Comparison

draft-wesplaap-deleg-01

vs.

draft-homburg-deleg-incremental-deleg-00

pspacek@isc.org

philip@nlnetlabs.nl

2024-11-04

Two† approaches

- New **parent-side-only** type
  - draft-wesplaap-deleg-01
  - new **DELEG type** … like DS

- Special **_deleg subtree** / special name
  - draft-homburg-deleg-incremental-deleg-00
  - SVCB RR type under _deleg, in the parent

In-domain delegation: legacy method

```
example.          IN  NS ns1.example.
ns1.example.      IN   A  192.0.2.1
```

In-domain delegation: legacy & parent type

NS

```
example.          IN  NS ns1.example.
ns1.example.      IN   A  192.0.2.1
```

type

```
example.          IN  DELEG 1 ns1.example. (
                                ipv4hint=192.0.2.1 )
```

In-domain delegation: 3 methods

```
example.          IN  NS ns1.example.
ns1.example.      IN   A  192.0.2.1
```

NS

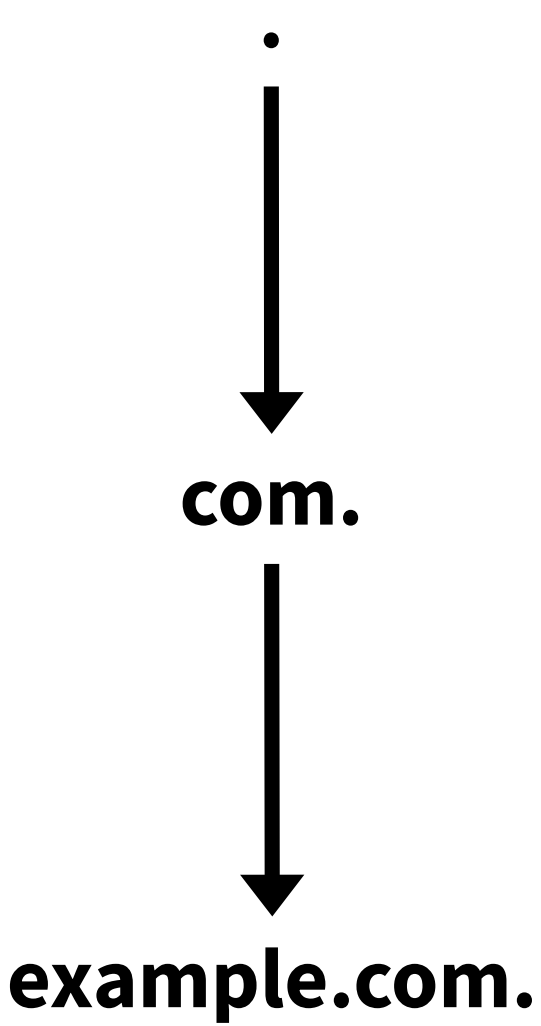
```
example.          IN  DELEG 1 ns1.example. (
                              ipv4hint=192.0.2.1 )
```

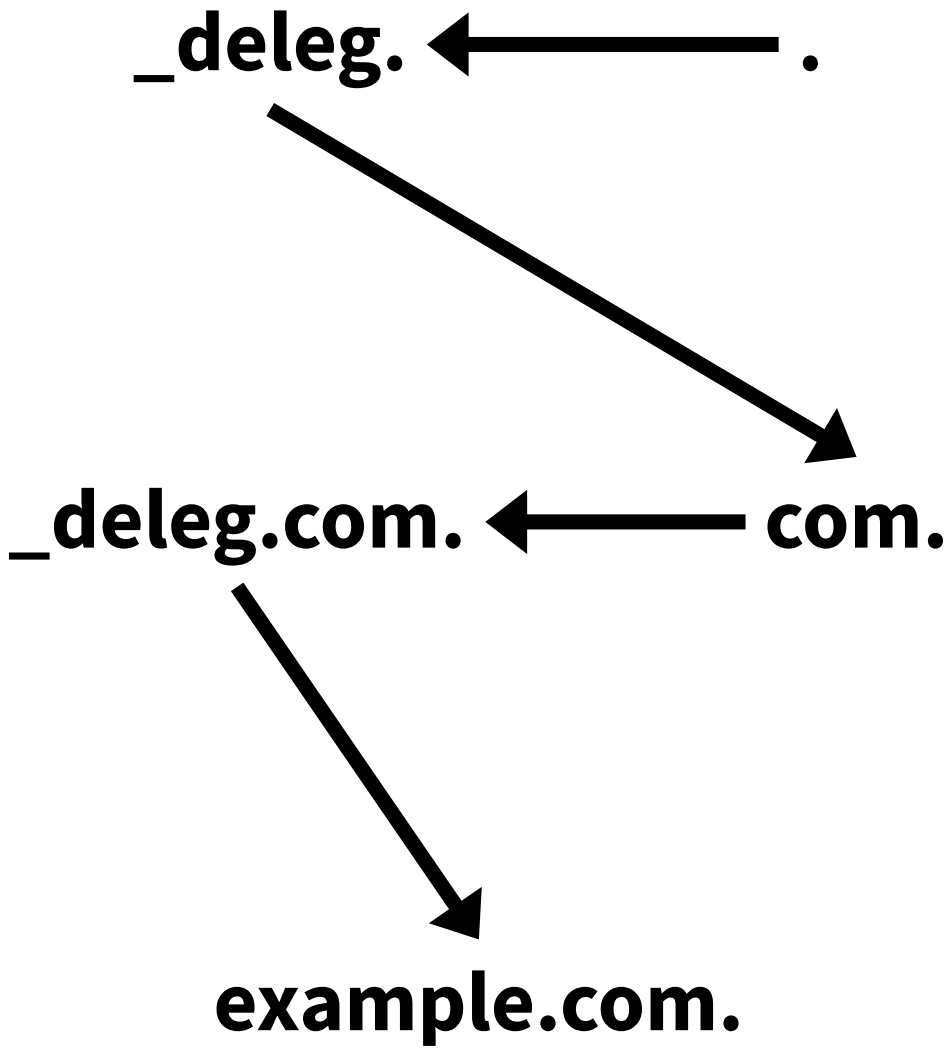
type

```
example._deleg.   IN  SVCB  1 ns1.example. (
                              ipv4hint=192.0.2.1 )
```

name

Parent-side RR vs. _deleg subtree





DNSSEC signer

- New special case for DELEG RR type
  - Exactly same as DS RR type
  - 12 lines of code in BIND

type

- Unaffected

name

DNSSEC validator

- New special case for DELEG RR type
  - Exactly same as DS RR type
  - 3 lines of code in BIND
    - Plus some more for downgrade protection / compatibility opt-out

type

- Unaffected

name

DNSSEC specials

- Downgrade protection for DELEG type
- New DNSKEY flag
  - "DELEG MUST be present or non-existence proven"
  - Much like NSEC3 opt-out, for DELEG

type

- None required

name

Delegation with alias ("include")

NS

```
; Not possible
```

type

```
example.          IN  DELEG 0 operator1.test.
example.          IN  DELEG 0 operator2.test.
```

name

```
example._deleg.   IN  CNAME operator1.test.
; or
example._deleg.   IN  SVCB 0 operator1.test.
```

Getting data into registry

- Unaffected by DNS protocol shape
- Both proposals – same expressivity
- EPP all the same
- Hardest of all?

Authoritative changes: code

- Required
- Sign & serve new DELEG RR type – much like DS
- BIND implementation +289 lines, -29 lines (incl. DNSSEC, copy&paste)

type

- Optional, optimization
- Up to 2 x (parallel) query reduction

name

Authoritative: www.example.com. A ➡ com.

- example. NS ns1.example.
- ns1.example. A 192.0.2.1
- example. DELEG 1 (ns1.example. ipv4hint=192.0.2.1)
- example. RRSIG DELEG …

type

- example. NS ns1.example.
- ns1.example. A 192.0.2.1
- example._deleg. SVCB 1 (ns1.example. ipv4hint=192.0.2.1)
- example._deleg. RRSIG SVCB …

name

optional optimization

Authoritative changes: deployment

type

- Parent zone servers only
- Deploy **before** "DELEG-supported" keyflag is set
- Partial deployment ➡ retry / **SERVFAIL**

---

name

- Completely optional
- Up to 2 x (parallel) query reduction
- Partial deployment ➡ an extra explicit query, **works**

Query www.example.com. – minimized

• com. DELEG	➡	.	(parent zones)
• example.com. DELEG	➡	com.	
• www.example.com. DELEG	➡	example.com.	
• com. NS	➡	.	(parent zones)
• com._deleg. SVCB	➡	.	
• example.com. NS	➡	com.	
• example._deleg.com. SVCB	➡	com.	
• www.example.com. NS	➡	example.com.	
• www._deleg.example.com. SVCB	➡	example.com.	

(empty caches)

name | type

Query www.example.com. – traditional

- www.example.com. A ➡ . (parent zones)
- www.example.com. A ➡ com.
- www.example.com. A ➡ example.com.

(empty caches)

name	type

- Protocol requires query name minimization (for parallel queries)
  - OR
- Sequential queries for _deleg SVCB and a non-_deleg name
  - Potentially doubles latency

Resolver algorithm

type

- Same as pre-DELEG
- Query name minimization optional

name

- Additional heuristics
- Discover, cache
  - _deleg subtree presence in parent zone
  - _deleg support on auth server
- Query name mimimization with parallel queries
  - OR sequential

Queries per delegation

- Always 1

type

name

Auth optimized	Parent signed	SVCB present	Zone hack	# queries
Yes	Yes	Yes	No	1
Yes	Yes	No	No	1
Yes	No	Yes	No	1
Yes	No	No	Wildcard	1
Yes	No	No	No	2
Yes	No	None in parent	No	1 + one per parent
No	Yes	Yes	No	2
No	Yes	No	No	2
No	No	Yes	No	2
No	No	No	No	2
No	No	None in parent	No	1 + one per parent

Summary

- "Attacks" lack of extensibility in the parent
- DNSSEC & auth code changes required
  - Deployment per parent zone at a time
- Optimal query count

type

- "Side-steps" lack of extensibility in the parent
- No DNSSEC changes
  - Unilateral deployment by resolvers
- Additional heuristics / complexity in resolver
- Additional queries / auth changes / wildcard hack

name

Parent-side RR vs. _deleg subtree