Automatic DNSSEC Bootstrapping

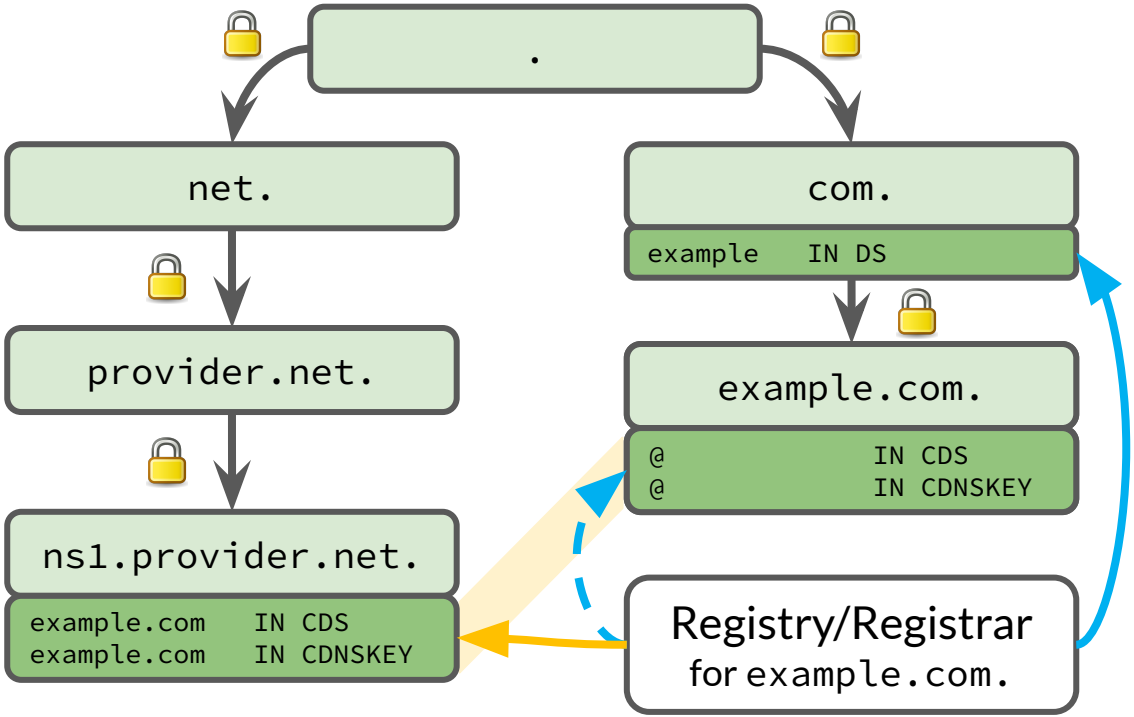
using Authenticated Signals from the Zone's Operator

draft-ietf-dnsop-dnssec-bootstrapping

IETF DNSOP Interim
May 24, 2022

Peter Thomassen (deSEC, Secure Systems Engineering)
Nils Wisiol (deSEC, Technische Universität Berlin)

Reminder: CDS Authentication via Trusted Hostname



💡 Use an **established chain of trust** (left) to take a detour

- co-publish
- authenticated, immediate
- no active on-wire attacker

Extends RFC 8078 to add authentication for initial DS

Status

- Adopted by DNSOP WG in April

- Wrote post for APNIC Blog to get the word out
  - https://blog.apnic.net/2022/03/08/authenticated-bootstrapping-of-dnssec-delegations/

- Implementations:
  - Prototype implementation: github.com/desec-io/dsbootstrap
  - CoCCA: implementation underway for 59 ccTLDs
  - GoDaddy: implementation planned after CDS scanning
  - .cl: implementation finished, waiting for internal approval
  - implementations by other registries and DNS operators under way

Open Issue 1: Support requirements

- Assume that a DNS operator supports the protocol
- Should the operator be REQUIRED to serve bootstrapping records for all their domains?
- Suggestion: No, as it won’t work with zones with secondary-only service

Open Issue 2: IANA action?

- Do we need a IANA action section to reserve the `_dsauth` label?

Open Issue 3: Delegations within a bootstrapping zone

- At IETF 112, it was discussed whether owner names of bootstrapping records should be
  - “plain”, e.g. **`example.co.uk`**`._dsauth.ns1.desec.io`, or
  - “hashed”, e.g. **`example.<`***`hash`***`(co.uk)>`**`._dsauth.ns1.desec.io`.

  There appears to be consensus to use the plain approach.

- **Plain approach causes ambiguities** with zone cuts underneath _dsauth label
  - CDS/CDNSKEY record ambiguous e.g. with zone cut at **`co.uk`**`._dsauth.ns1.desec.io`
  - Details: https://mailarchive.ietf.org/arch/msg/dnsop/FE5Sm5vzZtq9VgKxgkfmv4VuVI8/

- How to remove the ambiguity?
  - Ideally, solution would **preserve commonplace protocol guarantees**:
    all domains are equal (from a protocol perspective), **delegations are allowed everywhere**

Open Issue 3: Solution options – Decision needed

1. Use a different record type
   - e.g. BDS/BDNSKEY (“bootstrapping DS/DNSKEY”)
2. Use a hashed naming scheme to avoid the collision
   - ruled out
3. Allow bootstrapping only using the leaf domain under `_dsauth`
   - implies certain assumptions on the order in which delegations are done
   - treats some domains special: breaks bootstrapping for non-leaf domains under `_dsauth`
4. Disallow CDS/CDNSKEY usage for rollovers of subzones under `_dsauth`
   - “bootstrapping has precedence”
   - treats some domains special: breaks rollovers for domains under `_dsauth`
5. Disallow zone cuts underneath a `_dsauth` label entirely
   - treats some domains special: domains under `_dsauth` can’t be delegated
6. Use an underscore prefix for the actual signal
   - `_dsauth.example.co.uk._signal.ns1.desec.io`
   - would also allow other kinds of signals (multi-signer?)

Next steps

- Authors consider the protocol rather mature
  - Some things still needed!

- Document review and suggestions for improvement, especially
  - Section 3.3 (Triggers)
  - Section 6 (Security Considerations)

- Implementations!

Backup (some slides may be from previous revisions)

Detail: Transfer Trust from the DNS Operator

1. Create a **signaling mechanism for DNS operators**
   - **What?**
     - allow **publishing arbitrary information** about the zones they are authoritative for
     - in an **authenticated** fashion, **on a per-zone basis**
   - **How?**
     - use namespace **under each nameserver hostname**, e.g. _boot.ns1.desec.io
     - **require DNSSEC** under this namespace (requires nameserver domains to be secure)
     - under this namespace, **announcements** are made **using zone-specific owner names**

2. Use this mechanism to **publish an authentication signal**
   - start with **CDS/CDNSKEY records at the apex** of the target zone (RFC 8078)
   - **co-publish these records using the signaling mechanism** (signed with NS zone's keys)

3. **Validate** the target domain's CDS/CDNSKEY records **against this signal**
   - if successful: "transfer trust to the target domain" → **provision DS records** at the parent
   - **clean up** records when done

Technical Considerations

- No collision with primary use of CDS/CDNSKEY (those are apex-only)
- Add extra label: example.co.uk.**_dsauth**.ns1.provider.net
  - to enable delegation of signaling data to separate zone
- Name scheme features:
  - removes risk of accidentally modifying the nameserver's A/AAAA records
  - reduces churn on nameserver zone
  - allows splitting off DNS operations (e.g. online-signing with different key; delegate by parent)

Survey on Deployment Requirements

- DS bootstrapping **requires that NS targets** are **not part of the same zone**
  - **mostly the case:** > 99% of NS targets are out of bailiwick
    in bailiwick: < 0.33% for .com, < 0.72% for .net (thanks to John Levine)

- Secure signaling **requires NS targets** to be **in securely delegated zones**
  - How frequent is that?
  - For each domain in **Tranco Top 1M dataset**, extract
    a. whether the domain itself is **secure** (has validation path),
    b. **all NS targets** in the delegation,
    c. which NS targets are **secure** (if any),

    … and compute things like
    Bootstrappability: A domain is *bootstrappable* if b == c, but a == false

Survey on Deployment Requirements: Bootstrappability

```
Measurement failure rate.............................:   2.30%
Remaining sample size................................:  977007

Proportion of secure zones...........................:   5.43%
Proportion of signed zones...........................:   6.84%

Proportion of zones with all nameserver targets secure:  24.63%
Proportion of zones with ≥ 1 nameserver targets secure:  25.97%
```

bootstrappable:
domain is not secure *and* NS targets have validation path → signaling possible

```
Proportion of bootstrappable zones (all NS) ..........:  22.11%
Proportion of bootstrappable zones (≥ 1 NS) ..........:  23.07%
```

as of 22 October 2021

Survey on Deployment Requirements: by TLD, by Provider

tld	zones total count	signed rel.	secure rel.	bootstrappable rel.	bootstrappable abs.
com	513660	4.5%	3.4%	23.2%	119195
org	71332	4.8%	3.7%	17.8%	12664
net	46232	6.8%	5.4%	22.1%	10231
ru	32387	7.3%	2.0%	13.9%	4511
uk	21003	4.3%	3.4%	18.8%	3945
in	9595	7.3%	5.7%	28.3%	2719
io	7673	8.6%	6.2%	34.9%	2677
xyz	4054	6.1%	5.1%	55.6%	2254
co	7408	10.6%	8.7%	29.7%	2201
online	3202	3.3%	2.4%	68.1%	2180

ns_rname	zones total count	signed rel.	secure rel.	bootstrappable rel.	bootstrappable abs.
dns.cloudflare.com.	252145	6.1%	3.1%	76.5%	192895
dns.hostinger.com.	4141	0.1%	0.0%	87.8%	3634
hostmaster.nsone.net.	19911	1.1%	0.9%	12.9%	2568
nan	80403	9.2%	8.6%	2.6%	2066
hostmaster.cscdns.net.	6041	1.8%	1.7%	22.8%	1375
dns.openprovider.eu.	1290	1.0%	0.8%	91.7%	1183
postmaster.iij.ad.jp.	935	2.0%	2.0%	98.0%	916
nstld.verisign-grs.com.	8531	90.4%	90.4%	7.5%	637
root.v1.wpxhosting.com.	617	0.3%	0.3%	99.7%	615
nsadmin.nic.in.	771	29.4%	29.4%	70.6%	544

as of 22 October 2021, “nan” ns_rname means that referenced NS zones have more than one rname in their SOAs

	Bootstrapping Method		
	Manual	CDS/CDNSKEY	Proposed
Bootstrapping Involves			
zone operator Z	✓[1]	✓	✓
domain owner	✓	✗	✗
registrar	✓	✗	✗
registry	✓	✓	✓
Actors who can initialize keys			
Required parties (trusted)			
registrar	✓	✓[2]	✓[2]
NS zone operator	✗	(✓)	(✓)[3]
NS zone ancestors	✗	(✓)	(✓)
NS zone owner	✗	(✓)	(✓)
Others parties (untrusted)			
active on-wire attacker	depends	✓[4]	✗
social engineering attacker [1]	✓	✗	✗
Properties			
Prerequisites	out-of-band channel	MITM attack mitigation	suitable NS zone configuration
Authentication	bad in practice [1]	none	cryptographically
Duration	varies	days	minutes

Table 1: Comparison of methods for establishing a new secure delegation, dispaying a) entities involved in the bootstrapping of an individual insecure zone, b) attack surface towards trusted and untrusted third parties, and c) prerequisites, key material authentication, and bootstrapping duration. Key initialization within parentheses (✓) requires collusion across all NS zones. [1] For offline signing, only the signing key holder is involved. [2] Registry could refuse deployment through registrar. [3] Requires knowledge of private key. [4] Several vantage points and long time must be covered.