

ARQUITECTURAS DE REDES AVANZADAS

QUAGGA

José Luis Cánovas Sánchez



Open Source Routing

Resumen

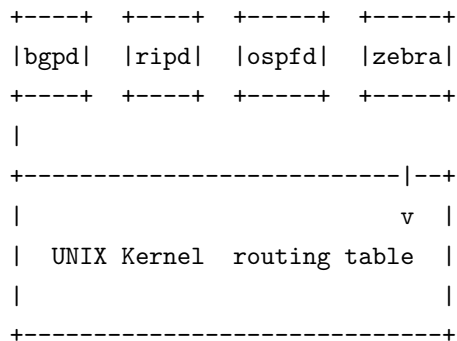
En este informe se redacta el desarrollo usando la herramienta QUAGGA para el despliegue del escenario de red IPv6.

Índice

1. Introducción	2
2. Topología de trabajo	2
3. Instalar Quagga	4
4. Configuración de OSPF	4
5. Configuración de RIPng	4
6. Configuración de BGP	4

1. Introducción

Quagga es un proyecto de software que provee de utilidades para encaminamiento de red en sistemas UNIX, proveyendo el demonio zebra originario de otro proyecto y que permite aplicar configuraciones de algoritmos de encaminamiento a las tablas de red de la máquina. Los algoritmos se ejecutan como demonios que toman la información de la red y aplican protocolos conocidos, tales como OSPF, RIP, IS-IS y BGP, en múltiples de sus versiones, y por medio de zebra los aplican a la red.



Quagga System Architecture

Vamos a configurar tres de estos protocolos con Quagga en cuatro máquinas que harán las veces de routers. En la sección 2 se muestra la topología que se va a implementar. En las secciones 3 a 6 se explicará cómo instalar y configurar Quagga con los mecanismos OSPF, RIP y BGP para IPv6.

La topología usada en este estudio de Quagga se asemeja mucho a la de la práctica 1 de ARA, pero se puede llevar a otros proyectos, como LEGO, fácilmente. Más adelante se muestra la topología en Figura 1, y con hacer corresponder R1 y R2 con los routers de las organizaciones de LEGO, las redes XXXX:0:1::/64 con las subredes de cada organización, y las subredes XXXX::/64 con una subred un nivel más interna, tenemos la topología de LEGO ampliada. La salida a internet se debería realizar con tunelado a IPv4 en R1 y R2, si no se dispone de IPv6 pública. En caso de tener un rango de direcciones IPv6 públicas, habría que cambiar los valores de cada subred para que sean válidos, pero queda fuera de este estudio.

2. Topología de trabajo

Partimos de dos Sistemas Autónomos con números 17 y 71 (curiosidad, son primos permutables entre sí) y mecanismo de encaminamiento interno por OSPF, con un área backbone y un área 1 stub para el AS17, un mecanismo basado en RIPng para el AS1 y BGP entre ambos AS, como se muestra en la Figura 1.

Esta topología lógica la llevamos a la práctica con Virtual Box, con cuatro máquinas Debian sin interfaz gráfica, y con tarjetas de red virtuales configuradas como redes internas del siguiente modo:

- R1: Adaptador 1 Red Interna *AS17* - Adaptador 2 Red Interna *interAS* - Adaptador 3 NAT.
- R2: Adaptador 1 Red Interna *AS71* - Adaptador 2 Red Interna *interAS* - Adaptador 3 NAT.

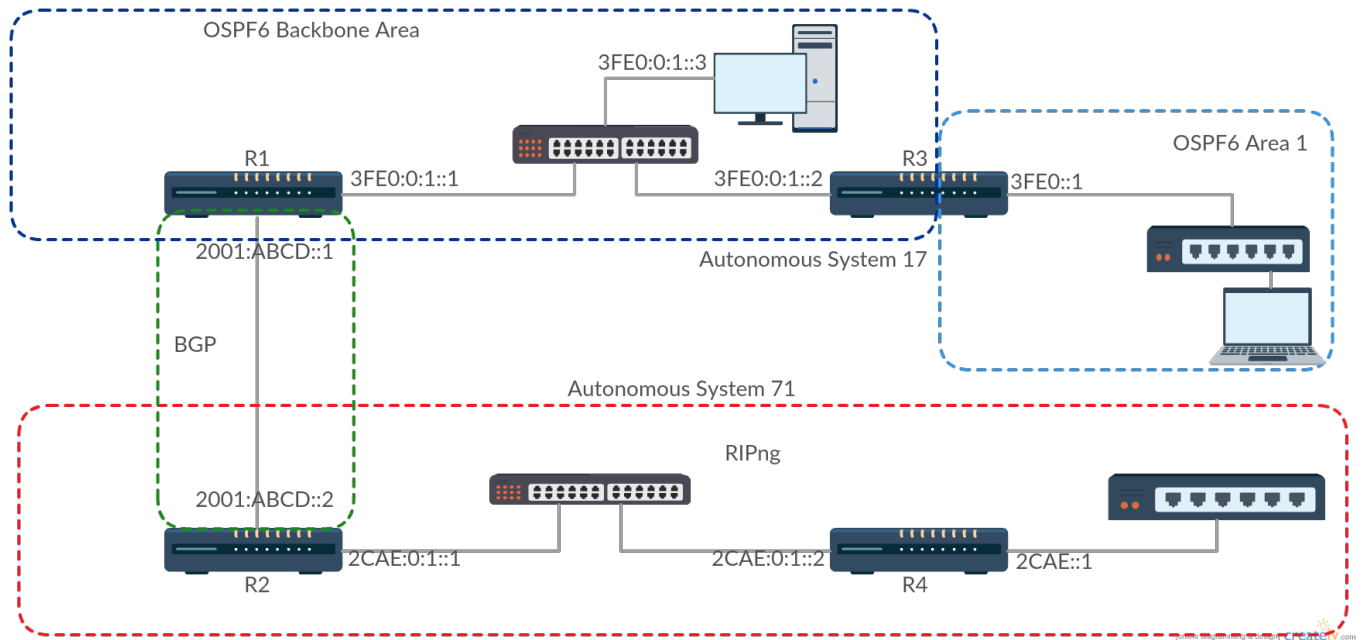


Figura 1: Topología

- R3: Adaptador 1 Red Interna *AS17* - Adaptador 2 Red Interna *AS17Area1* - Adaptador 3 NAT.
- R4: Adaptador 1 Red Interna *AS71* - Adaptador 2 Red Interna *AS71Subred* - Adaptador 3 NAT.

Con el NAT conseguimos salida a internet para poder descargar los ficheros de configuración. Con las otras conseguimos simular los switches y conectar cada router por las interfaces correspondientes.

La configuración de `/etc/network/interfaces` sería, por ejemplo para R1:

```
auto lo
iface lo inet loopback

auto eth0
iface eth0 inet6 static
address 3FE0:0:1::1
netmask 64

auto eth1
iface eth1 inet6 static
address 3FE0::1
netmask 64

auto eth2
iface eth2 inet dhcp
```

3. Instalar Quagga
4. Configuración de OSPF
5. Configuración de RIPng
6. Configuración de BGP