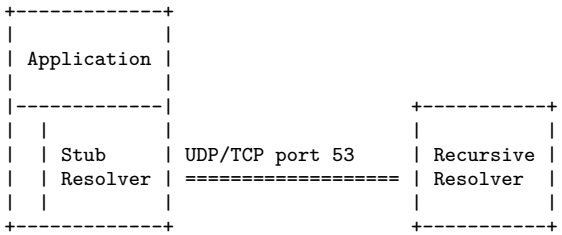
Control Options For DNS Client Proxies

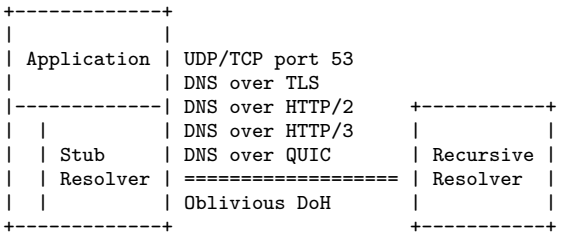
draft-homburg-add-codcp-00

Philip Homburg <philip@nlnetlabs.nl>
with Willem Toorop <willem@nlnetlabs.nl>

Traditional Stub Resolver



Modern Stub Resolver



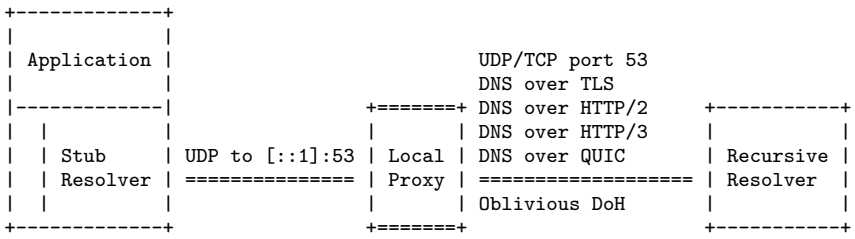
Background

- NLnet Labs was working on a connectbyname implementation. At the moment just a prototype.
- Asynchronous, Happy Eyeballs, DANE
- On top of getdns
- `https://nlnetlabs.nl/projects/connectbyname/about/`

Problems

- Many applications use a stub resvoler. How many libraries will implement all transports?
- DNS directly over UDP has almost no state. DoT, DoH, and DoQ require connection set up.
- Load on recursive resolver.
- Bad for shortlived applications, for example ping.

Solution



A proxy on the same system as the application.
Examples of existing proxies: unbound, stubby, dnsmasq, systemd-resolved.

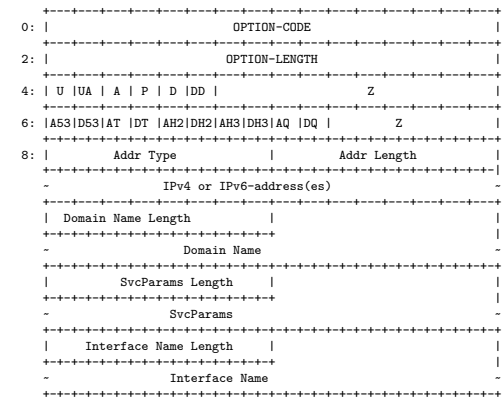
Problem

Applications have no control.

- How to specify that encryption is required?
- How to specify a DoH connection to a specific public DNS resolver?
- Feedback to the user if DNS resolution fails.
- Diagnostic tools

Proxy Control Option

New EDNS(0) Option



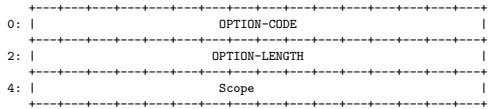
Notes

- Stateless, send proxy control options in every request
- Maximize control of the application
- Potential for caching
- Potential for local policies in the proxy
- No DNSSEC validation requirements for the proxy
- Proof of concept:

  `https://github.com/getdnsapi/getdns/tree/philip-proxy-config`

Proxy Scope Option

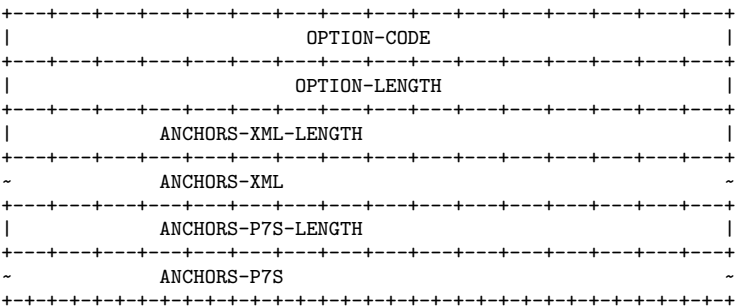
What if a legacy proxy forwards proxy control options unchanged?



Value	Scope
0	Undefined
1	Host local
2	Link local
3	Site local
4	Global

Support For Local Validation

The local proxy is not trusted, but can cache a signed trust anchor.



Feedback, Questions?

Contact me at <philip@nlnetlabs.nl>