Network Infrastructure for Critical DNS

Steve Gibbard
http://www.stevegibbward.com
scg@stevegibbward.com
Introduction

- No research here; just a how to.
  - This was intended as a ccNSO TECH Day talk, not an OARC one.
- DNS network architecture
  - Whose network infrastructure to use
  - Where and how should name servers be connected?
- Focusing on network infrastructure
  - Lots of important stuff happens on the servers too, but that’s not my area.
DNS is critical infrastructure

- Without DNS, nothing else works.
- Authoritative DNS needs to be as reliable as the most reliable parts of the network.
- DNS is a hierarchy. For a domain name to work, its servers and those for all zones above it must be reachable.
Reliability is best close to authoritative servers

- There’s less to break between the server and the user.
- Response times are faster.
ccTLDs are location-based

- It’s somewhat obvious where they should be reliable.
  - They’re depended on by users in their countries.
  - They may be used in neighboring/trading partner countries.
  - People outside may not care much.
- Local root servers are needed too.
Network partitions

- In a network partition, it’s good if local communications keep working.
  - In satellite-connected regions, international connectivity breaks frequently.
  - Outages are rarer in fiber-connected regions, but last longer.
  - Local phone calls work without international connectivity. Local Internet should too.
Notable incidents

- **Sri Lanka (2004)**
  - International fiber was cut in Colombo harbor.
  - Press reports described an outage of “Internet and long distance phone service.”
  - ccTLD hosted locally, but no root server (now fixed).

- **Burma/Myanmar (2007)**
  - International connectivity was cut off by the government.
  - Local connectivity kept working.
  - .MM worked inside but not outside.
Root Server Locations

Source: http://www.root-servers.org
Building DNS infrastructure

- Goals
- How to build it
- Topology
- Redundancy
Goals

- Who are you trying to serve?
  - Local users?
  - Users in other local areas?
  - The rest of the Internet?

- Your region’s topology:
  - Is everything well-connected, or a bunch of “islands?”
  - Servers in central location, or lots of places?
Whose infrastructure?

- Your own?
- Somebody else's?
  - Free global anycast services for ccTLDs provided by ISC, PCH, others
  - Several commercial anycast operators (now including Nominum…)
  - Lots of free unicast options
  - Mixing these for an easy large-scale global-build
- Mixture?
  - Your own servers in areas that matter most to you
  - Somebody else's global footprint
Where to put the servers

- In country options:
  - At a central location -- an exchange point
  - One in each ISP
  - At a common uplink location (like Miami for Latin America)

- In the rest of the world:
  - At major Internet hubs
  - At the other end of your ISPs’ international links
**Unicast/anycast:**

- This is mostly an issue of scale
- For small numbers of servers, unicast works well
- Having several service IP addresses *in different places* is good for reliability
- Anycast is required for larger numbers of servers
Unicast configuration

- Fairly trivial, from a network perspective
  - Plug your host or hosts into a network connection, and it will work
- Do make sure you have enough capacity
- Make sure you have network and power diversity between servers
- Use colocation providers close to your users
Anycast topology – keeping traffic local

- Backbone engineers are often good at keeping local traffic local.
- Anycast DNS operators aren’t so good at this.
  - Anycast looks like a backbone.
  - But, plugging servers into random networks is done in pursuit of network diversity.
  - Networks send traffic to customers first, regardless of geography.
There are four local J-Root servers in the Bay Area (www.root-servers.org)

Queries from 4Bay Area hosts are responded to by:

- jluepe1-elsel1 – Seoul, via Level(3)
- jluepe2-elbom1 – Mumbai, via GBLX
- jluepe1-eltpe1 – Taipei, via Asia Netcom peering
- jns4-sea1 – ICANN meeting network / NTT
Anycast can keep traffic local

- Consistent transit should be gotten from global ISPs
- Peering only locations work in areas where global transit isn’t available, but be careful
- No transit from non-global providers:
  - Insist on being treated like a peer
Routing Topology
Queries with consistent transit

Palo Alto

Ashburn

London

Hong Kong
Routing protocols - External

- Upstream peering via BGP
  - Single Global AS helps keep things consistent
  - Don’t propagate anycast routes between sites
  - Be careful about BGP attributes (e.g. MEDs), especially in a multi-vendor environment.
Routing protocols - internal

- Internal: BGP or your favorite IGP.
  - Internal routing scope should be limited and
  - Routes can be originated on servers for dynamic withdrawal. Use Quagga or BIRD
  - OSPF has wider support; BGP has better filtering
  - Dedicated load balancers are an option
  - If mixing, be careful about routing attributes
Redundancy

- More servers are better than fewer, if they’re manageable.
- There’s no contradiction between using your own servers and outsourcing.
- Monitoring:
  - Check zone serial numbers on all servers frequently.
  - If using anycast, monitor individual unicast management addresses.
  - Check response times from multiple locations.
Anycast Requirements

- Servers running Quagga (or BIRD)
- BGP capable routers
- IP transit from consistent providers in all sites
- Colocation space in all sites
- A /24 of address space per site, if using multiple transit providers
What should it look like when done?

| np. | 86400 IN NS   | ns-ext.isc.org. |
| np. | 86400 IN NS   | ns-ext.vix.com. |
| np. | 86400 IN NS   | sec1.apnic.net. |
| np. | 86400 IN NS   | shikhar.mos.com.np. |
| np. | 86400 IN NS   | yarrina.connect.com.au. |
| np. | 86400 IN NS   | np-ns.npix.net.np. |
| np. | 86400 IN NS   | np-ns.ripe.net. |
| np. | 86400 IN NS   | np-ns.anycast.pch.net. |
| np. | 86400 IN NS   | sec3.apnic.net. |
Further reading
Very old papers

- DNS infrastructure distribution

- Observations on anycast topology and performance.
Thanks!

Steve Gibbard
http://www.stevegibbard.com
scg@stevegibbard.com