ANYクエリには一部のレコードだけが得られる件

東 大亮　DNS Summer Day 2025　2025/6/27

目次

- ANYクエリとは
  - ANYクエリの問題点 (DNSリフレクション攻撃への悪用)
  - ANYクエリの問題点の克服
- ANY応答の最小化 (RFC 8482)
  - 最小化なんかして大丈夫？
  - RFC 8482ソフトウェア実装状況・利用しているオペレータ
  - DNSオペレータが注意すべきこと
- まとめ
- Appendix　RFC 8482の後方互換性について (トークでは省略)

ANYクエリとは

- 通常のDNSの問い合わせではレコードタイプを１つ指定するが、ANYタイプ (QTYPE=255)を指定することができる
- **伝統的な実装**のDNSサーバ(特に権威サーバ)は、ANYクエリに対してQNAMEに対応する全てのRRSetを応答する [RFC1035]

レコードタイプ(A)を指定して問い合わせ

```
% dig @ns-1442.awsdns-52.org. kantei.go.jp A

;; ANSWER SECTION:
kantei.go.jp.      60 IN A  3.166.244.21
kantei.go.jp.      60 IN A  3.166.244.42
kantei.go.jp.      60 IN A  3.166.244.101
kantei.go.jp.      60 IN A  3.166.244.31

;; MSG SIZE  rcvd: 245
```

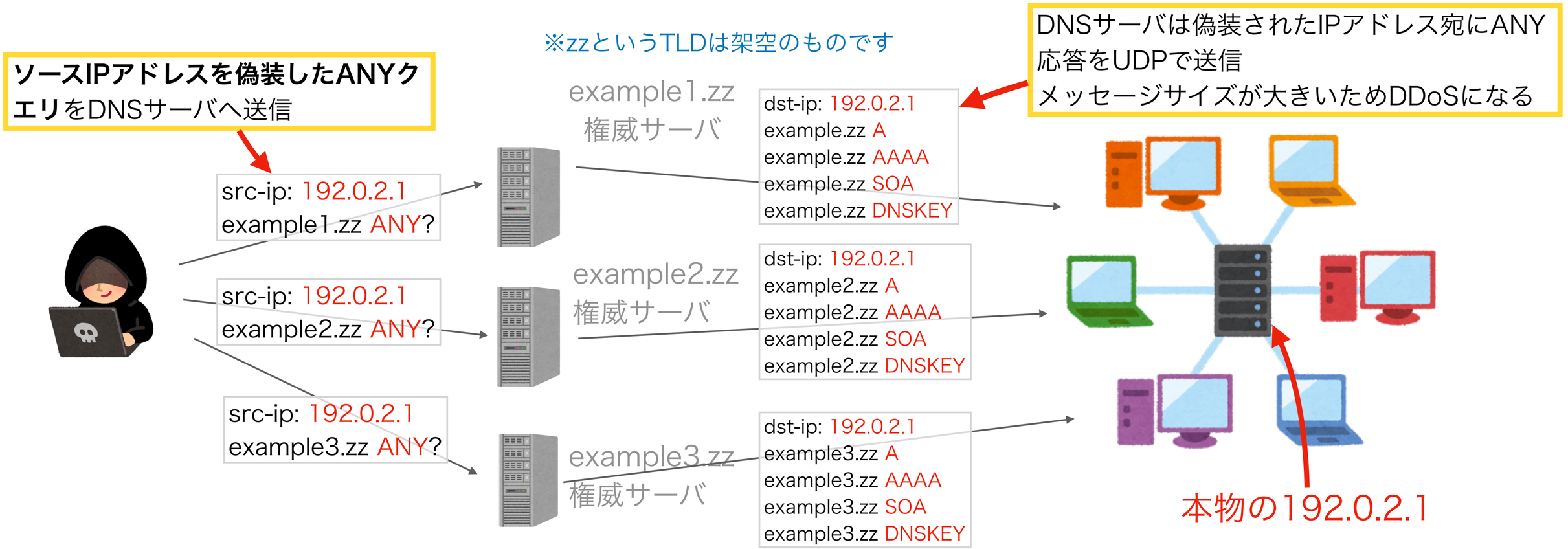
```
% dig @ns-1442.awsdns-52.org. kantei.go.jp ANY

;; ANSWER SECTION:
kantei.go.jp.      60 IN A  3.166.244.21
kantei.go.jp.      60 IN A  3.166.244.42
kantei.go.jp.      60 IN A  3.166.244.101
kantei.go.jp.      60 IN A  3.166.244.31
kantei.go.jp.      172800  IN NS ...
kantei.go.jp.      172800  IN NS ...
kantei.go.jp.      172800  IN NS ....
kantei.go.jp.      900   IN SOA  ...
kantei.go.jp.      300   IN MX ...
kantei.go.jp.      300   IN TXT  ...
kantei.go.jp.      300   IN TXT  ...
kantei.go.jp.      300   IN TXT  ...
kantei.go.jp.      3600  IN DNSKEY 256 3 13 ...
kantei.go.jp.      3600  IN DNSKEY 256 3 13 ...
kantei.go.jp.      3600  IN DNSKEY 256 3 13 ...

;; MSG SIZE  rcvd: 765
```

ANYで問合わせ

全部のRRSetが返る

ANYクエリの問題(とされている)点

- ANYクエリに対する伝統的な応答は全RRSetのため応答サイズが特に大きい
- DNSリフレクション攻撃の踏み台として悪用される例が多い（多かった）



ANYクエリの問題点の克服

- ANYのDDoS踏み台対策として、さまざまな対策が実装されてきたが、長年標準化されたものがなった

DNSサーバ側での対策例	実装例 (2025/6現在)
ANYクエリには応答しない (応答パケットを出さない・REFUSE or NOTIMPで応答)	一部のFirewall、DDoS対策製品 Unbound (refuse-any: yes設定時) 1.1.1.1 (CloudflareのDNSリゾルバサービス)
UDPのANY応答はレートリミット (RRL)	BIND9・NSD・Knot DNS 等（デフォルトでOFF)
UDPの応答サイズは一定以下とする (DNS Flag Day 2020) ※もともとリフレクション攻撃対策を目的にはしていないが、対策として一定の効果があったと考えられる	Flag Day以降、多くの DNSサーバのデフォルト設定が 1232バイトという上限設定しているようです
UDPのANYクエリには常にtruncate (TC=1)応答 (TCPで再クエリさせることでソースIPアドレス偽装に対抗)	AWS Route 53 PowerDNS
ANYクエリには、最小応答を返す **[RFC 8482]**	NSD・Knot DNS・BIND9 IIJ DNSプラットフォームサービス CloudFlare DNS Microsoft Azure DNS

kantei.go.jp は AWS Route53なので、ANYクエリを悪用した**DDoS攻撃の踏み台対策済**

ANYクエリの問題点の克服

- ANYのDDoS踏み台対策として、さまざまな対策が実装されてきたが、長年標準化されたものがなった

DNSサーバ側での対策例	実装例 (2025/6現在)
ANYクエリには応答しない (応答パケットを出さない・REFUSE or NOTIMPで応答)	一部のFirewall、DDoS対策製品 Unbound (refuse-any: yes設定時) 1.1.1.1 (CloudflareのDNSリゾルバサービス)
(UDPの)ANY応答はレートリミット (RRL)	BIND9・NSD・KnotDNS 等（デフォルトでOFF)
UDPの応答サイズは一定以下とする (DNS Flag Day 2020) ※もともとリフレクション攻撃対策を目的にはしていないが、対策として一定の効果があったと考えられる	Fl 定 32バイトという上限設
UDPのANYクエリには常にtruncate (TC=1)応答 (TCPで再クエリさせることでソースIP偽装に対抗)	A R
ANYクエリには、最小応答を返す **[RFC 8482]**	NSD・Knot DNS・BIND 9 IIJ DNSプラットフォームサービス CloudFlare DNS Microsoft Azure DNS

標準化され、急速に普及してきた
本方式にフォーカスして議論

ANY応答の最小化 (RFC8482)

- RFC 8482で、DNSサーバ (レスポンダ)は、ANYクエリに対して全てのRRSetを応答しなくてもよいことになった
- 主に2種類の最小化応答の方法を規定

応答メッセージのサイズが劇的に小さくなり、DNSリフレクション攻撃に悪用されてもその威力を軽減する

全RRSetを応答 (従来)

```
% dig @ns2.******.jp ******.JP ANY

;; ANSWER SECTION:
******.jp.      10800 IN SOA...
******.jp.      10800 IN RRSIG SOA ...
******.jp.      10800 IN NS ns2....
******.jp.      10800 IN NS ns1....
******.jp.      10800 IN RRSIG NS ...
******.jp.      300IN A  ...
   (SNIP)
******.jp.      10800 IN RRSIG NSEC3PARAM ...
******.jp.      300IN CAA...
******.jp.      300IN CAA...
******.jp.      300IN CAA...
******.jp.      300IN RRSIG CAA ...

;; MSG SIZE  rcvd: 1528
```

最小化① 一部のRRSetのみ応答 (RFC8482 4.1 or 4.3)

```
% dig @ns1.******.jp ******.JP ANY +dnssec

;; ANSWER SECTION:
******.jp.   300  IN A  ...
******.jp.   300  IN RRSIG A ...

;; MSG SIZE  rcvd: 159
```

最小化② HINFO応答 (RFC8482 4.2)

```
% dig @jeff.ns.cloudflare.com **********.com ANY

;; ANSWER SECTION:
**********.com.  3600  IN HINFO "RFC8482" ""

;; MSG SIZE  rcvd: 112
```

HINFOという、元々当該ホストの"CPU" "OS"を表現するためのレコード（現在はほぼ使用されていない）に、最小化された応答であることを示す短い文字列を記載したものを返す

ANY応答の最小化なんかして大丈夫？

?

今までは ANYクエリには全部のRRSetを応答していたのに、

- 一部のRRSetだけを返したり
- 存在もしないRRSet (HINFO)を返したり

大丈夫なの？ ANYクエリを使用するアプリケーション自体が少ないので問題になることは少ないだろうけど。

!

- **（後方）互換性が保たれるとされる**
- 「なぜ互換性が保たれるか」の説明は長くなるので Appendixで

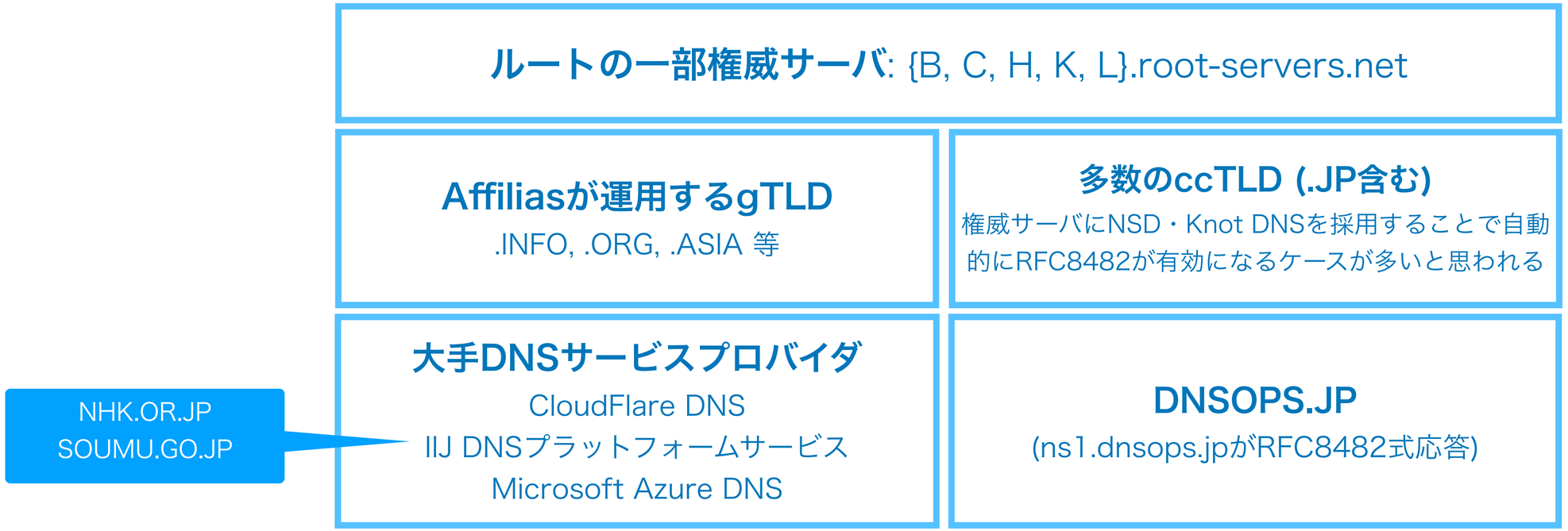
RFC 8482 実装状況

- RFC 8482を実装するDNSサーバソフトウェアの例

	実装概要	参考情報
BIND 9 (9.11以降)	• **UDPのANYクエリには 1つのRRSet (+RRSIG) だけ**を応答する (RFC8482 4.1, 4.4) • **デフォルトでOFF** (有効化設定: minimal-any yes)	• どのRRSetが選択されるかは不定 • TCPのANYクエリには従来通り全RRSetを返す → 最近の dig (9.12-) は ANYクエリをTCPで行うので RFC 8482を有効化したBIND9に気づかないかも
Knot DNS (3.0.0以降)	• **UDP, TCP両方のANYクエリに、1つのRRSet (+RRSIG) だけ**を応答する (RFC8482 4.1) • **常時 ON** (設定でOFFにできない)	• どのRRSetが選択されるかは不定
NSD (4.1.27以降)	• **UDP, TCP両方のANYクエリに、1つのRRSet (+RRSIG) だけ**を応答する (RFC8482 4.1) • **常時 ON** (設定でOFFにできない)	• RRSetの選択はA, AAAA, MX, SOAレコードを優先し、DNSKEY, NSECは低優先とする。 （応答メッセージサイズをなるべく小さくする）
dnsdist (1.9.0以降)	• ANYクエリに対し **HINFO応答** (RFC8482 4.2)**するためのActionオプション (typeForAny)** が用意 • デフォルトでRFC8482応答するわけではない	以下のコードで ANY-to-HINFO応答する。RRSIG合成はしない。 `addAction(QTypeRule(DNSQType.ANY), SpoofRawAction("\007rfc\056\052\056\050\000", { typeForAny=DNSQType.HINFO }))`

RFC 8482を使用しているオペレータの例

- 実装の仕方は様々で、ゾーンの全部のNSでRFC 8482を実装しているオペレータもあれば、一部のNSで実装するオペレータもある



RFC 8482でDNSオペレータが注意すべきこと？

RFC 8482が発行されて約５年が経過し実装も普及しつつあるが、現在のところ一般のインターネット利用者に影響を与える問題は発生していないように見える

一方、ANYクエリはDNSサーバの診断などに使われることがあるので、RFC8482について以下の点には留意したい

1. ANYクエリで全部のレコードが得られると思っているエンジニアが、そうでない応答で困惑するかも
2. ANYクエリを使用した DNSサーバチェックスクリプトが、全部のレコードが返答される前提で書かれていると、突然そのスクリプトがエラーになるかもしれない

実際、某 ccTLDオペレータから、
「ANYクエリに、全レコードが返る前提で書かれたスクリプトが動作しなくなって困るんだけど」
というクレームが、某OSSなDNSサーバのユーザMLに届いたことが・・・（しかも2回）

まとめ

- RFC 8482はそこそこ普及してきたため、オペレータやソフトウェア開発者は以下の点は考慮・留意しましょう。

DNSオペレータ	ANYクエリを診断やDNSサーバのチェックスクリプトで使用するときは、ANYが拒否されたり、RFC8482により一部のRRSetしか返らなかったり、謎のレコード(HINFO)が返る場合があることに留意する。
	BIND9利用者は、RFC 8482の有効化を検討する。 (minimal-any yes)
DNSを使用するソフトウェア開発者	ANYクエリは拒否されたり関係ないレコードが返る可能性があるので、実アプリケーションではANYの使用は避ける。
DNSサーバ（レスポンダ）開発者	御社のソフトウェアに RFC 8482の実装をお願いします！ (ANY-to-TCPでもいいけど)

END

Appendix
RFC 8482の後方互換性について

後方互換性

【英】backward compatibility

システム又はソフトウェアパッケージのより進んだバージョンから、余り進んではいないバージョンにデータを移動する能力

(JIS Z 6015:2022)

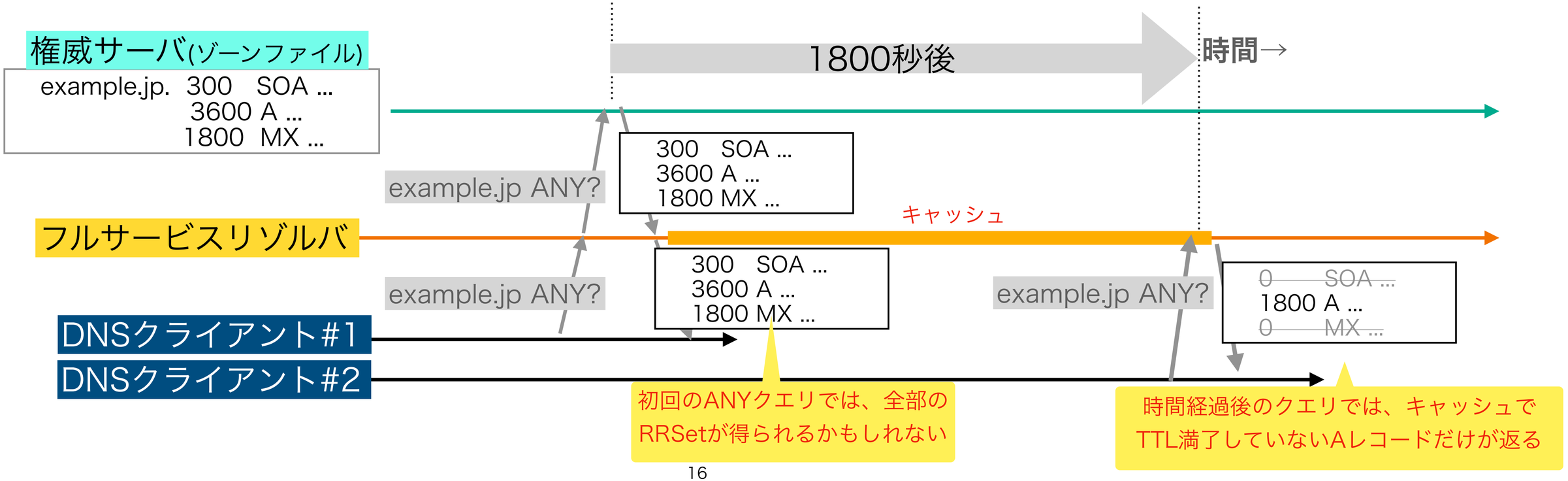
RFC 8482の制定経緯について

- RFC 8482 の最初期のI-D [draft-ogud-dnsop-any-notimp] は、ANYクエリにNOTIMPで応答することを主張し、事実上ANYクエリを禁止するものであった
- NOTIMP応答には、いくつかの問題が指摘された
  - そもそも、ANYクエリを使用する正当なアプリケーション (例: qmail)が存在し、後方互換性を確保せずに突然禁止するのは、それらのアプリが使用できなくなる乱暴な変更である (D.J. Bernstein; qmail 作者)
  - NOTIMP応答は、EDNS(0)の古い規格 [RFC2671]で「EDNS0未対応」と区別できず、そのような古い実装では追加の余計な(ENDS0無しの)クエリが行われる等の不効率なものとなる

→ 後方互換性を維持しつつ、簡潔なANYクエリ応答の最小化の方式が模索された

「一部のRRSetのみ応答」に変えても問題ないのか？

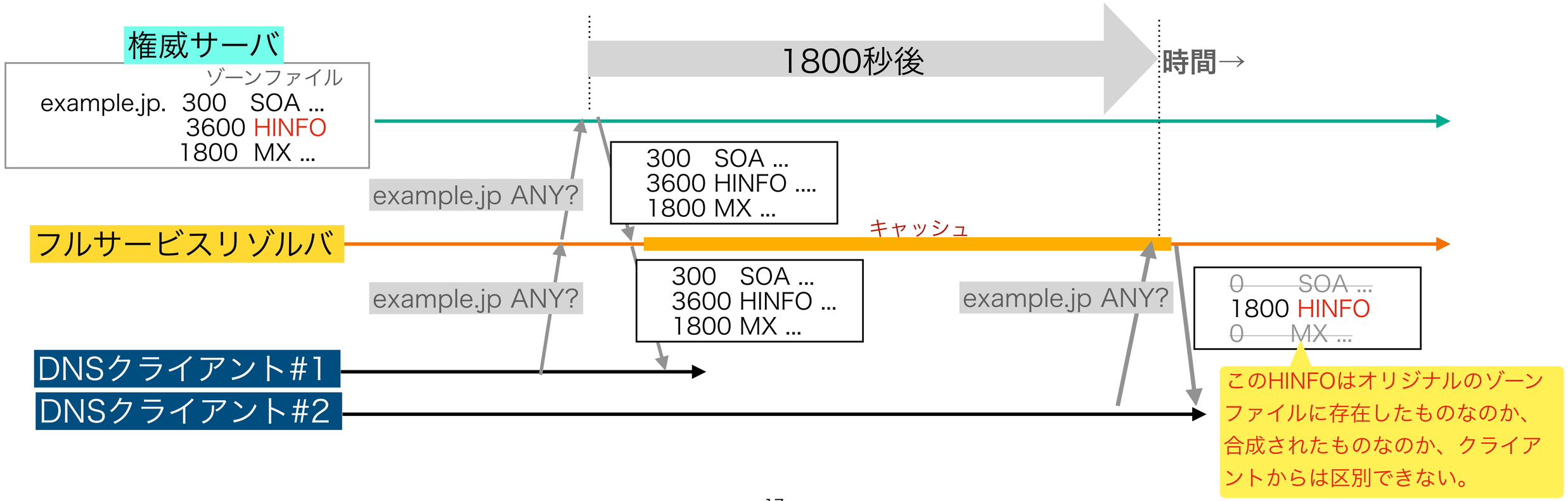
- 実はRFC 8482以前でも、フルサービスリゾルバにANYクエリで問い合わせても全部のRRSetを得られる保証は無かった
  - フルサービスリゾルバはその時点でキャッシュしている (=TTL満了していない)RRSetを応答するため
- DNSクライアントから見れば、RRSetが1個しか得られないのは、権威サーバが1個しか返していないためなのか、他のRRSetがキャッシュでexpireして1個なのか区別ができない→実質的に問題なし、とされる
  - qmailもMX/Aレコードを得るために初回はANYクエリを利用するが、それでMXが得られなければA/MX指定の通常クエリを使用するので問題ない（らしい）



「HINFOレコード」を勝手に合成して応答して問題ないのか？

- ANYクエリでは全てのRRSetを得られないため、
  - HINFOがオリジナルのゾーンファイルに存在したものなのか
  - 勝手に合成されたものか

DNSクライアントからはやはり区別できない→実質的に問題なし、とされる



RFC 8482のDNSSEC考慮は？

- 署名済ゾーンからRFC8482応答する場合は、RRSetに対応するRRSIGを添付することでDNSSEC検証は問題なく行われる
- なお、DNSSECでは、バリデータが「ANYクエリに対してownerの全RRSetが応答されているか」をチェックすることは義務付けられていない
  - NSEC/NSEC3を利用した（高コストな）チェックは不可能ではないが、それを実行するバリデータ実装は存在しないと考えられる