DELEG

born at IETF Hackathon 118
Hackathon 118

• ~ 20 people
  - DNS providers
  - TLD operators
  - implementors
  - historians

• RESINFO, NOTIFY to trigger CDS/CDNSKEY update

• Discussion
  - How to make the DNS more… better …?
  - Joke … DNS = Does Not Scale
What should we fix/improve?

- i'd not keep the records separated like NS+DS. One record should contain the NS name, address, capabilities, TLS… And if there are multiple those, each NS can have different capabilities and properties.
- Nameserver-specific DS (allows easy multi-signer: each NS can have independent DNSKEYs)
- Transport should be QUIC. Clients get session tickets and use them when needed. Do53 is used for priming/discovery towards an auto discovery address.
- Auto configure on local networks (multiple responders possible (routers/tunnel-providers/IoT gateways) - verification of functionality )
- Signed ADoT bootstrap on the parent side + EPP extensions to populate it
- Delegation has info to verify legitimacy of servers too, e.g. TLS certificate hashes or the certs.
- Zone Cuts. We need to define what is authoritative at the child and what is authoritative at the parent. NS was a failure we can not let happen again. Not sure on glues, but having an SVCB type record would help here
- If you have a better way to do name resolution, what transition plan can work?
- Get rid of sections. Just a collection of resource records with some ordering
- DNS text & wire formats might be changed to something used elsewhere and more structured, e.g. JSON+BSON(?) - but it is OK to keep the current
- Some way to ask and answer multiple things. Each question has a specific associated answer. E.g. “this is the A record for www.example”, “there is no AAAA record for www.example”, “I am not authoritative for www.random-thing”, “I’m not answering your 10th question as that’s too many”. These answers would also be encoded more explicitly than in the DNS. E.g. in the case of “the name you asked about has some data, but no data of the type you asked about” would be an explicit “NXRRSET” result, not inferred from “no error code” + “no answer” + “negative caching info in authority section”.
- Proper delegation objects: The current, un-signed NS + glue + DS is a mess
- Maybe structured as a DS2/NS2 record
- Secure Delegation (especially to secure transport servers) needs to come from the parents during delegation. Child information on this (NS/DS) is irrelevant.
- SVCB-DNS and maybe TLSA bootstrap info for ADoT
- Post-quantum DNSSEC
- we need a delegation record that handles delegated names, addresses, child NS capabilities, TLS certs, eventually DS
- possibly (a clone of) SVCB?
Hackaton self-imposed limits

• To an outsider, the DNS won’t change
• Keep name space as it is - the data model
• Keep management boundaries - zones
• Keep stub resolver model
  • (name, [class,] type) -> value

• **MUST** keep interoperability with the current DNS
  • ... and allow incremental evolution
Can we?

• There have been many attempts to overhaul the 1980’s DNS protocol, none of which went anywhere

• Events on the horizon – post-quantum? maybe?
  • What if we exceed message size limitations?

• Maybe we have the critical mass now?
Underlying problem

• Permitting new stuff and old stuff to coexist
  – MUST NOT break old clients
  – SHOULD allow (radical?) evolution

• NS RR usage for zone delegation
  • Not extensible
  • Half-secured ...

• Proposed new approach: **DELEG**
Proceed with caution

• Work-in-progress
• 3 days old ...
• Confusion and disagreements to be expected
  - Naming protocol parameters is hard :-(
DELEG – a new delegation example

• **In-bailwick** – principle, not a spec

```
$origin example.

a NS ns1.a.example.
a DS 01234 99 2 ABCDABCDABCD...
a DELEG 1 ns1.a.example. (ipv4hint=192.0.20.0 ipv6hint=2001:db8:1234::38 transport=dot ; just an example otherinfo=needed for handoff)
a RRSIG DELEG ...
```

• DELEG is authoritative on the parent side, signed like DS

6 Nov 2023
DELEG – a new delegation example

- **Out-of-bailwick** – principle, not a spec
  
  $origin example.
  
  sub NS ns1.a.example.
  sub DS 1234 99 2 ABCDABCDABCD...
  sub DELEG 0 config1.hoster.example.
  sub DELEG 0 cfg55.anotherhoster.example.

  $origin hoster.example.
  
  config1 SVCB 1 ns1.hoster.example. \(transport=do53\)
  config1 SVCB 1 ns2.hoster.example. \(transport=dot\)
  config1 SVCB 1 ns2.hoster.example. ( transport=dot \(wireversion=2\) )

multi-provider
DELEG and SVCB usage

• DELEG – much like the SVCB (Service Binding)
  − Creates zone cut (like NS)
  − Only at the parent, signed! (like DS)
  − Deviates from SVCB where needed – allows multiple aliases etc.
  − can point to set of SVCB (Service Binding) RRs
    • indirection to recognize role of DNS operator
• SVCB config can be shared by multiple delegations
  • Extra subtree – not necessarily tied to registry/registrar authorization
  • DNS provider can change it's own stuff
What is this good for?

• A first step – enables new things with “value add”

• Having a new way to do delegations is an enabler
  • A new publication protocol can be spoken (on another port, for example)
  • Anything is fair game on a new port
  • Everything from wire format up the stack *could* (but does not have to!) change

• Value add
  • Exposes DNS operator role
  • Meta-data about zone boundaries? Maybe??
Enabler

- Address problems with the current publication protocol (what we do over port 53)
  - UDP; Fixed-width header fields; constrained message structure; suboptimal compression; cruft (class, duplicate TTL and owner names, ...)

- Traffic Engineering in DNS
  - Can do better than EDNS Client Subnet (and the privacy headache)

- Provisioning side channel
  - Remove burden of shoving signals into DNS band, provide feedback
Value

• Recognizing operator’s role
  • Allows for DNS operator to represent DNS zone administrator (registrant in some language) on technical matters
  • Permits security association to be built between zone operator and delegating parent administration (registry in some cases) which enables use of a dedicated provisioning channel

• Recognizing boundaries
  • Help in addressing the ol’DBOUND problem?! Perhaps.
Will DELEG emerge once documented?

• Probably not
• Move from NS to DELEG is a change
• Operational deployment will require a reason
  • The enabling element is important
  • We also need to work on noticeable improvements
• DELEG is a foundational element
DELEG work-in-progress

• Discussion raging on a DNS-OARC’s Mattermost chat server
  • See: https://dns-oarc.net/oarc/services/chat
  • Channel name: DELEG-design
  • Draft work-in-progress: https://github.com/fl1ger/deleg/

• There is a significant path ahead

• It’s early
  • This work needs attention from non-IETF!
    • operators of all kinds
    • RRR involvement & regext