RIPE 81

QRATOR LABS

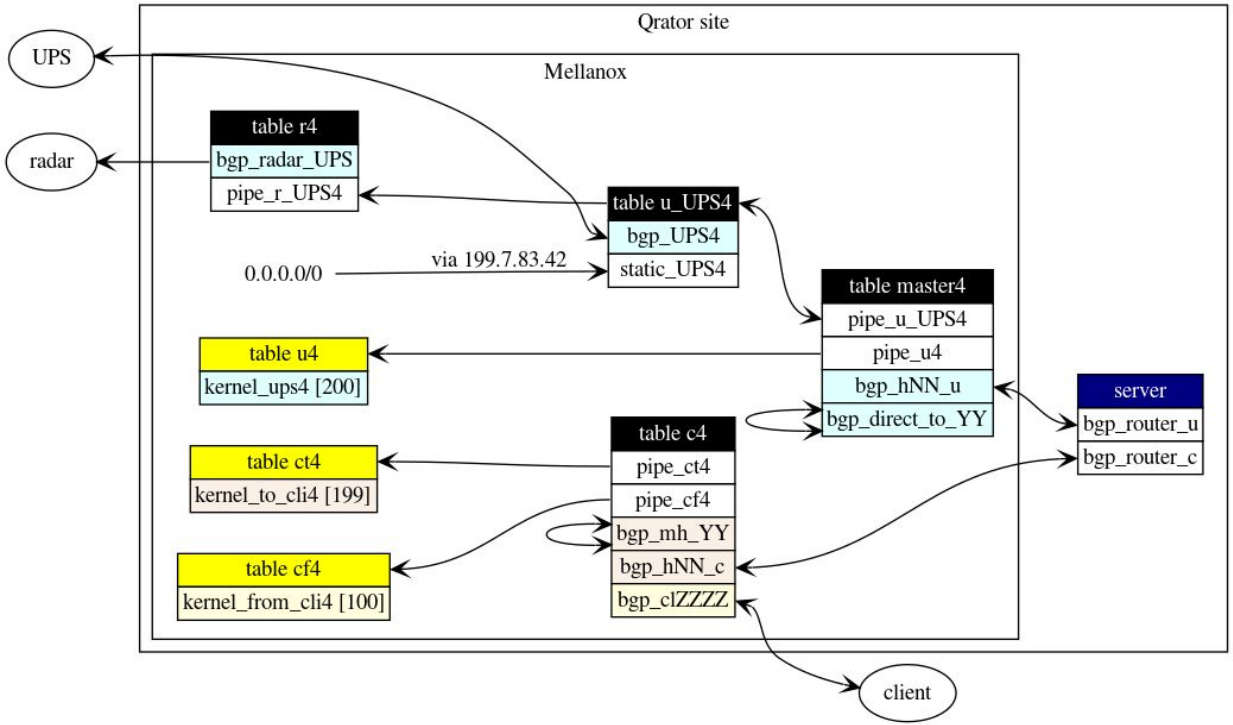
The birdist cookbook*

* BGP version

Alexander Zubkov

Bird at Qrator: Where

- Linux
- Servers
- Arista
- Mellanox



Bird at Qrator: What

- bird2: VRF, BGP bind, IPv4 & IPv6
- BGP
  - BFD
  - Large communities
  - FlowSpec
- filtering & tables 

I/O loop & timeouts

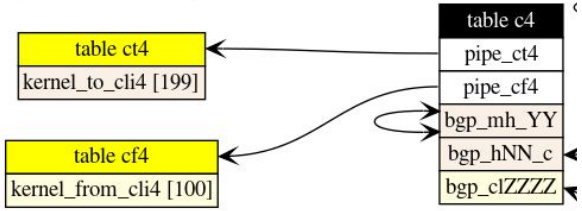
- Single thread: doesn’t scale to several cores
- IO loop, timeouts: BGP hold time, etc.

```
Jan 12 15:55:29 budic bird: I/O loop cycle took 9270 ms for 4 events
Jan 12 15:56:28 budic bird: I/O loop cycle took 9101 ms for 6 events
```

- Cause: full view, logging (file, syslog — blocking IO)
- Logging via UDP: fire-and-forget
  - https://static.qrator.net/bird/bird-log-udp.patch

Kernel protocol

- Kernel
  - 1 protocol — 1 kernel table
  - 1 protocol — 1 bird table
  - add tables & pipes
  - fixed proto (krt_source)
  - https://bird.network.cz/pipermail/bird-users/2020-June/014620.html




```
default via 192.0.2.1 dev eth0 proto bird metric 32
```

Protocol state

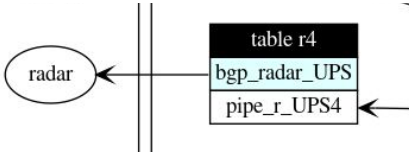
- configure → protocol state from config
  - sync admin state to config
  - patch: configure soft keeps protocol state
  - https://static.qrator.net/bird/bird-keep-state.patch
- strict bind: no address → down
  - https://bird.network.cz/pipermail/bird-users/2020-January/014138.html

```
bgp_01     BGP        ---        down   12:09:31.264  Error: No listening socket
```

  - Linux: IP_FREEBIND
  - https://static.qrator.net/bird/bird-nonlocal-bind-1-io.patch
    https://static.qrator.net/bird/bird-nonlocal-bind-2-bgp-always.patch

Same peer IP

- same neighbor address
- only 1 session active




```
bgp_radar_ntt BGP    ---      start  2020-06-30   Idle          BGP Error: Hold timer expired
bgp_radar_pccw BGP   ---      up   2020-06-30     Established
```

- no solution yet

eBGP first AS

- bird does not force on import
  - use filter
  - `bgp-enforce-first-as` (v2.0.8)
- export prepend (bird1 → bird2)
  - export filter (after → before prepend)
  - `rs client` (v2.0.3), before `rs client` → `rs client` only
  - https://bird.network.cz/pipermail/bird-users/2017-December/011774.htm
    https://bird.network.cz/pipermail/bird-users/2018-September/012751.html
    https://bird.network.cz/pipermail/bird-users/2018-November/012853.htmll
    https://bird.network.cz/pipermail/bird-users/2019-March/013200.html

direct/recursive

- Neighbor: direct|multihop <N>
  - eBGP → direct
  - iBGP → multihop
- Channel: gateway direct|recursive
  - direct → gw direct
  - multihop → gw recursive
  - ~~multihop + gw direct~~
- gateway direct
  - non direct bgp_next_hop → drop

gw recursive: unreachable

- Route selection: localpref, ASPATH, ...

```
192.0.2.0/24       unicast [bgp_direct 2020-06-19] * (100) [AS64500i]
    via 192.168.20.254 on vlan20
    BGP.as_path: 64500
    BGP.next_hop: 100.64.20.254
    BGP.local_pref: 140
                   unreachable [bgp_mh 2020-08-25 from 192.168.0.7] (100) [AS64500i]
    BGP.as_path: 64511 64510 64500
    BGP.next_hop: 100.64.21.254
    BGP.local_pref: 150
```

- Set gw attribute to some stub

Recursive routes

- BGP, static
- only 1 level of recursion
  - https://bird.network.cz/pipermail/bird-users/2019-September/013820.html
    https://bird.network.cz/pipermail/bird-users/2020-March/014297.html
- iBGP → multihop → recursive
  - no visible difference

```
<WARN> Next hop address 198.51.100.3 resolvable through recursive route for 198.51.100.0/24
```

- no multipath

```
protocol static {
    route 192.0.2.0/24 recursive 198.51.100.3;
}
```

direct

```
198.51.100.0/24     unicast [bgp1 18:52:07.294 from 192.168.0.2] * (100) [i]
    via 203.0.113.5 on vlan10
    Type: BGP univ
    BGP.origin: IGP
    BGP.as_path:
    BGP.next_hop: 203.0.113.5
    BGP.local_pref: 100
192.0.2.0/24        unicast [static1 18:45:15.336] * (200)
    via 203.0.113.5 on vlan10
    Type: static univ
```

```
protocol static {
     route 192.0.2.0/24 recursive 198.51.100.3;
}
```

direct

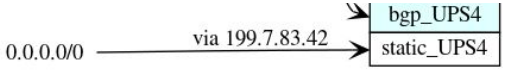
```
198.51.100.0/24     unicast [bgp1 18:52:07.294 from 192.168.0.2] * (100) [i]
    via 203.0.113.5 on vlan10
    Type: BGP univ
    BGP.origin: IGP
    BGP.as_path:
    BGP.next_hop: 203.0.113.5
    BGP.local_pref: 100
192.0.2.0/24        unicast [static1 18:45:15.336] * (200)
    via 203.0.113.5 on vlan10
    Type: static univ
```

recursive

```
198.51.100.0/24     unicast [bgp1 18:51:06.896 from 192.168.0.2] * (100) [i]
    via 203.0.113.5 on vlan10
    Type: BGP univ
    BGP.origin: IGP
    BGP.as_path:
    BGP.next_hop: 203.0.113.5
    BGP.local_pref: 100
192.0.2.0/24        unreachable [static1 18:45:15.336] * (200)
    Type: static univ
```

Aggregation

- no aggregation
- imitate: static recursive
  - filter unreachable
  - no original attributes




```
protocol static static_UPS4 {
    route 0.0.0.0/0 recursive 199.7.83.42 { … };
}
```

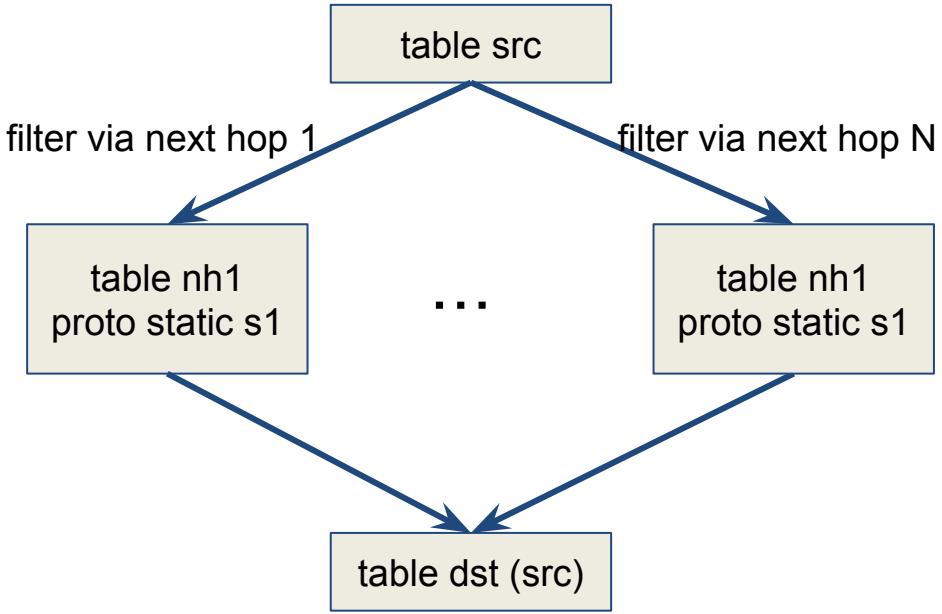
```
protocol static signal {
    route 100.64.5.1/32 recursive 0.0.0.0;
}
```

Multipath: protocols

- default: single route
- kernel
  - merge paths
  - best + equivalent
- bgp
  - add paths
  - all routes

Multipath: recursive

- recursive: only best
- static: imitate again
  - pipe to several tables
  - add static to each
  - pipe back



Bird on Mellanox

- switchdev: like a server
- map switch → Linux
  - VRF
- no offload
  - ipv4 route with ipv6 gw
- fixed stuff: VRF, BFD

Bird on Arista

- Linux, but proprietary
- userspace → data plane
  - kernel (multipath problems)
  - SDK
  - eAPI (cli over HTTP/JSON)
- protocol arista (SDK), bird 1.6
- bird2 → bird 1.6 → SDK
- bird2 → arista (BGP as an API)

Bird on Arista: BGP API

Arista

```
router bgp 64500
   distance bgp 32 32 32
   bgp transport listen-port 8179
   maximum-paths 32
   neighbor 127.0.0.2 remote-as 64500
   neighbor 127.0.0.2 transport remote-port 179
   neighbor 127.0.0.2 ebgp-multihop 1
   neighbor 127.0.0.2 additional-paths receive
   neighbor 127.0.0.2 route-map none out
   neighbor 127.0.0.2 maximum-routes 100000
   !
   address-family ipv4
     neighbor 127.0.0.2 activate
     neighbor 127.0.0.2 additional-paths receive
   !
```

Bird

```
protocol bgp {
     strict bind yes;
     neighbor 127.0.0.1 port 8179 as 64500;
     local 127.0.0.2 as 64500;
     multihop 1;
     rr client;
     ipv4 {
          next hop keep;
          add paths tx;
          export filter to_dataplane;
          import none;
     };
}
```

Bird on Arista: VRF

- netns
- several bird daemons
- route exchange?
- BGP over unix socket
  - haproxy: inet → unix

Bird on Arista: VRF

- netns
- several bird daemons
- route exchange?
- BGP over unix socket
  - haproxy: inet → unix

Bird

```
protocol bgp {
    strict bind yes;
    neighbor 127.0.0.111 port 8888;
    neighbor as 64500;
    local 127.0.0.1 as 64500;
    multihop 1;
}
```

haproxy

```
listen bgp-to
    bind 127.0.0.111:8888
    server vrf unix@/var/run/bgp-to-vrf.sock
listen bgp-from
    bind /var/run/bgp-from-vrf.sock
    server vrf 127.0.0.1:179 source 127.0.0.111
```

- Bird mailing list
  - https://bird.network.cz/?m_list
- My contacts
  - Alexander Zubkov
  - green@qrator.net