Dual Homing

Sergey Dremin

08.25.20

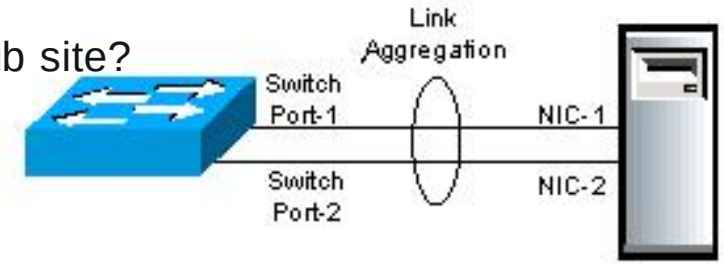
COMCAST

Agenda

- Historical Background
- Requirements
- Solutions Overview
- Detailed Design and Deployment
- Demo and QA

Historical Background

- First mass Comcast CDN caches deployed in regional data centers with two cables from two nics to one router, using ling aggregation group (LAG)
- Data center routers rarely need to be rebooted, that worked well.
- Then CDN caches deployed closer to customers in hub sites connected to one router, a residential u-ring router (RUR), in same configuration
- RURs deployed in pairs to provide uninterrupted residential and business internet service
- RURs need to be offlined every couple of months, causes CDN ops to need to monitor RUR maintenances, and drain sticky clients from caches connected to those RURs
- Why not connect to BOTH RURs in the hub site?



Dual Home - Basics

The objective of router redundancy is to increase availability, reduce maintenance, leverage internet address space flexibility, and ease installation of new servers.

The drawbacks of router redundancy compared with single homing is a more complex control plane and harder debug.

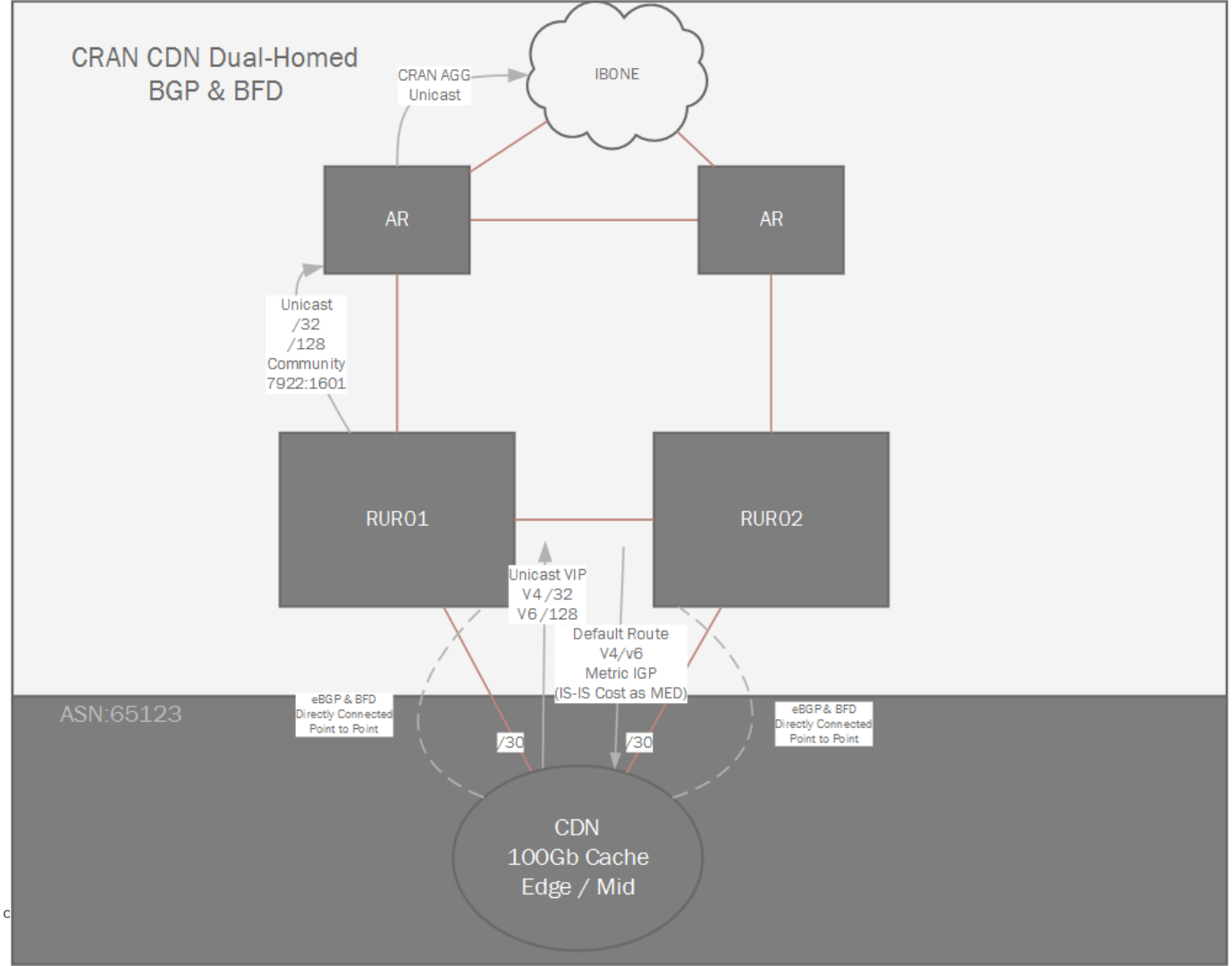
Several technologies were considered to archive router redundancy – BGP pairing, multi-router ling aggregation or MLAG, and simple bond based active-passive failover

Dual Home - Requirements

- Uplinks MUST terminate on different network nodes
- Solution MUST NOT require state, protocol or control-plane between network nodes
- Caches MUST be able to utilize either link at any time, during steady state
- Link failure MUST be transparent to clients
- Caches SHOULD not exceed half of the available bandwidth
- Caches MUST automatically return to a nominal state when a failed link is restored

Dual Home – Solution Overview

- BGP pairing between the cache and both routers on the two interfaces
- The cache tells the router that a V4/32 V6/128 VIP is available on a particular link
- If the link is up, the router forwards client requests to the cache on that link
- The router also forwards its default route to the cache along with a MED value to indicate a route cost



TPX Technology Product Xperience

Detailed Design and Deployment

- Caches have two 100 Gig interfaces connected to two different routers
- Both 100 Gig interfaces are configured with IP's (automatically during initial net install using router advertised ipv6 addresses to connect to the control plane) and are active.
- Both links are configured with LACP although each bundle only has a one member link. That is done for faster link down detection by the OS.
- Loopback VIP's are assigned to the caches(during net install).

Detailed Design and Deployment

- For CentOS 7, at boot time the default route is set to the first bonded interface that is configured.
- Once the system boots up, BIRD BGP daemon is setup, and it gets a default route from its BGP neighbors and propagates it into the Linux kernel routing table.
- The CDN traffic monitor connects to the VIP and establish the health of the cache.
- Then CDN traffic router directs clients to these addresses (instead of the interface IP's).

Detailed Design and Deployment

- To prevent the server from being unreachable if the chosen default interface is down, and BIRD/BGP is not providing a default route, each interface needs its own routing table.
- With two active interfaces, deciding which interface to send packets out of is not trivial as in a single active connection scenario, where a default gateways is statically defined, or provided by DHCP.
- On Centos 7 it is possible to create a default route with two different interfaces, aka a multi path or equal cost multi path route (ECMP). But if an interface is down, Centos 7 is not able to adjust a MP route and will send packets to a dead interface.
- Centos 7 must have a default route with one interface in it
- In this design BIRD and BGP specify that interface for the OS.
- Those interface specific routing tables specify to use an interface's gateway as the next hop if the packet is originating from that interface's ip address.
- The main routing table will contain the BGP provided routes for the VIP.

Detailed Design and Deployment

```
[root@sergey1-nightly ~]# ip rule
0:      from all lookup local
10:     from 96.96.18.190 lookup Tbond0
11:     from 96.96.18.194 lookup Tbond1
32766: from all lookup main
32767: from all lookup default
```

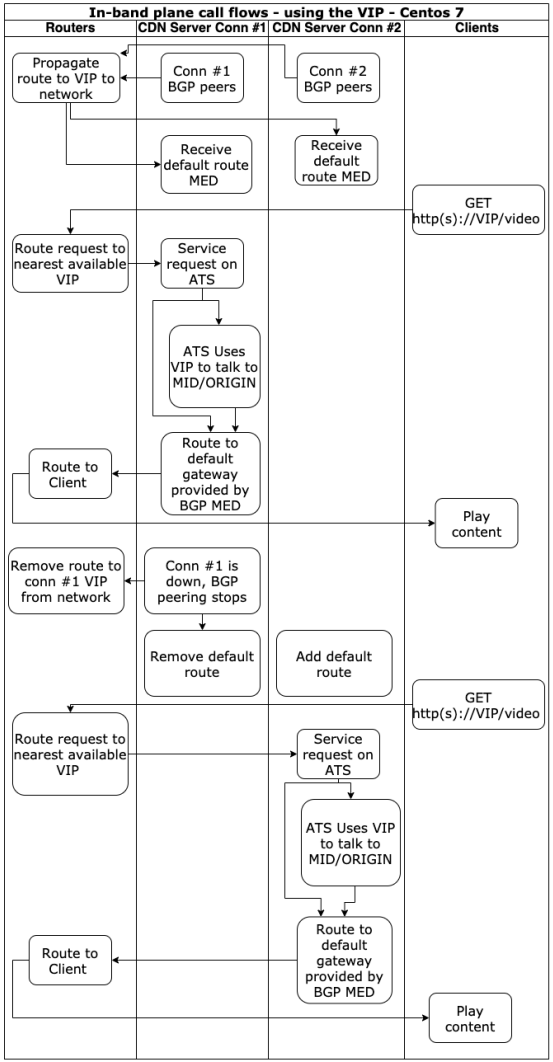
bond0: 96.96.18.190/30
bond1: 96.96.18.194/30

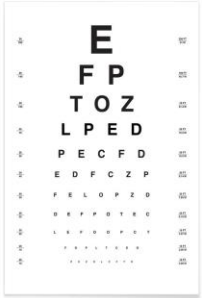
```
[root@sergey1-nightly ~]# ip route show table Tbond0
default via 96.96.18.189 dev bond0
96.96.18.188/30 dev bond0 scope link src 96.96.18.190
```

```
[sdremi200@sergey1-nightly ~]$ ip route show table Tbond1
default via 96.96.18.193 dev bond1
96.96.18.192/30 dev bond1 scope link src 96.96.18.194
```

Detailed Design and Deployment

[root@sergey1-nightly ~]# ip route show table main
default via 96.96.18.189 dev bond0 proto bird metric 1 ← from BIRD/BGP for VIP
default via 96.96.18.193 dev bond1 metric 2 ← **never actually used once BIRD/BGP IP**
96.96.18.188/30 dev bond0 proto kernel scope link src 96.96.18.190
96.96.18.192/30 dev bond1 proto kernel scope link src 96.96.18.194
unreachable 96.96.19.122 proto bird metric 1





Showing state diagram for a single server with two connections

1. Conn #1 used initially to receive and service requests
2. Conn #1 is down
3. Conn #2 takes over
4. Go back to 1 if conn #1 comes online

TPX Technology Product Xperience

Demo

- Toggle connections to a server and verify connectivity on VIP
- http://96.96.18.226/SampleDir/kelloggs.mp4

Future Work

- Multiple VIPs assigned to multiple servers allow the servers to steer traffic towards and away from each other reducing sticky client issues
- Traffic localization and load balancing using BGP advertised anycast

Thank You