REDES DE COMPUTADORAS

Tema 4: Ruteo dinámico BGP

Integrantes:

- *Izquierdo, Agustina*
- *Navarro, Matias Alejandro*

Carrera: *Ing. en computación*

Profesor: *Matías R. Cuenca del Rey*

Ayudantes alumnos: *Elisabeth Leonhardt - Andrés Serjoy - Mariano Aguero - Matthew Aguerreberry - Matias Kleiner - Agustin Montero - Ramiro Morales - Sergio Sulca - Natasha Tomattis*

Fecha: *11/04/2019*

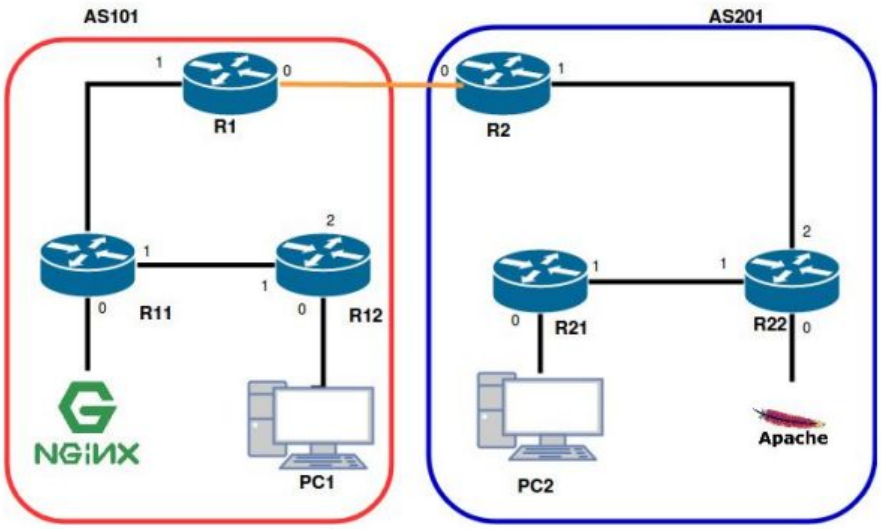
Izquierdo, Agustina
Navarro, Matías

Ejercicio: Ruteo dinámico BGP

Recomendaciones

- Lea con cuidado las consignas.
- Tenga certeza de los comandos que ejecuta.

Diagrama



Consignas

Configuración de interfaces

- Configurar interfaces de los routers y computadoras. En este práctico se solicita al alumno que complete todas las interfaces. Las direcciones IPs no se pueden repetir entre los distintos grupos de trabajo.
    - IPv4: 192.<group_number>.0.0/16
    - IPv6: aaaa:<group_number>::/64

Configuración de ruteo dinámico interno OSPF

- Configurar OSPF en todos los router dentro de un mismo AS.

Configuración de ruteo dinámico externo BGP

- Configurar BGP en los router R1 y R2 para IPv4 e IPv6.
- Publicar la red de los web servers de cada Sistema Autónomo.
- Comprobar la configuración haciendo consultas *http* (comando *curl*) entre clientes y los servidores web (*nginx* y *apache*) de distintos Sistemas Autónomos.

Recomendación

- *(OPCIONAL)* Las configuraciones de cada equipo deben ser provistas a través de un servidor TFTP.

Links de ayuda

Configuración de BGP en Cisco
http://www.cisco.com/c/en/us/support/docs/ip/border-gateway-protocol-bgp/23675-2 7.html
Ejemplo de redistribución de rutas OSPF en BGP
http://www.cisco.com/c/es_mx/support/docs/ip/border-gateway-protocol-bgp/5242-bgp-ospf-redis.pdf
Docker NGINX https://hub.docker.com/_/nginx Docker APACHE
https://hub.docker.com/_/httpd

Topología de la red

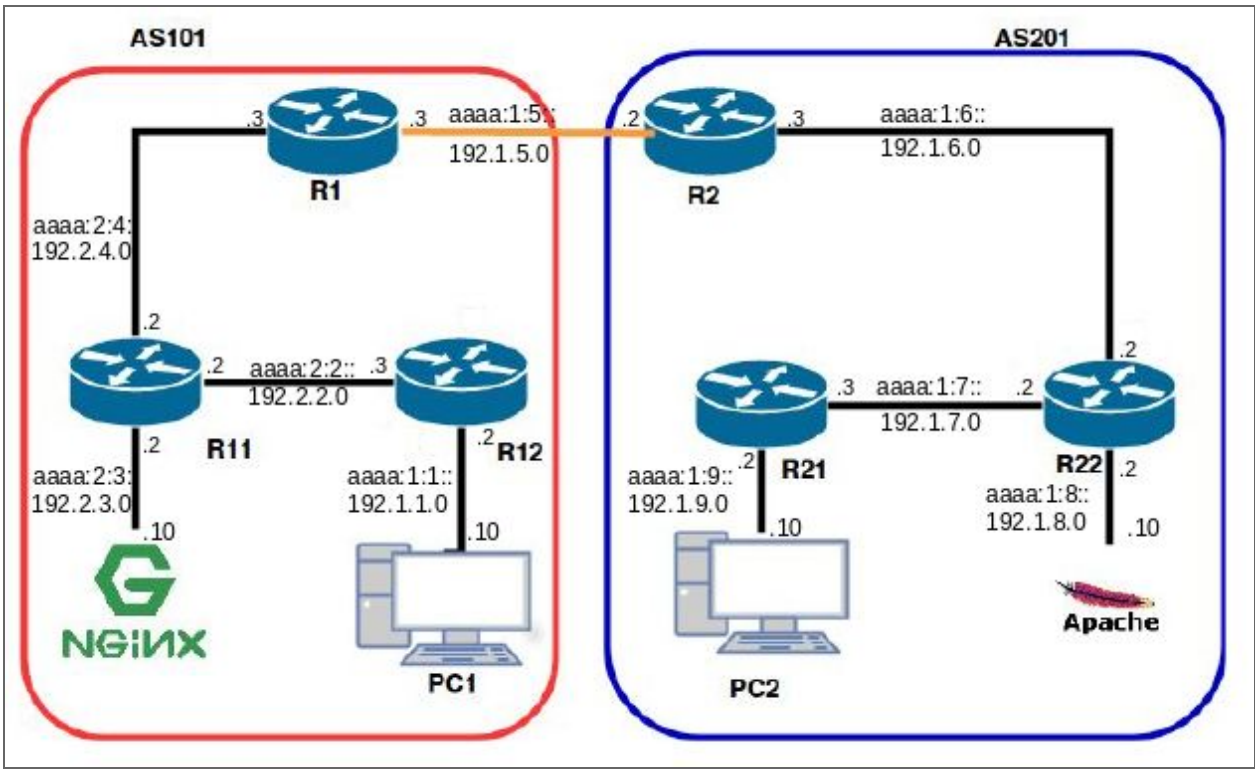


Tabla de interfaces por router

Sistema autónomo 101

Dispositivo	Interfaz de red	Dirección IP
Nginx	eth0	192.2.2.10
		aaaa:2:2::10
PC1	eth0	192.2.1.10
		aaaa:2:1::10
R12	eth0	192.2.1.2
		aaaa:2:1::2
	eth1	192.2.3.3
		aaaa:2:3::3
R11	eth0	192.2.2.2
		aaaa:2:2::2
	eth1	192.2.3.2
		aaaa:2:3::2
	eth2	192.2.4.2
		aaaa:2:4::2
R1	eth0	192.2.4.3
		aaaa:2:4::3
	eth1	192.2.5.3
		aaaa:2:5::3

Sistema autónomo 201

Dispositivo	Interfaz de red	Dirección IP
Apache	eth0	192.1.8.10
		aaaa:1:8::10
PC2	eth0	192.1.9.10
		aaaa:1:9::10
R21	eth0	192.1.7.3
		aaaa:1:3::3
	eth1	192.1.9.2
		aaaa:1:9::2
R22	eth0	192.1.6.2
		aaaa:1:6::2
	eth1	192.1.7.2
		aaaa:1:7::2
	eth2	192.1.8.2
		aaaa:1:8::2
R2	eth0	192.2.5.2
		aaaa:2:5::2
	eth1	192.1.6.3
		aaaa:1:6::3

Configuración de ruteo dinámico interno OSPF

| Se configuraron los routers virtuales (r11, r12, r21 y r22) pasando los archivos de configuración correspondientes como volúmenes. Además se abrieron los puertos para poder verificar las rutas aprendidas por OSPF.

```
r11:
  build: ./../images/ospf/.
  volumes:
    - ./../volumes/ospf/r11/zebra.conf:/etc/quagga/zebra.conf:ro
    - ./../volumes/ospf/r11/ospfd.conf:/etc/quagga/ospfd.conf:ro
    - ./../volumes/ospf/r11/ospf6d.conf:/etc/quagga/ospf6d.conf:ro
    - ./../volumes/ospf/supervisord.conf:/etc/supervisord.conf:ro
  image: ospf:20180508
  privileged: true
  ports:
    #admin
    - 10111:2601
    #ospf ipv4
    - 10114:2604
    #ospf ipv6
    - 10116:2606
  networks:
    r11_r12:
      ipv4_address: 192.2.3.2
      ipv6_address: aaaa:2:3::2
    r11_nginx:
      ipv4_address: 192.2.2.2
      ipv6_address: aaaa:2:2::2
    r11_r1:
      ipv4_address: 192.2.4.2
      ipv6_address: aaaa:2:4::2
```

Los web server se levantaron usando como base una imagen de docker hub, y se abrió el puerto 80 para recibir consultas HTTP.

```
services:
  nginx:
    image: nginx:1.15.10-alpine
    privileged: true
    ports:
      - 8080:80
    networks:
      r11_nginx:
        ipv4_address: 192.2.2.10
        ipv6_address: aaaa:2:2::10
```

En el makefile se automatiza el proceso de eliminar las rutas por defecto para que se utilicen las rutas aprendidas por OSPF, además se agrega la ruta por defecto en el web server para que salga por el router 11 (y en el caso de Apache, por el router 22).

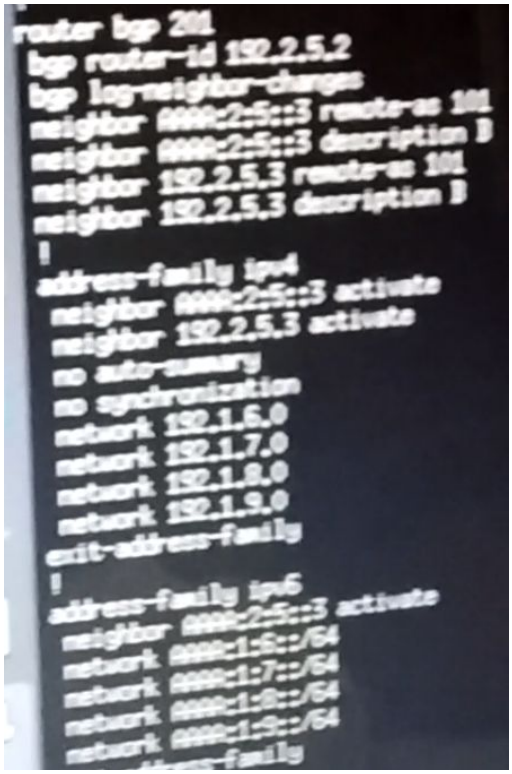
```
all : setup delete_routes

setup:
    docker-compose -p tp4 up -d

clean:
    docker network prune
    docker-compose -p tp4 down

delete_routes:
    docker exec -ti tp4_r11_1 ip -6 route del default;
    docker exec -ti tp4_r11_1 ip route del default;
    docker exec -ti tp4_r12_1 ip -6 route del default;
    docker exec -ti tp4_r12_1 ip route del default;
    docker exec -ti tp4_nginx_1 ip -6 route del default;
    docker exec -ti tp4_nginx_1 ip route del default;
    docker exec -ti tp4_nginx_1 ip route add default via 192.2.2.2;
    docker exec -ti tp4_nginx_1 ip -6 route add default via aaaa:2:2::2;
```

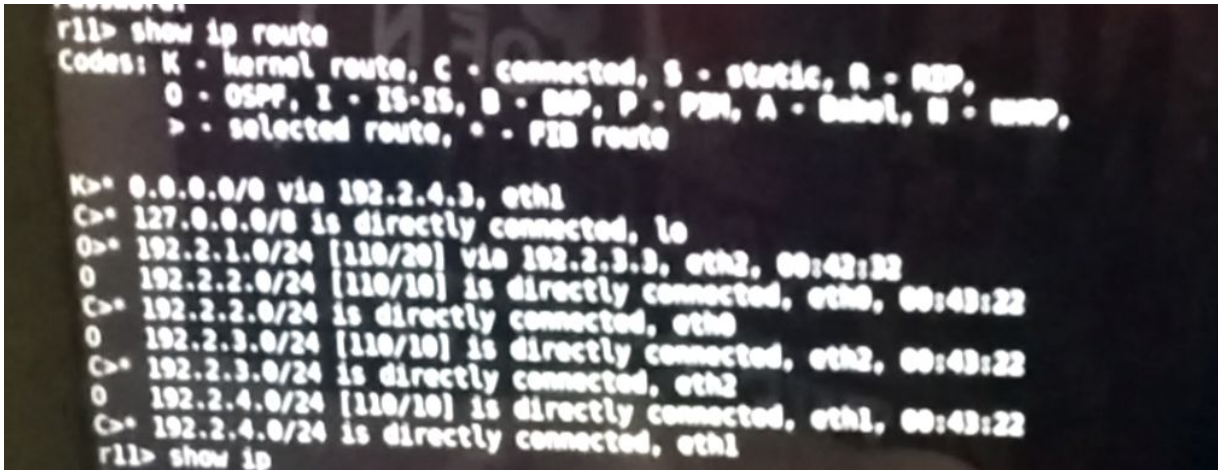
Configuración de ruteo dinámico externo BGP

Los router R1 y R2 físicos se configuraron mediante una consola serie desde Putty, con los siguientes comandos para bgp:

```
hostname r1
password admin
!
router bgp 101
  bgp router-id 192.2.5.3
!
  no auto-summary
  no synchronization
!
  neighbor aaaa:2:5::2 remote-as 201
  neighbor aaaa:2:5::2 description A
!
  neighbor 192.2.5.2 remote-as 201
  neighbor 192.2.5.2 description A
!
  network 192.2.1.0
  network 192.2.2.0
  network 192.2.3.0
  network 192.2.4.0
  neighbor 192.2.5.2 activate
!
  address-family ipv6
  network aaaa:2:1::/64
  network aaaa:2:2::/64
  network aaaa:2:3::/64
  network aaaa:2:4::/64
  neighbor aaaa:2:5::2 activate
!
```

Además se configuraron las interfaces de cada router y el ruteo IGP OSPF para que conozca la red de cada sistema autónomo. **Aclaración:** para conectar cada router entre sí se utilizó un **cable cruzado** debido a que los conectores TX y RX deben estar cruzados entre routers (TX de r1 a RX de r2 y viceversa)





Aquí se puede observar como uno de los Routers (R11) configurado con OSPF aprendió las rutas gracias a este protocolo.

Prueba de funcionamiento de las consultas HTTP a los webserver

Para comprobar que cada web server era accesible desde el otro sistema autónomo, se hizo un wget (GET al index.html) a la dirección del server especificando el puerto 80 (HTTP).