

IPv6-only iterative resolver utilising NAT64

draft-momoka-v6ops-ipv6-only-resolver

Momoka YAMAMOTO (momoka.my6@gmail.com)
Yasunobu TOYOTA (yas-nyan@sfc.wide.ad.jp)

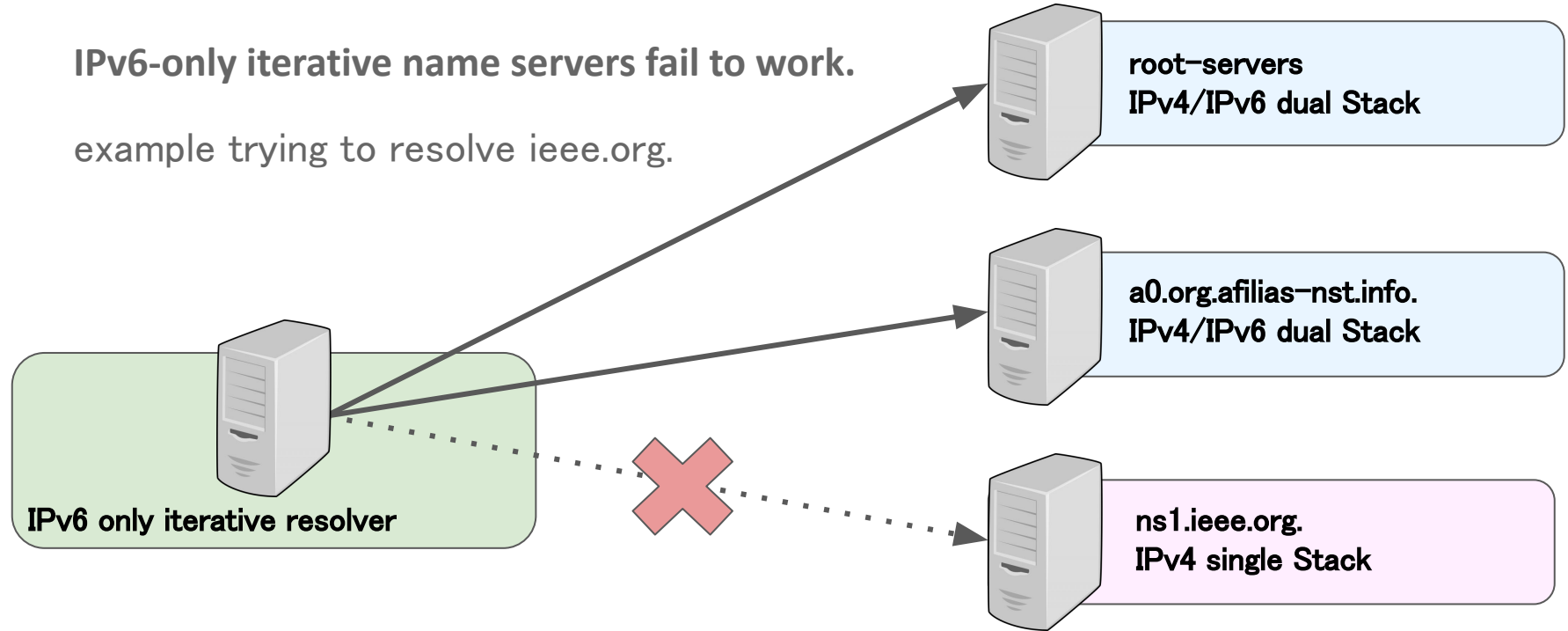
Motivation

- IPv6-only iterative resolvers are currently unusable 😞😞😞
- Because there are IPv4-only authoritative servers
- All other applications can use DNS64, but a resolver can't because it's the resolver.
- IPv6-only: Has only an IPv6 address

IPv6-only resolver cannot resolve all zones

IPv6-only iterative name servers fail to work.

example trying to resolve `ieee.org`.



Reference RFC3901 BCP91

4. DNS IPv6 Transport recommended Guidelines

In order to preserve namespace continuity, the following administrative policies are recommended:

- **every recursive name server SHOULD be either IPv4-only or dual stack,**

This rules out IPv6-only recursive servers. However, one might design configurations where a chain of IPv6-only name server forward queries to a set of dual stack recursive name server actually performing those recursive queries.

- **every DNS zone SHOULD be served by at least one IPv4-reachable authoritative name server.**

This rules out DNS zones served only by IPv6-only authoritative name servers.

We can fulfill the spirit of BCP91 with an IPv6-only resolver by doing the IPv4 to IPv6 translation inside the iterative resolver and making it “dual-stack”.

How it works

- If the resolver only finds an A record for an authoritative server,
- The resolver should perform address synthesis to the and make it IPv6.
 - This is done by applying the Pref64:: n to the IPv4 address to construct IPv4-converted IPv6 addresses as defined in RFC6052
- How to obtain the Pref64:: n of the NAT64
 - The iterative resolver can obtain the Pref64:: n used by the NAT64 of the network by either static configuration or by using discovery mechanisms. (The Port Control Protocol [RFC7225] or Router Advertisements [RFC8781]. Using the mechanisms described in [RFC7050] or [draft-hunek-v6ops-nat64-srv] may not function because these need a resolver to work.)

Considerations

Why not CLAT?

- Most applications can reach IPv4 Internet via DNS64/NAT64.
- An iterative server is the only application that cannot utilize DNS64.
- Using CLAT hinders the use of native IPv6.
- CLAT forces keeping IPv4 support.

Implementations

Not yet merged, but popular DNS software are implementing these features.

BIND

https://gitlab.isc.org/isc-projects/bind9/-/merge_requests/6334/commits

Unbound

<https://github.com/NLnetLabs/unbound/pull/722>