

Updates on

[draft-ietf-dnsop-cds-consistency](#)

IETF 117 – DNSOP WG

July 24, 2023

Peter Thomassen (deSEC, Secure Systems Engineering)

# Security Risks in Automatic Delegation/Trust Maintenance

- **CDS/CDNSKEY** spec says nothing about how parent should poll (RFC 7344)
  - Parents likely use standard resolution for retrieving CDS/CDNSKEY records from child
  - Used for **automatic DS management** (key rollovers, bootstrapping) → potential **security impact**
- **CSYNC** spec advocates limiting queries to just one auth (RFC 7477 Sec. 3.1)
  - Even suggests asking all (+ compare serial) for **freshness, not consistency** (Section 4.2)
  - Used for **delegation updates** (hostnames/glue, provider change) → potential **security impact**
- **Asking a single nameserver does not ensure consistency** across auths
  - This can go seriously wrong (even with domain lock!)
  - Example failure modes: **multi-homing, provider change, lame delegation hijack** (→ backup)
  - In multi-provider setups: **each party is a single point of failure**

**! Each nameserver publishing C\* records can break delegations !**

# New Failure Mode: **Replication Lag**

- In a **KSK rollover**, consider following series of events:
  1. ns1 has up-to-date zone data, **ns2 lags behind**
  2. Parent fetches new CDS/CDNSKEY records **from ns1** → **deploys new DS records**
  3. Zone owner monitors DS, detects deployment and continues rollover → removes old DNSKEY
  4. Next day, parent fetches CDS/CDNSKEY **from ns2** → **old DS gets deployed**

→ **broken (SERVFAIL)** ... but may work again the next day
- Correct behavior during rollovers: **wait until next step is safe**
  - Applies to zone operator
  - **But also applies to parent – don't break chain of trust!**
    - RFC 7344 Section 4.1: “MUST NOT break the current delegation if applied”
- Just one of many failure modes

# Updates since last IETF

- Basics unchanged: **process C\* RRsets only when consistent across auths**
- **NEW:** If response cannot be obtained, **SHOULD** try again at a later time
  - Exponential back-off **RECOMMENDED**
  - Prevents accidentally declaring a host permanently unreachable (might hide inconsistency)
  - **MAY** try from another vantage point to sidestep localized routing issue
- **NEW:** Explain that extra queries are only needed when records have changed
  - otherwise zero overhead
- Editorial changes
  - Including draft name change after adoption

# Open Issues

## 1. Consistency of the CSYNC serial field?

- Draft: "Parental Agent **MUST** [...] **ensure that the CSYNC rdata sets are equal**" across responses
- In multi-homing setups, providers will likely have different serials. **What now?** – Options:
  - a. Don't deal with CSYNC in this draft
  - b. Advise that in multi-provider setups, CSYNC serial processing should be turned off (don't set soaminimum flag, RFC 7477 Section 2.1.1). – **Still, how to achieve equality?**
  - c. **Should "consistency" really mean "equality"?** Perhaps:
    - Perform serial processing on a per-NS basis (seems most correct)
    - Only require equality of immediate flag + type bitmap + data RRsets (quite complex)

## 2. Require parents to check CSYNC updates don't break DNSSEC?

- CDS/CDNSKEY spec: "MUST NOT break the current delegation if applied" (RFC 7344 Sec. 4.1)
- CSYNC updates may have same effect (e.g. replace NS RRset so DNSKEYs become unavailable)
- Parents currently don't need to check. Should they?

# Backup

# Failure: Multi-homing

- Expectation: multi-homing guarantees provider independence!
- DS breakage (multi-signer):
  - Provider forgets to include other providers' keys in CDS/CDNSKEY (e.g. after key roll)
  - When processed by parent, **other providers' keys removed** from chain of trust  
→ **broken**
- NS breakage:
  - Provider publishes *incomplete* NS record set + CSYNC (e.g. after changing their hostnames)
  - When processed by parent, **other providers removed** from delegation  
→ **broken**

# Another Failure: **Provider Change**

- Unless going insecure, workflow requires **brief multi-signer period**:
  - Providers import each other's keys into their DNSKEY/CDS/CDNSKEY RRsets
  - DS update is triggered (via changed CDS/CDNSKEY records at old provider)
  - Once DS is updated: add new provider to NS record set (e.g. by old provider via CSYNC)  
→ **multi-signer mode fully operational** at this point
  - ... reverse steps to offboard old provider
- **Complication: New provider does not actually import any keys**
  - (Perhaps unaware of multi-signer and its intricacies)
  - Some “DNSSEC out-of-the-box” offers just **sign with fresh key pair + publish CDS/CDNSKEY**
  - From here, we're headed for “**multi-homing failure**”
    - **DS breakage** (other provider's keys removed)
    - **NS breakage** (other provider's nameservers removed)

# Failure Mode: **Lame Delegation Hijacking**

- EPP has a quirk that sometimes prevents removal of expired NS names
    - Registering expired name equivalent to on-wire attacker → **DNSSEC offers integrity protection**
    - **512K domains exposed** to this risk and **163K taken over** between 2011 and 2020  
(<https://dl.acm.org/doi/10.1145/3487552.3487816>)
  - C\* records enable new attack vector: **Full domain take-over**
    - Stage 1
      - Hijacker **publishes their own keys** via CDS/CDNSKEY
      - When processed by parent, responses from **remaining legitimate auths become bogus**  
→ **broken (availability)**
    - Stage 2
      - Hijacker **publishes NS and CSYNC** in child (all NS under their control)
      - When processed by parent, **remaining legitimate auths removed** from delegation  
→ **broken (integrity)**
- **Attacker now positioned as only party providing auth service for the victim domain**