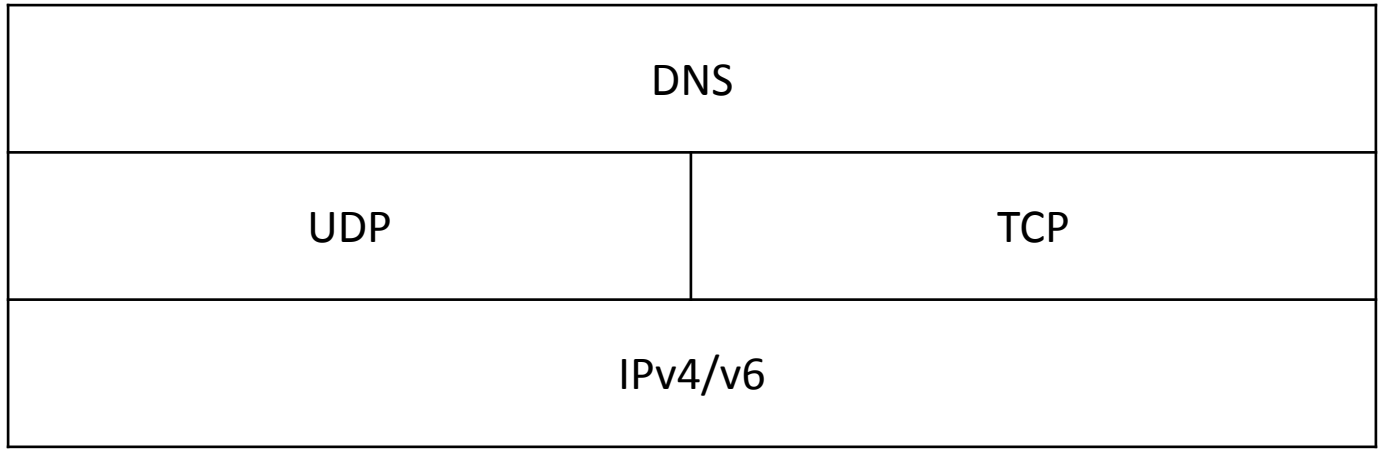
DoH/DoT入門

IIJ 山口崇徳

Internet Week 2019

DNS DAY

traditional DNS



DNS over UDP

- 基本のキ
- 複雑なネゴシエーション不要
  - ステートレス
  - パケット1往復で完了
- サイズ制限
  - 512バイト上限 → EDNS0 により緩和
  - IP fragment の問題
- パケット偽造されやすい
  - キャッシュポイズニング (IP アドレス、ポート番号、query id の偽造)
  - DNS amp (IP アドレスの偽造)
  - fragmentation attack (フラグメントしたパケットの2番目を偽造)

DNS over TCP

- RFC1123
  - UDP のサイズ制限を超える場合にかぎって TCP fallback
- RFC5966、RFC7766
  - UDP からの fallback でなくても TCP を使ってよい
  - が、現在でも fallback 以外で使われることはほぼない
- 3 way handshake
  - UDP よりやりとりが多い
  - UDP より偽造されにくい
- query pipelining
  - 1回の TCP 接続で複数クエリを送る
  - クエリを投げた順に応答が返ってくるとはかぎらない

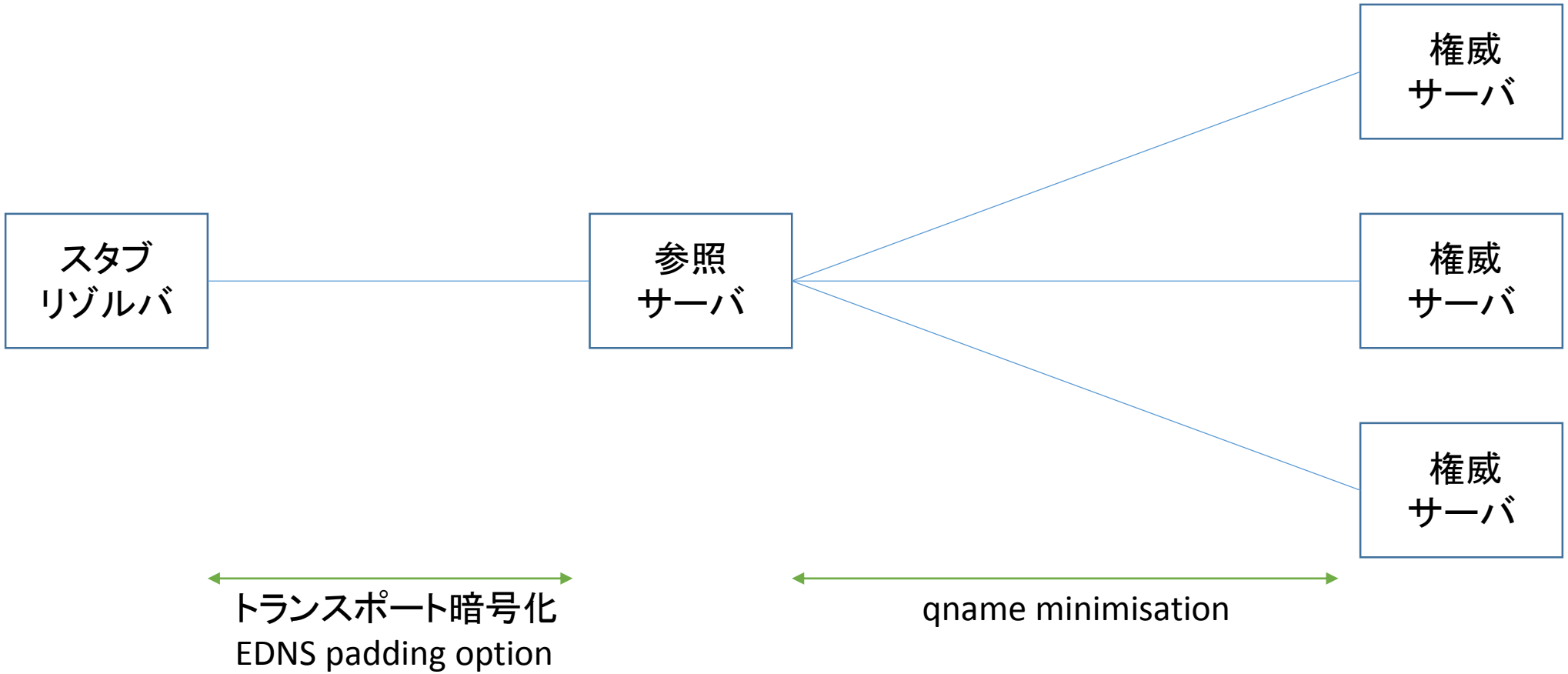
DNS とプライバシー (1)

- 昔: DNS は公開情報
  - 盗聴されないことよりも改竄されないことを重視 ⇒ DNSSEC
- スノーデン事件(2013)
  - DNS に関しても広範に監視がおこなわれていたことが発覚
  - RFC7258 Pervasive Monitoring is an Attack
- 公開情報とはいえ、どんな情報を欲しがっているのかは個人のプライバシー
  - 完全性だけでなく、機密性ももっと考慮しないといけないのでは?

DNS とプライバシー (2)

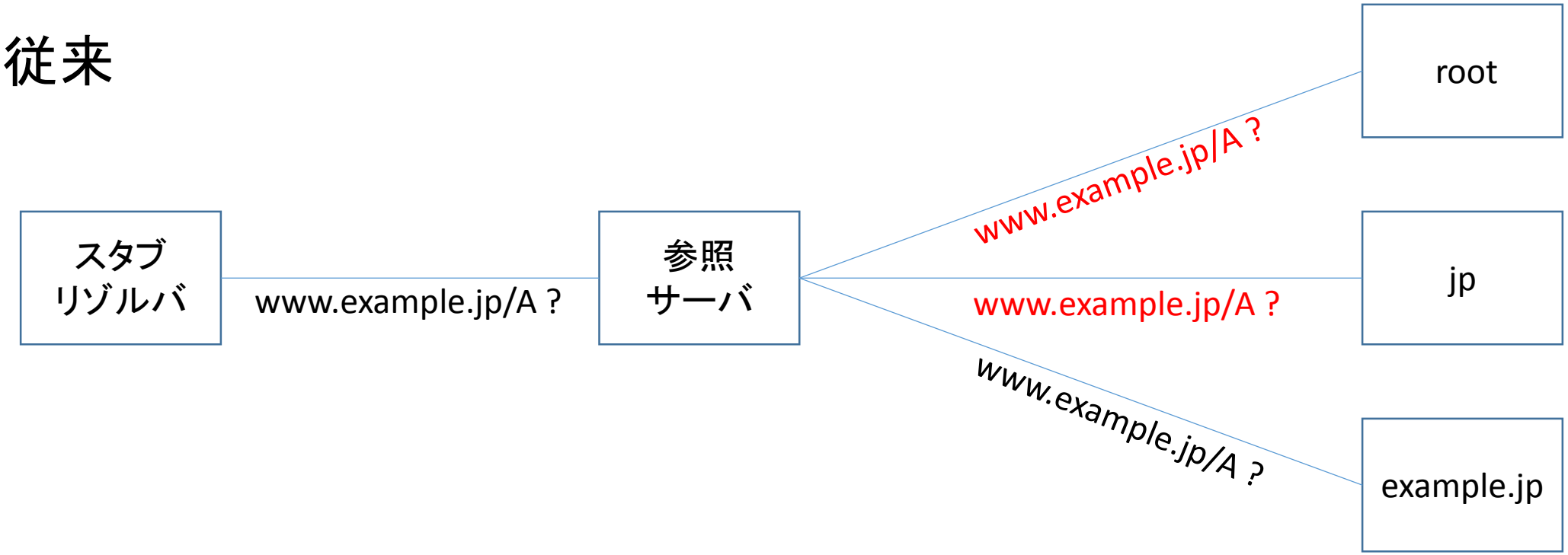
- IETF dprive (DNS PRIVate Exchange) WG (2014)
  - DNS に機密保持機能を追加することをミッションとする WG
  - https://datatracker.ietf.org/wg/dprive/
  - RFC7626 DNS Privacy Considerations
- さまざまなプロトコル拡張・修正
  - Qname minimisation (RFC7816)
  - EDNS(0) padding option (RFC7830、RFC8467)
  - DNS トランスポートの暗号化
  - など

DNS プライバシーのスコープ



Qname minimisation (1)

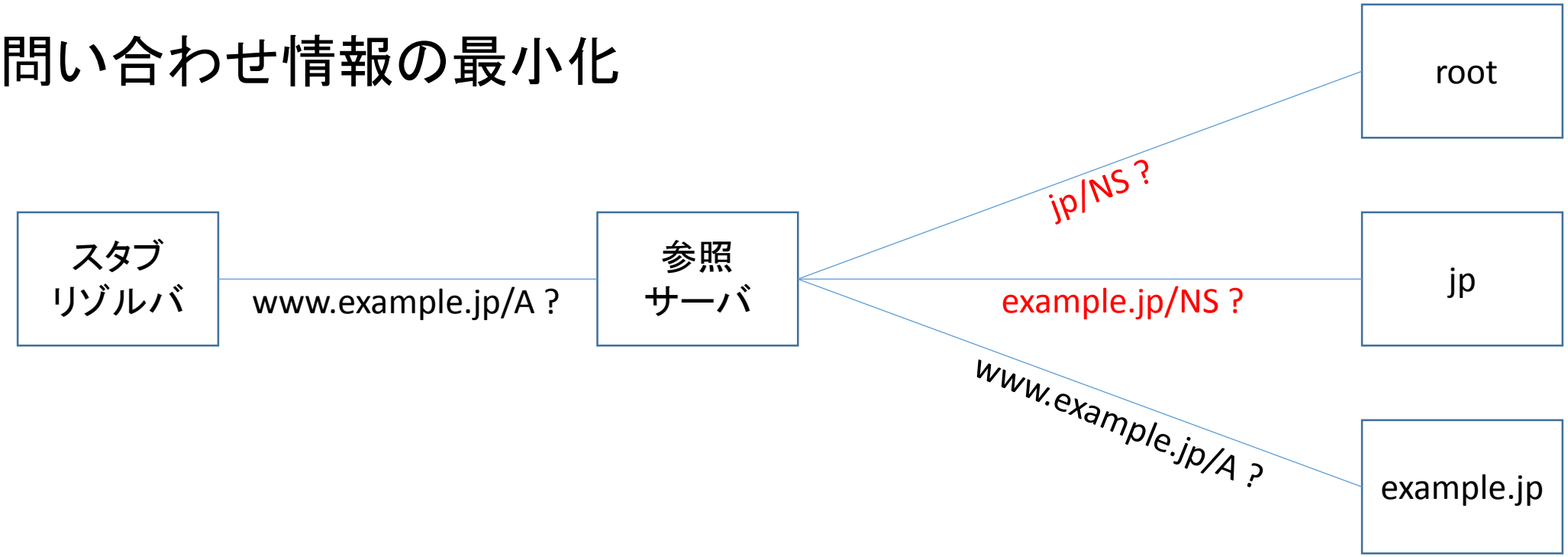
- 従来



- ルートサーバや jp. の権威サーバ、途中経路上の監視者は www.example.jp の情報を欲しがっているユーザの存在を知ることができる

Qname minimisation (2)

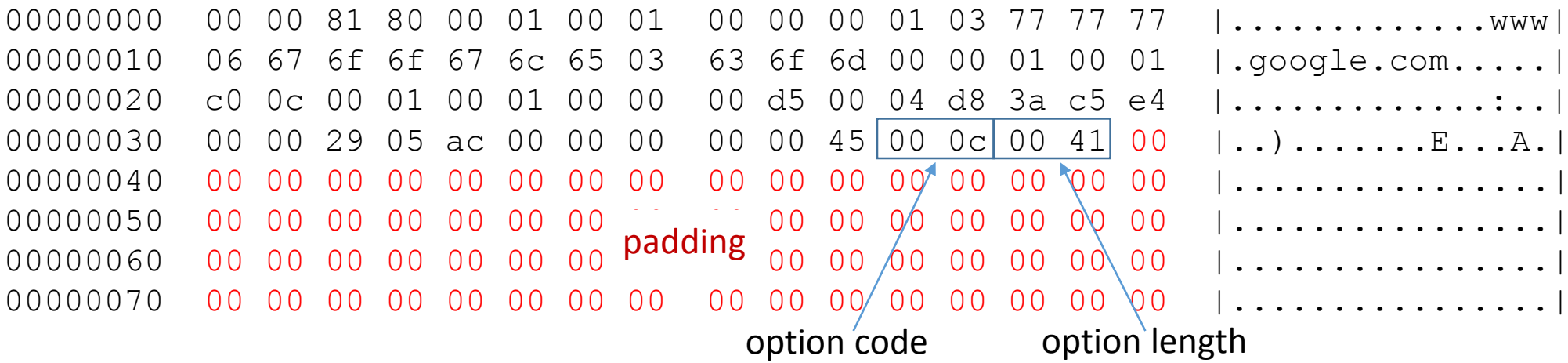
- 問い合わせ情報の最小化



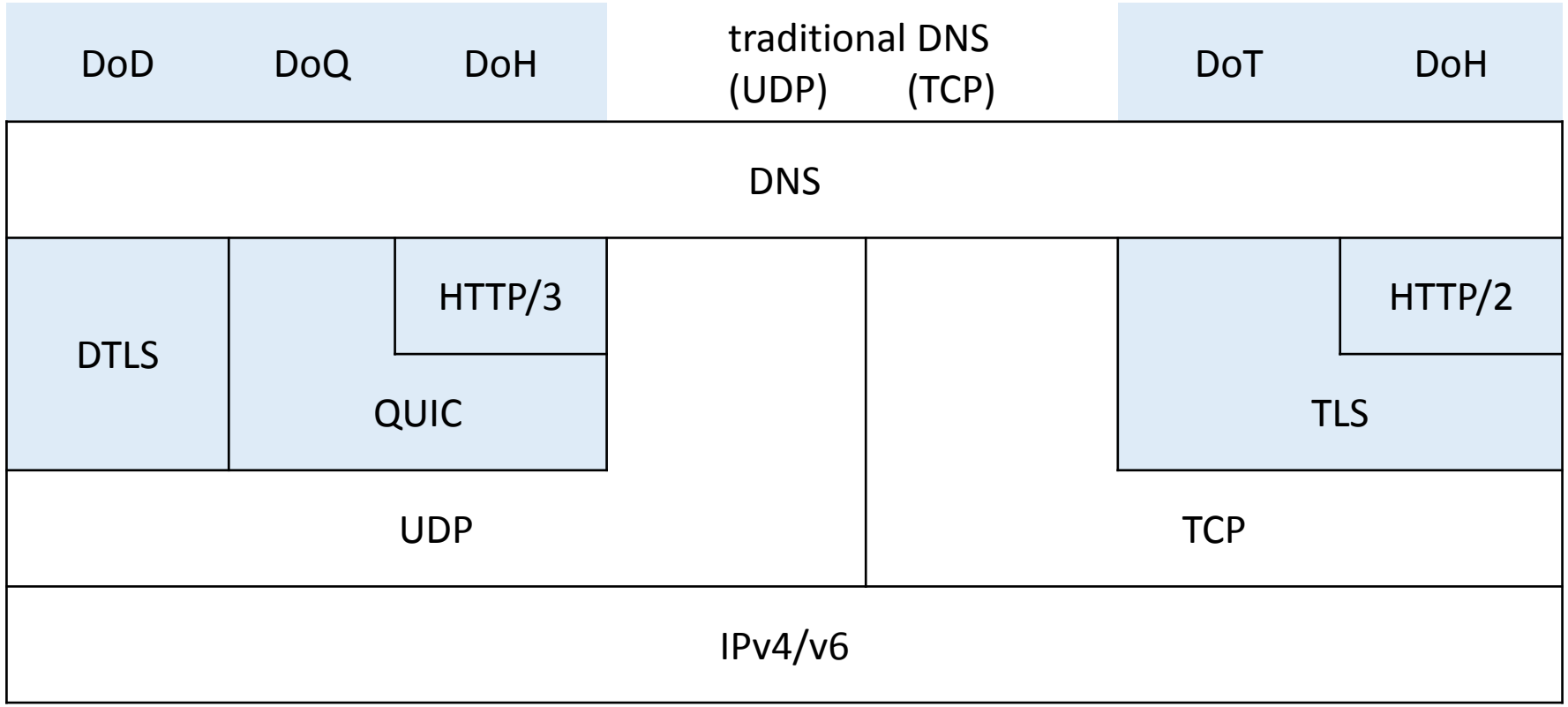
- 参照 - 権威間で www.example.jp の問い合わせ情報が必要以上に露出しない

EDNS(0) padding option

- DNS message は可変長
- が、問い合わせ/応答のサイズは問い合わせ内容によりほぼ固定
  - 暗号化されていても、サイズが推測の手がかりになる可能性
- ダミーデータを padding することでサイズからの推測を防止
  - トランスポート暗号化と併用



トランスポート暗号化


DoD
DoQ
DoH
traditional DNS
(UDP)
(TCP)
DoT
DoH
DNS
DTLS
HTTP/3
QUIC
HTTP/2
TLS
UDP
TCP
IPv4/v6

DNS over (D)TLS

- RFC7858 (TLS)、RFC8094 (DTLS)
- 853/tcp (TLS)、853/udp (DTLS)
  - 暗号化されていてもポート番号を見れば DoT ということはわかる
- DNS の TCP/UDP wire format をそのまま TLS/DTLS セッションに載せたもの
- SMTP の STARTTLS や HTTP の Upgrade のような、平文ポートに接続してから TLS に移行する仕組み(opportunistic encryption)は存在しない
- OS全体にかかわる設定で使われるケースが多い

DNS over HTTPS (1)

- RFC8484
- DNS の UDP wire format をそのまま HTTPS に載せたもの
  - content-type: application/dns-message
- HTTP(S) ではなく、HTTPS
  - TLS はオプションではなく、必須
- HTTP/1.1 非推奨
- server push 可
  - 対応してる実装があるかどうかは未確認

DNS over HTTPS (2)

- request
  - 通常は POST だが GET でも可

```
:method: POST
:path: /dns-query
accept: application/dns-message
content-type: application/dns-message

(DNS wire format)
```

```
:method: GET
:path: /dns-query?dns=(base64(DNS wire format))
accept: application/dns-message
```

- response
  - application/dns-message で wire format を応答
  - それ以外の content-type による応答も許容
    - accept ヘッダでコンテンツネゴシエーション
    - RFC8484 で定義されてるのは application/dns-message だけ

もうひとつの DNS over HTTPS

- google 独自形式
  - https://developers.google.com/speed/public-dns/docs/dns-over-https
  - cloudflare でも使えたりする
- GET の query string に問い合わせ内容、JSON で応答
  - 既存の DNS ライブラリ不要で、HTTPS アクセスと JSON の解釈ができればよいので、用途によってはこっちのほうが便利かも
  - RFC8484 以前からあったので、こっちを DoH と誤解してる人も多そう
- google は6月から RFC8484 方式の DoH にも対応

その他のトランスポート暗号化

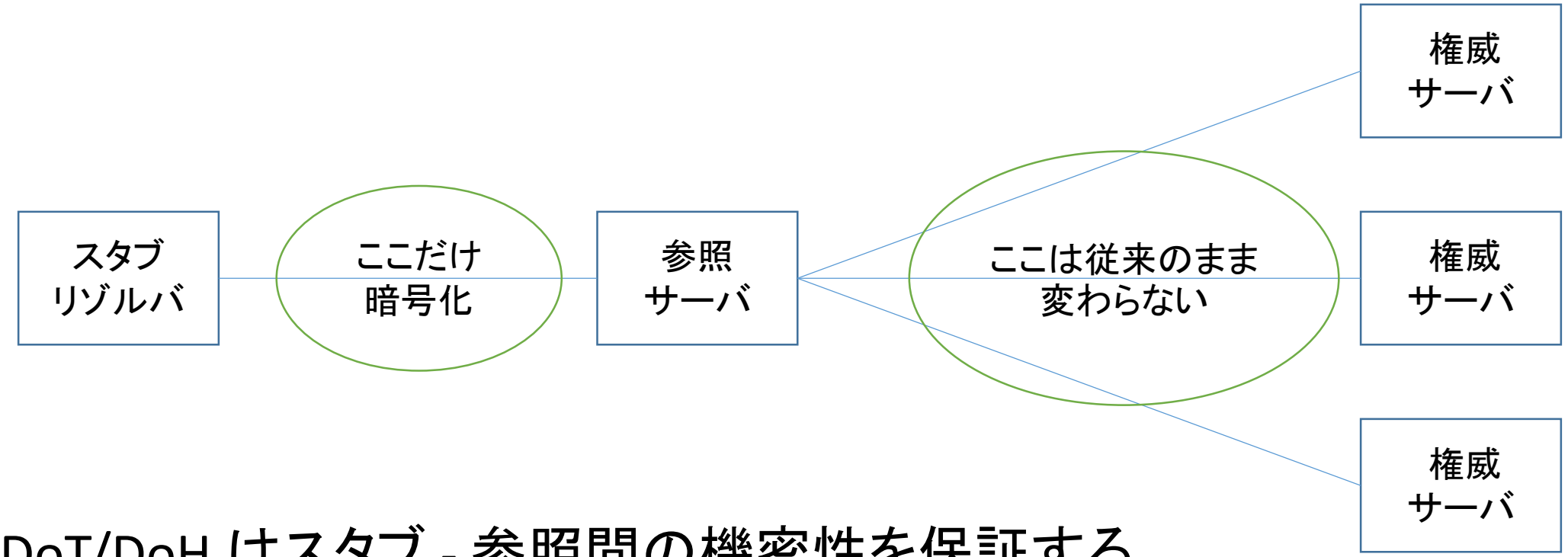
- DNS over QUIC (DoQ)
  - https://tools.ietf.org/html/draft-huitema-quic-dnsoquic-07
  - 784/udp (正式に IANA から割り当てられたわけではない)
- XFR over TLS (XoT)
  - https://tools.ietf.org/html/draft-ietf-dprive-xfr-over-tls-00
  - 権威サーバ間のゾーン転送の TLS 化
- 非 TLS による暗号化
  - DNSCurve: 参照 - 権威間
  - DNSCrypt: スタブ - 参照間
  - いずれも IETF で標準化議論する方向には興味なさげ

traditional DNS との違い

- トランスポート層のプロトコルが変わっただけ
- DNS そのものの syntax、semantics は変更されていない

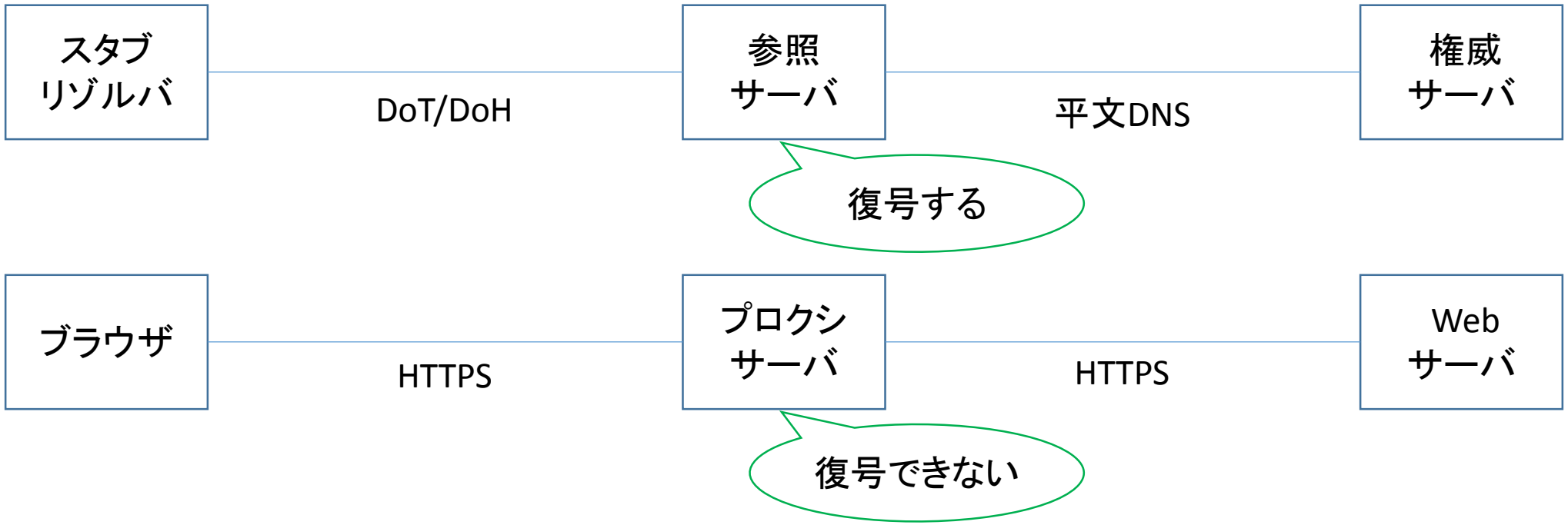
- ただし DoH では
  - server push 可能、すなわちクエリされてないレスポンスを返すことができる
    - A レコードを聞かれたら、聞かれなくても AAAA の応答も push しちゃう、とか
  - query id はランダムではなく 0 固定にすべき(SHOULD)
    - HTTP キャッシュを有効活用できるようにするため、とのこと
    - HTTP proxy ではなく、スタブレゾルバ(Webブラウザ)でのキャッシュと思われ

トランスポート暗号化のスコープ



- DoT/DoH はスタブ - 参照間の機密性を保証する
  - (DNSSEC なしなら)参照サーバの持つ情報が正しいという保証がない
  - 一般に TLS は完全性も保証するが、権威まで含めた DNS の系全体で考えれば DoT/DoH に完全性があるとはいえない

HTTP over TLS (aka HTTPS) との違い



- DoT/DoH は参照サーバで復号されて平文になる
- HTTPS はプロクシサーバで復号できず宛先ホスト名とポート番号しかわからない

DNSSEC vs DoT/DoH

- DNSSEC = DNS SECurity extensions
  - 一言でいうと、「電子署名つき DNS」
  - 暗号化しない = 機密性なし

- スコープが異なる
  - DNSSEC ⇒ 完全性の保証、否認防止
  - DoT/DoH ⇒ 機密性の保証

- DNSSEC が守るもの ≠ DoT/DoH が守るもの
  - DoT/DoH があるから DNSSEC なんかいらない、とはならない(vice versa)

DoT vs DoH

- HTTP のレイヤが必要ない分だけ DoT の方がシンプル
  - そのかわり HTTP/2 の再送制御などの恩恵はない
- DoT は専用ポート(853/tcp)を使う
  - 通信の中身がわからなくても、DoT を使ってることはわかる
- DoH は他の HTTPS の通信に紛れて見つけづらい
- 現状のおおまかな使い分け
  - DoT: OS 全体の名前解決
  - DoH: アプリケーション(Webブラウザ)独自の名前解決

DoT/DoH bootstrap

- ネットワークにつなぐと traditional DNS が自動設定される
  - IP アドレスなどの自動設定プロトコルに DNS サーバを指定するオプション
    - DHCP/DHCPv6、IPCP、IPv6 RA
    - ネットワーク内部から悪意の攻撃を受けることには無力なプロトコル

- DoT/DoH を自動設定する仕組みは、ない
  - ネットワーク管理者が多数のクライアントを一斉に DoT/DoH に対応させるようなことができない
  - 標準化に向けた議論は始まっている
    - 安全でないプロトコルで入手した情報を「とりあえず信じてみる」方向になる(?)

ニワトリとタマゴ問題 (1)

- traditional DNS の設定は IP アドレスで
- DoT/DoH の設定にはホスト名が必要
  - 証明書の CommonName、SubjectAltName がマッチする必要があるので
    - IP アドレス証明書というのもなくはないけど…
  - Android9 の設定 UI は IP アドレスを入力すると保存ボタンを押せない
- じゃあ、そのホスト名の名前解決はどうやるの?

ニワトリとタマゴ問題 (2)

- DoT/DoH サーバの名前解決のときだけ traditional DNS を使う実装が多い
  - つまり、traditional DNS は今後もなくならない(なくせない)
  - 複数の IP アドレスをひとつのホスト名で集約できるというメリットも

- その他
  - IP アドレスとホスト名のペアで設定するもの
  - IP アドレスと証明書の pin-sha256 のペアで設定するもの

参照 - 権威サーバ間の暗号化 (1)

- 実現させる方向で議論は始まったっぽい
  - 現時点で I-D は何本か出ているが、DPRIVE WG としてではなく、個人ドラフト
  - Authoritative DNS-over-TLS Operational Considerations
    - https://tools.ietf.org/html/draft-hal-adot-operational-considerations-02
  - DNS Privacy Requirements for Exchanges between Recursive Resolvers and Authoritative Servers
    - https://tools.ietf.org/html/draft-lmo-dprive-phase2-requirements-01

- で、ほんとに実現できるの?

参照 - 権威サーバ間の暗号化 (2)

- 参照サーバは権威サーバが暗号化対応していることをどうやって知ればいいのか?
  - たとえば、「まず TLS で試して対応してなかったら平文にフォールバック」という方式だと、MITM で TLS ハンドシェイクを失敗させることで、実際には TLS 対応していても平文にさせられてしまう
  - Signaling That an Authoritative DNS server offers DoT
    - https://tools.ietf.org/html/draft-levine-dprive-signal-02
- あるゾーンの権威サーバが暗号化対応しているとき、その上位ゾーンが暗号化非対応なケースを許容するか?
  - 上位の権威サーバからの委任応答を MITM で改竄することで、暗号化されていない偽サーバにクエリを誘導できてしまう
  - 結局完全性も機密性も保証できない

DoT/DoH のユースケース

- 自動設定される参照 DNS が信頼できないとき、かわりに public DNS を利用する
  - 公衆 wifi とかホテルの客室インターネットとか
  - 参照 DNS が意に反するブロッキングをやってるとか
- public DNS への途中経路での中間者攻撃を回避 → DoT/DoH

- 将来的には、使い分けせずぜんぶ暗号化という方向になる?

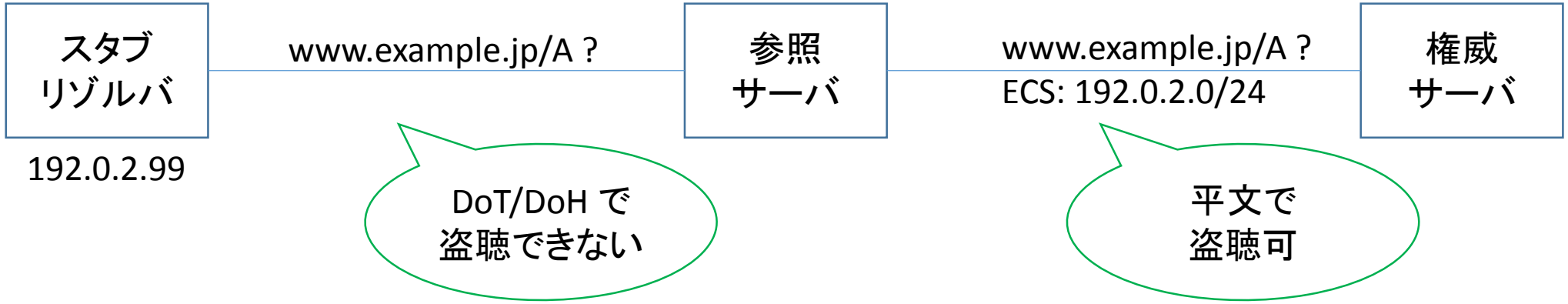
EDNS client subnet (1)

- CDN 屋さんはユーザにもっとも近いサーバから配信したい
  - 名前解決してきた参照サーバのアドレス(≒利用している ISP)からユーザの所在を推定
- CDN 屋さんとしてはもっと高い精度でユーザの所在を把握したい

- RFC7871 EDNS Client Subnet (ECS)
  - 参照サーバと権威サーバの間の通信にクライアントの IP アドレス(を /24 程度に丸めたもの)を付加情報として追加
  - 権威サーバはそれを参考に適切な配信サーバを応答する

EDNS client subnet (2)

- google public DNS は DoT/DoH に対応している
- google public DNS は ECS にも対応している
- google public DNS を DoT/DoH で利用すれば、ユーザのプライバシーは保護されるといえるのだろうか?



DoT/DoH でユーザトラッキング

- traditional DNS
  - ユーザを追跡できる情報は IP アドレスだけ
  - IPv4: CGN (carrier grade NAT) の普及 / IPv6: temporary address
- DoT/DoH
  - IP アドレスが変わっても TLS セッションで追跡できる可能性
- DoH
  - HTTP のレイヤで追跡できる可能性
    - RFC8484 では cookie その他のフィンガープリント情報は使うべきではないとされている
    - が、禁止まではしていない
- DoT/DoH では参照サーバの得られるユーザ情報が増えている

DoT/DoH でプライバシーは守れるのか (1)

- DoT/DoH で経路上の監視者から DNS を盗聴される危険が減る
  - 権力による検閲からの回避
  - マルウェアにとっても活動が隠蔽されて都合がいい

- 参照サーバはすべてのクエリを知ることができる
  - DoT/DoH なクエリも復号できる
  - クエリを集計して別の用途に利用することも、応答を書き換えることも(やろうと思えば)できる
  - DoT/DoH ならではの話ではなく、traditional DNS でも同じ
    - が、traditional DNS では機密性は元からスコープ外だった

DoT/DoH でプライバシーは守れるのか (2)

- 膨大なプライバシーが参照サーバ運用者の手元に
  - 現状では DoT/DoH に対応しているのは public DNS 以外にほとんどない
  - DoT/DoH への移行 = public DNS への集中化・寡占化

- public DNS に集積されたプライバシーは捨てられるのか活用されるのか? 漏洩の懸念は?
  - ほとんどの public DNS は日本の事業者ではない
  - 日本では法律上できないことも海外事業者はできてしまう(ことがある)
  - public DNS のポリシーしだい

まとめ

- トランスポート暗号化のスコープ
  - ○ 機密性　　○ スタブリゾルバ - 参照サーバ間
  - × 完全性　　× 参照サーバ - 権威サーバ間

- 守備範囲はそんなに広くない
- 限界をわきまえて使いましょう

- たまには DNSSEC のことも思い出してあげてください