Announcing CDS/CSYNC Scanner Existence and Capabilities

draft-berra-announce-scanner-02

Erik Bergström Leon Fernandez Johan Stenstam

The Swedish Internet Foundation

November 3, 2025

There Have Been Changes From -00 to -02

When I presented the initial version of this draft I said that this was very a simple but also useful idea.

- For operators of child zones where the parent runs a CDS and/or CSYNC scanner.

There Have Been Changes From -00 to -02

When I presented the initial version of this draft I said that this was very a simple but also useful idea.

- For operators of child zones where the parent runs a CDS and/or CSYNC scanner.
- It is still simple, but that has not stopped us from changing the mechanism slightly.

Changes: Put Details at Target

In the initial version of the draft the semantics of the `DSYNC NOTIFY port` and `target` fields were overloaded to signal the absence of a scanner and/or the scanner interval.

```
_dsync.se.        IN DSYNC CDS   NOTIFY 1440 scanner.iis.se.
_dsync.se.        IN DSYNC CDS   NOTIFY 0 .
```

- `port="0"` and `target="."` signaled the absence of a scanner and/or a notification receiver, respectively.

Changes: Put Details at Target

In the initial version of the draft the semantics of the `DSYNC NOTIFY port` and `target` fields were overloaded to signal the absence of a scanner and/or the scanner interval.

```
_dsync.se.        IN DSYNC CDS   NOTIFY 1440 scanner.iis.se.
_dsync.se.        IN DSYNC CDS   NOTIFY 0 .
```

- `port="0"` and `target="."` signaled the absence of a scanner and/or a notification receiver, respectively.

Changes: Put Details at Target

In the initial version of the draft the semantics of the `DSYNC NOTIFY port` and `target` fields were overloaded to signal the absence of a scanner and/or the scanner interval.

```
_dsync.se.        IN DSYNC CDS    NOTIFY 1440 scanner.iis.se.
_dsync.se.        IN DSYNC CDS    NOTIFY 0 .
```

- `port="0"` and `target="."` signaled the absence of a scanner and/or a notification receiver, respectively.

Instead, the scanner signaling has been moved to a new `DSYNC scheme`: `SCANNER`. The mechanism for providing details about a service has been standardized to use an `SVCB` record located at the `DSYNC target`:

```
_dsync.se.        IN DSYNC ANY    SCANNER 0 scanner.iis.se.
scanner.iis.se.   IN SVCB . 0 ( interval=86400 )
```

Changes: Put Details at Target

In the initial version of the draft the semantics of the `DSYNC NOTIFY port` and `target` fields were overloaded to signal the absence of a scanner and/or the scanner interval.

```
_dsync.se.        IN DSYNC CDS    NOTIFY 1440 scanner.iis.se.
_dsync.se.        IN DSYNC CDS    NOTIFY 0 .
```

- `port="0"` and `target="."` signaled the absence of a scanner and/or a notification receiver, respectively.

Instead, the scanner signaling has been moved to a new `DSYNC scheme`: `SCANNER`. The mechanism for providing details about a service has been standardized to use an `SVCB` record located at the `DSYNC target`:

```
_dsync.se.        IN DSYNC ANY    SCANNER 0 scanner.iis.se.
scanner.iis.se.   IN SVCB . 0 ( interval=86400 )
```

There is a scanner

Scanner interval

Changes, cont'd: Add Support For Bootstrap Mechanisms

Using the Swedish registry `.SE` as an example, we provide automatic DNSSEC bootstrap using RFC 8078 semantics

- "Query for the `CDS` RR from multiple vantage points, to all auth NS, multiple times"

But now RFC 9615 provides an alternative mechanism:

- "Validate the **signed** `CDS` RR located at `{magic name}`, where `{magic name}` is `_dsboot.{child}._signal.{nsname}`"

With multiple DNSSEC bootstrap mechanisms, the parent needs a method to signal which (if any) mechanisms it supports.

```
_dsync.se.        IN DSYNC ANY   SCANNER 0 scanner.iis.se.
scanner.iis.se.   IN SVCB . 0 (bootstrap="rfc8078,rfc9615" interval=86400)
```

Summary of Changes

1. New `DSYNC scheme: SCANNER`

```
_dsync.se.        IN DSYNC ANY   SCANNER 0 scanner.iis.se.
```

2. Locate details about a service at the `DSYNC target` instead of overloading the `DSYNC port` and `target` fields.
3. Add mechanism for announcing support for multiple DNSSEC bootstrap mechanisms.

```
_dsync.se.        IN DSYNC ANY   SCANNER 0 scanner.iis.se.
scanner.iis.se.   IN SVCB . 0 (bootstrap="rfc8078,rfc9615" interval=86400)
```

4. When there is a `NOTIFY` **Receiver**, then similar bootstrap details are needed for the `DSYNC NOTIFY scheme`:

```
_dsync.se.                IN DSYNC CDS NOTIFY 5301 notify-receiver.iis.se.
notify-receiver.iis.se.   IN SVCB . 0 (bootstrap="rfc8078,rfc9615")
```

Delegation Synchronization via DNS UPDATE

draft-johani-delegation-mgmt-via-ddns-02

Erik Bergström Leon Fernandez Johan Stenstam

The Swedish Internet Foundation

November 3, 2025

While Waiting for the DSYNC Record

We put this draft on a back burner while waiting for the DSYNC record to be standardized.

- Now that DSYNC is done it is time to revive the work on automatic delegation synchronization for unsigned child zones.

While Waiting for the DSYNC Record

We put this draft on a back burner while waiting for the DSYNC record to be standardized.

- Now that DSYNC is done it is time to revive the work on automatic delegation synchronization for unsigned child zones.

The DNS protocol supports parent-side delegation management via DNS UPDATE since RFCs 2136 (1997), 2931 and 3007 (2000).

- **This functionality has been available for 25+ years!**

While Waiting for the DSYNC Record

We put this draft on a back burner while waiting for the DSYNC record to be standardized.

- Now that DSYNC is done it is time to revive the work on automatic delegation synchronization for unsigned child zones.

The DNS protocol supports parent-side delegation management via DNS UPDATE since RFCs 2136 (1997), 2931 and 3007 (2000).

- **This functionality has been available for 25+ years!**

The contribution of this draft is two-fold:

- Using the DSYNC scheme UPDATE it standardizes the mechanism for locating where to send the DNS UPDATEs.
- It describes scalable mechanisms for automatically bootstrapping the child SIG(0) public key by the parent.

Bootstrapping The SIG(0) Public Key?

A primary concern (or perhaps objection) with this draft has been the method used to automatically bootstrap the SIG(0) public key as **trusted** by the parent.

- The previous version of this draft used similar logic as documented in RFC 8078 to bootstrap DNSSEC for a previously unsigned child zone.
- However, RFC 9615 provides an alternative mechanism which we adapted: "Validate the **signed** `KEY` RR located at `{magic name}`", where `{magic name}` is `_sig0key.{child}._signal.{nsname}`

However, that immediately prompts a need for a mechanism to signal which bootstrap mechanisms for the SIG(0) public key that the parent supports.

- **Hmm. . . it feels like we've seen this before. . .**

Symmetry Is Nice

In symmetry with the `DSYNC SCANNER` and `DSYNC NOTIFY` cases

```
_dsync.se.              IN DSYNC CDS SCANNER 0 scanner.iis.se.
scanner.iis.se.         IN SVCB . 0 (bootstrap="rfc8078,rfc9615" interval=86400)
```

```
_dsync.se.                IN DSYNC CDS NOTIFY 5301 notify-receiver.iis.se.
notify-receiver.iis.se. IN SVCB . 0 ( bootstrap="rfc8078,rfc9615" )
```

. . . we propose to use the following mechanism for signaling support for `DSYNC UPDATE` SIG(0) key bootstrap mechanisms:

```
_dsync.se.                IN DSYNC ANY UPDATE 5302 update-receiver.iis.se.
update-receiver.iis.se. IN SVCB . 0 ( bootstrap="at-apex,at-ns" )
```

Apart from symmetry, this has the advantage of simplifying the KeyState EDNS(0) Option that is used for communication about the SIG(0) public key between the child and the parent.

Symmetry Is Nice

This is not optimal. Should be unified.

In symmetry with the DSYNC SCANNER and DSYNC NOTIFY cases

```
_dsync.se.              IN DSYNC CDS SCANNER 0 scanner.iis.se.
scanner.iis.se.         IN SVCB . 0 (bootstrap="rfc8078,rfc9615" interval=86400)
```

```
_dsync.se.                IN DSYNC CDS NOTIFY 5301 notify-receiver.iis.se.
notify-receiver.iis.se. IN SVCB . 0 ( bootstrap="rfc8078,rfc9615" )
```

. . . we propose to use the following mechanism for signaling support for DSYNC UPDATE SIG(0) key bootstrap mechanisms:

```
_dsync.se.                IN DSYNC ANY UPDATE 5302 update-receiver.iis.se.
update-receiver.iis.se. IN SVCB . 0 ( bootstrap="at-apex,at-ns" )
```

Apart from symmetry, this has the advantage of simplifying the KeyState EDNS(0) Option that is used for communication about the SIG(0) public key between the child and the parent.

Summary of Changes

1. Support for a new RFC 9615-based SIG(0) key bootstrap mechanism,
   - This mechanism is provably secure.
2. Standardized mechanism across the three `DSYNC` schemes `NOTIFY`, `SCANNER` and `UPDATE` for announcing service details at the `DSYNC` `target`.
3. Simplified the KeyState EDNS(0) Option that is used for communication about the SIG(0) public key between the child and the parent.

Questions?

Thank You & Contact Information

erik.bergstrom
leon.fernandez
johan.stenstam
@internetstiftelsen.se

Source code https://github.com/johanix/tdns