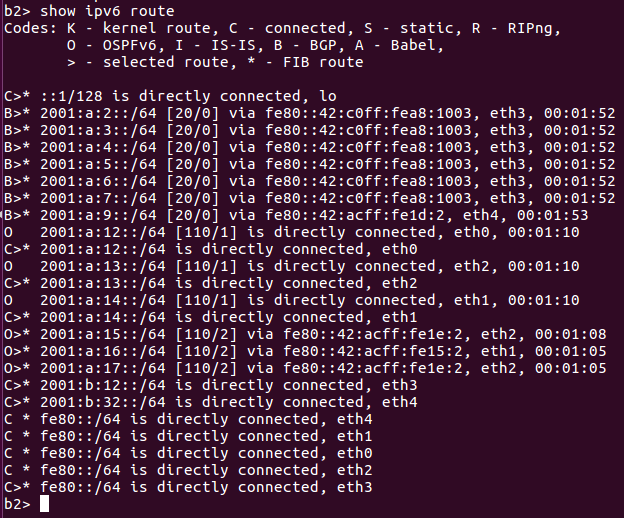


Tabla de ruteo de b3

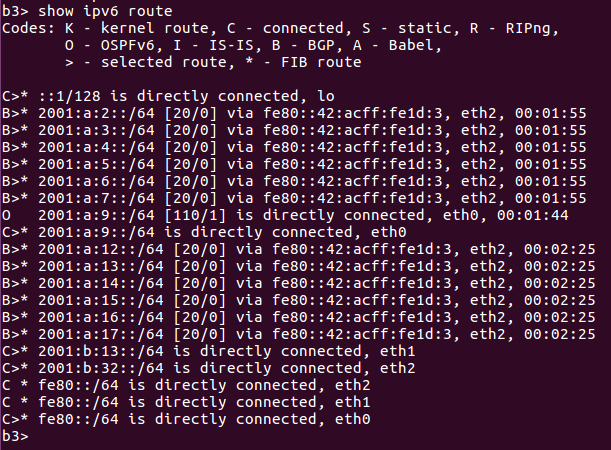
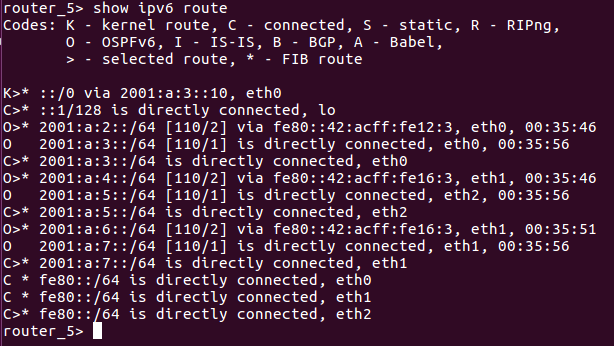


Tabla de ruteo de r5 (AS101)



10.- Identificar y Analizar los mensajes de BGP. Obtener screenshots y explicarlos

Todos los mensajes BGP tienen el mismo encabezado de tamaño fijo, que contiene un campo de marcador que se usa tanto para la sincronización como para la autenticación, un campo de longitud que indica la longitud del paquete y un campo de tipo que indica el tipo de mensaje (por ejemplo, open, update, notification, keepalive, etc.).  
  
Esta sección trata sobre los siguientes temas:  
  
Open Messages  
Update Messages  
Keepalive Messages  
Notification Messages  
Route-Refresh Messages

**Open Messages**

Después de establecer una conexión TCP entre dos sistemas BGP, intercambian Open Messages BGP para crear una conexión BGP entre ellos. Una vez que se establece la conexión, los dos sistemas pueden intercambiar mensajes BGP y tráfico de datos.

Los Open Messages consisten en el encabezado BGP más los siguientes campos:  
  
***Versión***: el número de versión actual de BGP es 4.  
***Local AS number***: usted configura esto al incluir la declaración del sistema autónomo en el nivel de jerarquía [edit-routing-options] o [edit logical-systems logical-system-name routing-options] .  
***Hold time*** *-* Valor propuesto de tiempo de espera. Configura el tiempo de espera local con la instrucción BGP hold-time.  
***BGP Identifier***: dirección IP del sistema BGP. Esta dirección se determina cuando el sistema se inicia y es el mismo para cada interfaz local y cada par BGP. Puede configurar el identificador BGP al incluir la instrucción id del enrutador en el nivel de jerarquía [edit routing-options] o [edit logical-systems logical-system-name routing-options] . Por defecto, BGP usa la dirección IP de la primera interfaz que encuentra en el enrutador.  
***Parameter field length and the parameter itself***: estos son campos opcionales.

**Update Messages**

Los sistemas BGP envían mensajes de actualización para intercambiar información de alcance de la red. Los sistemas BGP usan esta información para construir un gráfico que describe las relaciones entre todos los AS conocidos.  
  
Los mensajes de actualización consisten en el encabezado BGP más los siguientes campos opcionales:  
  
***Unfeasible routes length***: longitud del campo de rutas retiradas  
***Withdrawn routes***: prefijos de direcciones IP para las rutas que se retiran del servicio porque ya no se consideran accesibles  
***Total path attributes length***: longitud del campo de atributos de ruta; enumera los atributos de ruta para una ruta factible a un destino  
***Path attributes***: propiedades de las rutas, incluido el origen de la ruta, el Multiple Exit Discriminator (discriminador de salida múltiple) (MED), la preferencia del sistema de origen para la ruta e información sobre agregación, comunidades, confederaciones y reflexión de ruta.  
***Network layer reachability information* (NLRI)**: prefijos de la dirección IP de las rutas factibles anunciadas en el mensaje de actualización

**Keepalive Messages**

Los sistemas BGP intercambian Keepalive Messages para determinar si un enlace o host ha fallado o ya no está disponible. Los mensajes de Keepalive se intercambian con la frecuencia suficiente para que el temporizador de espera no caduque. Estos mensajes consisten solo en el encabezado BGP.

**Notification Messages**

Los sistemas BGP envían mensajes de notificación cuando se detecta una condición de error. Después de enviar el mensaje, la sesión BGP y la conexión TCP entre los sistemas BGP se cierran. Los mensajes de notificación consisten en el encabezado BGP más el código de error y el subcódigo, y los datos que describen el error.  
  
**Route-Refresh Messages**

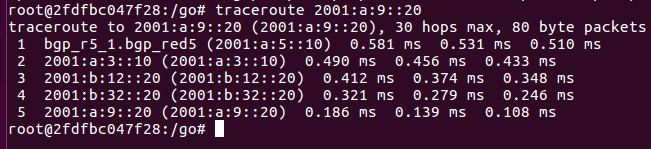
Los sistemas BGP envían mensajes de actualización de ruta a un par solo si han recibido el anuncio de capacidad de actualización de ruta del par. Un sistema BGP debe anunciar la capacidad de actualización de ruta a sus pares utilizando el anuncio de capacidades BGP si desea recibir mensajes de actualización de ruta. Este mensaje opcional se envía para solicitar actualizaciones dinámicas, entrantes de rutas BGP de pares BGP o para enviar actualizaciones de rutas salientes a un interlocutor BGP.  
  
Los mensajes de actualización de ruta constan de los siguientes campos:  
  
***Address Family Identifier*** (AFI-Address) (16 bits).  
***Campo Res-Reserved*** (8-bit), que debe ser puesto a 0 por el emisor e ignorado por el receptor.  
***Subsequent Address Family Identifier*** (Identificador de familia de direcciones subsiguientes SAFI) (8 bits).

Si un par sin la capacidad de actualización de ruta recibe un mensaje de solicitud de actualización de ruta desde un par remoto, el receptor ignora el mensaje.

11.- Aplicar políticas de distribución en BGP:

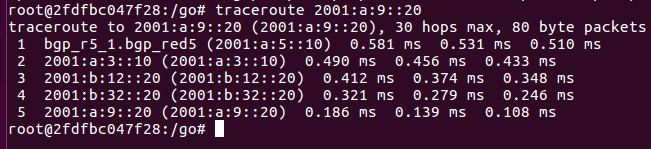
11.1.- Para que el sistema autónomo AS101 prefiera llegar a AS303 a través de AS202

Se agrego el comando neighbor 2001:b:12::20 weight 500 en el router b1 (as 101) para que prefiera llegar al as303 a través de as202:



11.2.- El sistema autónomo AS303(AS2727 en nuestro caso) bloquee las rutas aprendidas del sistema autónomo AS101(AS1919 en nuestro caso).

Se eliminó en el router b1 (AS 101) el neighbor que estaba conectado directamente a b3 (AS 303). Se hizo ping desde un dispositivo del AS 101 al AS 303 y se verificó que pasa por el AS 202



#### Comandos de ayuda

##### Docker: Listar las instancias

Las instancias en docker se listan con el comando “docker ps”.

|  |
| --- |
| $ sudo docker ps  CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES  $ sudo docker ps -a  CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES  4f8b4d122873 ospf:20180419 "/usr/bin/supervisord" 16 minutes ago Exited (0) 6 seconds ago ospf\_r1\_1  953c9ce6139a ospf:20180419 "/usr/bin/supervisord" 16 minutes ago Exited (0) 6 seconds ago ospf\_r2\_1 |

##### Docker: Eliminar containers

Las instancias en docker se pueden eliminar.

|  |
| --- |
| $ sudo docker rm -f ospf\_r1\_1 ospf\_r2\_1  ospf\_r1\_1  ospf\_r2\_1 |

##### Docker: Borrar todas las redes que no se usan

Usualmente cuando se quiere regenerar un entorno con docker-compose, es necesario eliminar los recursos creados previamente. En este caso, se muestra como ejemplo como borrar todas las redes que ya no se utilizan.

|  |
| --- |
| $ sudo docker network prune  WARNING! This will remove all networks not used by at least one container.  Are you sure you want to continue? [y/N] y  Deleted Networks:  ospf\_nr1  ospf\_oam  ospf\_nr2 |