DNS入門

DoT/DoHに関する入門と動向編

InternetWeek 2020 DNS DAY

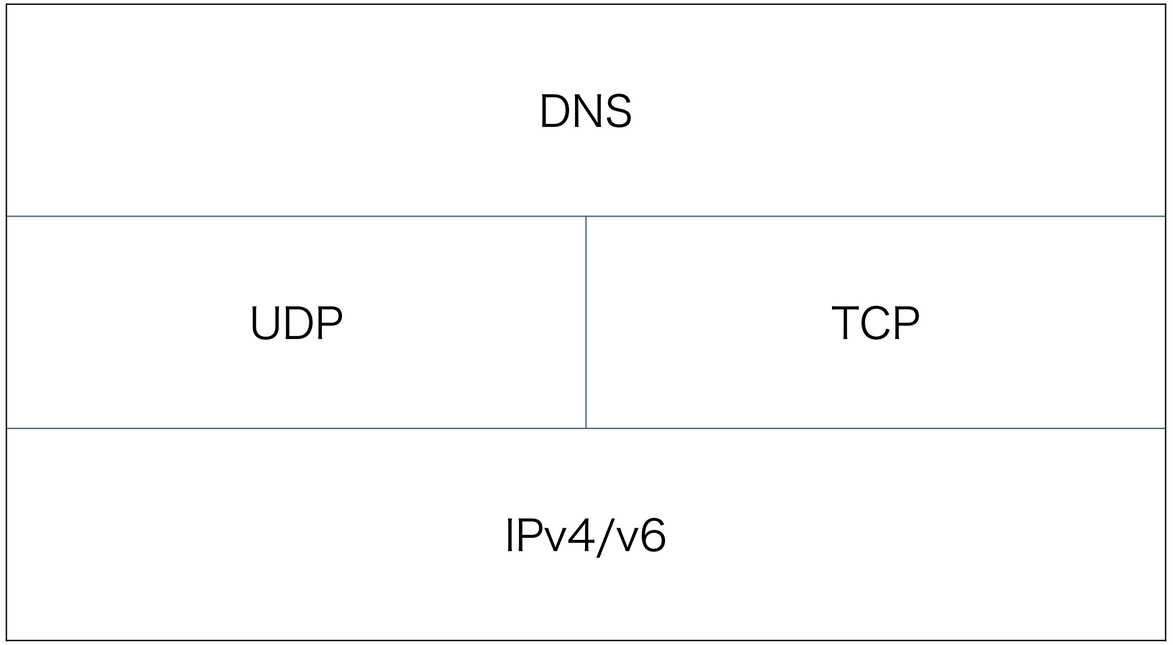
山口崇徳@IIJ

はじめに

- English speakerは「でぃーおーえいち」「でぃーおーてぃー」ではなく、「どー」「どっと」と発音するようです
  - が、日本人にとっては「でぃーおーえっち」のほうがわかりやすいと思うので、今回もそう喋ります

- ここ数年ほど、DoH/DoTについて発表させてもらう機会が多いし、会社でもいろいろ始めてるけれど、個人的には推進派のつもりはありません

従来のDNS
DNS over UDP/TCP (Do53)



従来のDNS
DNS over UDP/TCP (Do53)

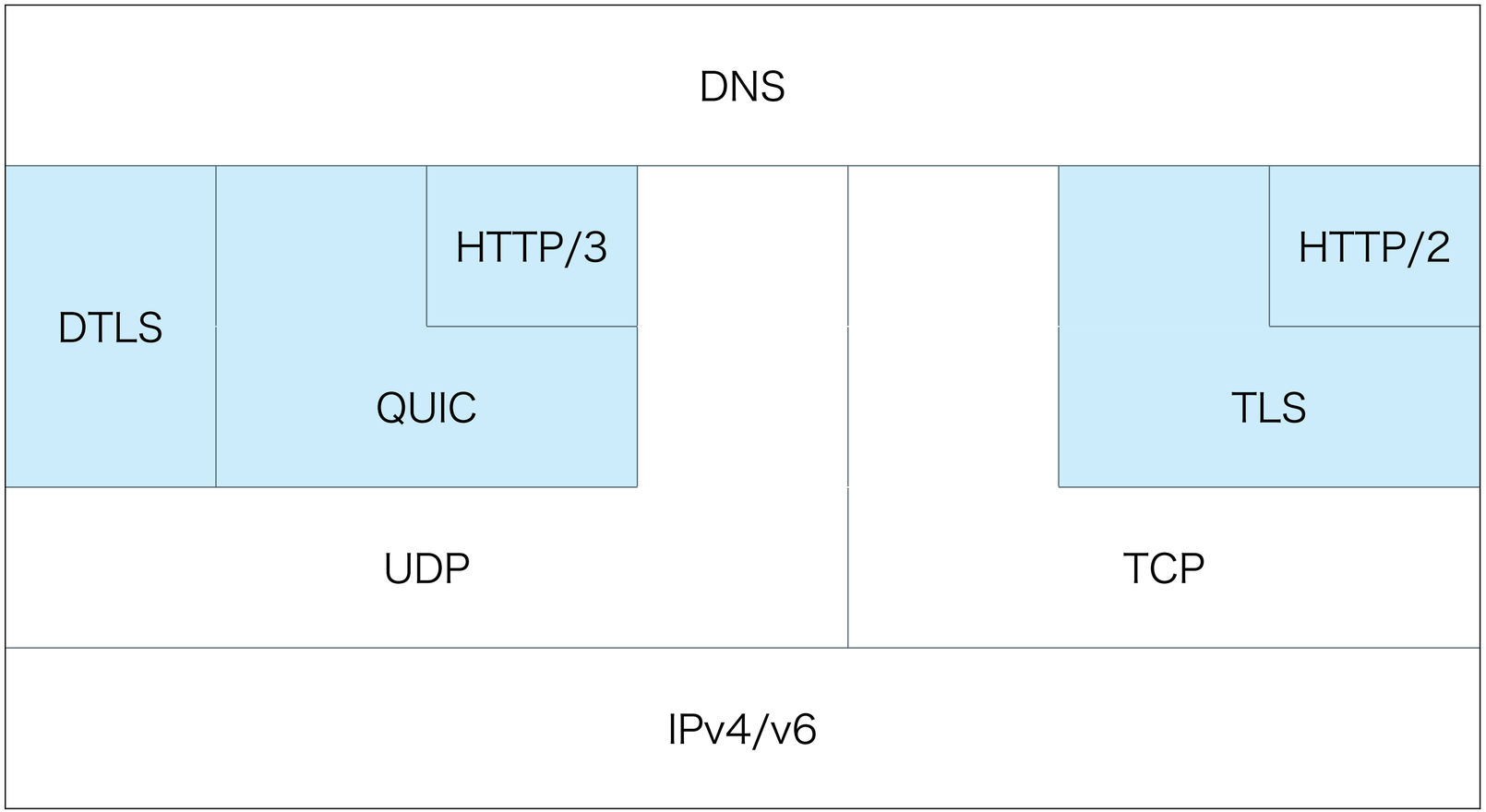
- 基本的にUDP
  - 512バイト制限 → EDNS0で緩和
  - 複雑なハンドシェイクが不要なかわりにパケット偽造に弱い
    - キャッシュポイズニング、DNS amp、fragmentation attack
- ときどきTCP
  - UDPより偽造されにくいがハンドシェイクが必要
- いずれにせよ、平文
  - 経路上での改竄や盗聴に弱い

DNSとプライバシー

- 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についても広範に監視さ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
  - 改竄されないことだけでなく、盗聴されないことも考慮されるべき → 暗号化
  - RFC7626 DNS Privacy Considerations

トランスポート暗号化

DNS over TLS/HTTPS/…



DNS over TLS/DTLS/QUIC

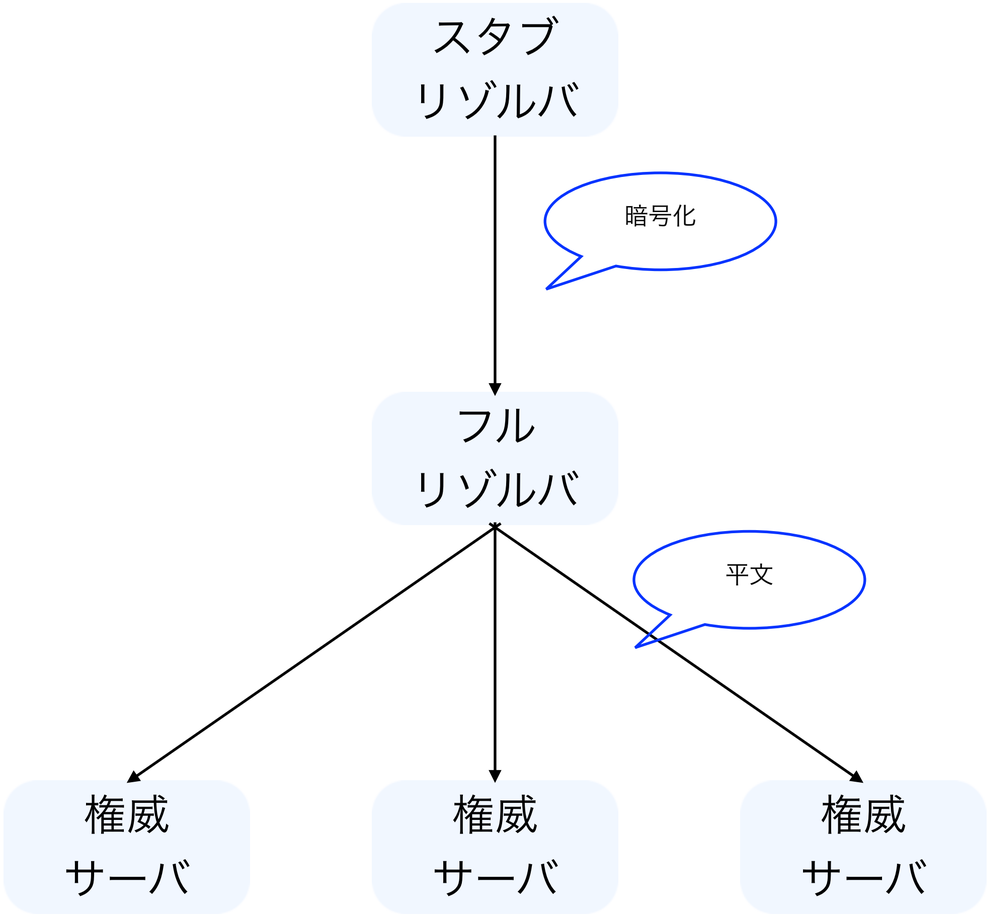
- DNSのTCP/UDP wire formatをそのままTLS/DTLS, QUICに載せたもの
  - TLS: RFC7858, 853/tcp
  - DTLS: RFC8094, 853/udp
  - QUIC: draft-ietf-dprive-dnsoquic, 784/udp (仮)
- それぞれ専用ポート番号を持つ
  - 暗号化されていても、ポート番号を見ればDNSのやりとりということはわかる
- SMTPのSTARTTLSのような、平文ポートでの接続中にTLSに移行する仕組みは存在しない

DNS over HTTPS

- DNSのUDP wire formatをそのままHTTPSに載せたもの
  - RFC8484
  - content-type: application/dns-message
  - 通常はPOSTメソッドでクエリを送るが、GETでも可
    - GETでは query string に dns=(base64(DNS wire format)) を付加
- 平文HTTPでの利用は禁止
- HTTP/1.1非推奨

暗号化DNSのスコープ

- 現状はスタブ - フルリゾルバ間だけ暗号化
  - フルリゾルバ - 権威間は平文のまま
- (DNSSECなしなら)フルリゾルバの持っている情報が正しいという保証がない
  - 完全性のない情報を暗号化しても完全性は得られない
- スタブ - フルリゾルバ間の機密性の保証のみ
  - キャッシュポイズニング対策にはならない



DNSSEC vs DoH/DoT

- 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
  - DoH/DoT: 機密性の保証
- DNSSECが守るもの ≠ DoH/DoTが守るもの
  - DoH/DoTがあるから DNSSECなんかいらない、とはならない(逆もまた然り)

DoH/DoTで変わること(1)

- 暗号化っていうか、それ以前にTCP化
  - ステートレス → ステートフル
  - kaminsky attackが発見されたときですらUDPにこだわったDNS業界なのに!
- TLS化
  - DNSメッセージのサイズはもともと大きくない(数百バイト以下)
  - 相対的に、暗号化よりもハンドシェイクや鍵交換のオーバーヘッドが大きくなる
- インシデント/障害時の調査が困難
  - パケットキャプチャによる調査は絶望的

DoH/DoTで変わること(2)

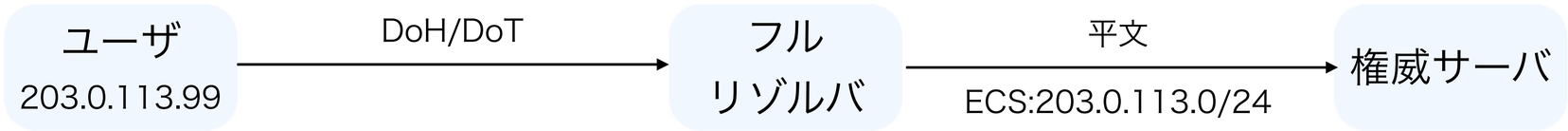
- ユーザのDNSのやりとりが外部から盗聴できなくなる
  - マルウェアによる名前解決も暗号化されて検知不能に
  - ただし、DoH/DoTサーバ自身は復号可能
- DNSによるフィルタリング/ブロッキングがやりづらくなる
  - ファイアウォールなどでの遮断は不可能に
  - ただし、DoH/DoTサーバ自身によるフィルタリング/ブロッキングは可能
- ブラウザがOSの設定と異なるフルリゾルバ設定を持つ
  - DNSによるフィルタリングの回避

DoH/DoTで変わらないこと

- あくまでトランスポートプロトコルが変わっただけ
  - フルリゾルバ - 権威間はトランスポートも変わらない
- DNSメッセージそのもののsyntax、semanticsは変更なし

EDNS Client Subnet
RFC7871

- CDN事業者がユーザに最も近いエッジノードからコンテンツを配信できるようにするためのEDNS拡張
  - フルリゾルバから権威サーバへのクエリにクライアントのIPアドレス(を/24程度に丸めたもの)を拡張情報として追加
  - 権威サーバはそれを参考に動的に応答を変更する



- せっかく暗号化したのに、ECS情報が載る経路が平文でユーザのプライバシーを守ったことになるの?

public DNSとDoH/DoT

- ISPがユーザに対してDoH/DoTサービスを提供するメリットがない
  - ISP網内の通信は平文でも外部からの盗聴の危険は小さい
  - 仮にISPがDoH/DoTサービスを提供したとしても、自動設定の仕組みがなく、Same-Provider Auto-Upgradeも機能しないため、ほとんど利用されない
- つまり、DoH/DoTは(現状では)ほぼpublic DNS専用プロトコル
  - プライバシーの懸念 https://www.janog.gr.jp/meeting/janog45/program/publicdns
  - CDNの配信効率への影響 https://www.janog.gr.jp/meeting/janog46/ecs/

DoH/DoT のパフォーマンス

- DNSのレイテンシはパケットの往復時間が支配的
  - 1往復の時間を小さくするか、往復回数自体を減らすと速くなる
- 原理的には、ハンドシェイク不要でパケット1往復で完結するUDPと、TCP/TLSのハンドシェイクが必要な暗号化では数倍の違い
- が、Webページのロード時間で見れば暗号化してもほとんど変わらない
  - https://www.cs.princeton.edu/~ahounsel/publications/www20.pdf
  - https://dl.acm.org/doi/pdf/10.1145/3355369.3355575
  - https://www.samknows.com/blog/dns-over-https-performance

各種実装の対応状況

- かなりサーバ側の実装が進んでいる

- ただし、実際に使うかどうかは別問題
  - TLS/HTTPS化以前に、UDP → TCPという時点で特大のインパクト
  - ただインストールして設定して終わり、ではなく、ネットワークやサーバの構成から根本的に設計しなおす必要あり

BIND

- 開発バージョンの9.17で対応 → 9.16にもバックポートする予定、らしい
  - https://gitlab.isc.org/isc-projects/bind9/-/wikis/BIND-9.17-Plan
  - https://gitlab.isc.org/isc-projects/bind9/-/wikis/DoH/DOH-and-DoT-Design
- が、9.17.6リリース時点で未対応
  - 現時点のマイルストーンでは9.17.8 (2020/12)が目標らしいが…

Unbound

- 1.12.0 (最新版)でDoHに対応
  - ./configure —with-libnghttp2
- DoTもRFC7858よりはるか以前から対応済み

Knot Resolver

- 4.0.0 (2019/04)でDoHサポート
- DoTもかなり初期からサポート

- ResolverじゃないほうのKnot DNSは権威なのでDoH/DoT非対応
  - kdig (knot版dig)がDoH/DoTを喋れるので、テスト用におすすめ

```
% kdig @public.dns.iij.jp +https=/dns-query www.iij.ad.jp (DoH)

% kdig @public.dns.iij.jp +tls www.iij.ad.jp (DoT)
```

その他各種サーバ実装

- Windows DNS Server
  - 未対応っぽい
- PowerDNS Recursor
  - 未対応
- NSD
  - DoT対応済み(権威なのに…)
- dnsdist
  - DoH/DoTとも対応済み

public DNS

- 主要public DNSプロバイダはほぼDoH/DoTどちらにも対応済み
  - Google, Cloudflare, Quad9,…
  - OpenDNSはDoHのみ
  - IIJ はDoH/DoTのみ(Do53非対応)

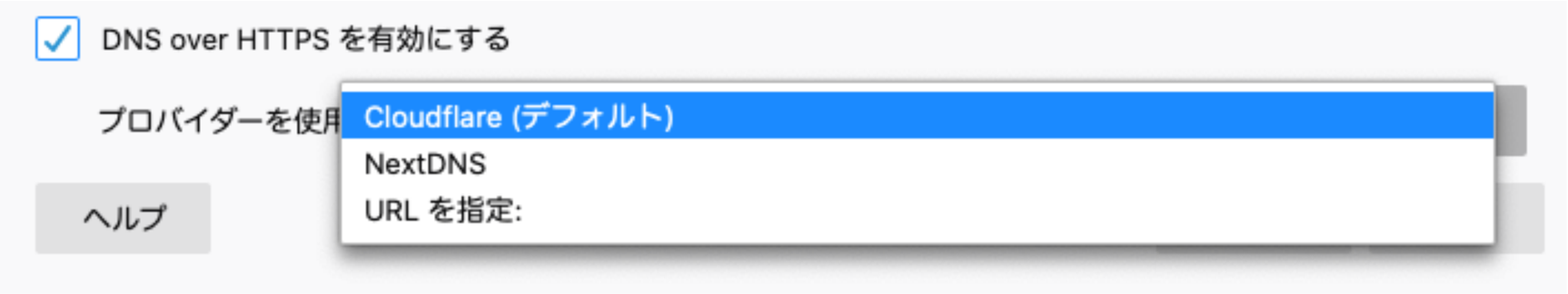
- ASをまたいでアクセスするpublic DNSは、同一AS内に閉じるISPのフルリゾルバよりも盗聴の危険が大きく、DoH/DoT対応のクライアントが出揃ってきた現在は積極的に暗号化すべき

各種実装の対応状況

- クライアント側(ブラウザ、OS)も対応が進んでいる
- 対応方針はまちまち

Firefox

- Firefox62でDoHに対応



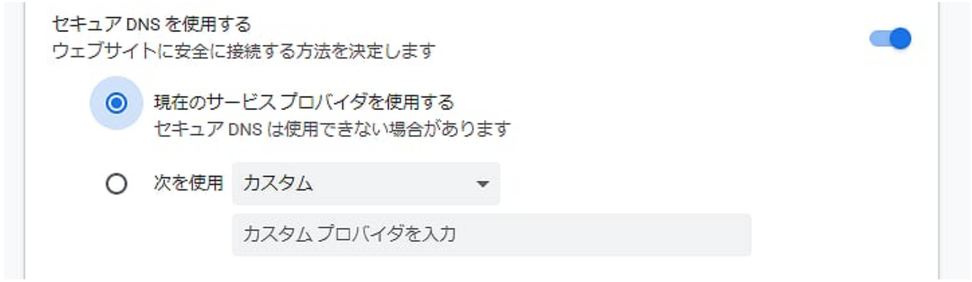
- 一部地域ではデフォルトでOSの設定を無視してお仕着せDoHプロバイダが使われる
  - どうやって地域判定してるのかは不明
  - 特定ドメインの名前解決が失敗する場合はDoH無効に(canary domain)

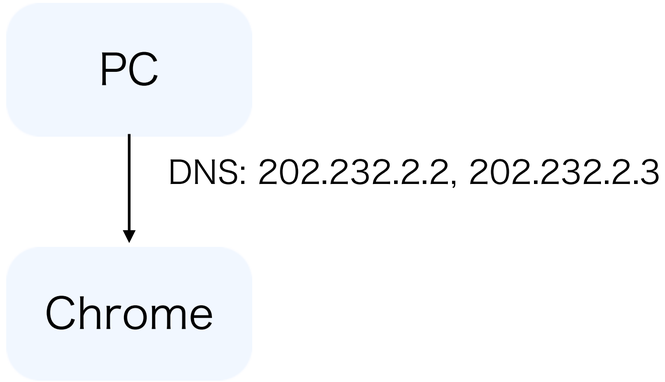
Firefox
なぜ批判されるのか

- DoHを実装したこと自体はほとんど批判されていない
- 「OS設定を無視して上書き設定する」という方針が批判されている
  - たとえば、DNSによるペアレンタルコントロール等が無効になってしまう
- Umbrella (DNSによるドメインフィルタリングサービス、商用版OpenDNS)はcanary domainの名前解決をブロッキングしてFirefoxのDoHが使われないようにしている
  - https://umbrella.cisco.com/blog/doh-dns-over-https-to-block-or-not-to-block

Chrome

- Chrome79で実験的にDoH対応
- Chrome83で正式対応
- デフォルトでは、OSに設定されているフルリゾルバがDoHにも対応している場合、DoHが優先される(Same-Provider Auto-Upgrade)
  - OS設定のリゾルバとDoHサーバは同一ポリシーで運用されている前提なので、Firefoxのような問題はない
  - ただし…

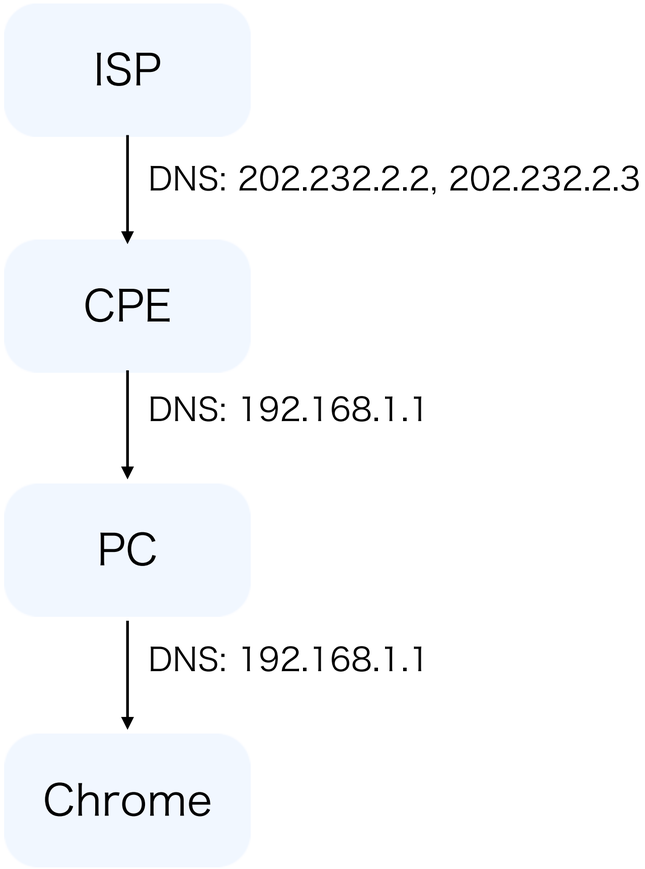




DoH: https://per.dns.iij.jp/

Chrome
Same-Provider Auto-Upgrade

- ほとんどのホームルータはDNS forwarder
  - PCに自動設定されるフルリゾルバは、ISPが配ってるものではなく、ホームルータのプライベートアドレス
  - ホームルータとDoHサーバはsame providerではない
  - ISPがDoHを提供しても、アップグレードされない
- auto upgradeできるフルリゾルバとDoHサーバの対応表は設定では変更できない
  - Chromeのソースを修正してもらう必要がある

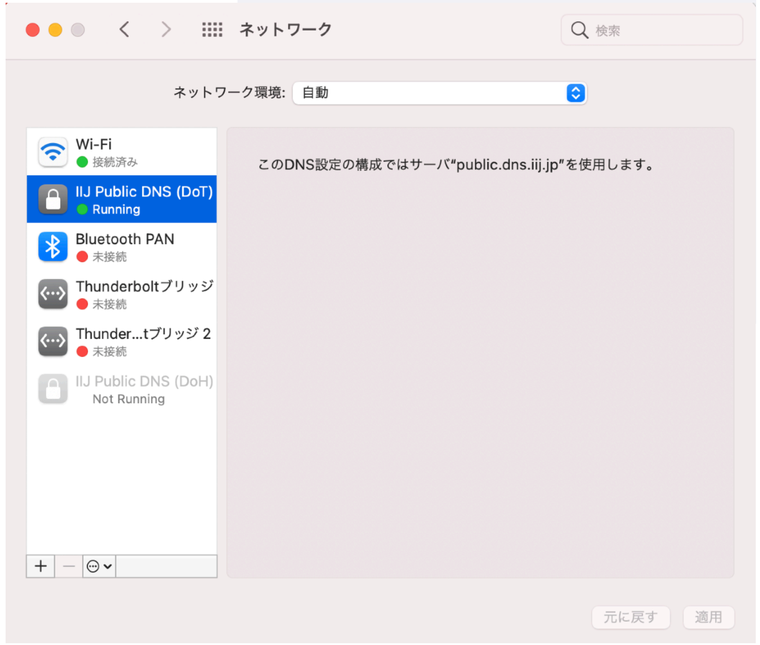


Android

- Android9からDoT対応
  - モードが自動の場合、PPP、DHCP等で自動設定された平文フルリゾルバにDoTでクエリを投げてみて、使えたらそちらを使う
- Intra: ほぼ公式といえるDoHアプリ
  - https://play.google.com/store/apps/details?id=app.intra&hl=ja&gl=US
- Chrome85からAndroid版でもDoHが利用可能に

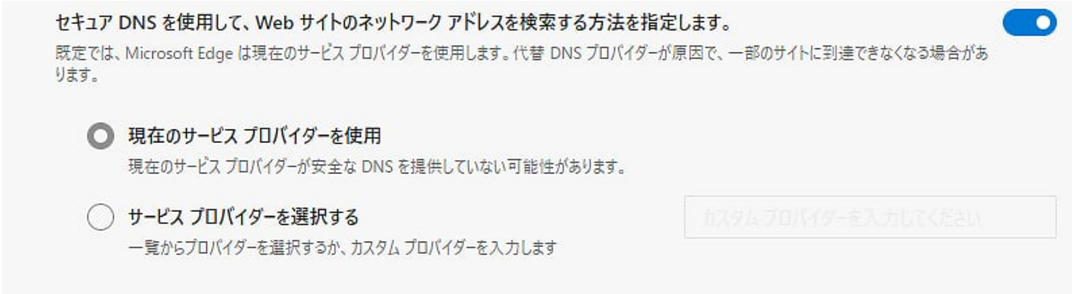
iOS/macOS

- iOS14、macOS11から DoH/DoTに対応
- 設定は手入力ではなく、プロファイルをインポート
  - 同一プロファイルをiOS/macOSで共用可
  - MDMで管理される業務端末では一括設定も可能
- 特定ドメインの名前解決だけ特定のDoH/DoTサーバを利用する/しないといった細かい設定が可能
- OS全体の設定とは別に、アプリが個別にDoH/DoTサーバに名前解決しにいくためのAPIが追加



Windows

- EdgeでDoHが利用可能
  - ベースがChromeなので動作も同じ
  - Same-Provider Auto-Upgrade



- OSとしてのDoHは、Windows10 21H1で対応とのこと
  - 現在Insider Previewでテスト中
  - Chrome/Edgeと同じくSame-Provider Auto-Upgradeを採用
    - IP アドレスとDoH URLの対応表はレジストリに格納される
- DoTは予定なさげ

DoH/DoTに欠けているもの (1)
フルリゾルバ - 権威間の暗号化

- 現状はスタブ - フルリゾルバ間のみ
- IETF dprive WGにおける “Phase 2” として重点的に取り組み
  - DNS Privacy Requirements for Exchanges between Recursive Resolvers and Authoritative Servers
    - https://www.ietf.org/archive/id/draft-ietf-dprive-phase2-requirements-02.txt
  - まだ要件整理の段階で標準化はだいぶ先になりそう…

DoH/DoTに欠けているもの (2)
クライアント設定の自動化

- DHCPやPPPなどで自動設定されるDNSサーバは平文のもの
  - ネットワーク管理者が「このDoHサーバ使ってね」と設定を配布することができない
  - Firefoxがお仕着せ設定をデフォルト有効にしたり、Same-Provider Auto-Upgradeなるものが考案されるのも、自動設定のための標準が存在しないから
- IETF add (Adaptive DNS Discovery) WGで議論
  - ドラフトはたくさん出てるけど、まだまだ時間がかかりそう…
  - https://datatracker.ietf.org/wg/add/documents/

DoH/DoTに欠けているもの (3)

- 権威暗号化と設定自動化、いずれも「サーバが暗号化に対応していることをどうやって安全に伝える/見つけるか」が問題
  - まずDoTを試してダメなら平文にフォールバック → MITMでTLS ハンドシェイクを失敗させれば暗号化対応していても平文になってしまう
  - 暗号化されていることを示す情報を載せる → MITMでその情報が改竄されたら暗号化対応していても平文になってしまう
  - DNSSECは後者(DSレコード)
    - DSレコード自体がDNSSEC署名されるので改竄できない
- 厳密にやるのであれば、標準化と普及にはDNSSEC並の困難が予想される

まとめ

- スタブ - フルリゾルバ間の機密性を保証するだけで、完全性はない
  - 過度な期待は禁物
  - フルリゾルバ - 権威間の暗号化は当分来ない
- OS/ブラウザがDoH/DoTに対応するのは当たり前という状況に
  - あとはWindowsが正式対応するのを待つのみ(来年春)
- ISPのフルリゾルバがDoH/DoTに対応するメリットがないので、実質的にpublic DNS専用プロトコルという状況がしばらく続きそう
  - 自動設定プロトコルが標準化・普及しないとユーザに使ってもらえない