DELEG

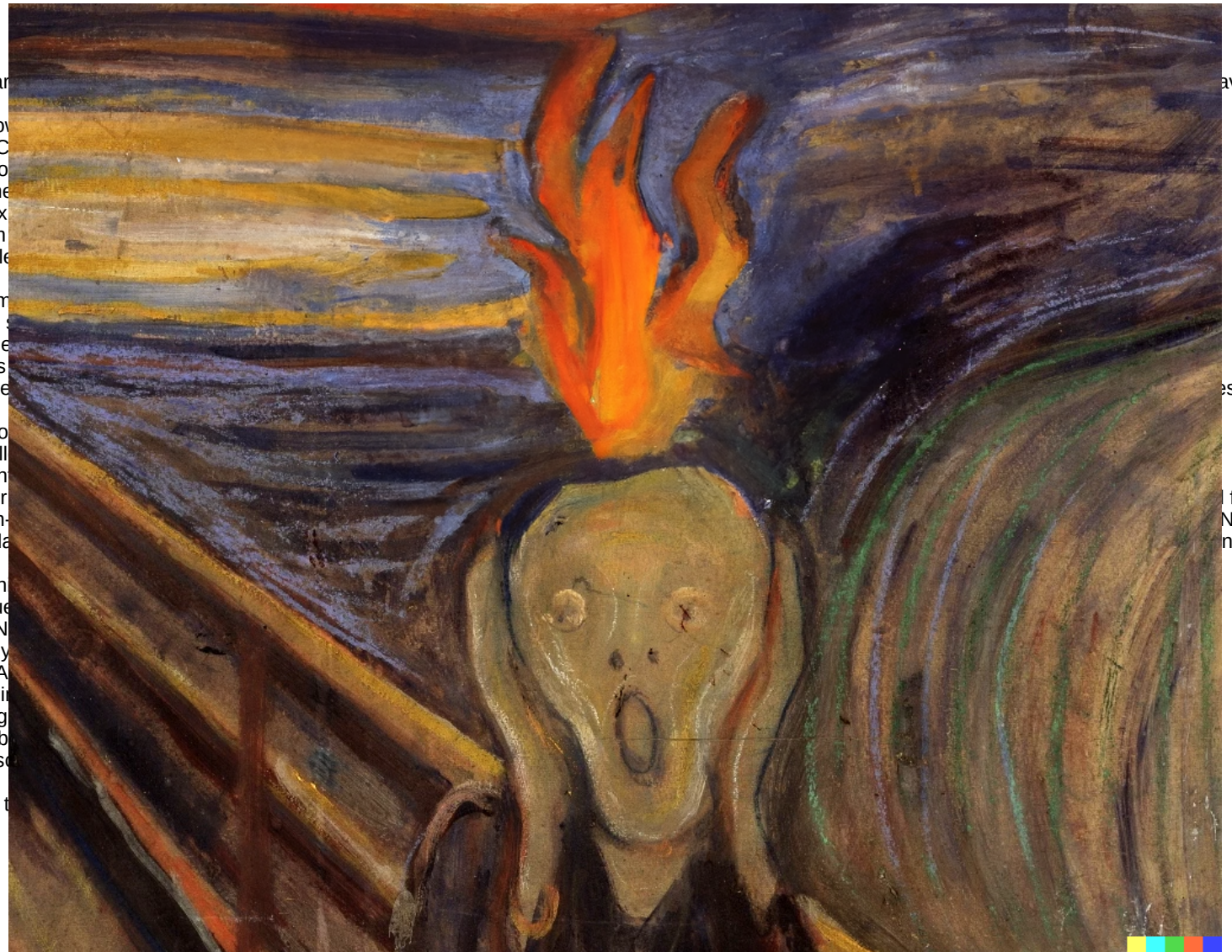
born at IETF Hackathon 118

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 separ…ave different capabilities and properties.
- Nameserver-specific DS (allo…
- Transport should be QUIC. C…
- Auto configure on local netwo…
- Signed ADoT bootstrap on the…
- main concern - protocol inflex…
- Much improved errors, and in …
- Delegation has info to verify le…
- simpler to operate over time.
- No name compression. If com…
- Using QUIC allows message s…
- and is signed (or otherwise se…
- I don't like Do53. Priming has…
- Zone Cuts. We need to define…es, but having an SVCB type record would help here
- If you have a better way to do…
- Get rid of sections. Just a coll…
- DNS text & wire formats migh…
- Some way to ask and answer …l for www.example", "I am not authoritative for www.random-…NS. E.g. in the case of "the name you asked about has some da…negative caching info in authority section".
- Proper delegation objects: Th…
- currently, un-signed NS + glue…
- Maybe structured as a DS2/N…
- Secure Delegation (especially…
- SVCB-DNS and maybe TLSA…
- the *key thoughts* of how to i…
- the new delegation record sig…
- Local network transport can b…
- CBOR / or otherwise self-desc…
- Post-quantum DNSSEC
- we need a delegation record t…
- 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 ...
```

all delegation info in one place

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

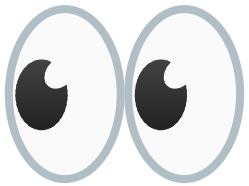
DELEG – a new delegation example

- **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.
```

multi-provider

```
$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 )
```



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 **necessairly** 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