

IPv6, DNS och rootserverar  
eller  
”tänkte inte på det...”

Johan Ihrén

Autonomica

johani@autonomica.se

## v6 induced name space fragmentation

- With v6 roots it becomes possible to have divergence between the name spaces visible in each "universe"
- That's bad
- Possible fixes encompass the entire spectrum from only recommendations ("don't do this"), via active validation (of delegations), to kludgy bridging services. Choose your poison. This **will** hurt.

# Namespace fragmentation: consequences of deploying v6 roots

Johan Ihrén  
Autonomica

Namespace fragmentation:  
what is this "namespace", anyway?

Johan Ihrén  
Autonomica

## Namespace fragmentation caused by new transports

- Let's assume the following:
  - v6 roots are deployed (i.e. nameservers for the DNS root available over v6 transport)
  - presumably the contents of the root zone is slightly modified with address records for the v6 roots, but that is not the issue here
    - the issue is **DNS transport**, not **DNS data**

## ... new transports, cont'd

- DNS "delegations" of subdomains go to nameservers referenced by name (not address)
- Possibly, when looking up the address corresponding to this name, only a v6 address will be found
  - or only a v4 address, same problem
- This will cause problems for resolvers that can only follow referrals within one "universe" (i.e. on one transport)

## ... new transports, cont'd

- In a perfect world this will not be a problem
  - because every single DNS "zone" will be served by servers available over both transports
- The world is **not even close** to being perfect
- Therefore "single stack" clients will see only part of the namespace
  - the "real" Internet namespace would be the union of the v4 and v6 namespaces

## ... new transports, cont'd

- Is this a problem?
- Are there any disadvantages to solving the problem?
- What about solving only half the problem (v6 client to v4 server)?
- Is it really our problem?

## Is this a problem?

- Obviously users (both "producers" and "consumers") would appreciate the availability of the same namespace regardless of transport.
  - They would also like a free lunch.
- There is a difference between reachability of a service and ability to lookup the service.
  - Being able to lookup the service enables the client to make an informed decision.

## There is **no** perfect solution

- Bogging down in a quest for perfection will not help us.
- A compromise (in the sense of "not perfect") is needed.
- The compromise **must** end up with the zones deployed over v6 transport.

## Guiding principle: coherency

- Coherence require that the answer to a query **must** be the same regardless of how the answer was obtained.
  - The Internet has a long tradition of being able to cope with outages, failures and brokenness.
  - It is much harder to try to cope with lying.

## Guiding principle: autonomy

- We will have to live with whatever solution we decide on for several decades.
  - If during that time the zone owners don't realize (or agree) to deploy their zones in v6 we will fail.
  - Ultimately **they** are responsible for deploying **their** zones. Our job is to create incentive for them to do the right thing.

## Deployment vs lookup

- Lookups from v6 works today, since the entire tree is available in v4 and all resolvers knows v4.
- Lookups would continue to work tomorrow if only the resolvers stayed dual-stack.
  - I'm only talking about **full service** resolvers, not stub resolvers, nor forwarding caches.
  - This will not help v4 if zones migrate to v6 only.

## Balkanization vs NAT/bridging

- "Balkanization" (i.e. uneven zone availability) is unknown territory.
  - It may create new failure modes for apps and services
  - It may drive deployment of zones over v6 transport
  - It will cause confusion
- Bridging is a known evil. It has a number of drawbacks and will only work to a certain extent. But to that extent it does keep the namespace together.

## Disadvantages to a "working" bridging solution

- Without bridging, deployment of zones over v6 becomes necessary
  - or v6 will fail utterly (still possible)
- With bridging, deployment may never happen
  - bridging will have to stay forever (working progressively worse as the v6 part grows but impossible to get out of)

## Where to solve problems.

- Close to the resolver (i.e. client)?
- Close to the server?
- On the v4/v6 border (as a network wide "service")?
- The cost of maintaining bridging should somehow (but how?) end up at the doorstep of the zone owners (to create incentive to deploy).

## One way solutions?

- Bridging from a v6 client to a v4 server is easier than the opposite way around.
- This could be taken as an argument that by keeping the entire namespace available over v4 transport everything is fine.
  - That is a false argument, since at some (remote) point the assumption of v4 prevalence will be false.
  - An assumption that everything is always available over v4 transport will eventually be false.

## Whose problem is this, really?

- "DNS is a distributed, autonomous, coherent, reliable database."
- The autonomy follows from the process of delegation. The parent gives up **responsibility** for the contents of the subdomain at the time of delegation.
  - one of the responsibilities is selection of servers.

## Questions to the audience.

- Will we ever get zones deployed if we create a "DNS bridge"?
- Would it be possible to keep lookups "working" with a combined recommendation of
  - every zone must have v4 servers (and eventually v6 also)
  - full service resolvers should become dual-stack
- What amount of "breakage" is acceptable?
  - let's not pretend there is no breakage on the net, although this would be a new type