

Building a small Data Centre

Cause we're not all Facebook, Google, Amazon, Microsoft...

Karl Brumund, Dyn
RIPE71

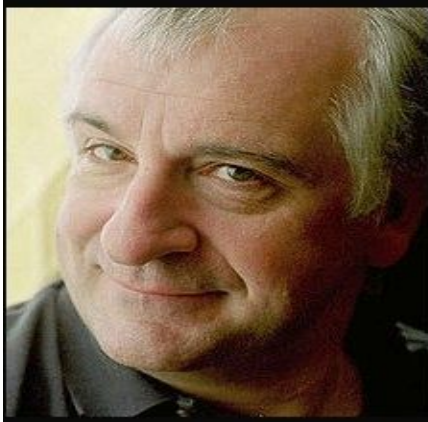
Dyn

- what we do
 - DNS, email, Internet Intelligence
- from where
 - 28 sites, 100s of probes, clouds
 - 4 core sites
 - building regional core sites in EU and AP
- what this talk is about
 - new core site network



First what not to do

it was a learning experience...



A learning experience is one of those things that say, You know that thing you just did? Don't do that.

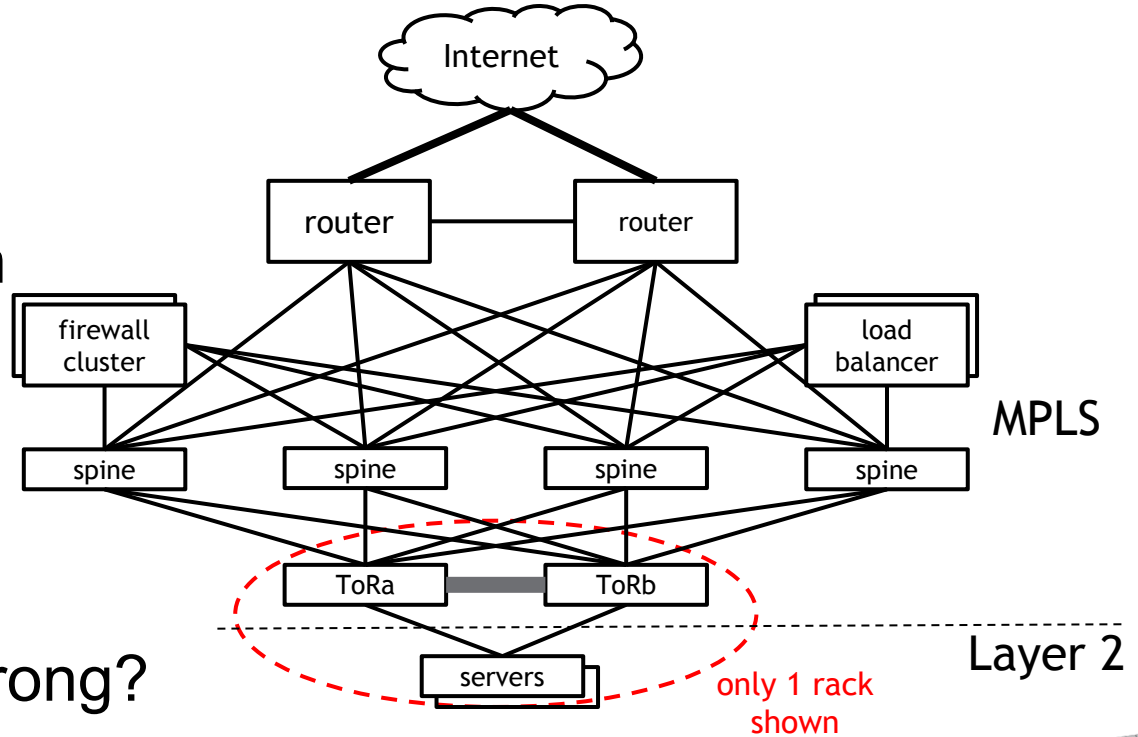
(Douglas Adams)



Design, version 1.0

Physical

- CLOS design
- redundancy
- lots of bandwidth
- looks good
- buy
- install
- configure
- what could go wrong?



Design, version 1.0

Logical

- MPLS is great for everything
- let's use MPLS VPNs
 - ToR switches are PEs
- 10G ToR switch with MPLS ✓
- 10G ToR switch with 6VPE ✗
- “IPv6 wasn't a requirement.”



reboot time

- let's start over
- this time lets engineer it



Define the Problem

- legacy DCs were good, but didn't scale
 - Bandwidth, Redundancy, Security
- legacy servers & apps = more brownfield than green
- **but** we're not building DCs with 1000s of servers
 - want it good, fast and cheap **enough**
 - need 20 racks now, 200 tomorrow



Get Requirements

- good
 - scalable and supportable by existing teams
 - standard protocols; not proprietary
- fast
- cheap
 - not too expensive
- fits us
 - can't move everything to VMs or overlay today
- just works
 - so I'm not paged at 3am



Things we had to figure out

1. Routing
 - actually make it work this time, including IPv6
2. Security
 - let's do better
3. Service Mobility
 - be able to move/upgrade instances easily



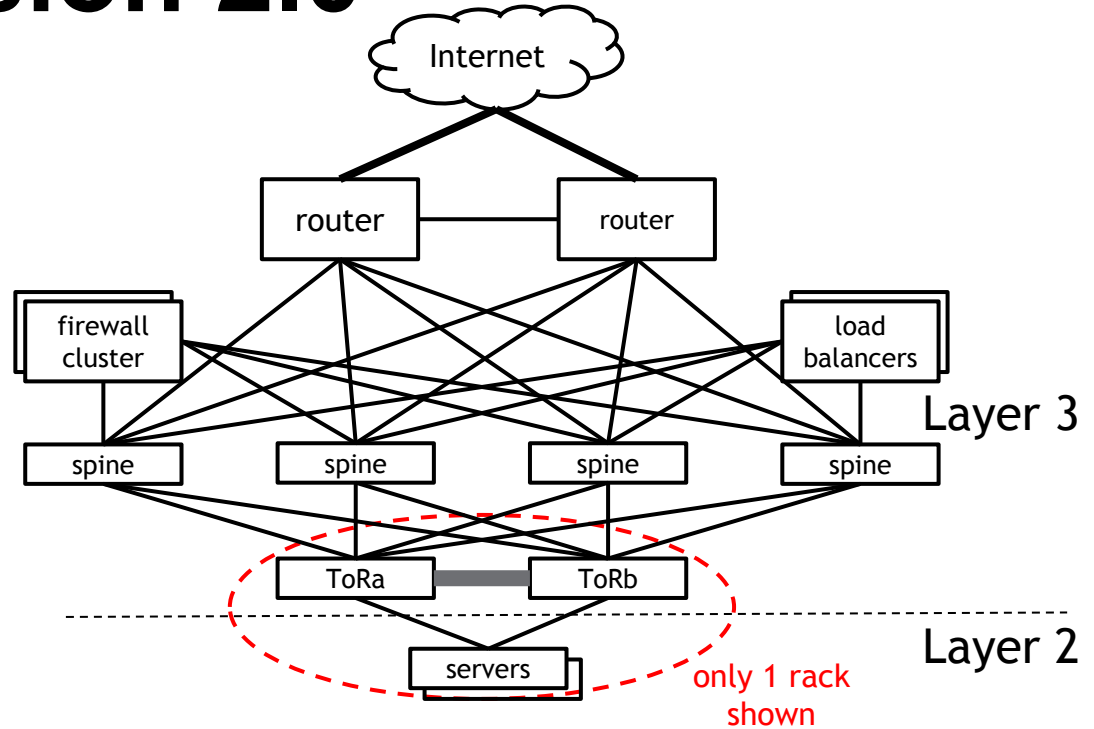
Design, version 2.0

Physical

see version 1.0

I can work with this

No money to rebuy



Design, version 2.0

Logical

- we still like layer 3, don't want layer 2
 - service mobility?
- not everything on the Internet please
 - need multiple routing tables
 - VRF-lite/virtual-routers can work
 - multiple IGP/BGP
 - RIB/FIB scaling
- we're still not ready for an overlay network



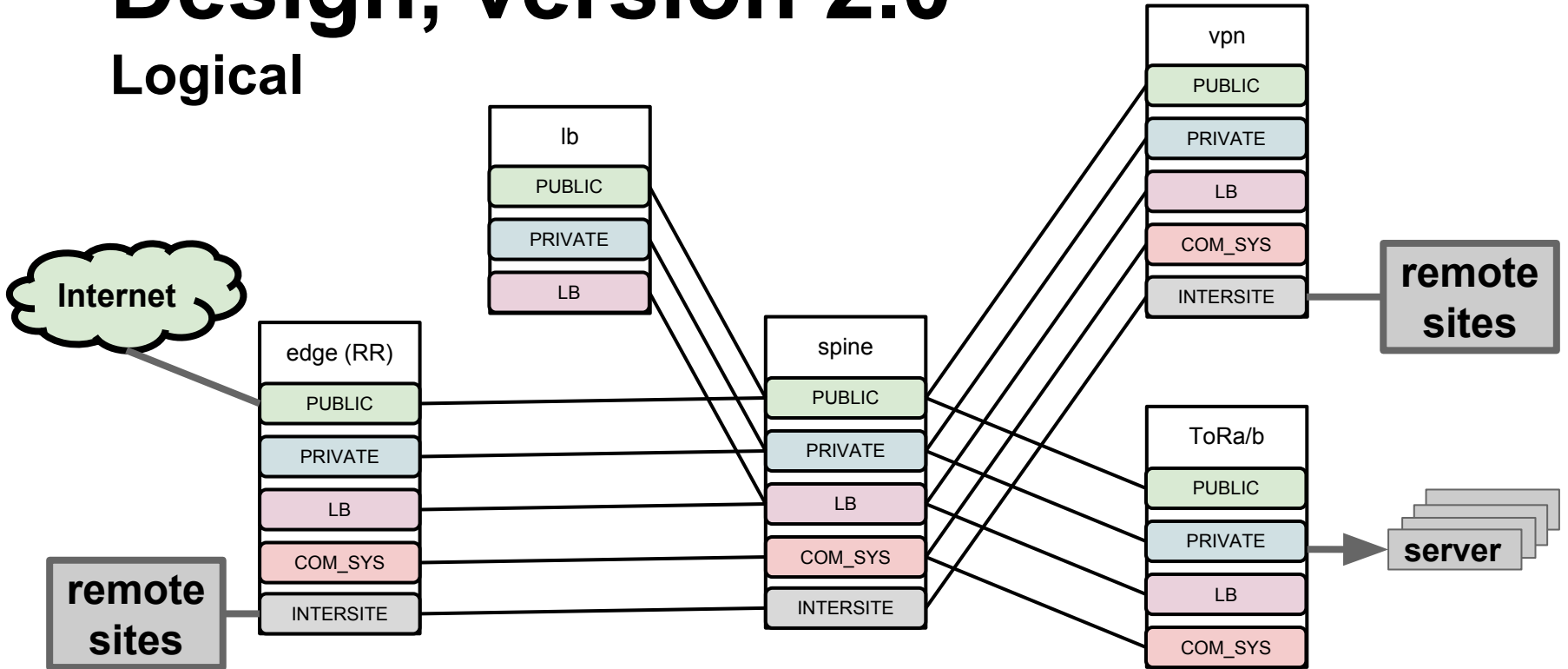
How many routing tables?

1. Internet accessible (PUBLIC)
2. not Internet accessible (PRIVATE)
3. load-balanced servers (LB)
4. between sites (INTERSITE)
5. test, isolated from Production (QA)
6. CI pipeline common systems (COM_SYS)



Design, version 2.0

Logical



eBGP or iBGP?

- iBGP (+IGP) works ok for us
 - can use RRs to scale
 - staff understand this model
- eBGP session count a concern
 - multiple routing tables
 - really cheap L3 spines (Design 1.0 reuse)
 - eBGP might work as well, just didn't try it
 - ref: NANOG55, Microsoft, Lapukhov.pdf



What IGP?

- OSPFv2/v3 or OSPFv3 or IS-IS
 - we picked OSPFv2/v3
 - any choice would have worked
- draft-ietf-v6ops-design-choices-08



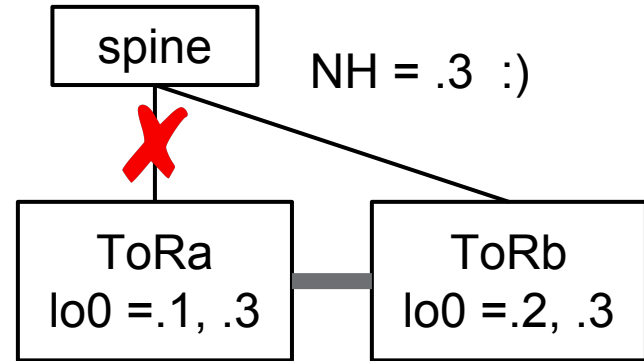
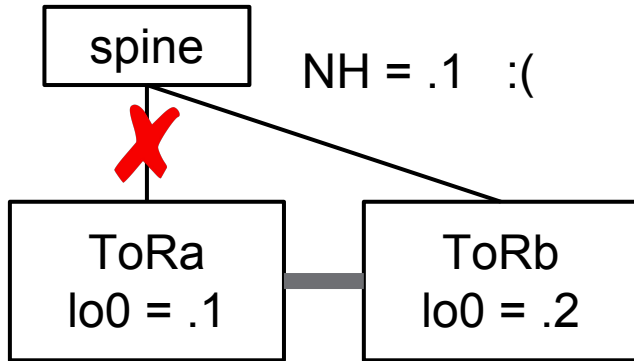
Route Exchange

- from one instance to another
- route-exchange can become confusing fast
- BGP communities make it manageable
- keep it as simple as possible
- mostly on spines for us



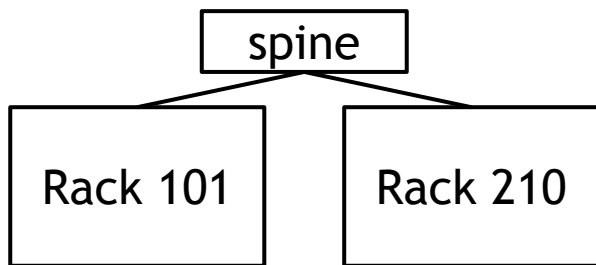
Routing Details

- pair of ToR switches = blackholing potential
 - RR can only send 1 route to spine, picks ToRa
 - breaks when spine - ToRa link is down
 - BGP next-hop = per-rack lo0 on both ToRa/b



Anycast ECMP

- ECMP for anycast IPs in multiple racks
 - spines only get one best route from RRs
 - would send all traffic to a single rack
 - we really only have a few anycast routes
 - put them into OSPF! :)
 - instances announce “ANYCAST” community



spine route table

- iBGP route from RR = Rack 101 only
- OSPF route = Rack 101, Rack 210

Security

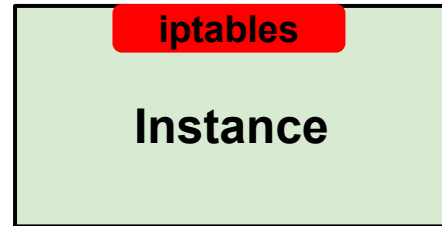
- legacy design had ACLs and firewalls
- network security is clearly a problem
- so get rid of the problem

No more security in the network



Security

- network moves packets, not filter them
- security directly on the instance (server or VM)
- service owner responsible for their own security
- blast radius limited to a single instance
- less network state



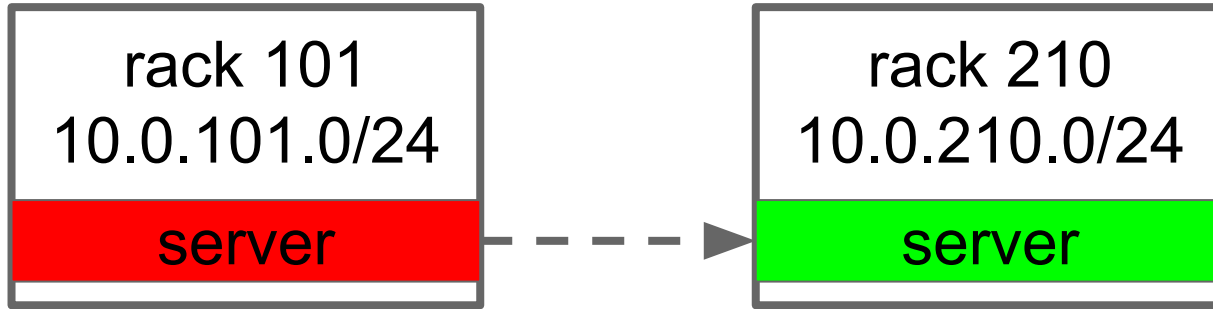
How we deploy security

- install base security when instance built
 - ssh and monitoring, rest blocked
- service owners add the rules they need
 - CI pipeline makes this easy
- automated audits and verification
- **needed to educate *and convince* service owners**
 - many meetings over many months



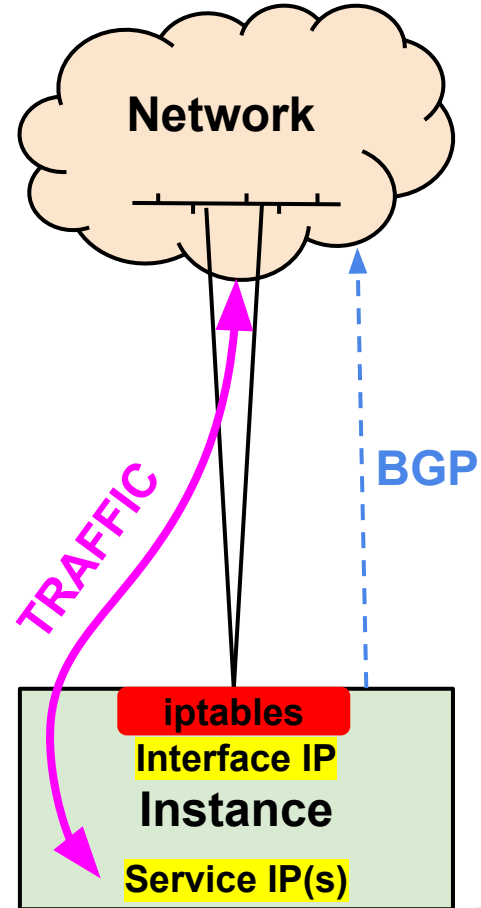
Service Mobility

- Layer 3 means per rack IP subnets
- moving an instance means renumbering interfaces
- what if the IP(s) of the service didn't change?
 - instances announce service IP(s)



Service IPs

- service IP(s) on dummy0
- exabgp announces service IP(s)
 - many applications work
 - some can't bind outbound
- seemed like a really good idea
- didn't go as smooth as hoped



Network Deployment

- ToR switches fully automated
 - trivial to add more as DC grows
 - any manual changes are overwritten
 - ref: NANOG63, Kipper, cvicente
- rest of network is semi-automated
 - partially controlled by Kipper
 - partially manual, but being automated



What We Learned - Design

- A design documented in advance is good.
- A design that can be implemented is better.
- Design it right, not just easy.
- Validate as much as you can before you deploy.
- Integrating legacy into new is hard.
 - Integrating legacy cruft is harder.
- Everything is YMMV.



What We Learned - Network

- Cheap L3 switches are great
 - beware limitations (RIB, FIB, TCAM, features)
- Multiple routing tables are a pain; a few is ok.
- Automation is your friend. Seriously. Do it!
- BGP communities make routing scalable and sane.
- There is no such thing as partially in production.
- Staff experience levels are really important.



What We Learned - Security

- Moving security to instances was the right decision.
- Commercial solutions to deploy and audit suck.
 - IPv6 support is lacking. Hello vendors?
 - We rolled our own because we had to.
- Many service owners don't know flows of their code.
 - never had to care before; network managed it
 - service owners now own their security



What We Learned - Users

- People don't like change.
- People really hate change if they have to do more.
- Need to be involved with dev squads to help them deploy properly into new network.
- Educating users on changes is ~~as much~~ work as ***a lot more*** building a network.



Summary

- Many different ways to build DCs and networks.
- This solution works for us. YMMV
- Our network moves bits to servers running apps delivering services. Our customers buy services.
- User, business, legacy >> network





Thank you

kbrumund@dyn.com

**For more information on
Dyn's services visit dyn.com**

**INTERNET
PERFORMANCE.
DELIVERED.**

dyn.com [@dyn](https://twitter.com/dyn)