CDS/CDNSKEYレコードのこれから

Internet Week 2023
2023/11/21

永井祐弥

自己紹介

■名前

永井 祐弥

■所属

個人

■略歴

2003年にdjbdnsに触れ始める。2010年のDNSSEC実証実験にも参加。

2012年から大手国内レジストラでレジストリ、レジストラ、ホスティングサービス等のDNSとDNSに関連するサービスの開発、運用を担当していた。

DNSSECは自社サービスへの導入のほか、現在は総務省の調査事業(*1)の一環としてDNSSEC有識者会議参画メンバーに加わり活動中。

(*1) ISPにおけるネットワークセキュリティ技術の導入に関する調査

CDS/CDNSKEYのメカニズム

CDS/CDNSKEYは子ゾーンのCDSレコード、CDNSKEYレコードを親ゾーンへ反映するための仕組み（RFC7344、RFC8078）

親ゾーンは子ゾーンのCDS/CDNSKEYを反映

親ゾーン (example.)

```
child.example. 86400 IN NS ns1.example.com.
child.example. 86400 IN NS ns2.example.com.
child.example. 86400 IN DS 12345 13 2 340A8516F584E68E72C35F168A546B80E...
```

子ゾーンはCDSレコードかCDNSKEYレコード
又はその両方をゾーンファイルに記述

子ゾーン (child.example.)

```
child.example. 86400 IN SOA ns1.example.com. postmaster.child.example. ...
child.example. 86400 IN NS ns1.example.com.
child.example. 86400 IN NS ns2.example.com.
child.example. 86400 IN DNSKEY 256 3 13 JLskW2IEf8kDj3f30e1aqvqTG+qp42Y...
child.example. 86400 IN DNSKEY 257 3 13 PeLZkjU27HV3XiC41dUGsDuyP6pPD7a...
child.example. 86400 IN CDNSKEY 257 3 13 PeLZkjU27HV3XiC41dUGsDuyP6pPD7...
child.example. 86400 IN CDS 12345 13 2 340A8516F584E68E72C35F168A546B80...
```

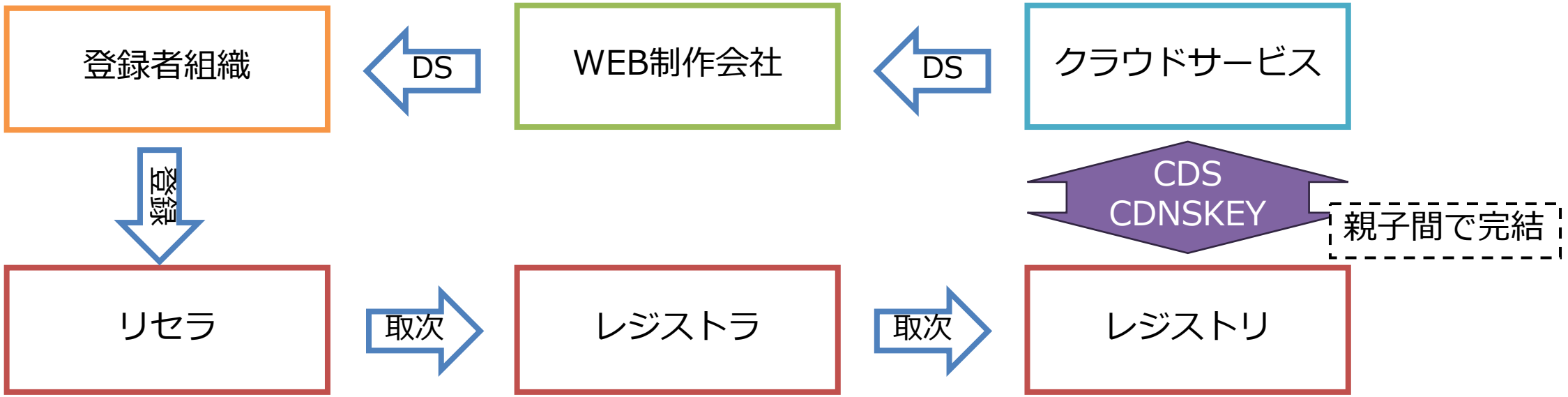
どちらも同じ値

CDS/CDNSKEYの機能
1. DSレコードの登録（DNSSEC有効化）
2. DSレコードの変更（鍵の更新）
3. DSレコードの削除（DNSSECの無効化）

CDS/CDNSKEYのメリット

CDS/CDNSKEYはDS取り次ぎ申請の代替手段として用いることが可能

- 例えばDNSプロバイダによるDNSSECサービスを利用者が意識することなく自動的にDNSSECを有効化したり、ドメイン名登録者とゾーン管理者が異なる場合ではDS取り次ぎ申請に係る手続きを簡略化することが出来る

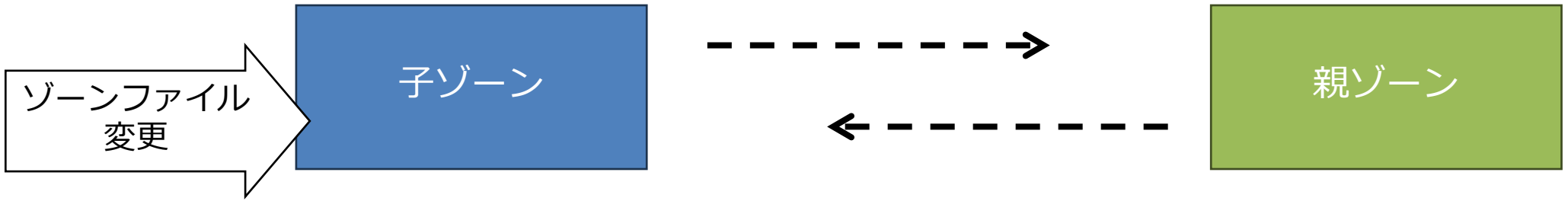


CDS/CDNSKEYの反映トリガー

子ゾーンのCDS/CDNSKEYが登録、更新、削除されたことを
親ゾーンが知るには、親と子のどちらかがアクションする必要がある

1. 子側から親側に申請
2. 親側によるポーリング等の定期的な確認プロセス（スキャン）

(1)CDS/CDNSKEY登録後の子側のアクション
・申請（Web UI、API、メール、Notify、etc）



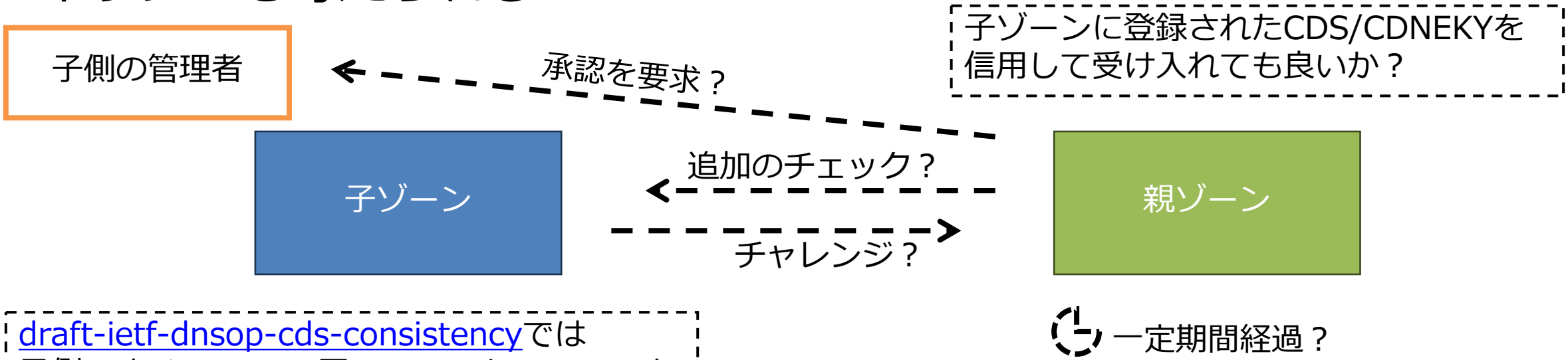
draft-ietf-dnsop-generalized-notifyでは
既存のNOTIFYメカニズムの拡張し、NOTIFY
レコード（仮）の追加を提案している

(2)親側の定期的なアクション
・ポーリング（DNS）

親エンティティのポリシー

親側のエンティティはポリシーに従ってCDS/CDNSKEYを受け入れる

- 本物の子ゾーン管理者による変更であることを検証、確認するかもしれないし、無条件に受け入れるというポリシーもありうる
- あるいは受け入れにチャレンジ（例えばトークン）を要求するポリシーも考えられる



draft-ietf-dnsop-cds-consistencyでは
子側のすべてのNSで同一のCDS/CDNSKEYを
求めることを提案している

CDS/CDNSKEYの課題（エンティティ）

CDS/CDNSKEYを機能提供するために必要な検討項目

- CDS/CDNSKEY受け入れポリシー策定の難しさ
  - 本当にDSレコードを登録/更新/削除しても良いのか？
  - 同意をどの段階で得るか？
- ポーリングの場合、スキャンを実行するためのリソース
  - 1日XX回？ あるいは、NNN秒毎/件数？
  - 子ゾーンのDNS応答が偽装されていないことの検証
  - DNSSECを理解しているエンジニア（プログラマ）による実装
- オペレーター、カスタマーサポートの教育
  - 提供機能を説明できるだけのドキュメント、運用ナレッジの構築
- マーケットの需要
  - 開発コストに見合う提供価値があるのか？

CDS/CDNSKEYの課題（コミュニティ）

CDS/CDNSKEYにおけるコミュニティの課題

- レジストリロックやレジストラロックによるドメイン名登録情報の変更禁止と、CDS/CDNSKEYによるメカニズムのどちらを優先するべきか
- レジストリとレジストラ（あるいはリセラ）のどちらもCDS/CDNSKEYに対応済みの場合に競合しないような関係（システム）が築けるかどうか
- レジストリやレジストラ（あるいはリセラ）は子側へ状況を通知するべきかどうか（例えば正常に処理出来た場合や、CDS/CDNSKEYの不備、親側のポリシーに違反している場合のエラーなど）
- 親側のポリシーを子側が自動的に知る術が存在しない（許可されているCDS/CDNSKEYのパラメタや個々のドメイン名の許可状況など）
- DNSSECを安全に有効化する方法について明示的なガイドラインが存在しない（draft-ietf-dnsop-dnssec-bootstrappingでは初回の自動登録について提案している）

将来的な予測

現時点ではエンティティ次第

- レジストリ視点では提供可能性あり
  - 新gTLDレジストリは既存レジストリとの差別化を図るかもしれない
  - ICANNにより新しいポリシーが追加されるかもしれない
- レジストラ/リセラ視点では慎重傾向か
  - レジストラやリセラはレジストリに先駆けて機能提供する可能性はある
  - レジストリが提供した場合に機能が重複するため、
    機能の開発には消極的かあるいは慎重な判断になりそう
- DNSプロバイダ視点では親ゾーンにアプローチしたい
  - 登録、鍵更新を自動化できるメリットは大きい
  - 今後レジストリ、レジストラに対して機能提供を求めることは考えられる

CDS/CDNSKEYの活用

CDS/CDNSKEYを期待する場面として、システムによる自動登録を行う時が挙げられる

- 通常のDS取り次ぎではレジストラ/リセラのUIを手動で操作するか、APIなどを利用するため、個人的な利用には差し支えない
- DNSプロバイダのように多数のドメイン名を扱うゾーン管理者の場合は、CDS/CDNSKEYのように統一されたメカニズムの方が都合が良い
- マルチ署名DNSSECモデル（RFC8901）では同じゾーンを複数のDNSプロバイダが各々の署名を共存する方法について情報提供されており、例えばDNSプロバイダの変更をマルチ署名DNSSECモデルとCDS/CDNSKEYを組み合わせて利用することでDS取り次ぎ機能に依存せずに移行出来るようになる

まとめ

CDS/CDNSKEY機能が提供されると...

- DSレコードの取り次ぎが自動化される
  - DNSSECの鍵更新（キーロールオーバー）、有効化、無効化

- 現状はメカニズムが先行しておりポリシーが追いついていない
  - 既存ポリシーのアップデート
  - 誰が、いつ、どのように？

- CDS/CDNSKEYが期待されている場面
  - DNSプロバイダ
  - マルチ署名DNSSECモデル

ご清聴ありがとうございました