Dense Anycast Deployment of DNS Authority Servers

RIPE 64
Ljubljana, April 2012
Dave Knight
Contents

• Background
• Expansion of L root
• Redesign
• Further Work
Background
In the beginning

• Authority service provided with a discrete server per entry in the NS RRSet

• Practical limit to the number of entries in the NS RRSet

• Small number of servers means high RTT for many users
In the beginning (2)

- An unreachable server means resolvers must retry
- Scaling and performance characteristics are not great
Anycast

- Anycast allows copies of a single server to exist at multiple points in the topology
- Some root servers have been anycast for well over a decade
- Typical model involves a stack of servers and network equipment hosted at IXP locations
Anycast enables...

• Avoid the limitation on number of servers
• Take the service closer to the user, lower RTT
• More query handling capacity
• Flexibility, it’s easy to add more servers, and easy to disable a broken server
• Keep attack traffic closer to the source
• Anycast since 2007

• Old model had 3 big nodes at IXP locations in Los Angeles, Miami, Prague

• Big nodes have one router and many servers
Expansion of L root
Why?

- Present in North America and Europe, wanted to take the service closer to underserved regions
- Wanted to significantly increase our query handling capacity
How?

• Change from a small number of big nodes to many small nodes

• Lots of nodes would mean lots of work, a redesign was required to significantly reduce opex
Where?

- No one good heuristic for locating nodes
  - Typical model says IXPs are good places to be, interested parties will peer
  - L is just one of 13 root servers, can add diversity by doing something different
  - We decided to go directly into eyeball networks, interested parties will provide a server
When?

- Field trial in 2011
  - Deployed 30 small nodes
- Working concurrently on new, fully automated platform
- Began rollout of the new platform in 2012
- Deployed at 60 locations in March
What?

- 21 physical servers at 20 locations.
- Quick win by deploying into virtual machines operated by PCH. Rolled out 146 VMs at 40 locations during March.
Before March 2012

68 servers at 37 locations
Deployed in March 2012

21 servers at 20 locations, 146 VMs at 40 locations
Now

89 physical servers and 146 VMs at 97 locations
Redesign
Old platform

- Servers run CentOS
- Some in-house scripts to automate things but installation and administration largely done by manual effort
New platform

• Servers run Ubuntu
  ▸ More flexible
  ▸ Prefer Debian package management

• Fully automated install

• Fully automated administration with Puppet

• Can treat the whole cloud as one system with one set of controls
Puppet

- Open Source IT Automation Software
  - Ensures that all systems have correct packages and configuration
  - Ensures easy system-wide application of configuration policy
  - Clients check in periodically and have policy enforced
Configuration

• Single YAML configuration file per server
  ‣ Used by Puppet and in-house Perl scripts

• Describes
  ‣ Location, Network info, Zone transfer auth
Example configuration

\texttt{yxu01.l.root-servers.org.yaml}

\texttt{location:}
  \texttt{town: London}
  \texttt{state: Ontario}
  \texttt{country: Canada}

\texttt{drac:}
  \texttt{addr4: 192.0.2.1/24}
  \texttt{gw4: 192.0.2.254}

\texttt{interfaces:}
  \texttt{eth0:}
    \texttt{addr4: 192.0.2.2/24}
    \texttt{gw4: 192.0.2.254}

\texttt{bgp:}
  \texttt{as: 64496}
  \texttt{desc: Example Limited}
  \texttt{addr4:}
    \texttt{192.0.2.254}
  \texttt{enable_advertisements: true}

\texttt{tsig:}
  \texttt{name: yxu01-20120101.}
  \texttt{algo: hmac-sha256}
  \texttt{data: tsigsecret=}

Wednesday, 18 April, 12
Install

• Host acquires and racks the server, does basic configuration of the DRAC

• ICANN DNS Ops run the installer
  ▸ Generates custom install ISO, attaches it to the DRAC as a virtual CD, tells the DRAC to boot

• Custom install CD
  ▸ Fully automated install, bootstraps Puppet, which starts working on first boot
Post install

- Some pre-flight checks to verify that the nameserver is working correctly, then we update the configuration and the node goes live

```yaml
bgp:
  as: 64496
  desc: Example Limited
  addr4:
    - 192.0.2.254
  enable_advertisements: false
  enable_advertisements: true
```
Puppet Modules

- Our Puppet code is organized into modules, one per service, for example, every server runs the NSD nameserver software, so we have a module for that which contains
  - A Puppet manifest which describes the policy regarding required packages, files, permissions, controls, dependencies, etc
  - A ruby template for the NSD config file
Monitoring

• Currently use Intermapper for alerts

• Puppet is well integrated with Nagios, we will migrate to use Nagios for alerts
  ‣ In the manifest for a service we can simply state that the service should be monitored by Nagios and it is
  ‣ A new node is therefore being monitored with Nagios as soon as the install completes
Monitoring (2)

- We use Observium to track resource usage, essentially everything we can graph with SNMP: Disk, Network, etc.

![Graphs showing resource usage over time](image-url)
Monitoring (3)

• We have custom scripts generating reports on the state of all nameservers we have any involvement with, the configuration of the L root portion of that is automatically generated from Puppet data.

• And of course we rely on RIPE NCC’s DNSMON
Measurement

- We monitor DNS traffic with DSC
  - Every server collects DSC stats locally and uploads them to a central server
  - We publish our DSC stats
  - http://dns.icann.org/lroot/stats/
Aggregate traffic at all nodes, by Query Type
Measurement (3)

- Will do ongoing capture of query packets
- Periodically upload those as part of a DITL
Further Work
Continue the migration

- Around 70 L root servers still running CentOS, they will be reinstalled with Ubuntu over the next few weeks
- Much of the back office services still run on CentOS: DSC, Observium, etc. They will be migrated to Ubuntu+Puppet
- Goal is to have a process where the initial deployment is the only manual step required in adding a server
Further Expansion

• We continue to deploy more nodes
• Interested parties can find more information at

http://dns.icann.org/
Questions?

dave.knight@icann.org