

Automatizando BIRD y tener BGP sin morir en el intento.

Ariel Antigua

ariel@antigua.pw

<https://aanetworks.org>

AS207036

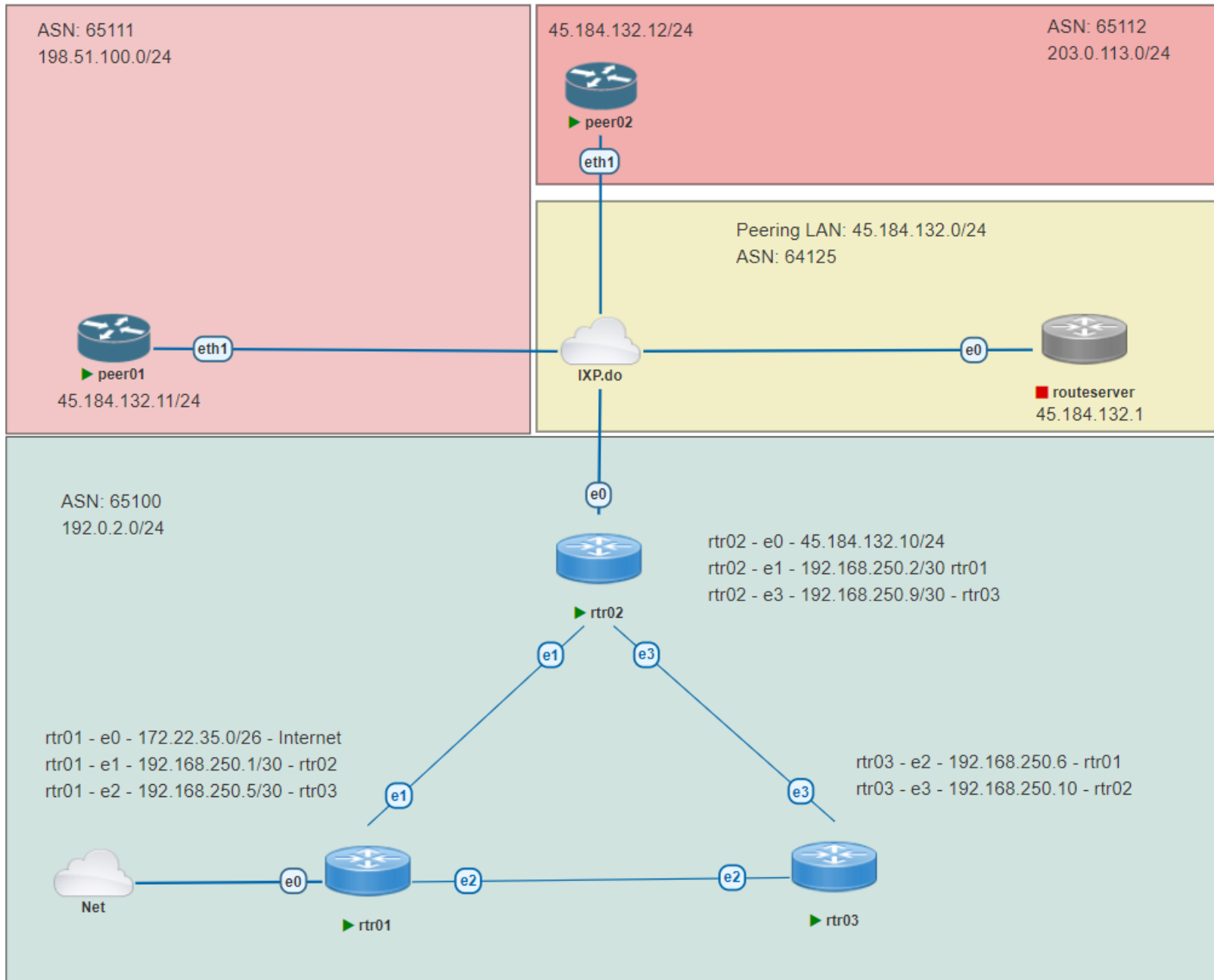


BIRD (BIRD Internet Routing Daemon)

- Es una suite que nos permite convertir un servidor Linux/BSD en un enrutador con soporte para BGP y otros protocolos de enrutamiento.
- En nuestro caso, configuraremos **BIRD** usando una herramienta llamada **BCG** (BIRD Config Generator)
<https://github.com/aredan/bcg/tree/core-rtr>
- Git para almacenar las configuraciones de BCG (opcional).
- Con Ansible podemos enviar el archivo config.yml a cada router (opcional).

Escenario de red.

- Administrar un servidor Linux el cual hemos convertido en router usando BIRD, es una tarea relativamente fácil, el tema se complica cuando contamos con tres o más routers con varias conexiones a diferentes ASN (Números Autónomos) o incluso conexión a un IXP local.
- En cada servidor Linux (Ubuntu) contamos con la herramienta BCG, toda la configuración necesaria proviene desde un archivo YAML llamado **config.yml** y ubicado por defecto en `/etc/bcg/`.
- Por cada router con BIRD debemos tener un archivo `config.yml` con las configuraciones únicas de este archivo, la mejor forma de administrar estas configuraciones es almacenando dicho archivo en un repositorio Git.

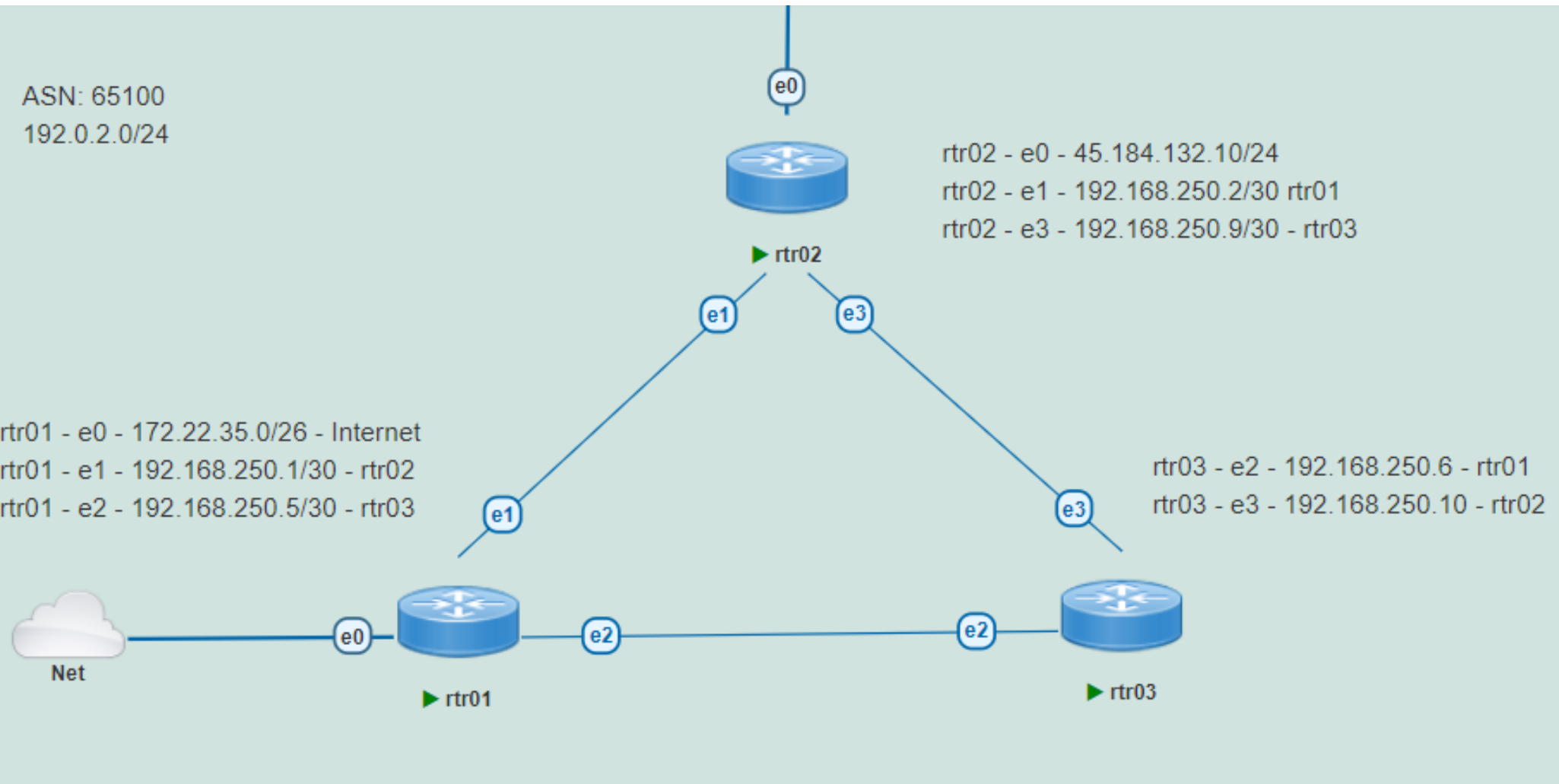


ASN: 65100
192.0.2.0/24

rtr01 - e0 - 172.22.35.0/26 - Internet
rtr01 - e1 - 192.168.250.1/30 - rtr02
rtr01 - e2 - 192.168.250.5/30 - rtr03

rtr02 - e0 - 45.184.132.10/24
rtr02 - e1 - 192.168.250.2/30 - rtr01
rtr02 - e3 - 192.168.250.9/30 - rtr03

rtr03 - e2 - 192.168.250.6 - rtr01
rtr03 - e3 - 192.168.250.10 - rtr02



config.yml

asn: 65100

router-id: 192.0.2.2

prefixes:

- 192.0.2.0/24
- 2001:db8::/48

peers:

IXPDO:

asn: 64125

type: peer

neighbors:

- 45.184.132.1
- 2801:182:132::1

Global Configuration Options

Option	Usage
asn	ASN of this router
router-id	Router ID of this router
prefixes	List of prefixes to originate
irrdb	IRRDB to query prefix sets from (default is rr.ntt.net which includes generated route objects from RPKI ROAs)
rtr-server	IP address or hostname of RPKI RTR server (default is 127.0.0.1)
keep-filtered	Should BIRD keep filtered routes
peers	Map of name to peer (see below)
merge-paths	Enable merge paths on kernel export
pref-src4	Preferred source IPv4 to export to kernel
pref-src6	Preferred source IPv6 to export to kernel
filter-default	Should default routes be denied?
enable-default	Add static default routes

config.yml

Peer Configuration Options

Option	Usage
asn	Neighbor ASN
type	Type of peer (core, upstream, peer, downstream, import-valid)
local-pref	BGP LOCAL_PREF
disabled	Should neighbor sessions be disabled?
passive	Should neighbor sessions listen passively for BGP TCP connections?
multihop	Should neighbor sessions allow multihop?
password	BGP MD5 Password
port	BGP Port (default 179)
neighbors	List of neighbor IP addresses
pre-import	List of BIRD expressions to execute after the prefilter and before the prefix filter
pre-export	List of BIRD expressions to execute before the export filter
prepends	Number of times to prepend local AS to
import-limit4	Maximum number of IPv4 prefixes to allow before disabling the session
import-limit6	Maximum number of IPv6 prefixes to allow before disabling the session
skip-filter	Disable the universal bogon filter (Dangerous!)

rs-client	Enable route server client
rr-client	Enable route reflector client
bfd	Enable BFD
session-global	String to add to session global config
enforce-first-as	Reject routes that don't have the peer ASN as the first ASN in path
enforce-peer-nexthop	Reject routes where the next hop doesn't match the neighbor address
export-default	Should a default route be sent over the session? (default false)
no-specifics	Don't send specific routes (default false, make sure to enable export-default or else no routes will be exported)
allow-blackholes	Accept community (ASN,1,666) to blackhole /32 and /128 prefixes
strip-private-asns	Should private ASNs be stripped from path before exporting? (in range [64512..65534, 4200000000..4294967294])
communities	List of BGP communities to add on export (two comma-separated values per list element; example 0,0)
large-communities	List of BGP large communities to add on export (three comma-separated values per list element; example 0,0,0)

Dependencias de BCG

- Para poder ejecutar **bcg** debemos tener instalado **bgpq4** en nuestro servidor Linux, esto es debido a que gran parte de los filtros son creados con información solicitada a servicios tales como **PeeringDB** y algunos **IRR**.
- En Ubuntu server la versión disponible en apt es la versión 3 (bgpq3), debemos instalar la versión 4 (bgpq4) de manera manual si es el caso que estamos corriendo Ubuntu 20.04.

Ejecutando BCG

- Llega el día que adquirimos una nueva conexión a un IXP (IXP.DO) y nos asignaron la dirección IP 45.184.132.10/24, necesitamos definir este nuevo peer de BGP en nuestro config.yml y luego ejecutar el binario **bcg** para crear las configuraciones que serán usadas por BIRD.

```
/usr/bin/bcg -dryRun
```

Esta opción nos permite validar que toda la configuración en config.yml esta correcta.

```
/usr/bin/bcg
```

Usage for bcg <https://github.com/natesales/bcg>:

- config string
configuration file in YAML, TOML, or JSON format (default "/etc/bcg/config.yml")
- debug
show debugging messages
- dryRun
skip modifying BIRD config
- noGenerateUi
disable generating web UI
- noConfigure
disable configuring bird at end of run
- output string
directory to write output files to (default "/etc/bird/")
- socket string
BIRD control socket (default "/run/bird/bird.ctl")
- uiFile string
file to store web UI (default "/tmp/bcg-ui.html")

```
root@rtr02:/etc/bird#
root@rtr02:/etc/bird# /home/eve/bcg
INFO[0000] Starting bcg dev
INFO[0000] Using YAML configuration format
INFO[0000] Using IRRDB server rr.ntt.net
INFO[0000] Using RTR server 127.0.0.1
INFO[0000] Using RTR port 8282
INFO[0000] Checking config for IXPDO AS64125
INFO[0000] [IXPDO] type: import-valid
INFO[0000] [IXPDO] has no IPv4 import limit configured. Setting to 1000000
INFO[0000] [IXPDO] has no IPv6 import limit configured. Setting to 150000
INFO[0000] [IXPDO] local pref: 100
INFO[0000] [IXPDO] max prefixes: IPv4 1000000, IPv6 150000
INFO[0000] [IXPDO] export-default: false
INFO[0000] [IXPDO] no-specifics: false
INFO[0000] [IXPDO] allow-blackholes: false
INFO[0000] [IXPDO] strip-private-asns: false
INFO[0000] [IXPDO] communities:
INFO[0000] [IXPDO] large-communities:
INFO[0000] [IXPDO] neighbors: 45.184.132.1, 2801:182:132::1
INFO[0000] [IXPDO] Writing config
INFO[0000] [IXPDO] Wrote config
INFO[0000] reconfiguring bird
INFO[0000] Connecting to BIRD socket
INFO[0000] Connected to BIRD socket
INFO[0000] BIRD init response: 0001 BIRD 2.0.7 ready.
INFO[0000] Sending BIRD command: configure
INFO[0000] Sent BIRD command: configure
INFO[0000] BIRD response: 0002-Reading configuration from /etc/bird/bird.conf
0003 Reconfigured
root@rtr02:/etc/bird#
```

bird.conf

- Es el principal archivo de configuración, tenemos la opción de incluir otros archivos y de esta manera dividir la configuración para entenderla mejor.
- En el caso de bcg, tendremos bird.conf y los archivos de configuración de cada peer.

```
include "manual*.conf"
```

```
include "AS*.conf"
```

bird.conf

- En caso de necesitar configuración de BIRD que no este soportada por bcg, se puede agregar en un archivo con el prefijo “manual*.conf” y esta configuración sera cargada en bird.

manual_OSPF.conf

Dentro de este archivo podríamos colocar la información necesaria para configurar el protocolo de OSPF en BIRD si es necesario.

Archivos de configuración para BIRD.

- Dependiendo de la cantidad de peers que sean agregados en config.yml, bcg generara un archivo de configuración específico para cada peer.

AS###_NOMBRE.conf

- AS54125_IXPDO.conf

Actualizemos config.yml

- Decidimos hacer peer directo con uno de los miembros del IXP (puede ser cualquier otro con una conexión dedicada), la razón de este caso particular puede ser que el ISP no exporta todas sus rutas al route-server o simplemente quiere tener mejor capacidad con nuestra red.
- Que tal si lo hacemos en vivo !!

- Demo en vivo.



Preguntas y Respuestas !

- Que te pareció el tema?
- Usarías BIRD en producción ?