TLP:CLEAR

DNS の基礎の基礎

Keio University
Wataru Ohgai (alt@sfc.wide.ad.jp)

TLP:CLEAR

誰?

大谷 亘(おおがい・わたる) a.k.a. alt

- 慶應義塾大学政策・メディア研究科
- WIDE Project メンバー
- ISOC-JP プログラム委員

https://oru.to/aboutme

TLP:CLEAR

前置き

参照 RFC 番号

SFC KEIO UNIVERSITY

- alt の解釈を元に作成しています
- 資料中の情報は作成当時のものです
- 書いていない詳細は RFC や参考文献を参照してください
  - RFC を読んでも書いていないときもあります
- このスライドは時たまアップデートされます
- このスライドでは基礎の基礎を***理解できません***

TLP:CLEAR

TLP:CLEAR

アジェンダ

TLP:CLEAR

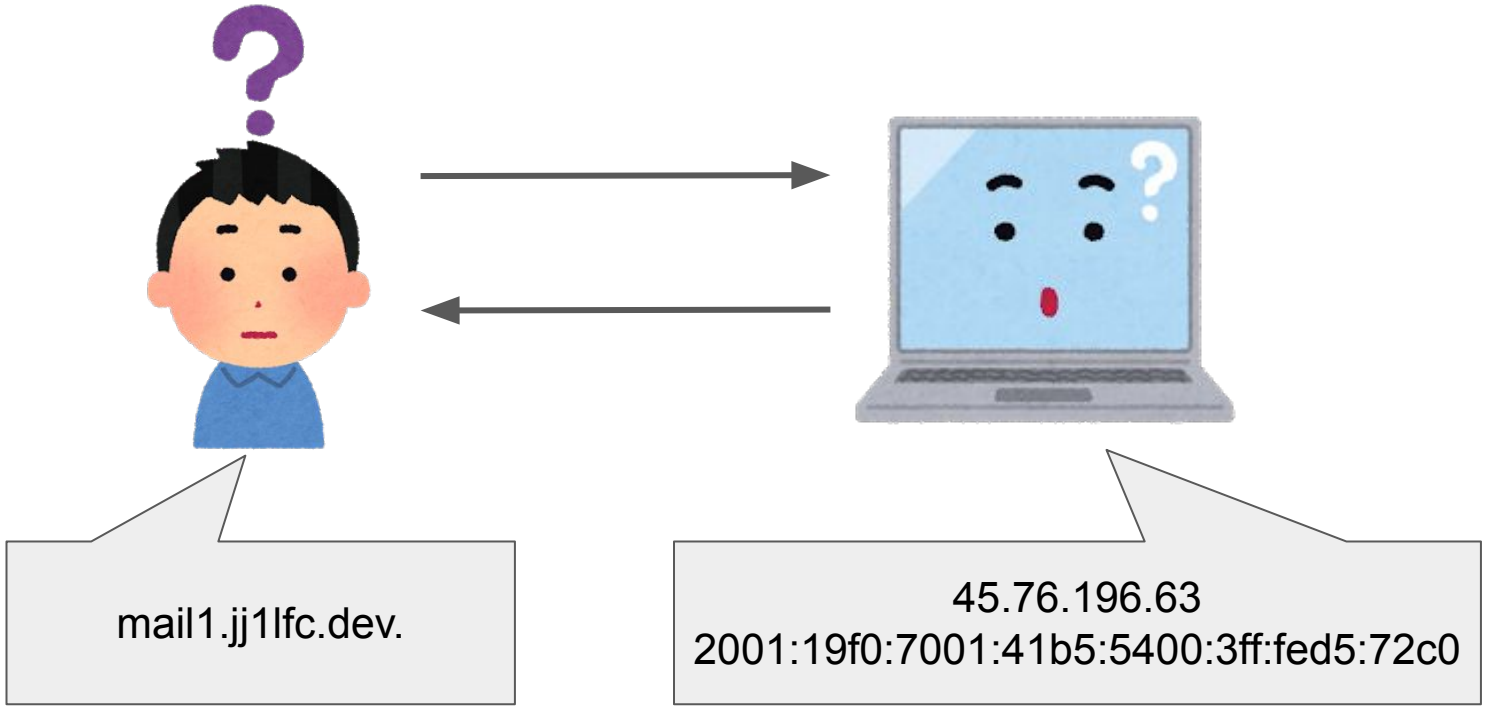
Why DNS?

DNS を使う理由とそのガバナンス

TLP:CLEAR

TLP:CLEAR

Why DNS? - ドメイン名と IP アドレス



TLP:CLEAR

TLP:CLEAR

Why DNS? - HOSTS.TXT

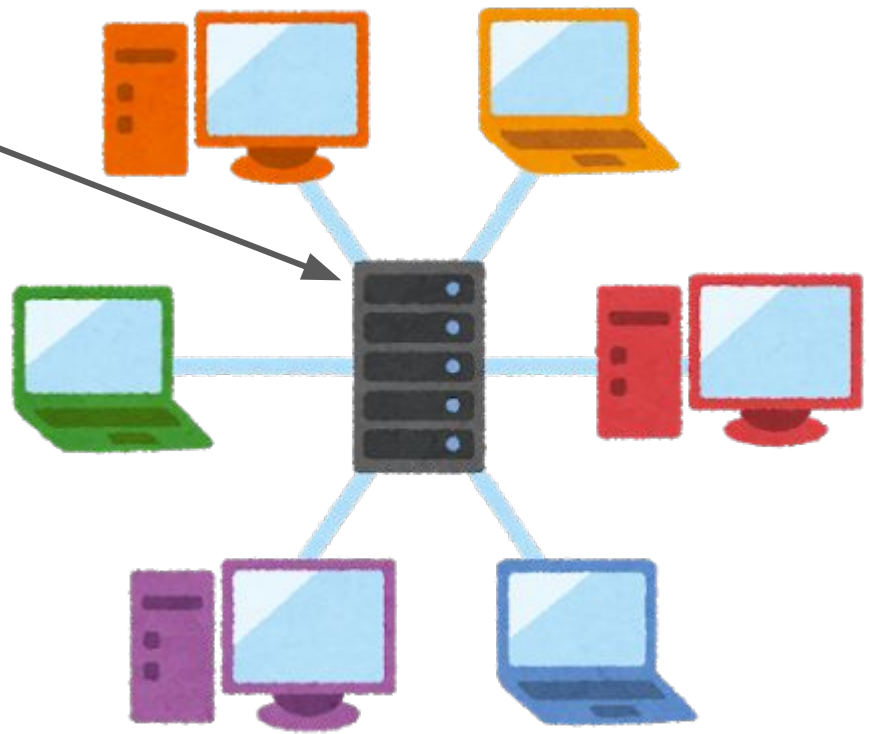
952, 953

HOSTS.TXT

127.0.0.1 localhost.
128.199.228.253 mail1.jj1lfc.dev.
203.178.136.59 wide.ad.jp.
131.113.134.133 keio.jp.
…

SRI-NIC
26.0.0.73
10.0.0.51

ARPANET



(※実際に使われていた HOSTS.TXT とは異なる)

TLP:CLEAR

TLP:CLEAR

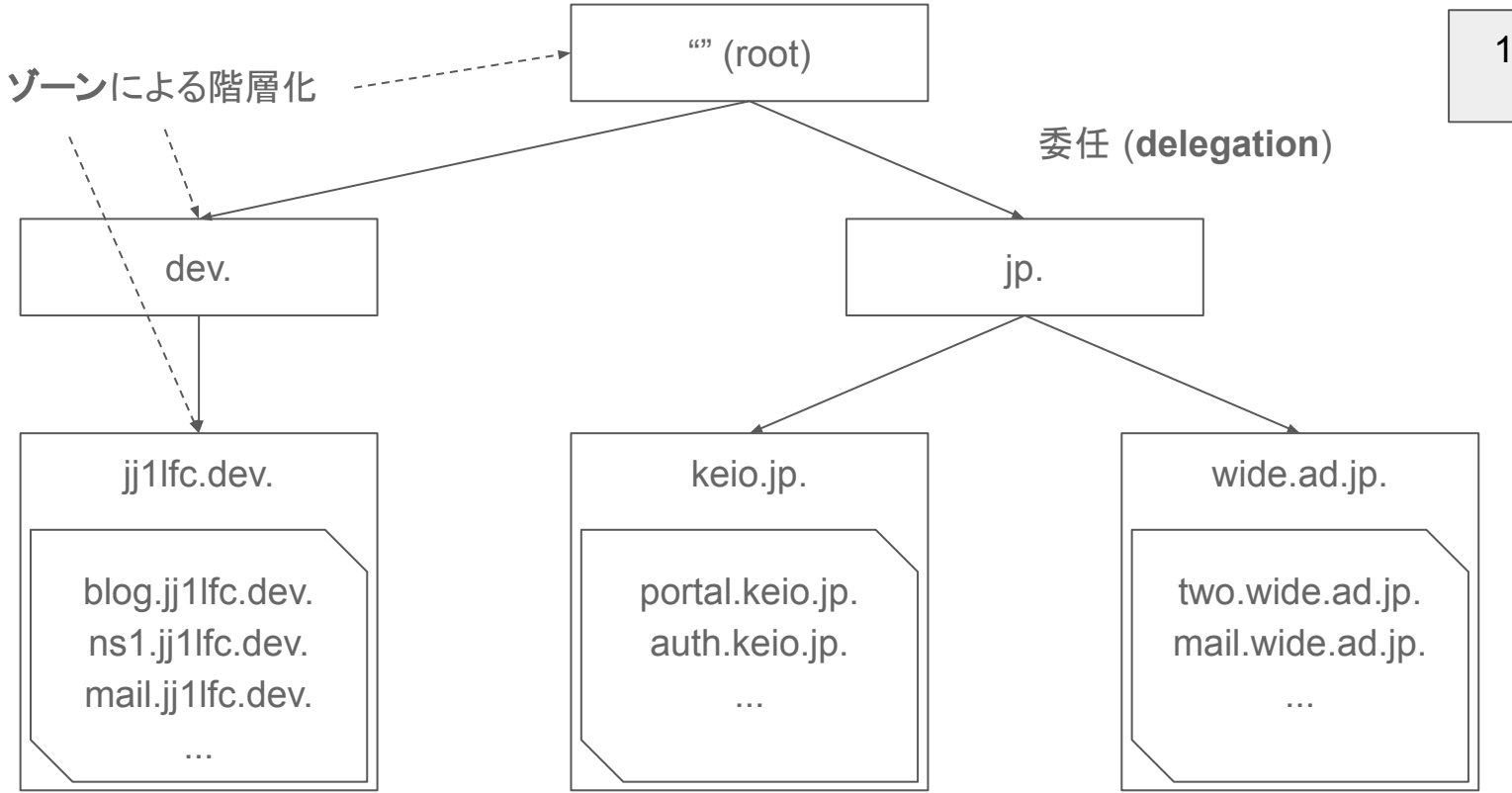
Why DNS? - ドメイン名

インターネット上の名前空間 (識別子) を階層的に示す

TLP:CLEAR

TLP:CLEAR

Why DNS? - 階層化と委任による分散管理



1033, 1034, 2181

TLP:CLEAR

TLP:CLEAR

DNS の構成要素

1. 名前空間 (DOMAIN NAME SPACE) とリソースレコード
   - 木構造を構成する名前空間と名前に紐付いたデータ
2. ネームサーバ (NAME SERVERS)
   - ドメインの木構造を保持し情報を定義するプログラム
3. リゾルバ (RESOLVERS)
   - クライアントのリクエストに応じてネームサーバからの情報を展開するプログラム

TLP:CLEAR

TLP:CLEAR

ICANN

- DNS ルートサーバの運用
- インターネット上の識別子の割り当て・登録

を行う

IETF: インターネット標準の検討・標準化
RSSAC: ルートサーバの運用・管理などの検討・助言
GNSO: gTLD の管理
ccNSO: ccTLD の管理
ALAC: インターネットユーザからの声を集める

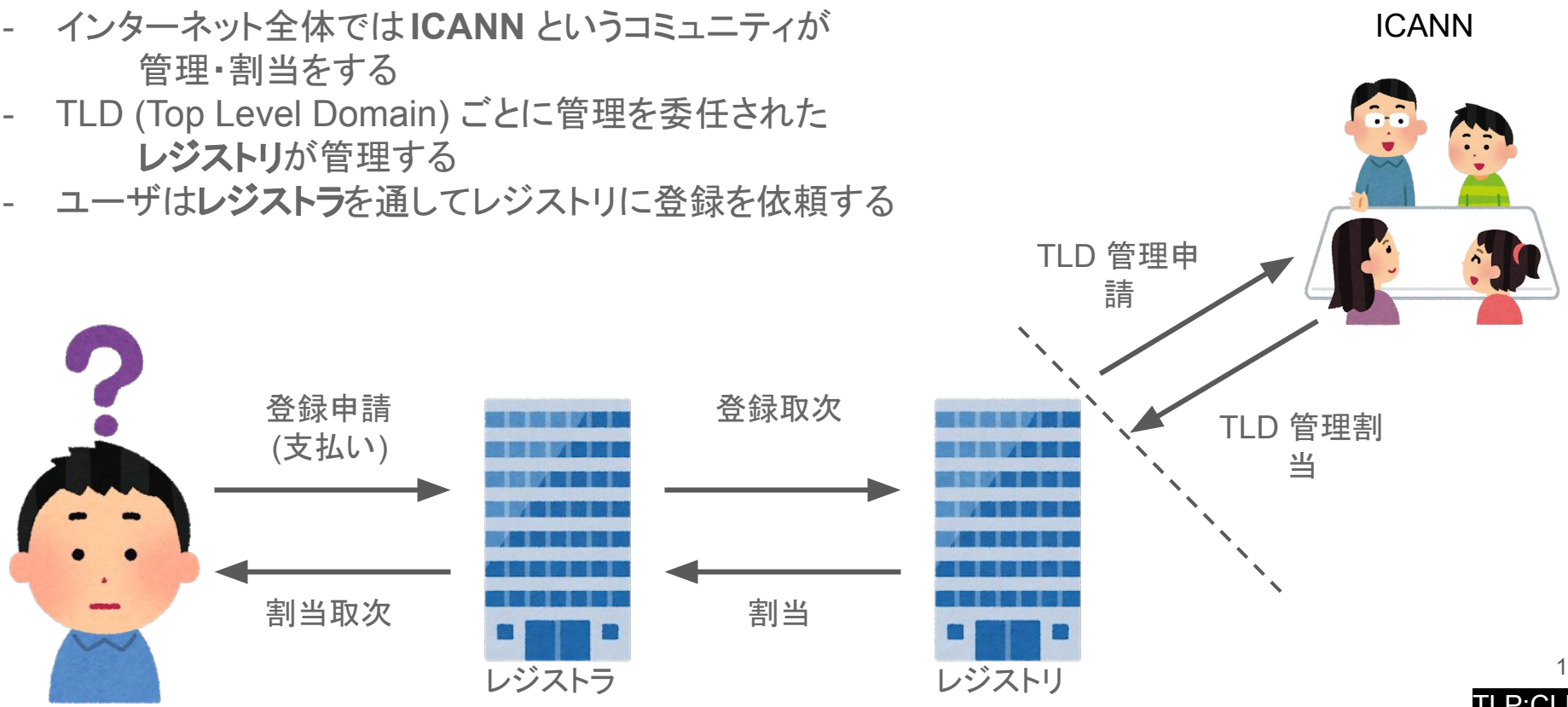
などの組織

TLP:CLEAR

TLP:CLEAR

ドメイン名を登録する

- インターネット全体では**ICANN** というコミュニティが管理・割当をする
- TLD (Top Level Domain) ごとに管理を委任された**レジストリ**が管理する
- ユーザは**レジストラ**を通してレジストリに登録を依頼する



TLP:CLEAR

TLP:CLEAR

レジストリ・レジストラ

レジストリ:

ドメイン名の割り当て・登録を行う組織
通常，1TLD に 1 組織であることが多い

レジストラ:

登録者からの依頼を確認しレジストリに取り次ぐ組織
1 TLD に複数組織存在できる

(さらに登録者-レジストラ間にリセラが入ることも)

TLP:CLEAR

TLP:CLEAR

Why DNS?

まとめ

- HOSTS.TXT→限界
- 階層化と委任
  - ゾーン
- DNS の 3 つの要素
  - 名前空間と RR
  - ネームサーバ
  - リゾルバ
- ICANN と
  レジストリ/レジストラ

TLP:CLEAR

TLP:CLEAR

ゾーンとリソースレコード

ゾーンを構成する概念

TLP:CLEAR

TLP:CLEAR

リソースレコードの構成

1035, 2181

SFC KEIO UNIVERSITY

mail1.jj1lfc.dev.	3600	IN	A	128.199.228.253
(ドメイン名)	(TTL)	(クラス)	(タイプ)	(データ)

- **ドメイン名**: ラベルを . で接続した名前
- **TTL**: リカーシブサーバがキャッシュできる時間 (秒)
- **クラス**: ネットワークの種別. IN や CH など
- **タイプ**: 情報の種別. A, AAAA, NS, TXT など
- **データ**: 情報の中身

単一のレコード (行): **リソースレコード** (**RR**)

ドメイン名・クラス・タイプが同じ RRs: **リソースレコードセット** (**RRset**)

RRset 内のすべての RRs の TTL は同じ (must)

TLP:CLEAR

TLP:CLEAR

タイプの種類 (一部)

1035 等

A: IPv4 アドレス
AAAA: IPv6 アドレス
NS: ゾーンの権威サーバ・切断点
SOA: ゾーン頂点と権威情報
MX: メールゲートウェイ
CNAME: 本名 (再帰検索を継続する)
PTR: 本名 (再帰検索をやめる) →主に逆引きに利用
TXT: 任意の文字列

TLP:CLEAR

TLP:CLEAR

ゾーンの要素

1034, 2181

SFC KEIO UNIVERSITY

1. ゾーン内のすべての権威あるデータ
2. ゾーン頂点を表すデータ
   - SOA レコードと (権威ある) NS レコード
3. 下部ゾーンを表すデータ
   - 下部ゾーンの NS レコード (権威はない)
   - 下部ゾーンの権威サーバが返すレコードと全く同じ (should)
4. 下部ゾーンの権威サーバにアクセスするためのデータ
   - グルーレコード (権威はない)
   - 必ずしも必要とは限らない

TLP:CLEAR

TLP:CLEAR

NS レコードの構成

1034, 1035

jj1lfc.dev.	600	IN	NS	ns1.jj1lfc.dev.
jj1lfc.dev.	600	IN	NS	ns2.jj1lfc.dev.

- 委任元 (親)・委任先で同じレコードをもつ
  - ゾーンの切断点を表す
  - 委任元 (親) が持つレコードには権威はない
- すべての権威サーバは全く同じ返答をする
  - **ゾーン転送 (zone transfer)** によって冪等性を図る
  - プライマリを一つ決め，他 (セカンダリ) はプライマリからゾーン転送を受ける

TLP:CLEAR

TLP:CLEAR

ゾーン転送

5936, 1995,
1996 等

転送方法の種類

- **Refresh**:　セカンダリからプライマリにリクエスト
- **Notify**:　　プライマリからセカンダリに更新の存在を通知

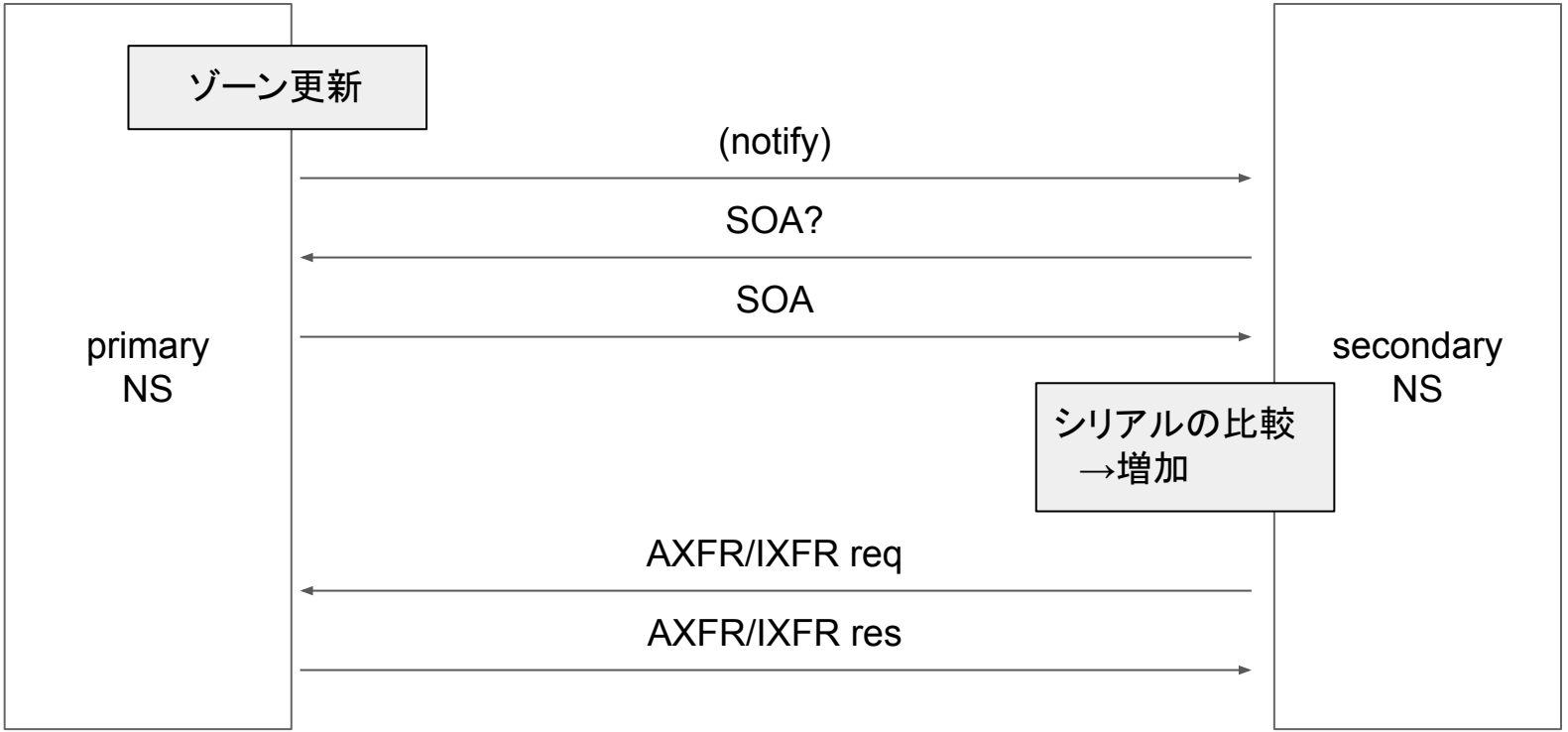
転送クエリの種類

- **AXFR**: ゾーンファイルの全てを転送
- **IXFR**:　ゾーンファイルの差分を転送

TLP:CLEAR

TLP:CLEAR

ゾーン転送



TLP:CLEAR

TLP:CLEAR

SOA レコードの構成

1034, 1035

```
jj1lfc.dev.         600         IN        SOA (
      ns1.jj1lfc.dev.           ;MNAME
      dnsadm.m.jj1lfc.dev.      ;RNAME
      1613691741                ;SERIAL
      600                       ;REFRESH
      600                       ;RETRY
      259200                    ;EXPIRE
      300                       ;MINIMUM
)
```


TLP:CLEAR

TLP:CLEAR

SOA レコードの意味

1034, 1035, 1982

SFC KEIO UNIVERSITY

- **MNAME**: プライマリサーバの絶対ドメイン名
- **RNAME**: ゾーン管理者のメアド (絶対ドメイン名形式)
- **SERIAL**: ゾーンシリアル (0-4294967295, 32bit)
- **REFRESH**: セカンダリがプライマリにアップデートを確認しにいく間隔
- **RETRY**: ゾーン転送失敗時の再実行までの間隔
- **EXPIRE**: ゾーン転送失敗が続いたとき，セカンダリが持っているデータを破棄するまでの猶予期間
- **MINIMUM**: 否定応答の TTL (※これは否定応答の TTL ではない)

TLP:CLEAR

TLP:CLEAR

ゾーンと RR

まとめ

- RR の構成
- ゾーンの 4 つの要素
  - 権威あるデータ
  - 権威を示すデータ
  - 下部ゾーンを表すデータ
  - グルーレコード
- ゾーン転送
- NS レコード
- SOA レコード

TLP:CLEAR

TLP:CLEAR

名前解決

リカーシブサーバになって考える名前解決の仕組み

TLP:CLEAR

TLP:CLEAR

名前解決の関係者

1034

SFC KEIO UNIVERSITY

クライアント

スタブリゾルバ

OS やブラウザなど
クライアントのこと

通信したい相手

“” (root) **権威サーバ**

(プロキシリゾルバ)

場合によって
組織管理者が設置

dev 権威サーバ

フルサービスリゾルバ

ゾーン管理者が設置
レジストラの管理画面などでいう
「DNS サーバ」がこれ

jj1lfc.dev 権威サーバ

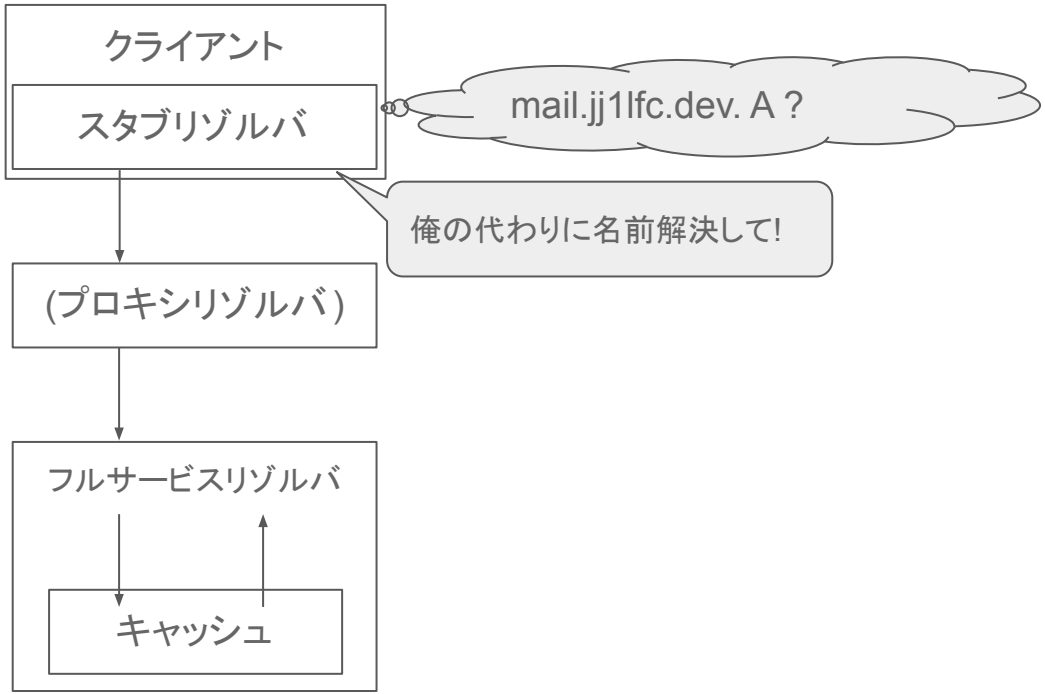
リカーシブサーバなどとも呼ばれる
キャッシュを持つ / ISP・組織管理者などが設置
クライアントや DHCP で設定する「DNS サーバ」

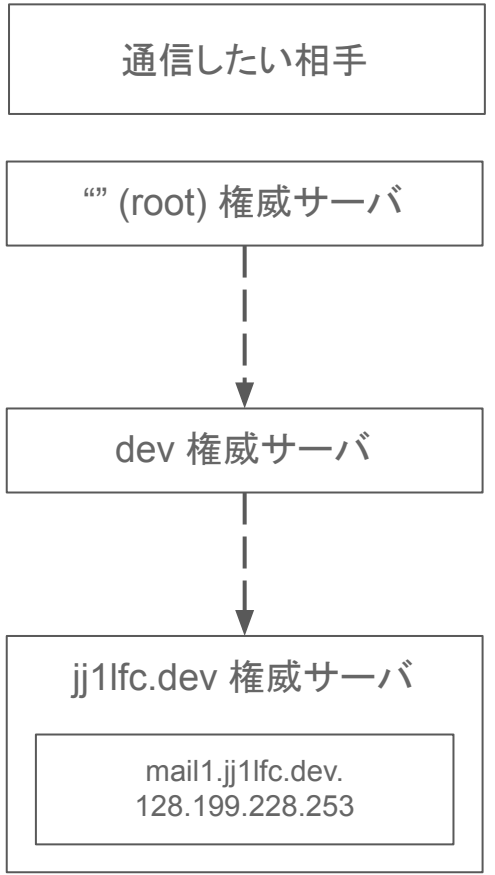
TLP:CLEAR

TLP:CLEAR

名前解決 (1)

1034





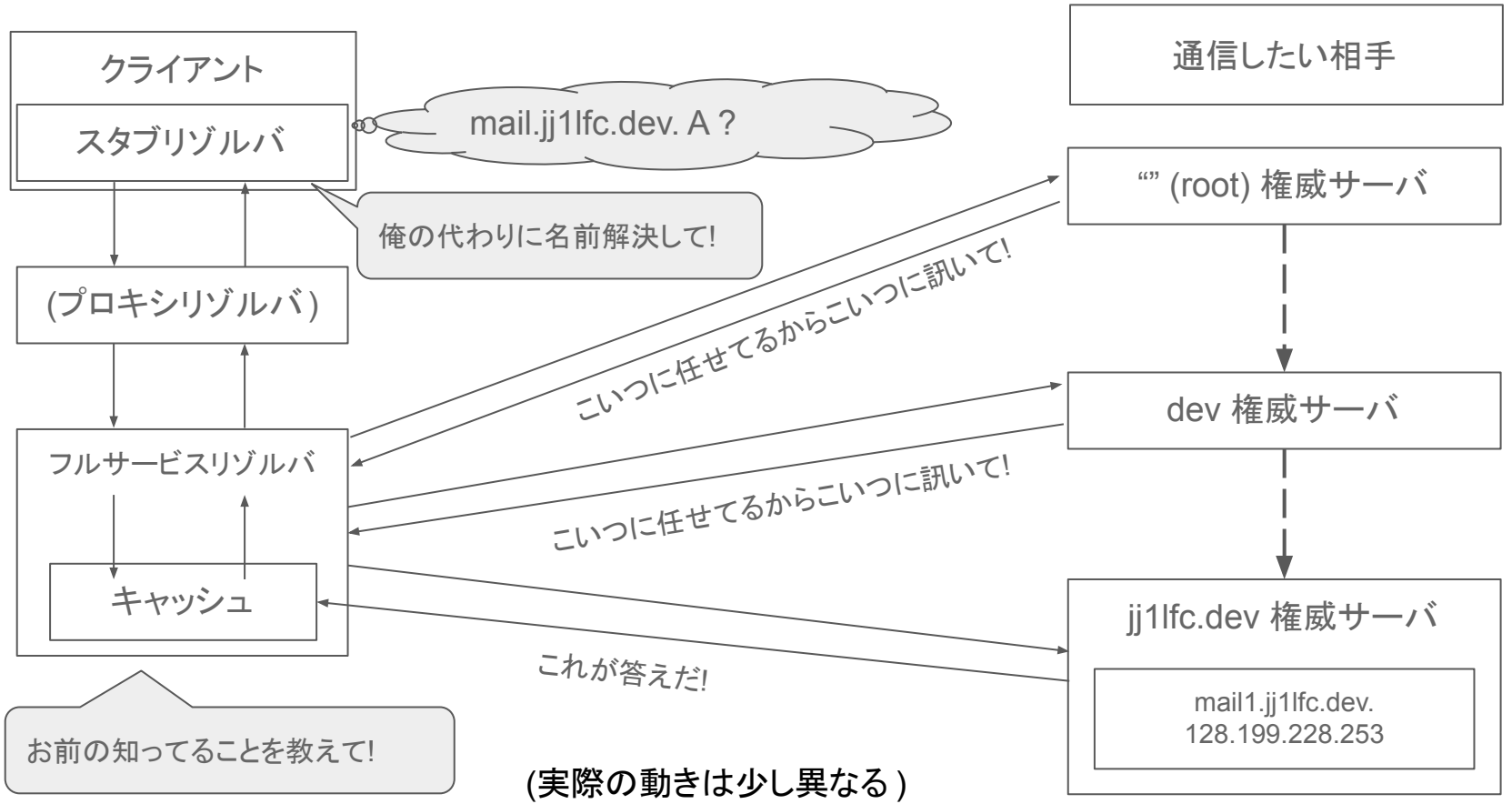
(実際の動きは少し異なる)

TLP:CLEAR

TLP:CLEAR

名前解決 (2)

1034



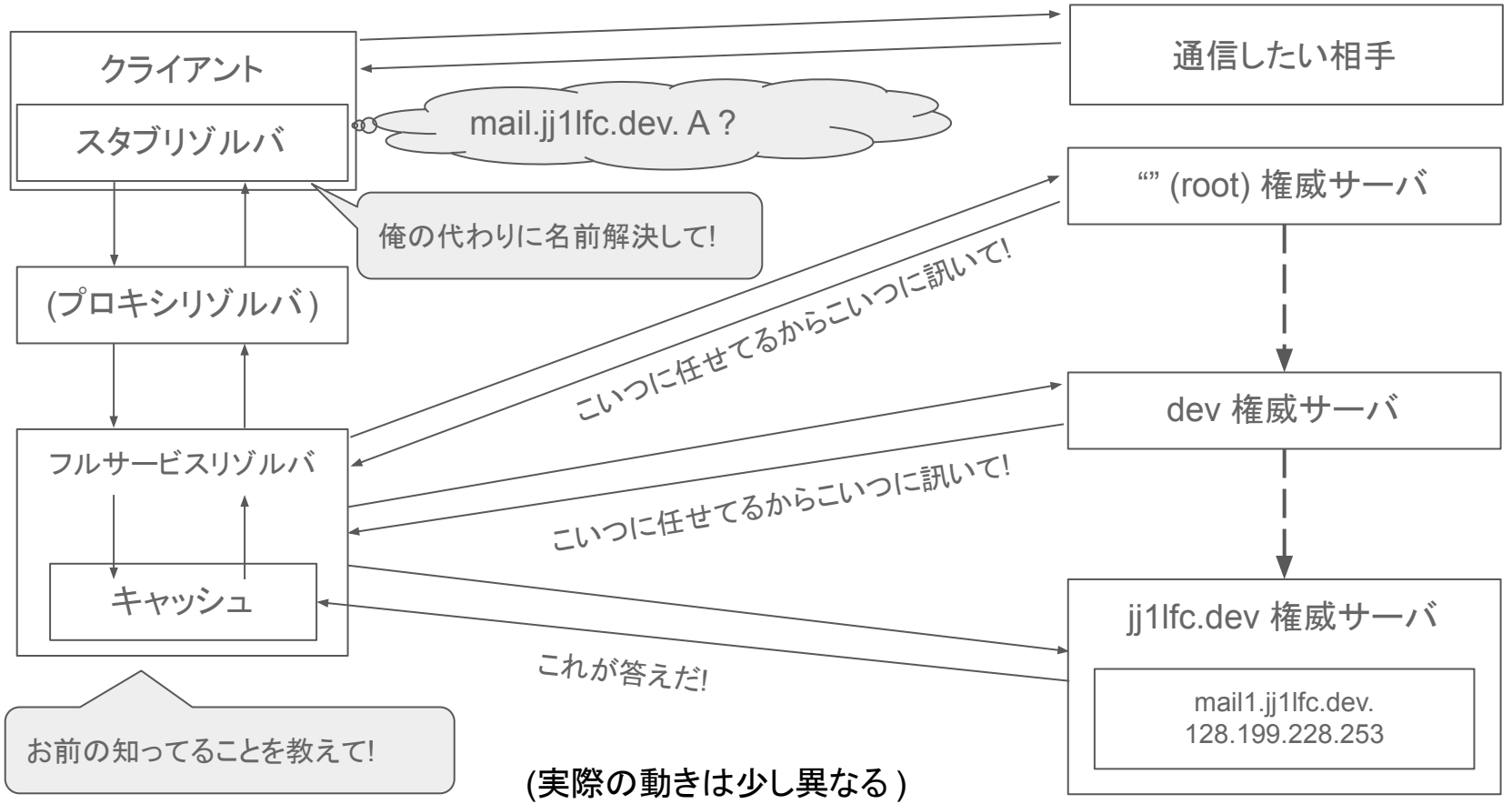
(実際の動きは少し異なる)

TLP:CLEAR

TLP:CLEAR

名前解決 (3)

1034



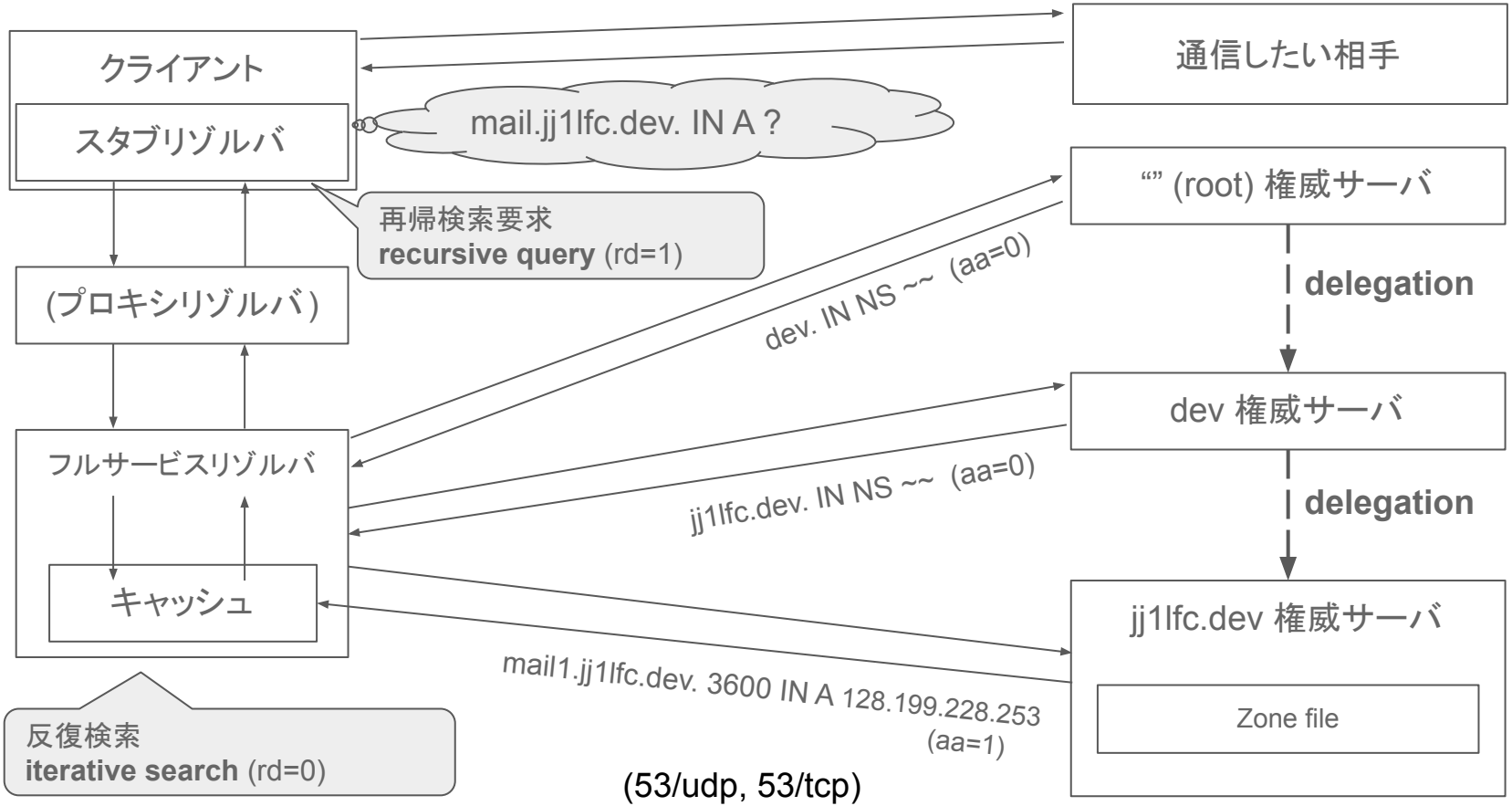
(実際の動きは少し異なる)

TLP:CLEAR

TLP:CLEAR

名前解決の用語

1034



TLP:CLEAR

TLP:CLEAR

応答の種類 (一部)

1035, 2308,
8020 等

SFC KEIO UNIVERSITY

NOERROR: 正常応答 (RCODE 0)

SERVFAIL: サーバで何らかのエラー (RCODE 2)

REFUSED: 応答拒否 (RCODE 5)

NXDOMAIN: 要求されたドメイン名が存在しない (否定応答)

(**NODATA**: 要求されたデータが存在しない)

など

TLP:CLEAR

TLP:CLEAR

5 種類の DNS 応答 (簡略版)

1. 仮名をクエリしていたため，再度クエリを変えて訊き直さねばならない
2. クエリしたタイプのドメイン名は存在せず，他のタイプでも存在しない
3. クエリしたドメイン名には一つまたは複数のレコードが存在する
4. 訊いた権威サーバが答えを持っていなかったため，他の権威サーバに訊く必要がある
5. クエリしたタイプのドメイン名は存在しないが，他のタイプで存在するかもしれない

※すべてリカーシブサーバの視点

TLP:CLEAR

TLP:CLEAR

(権威サーバへの問い合わせ方法・概略)

dig [@サーバ] [ドメイン名] [タイプ] [クラス] [オプション]

e.g.) `dig @165.22.250.24 mail1.jj1lfc.dev. AAAA IN +norec`

drill [-o オプション] [ドメイン名] [@サーバ] [タイプ] [クラス]

e.g.) `drill -o rd mail1.jj1lfc.dev. @165.22.250.24 AAAA IN`

※どちらも [] は順不同

TLP:CLEAR

TLP:CLEAR

リカーシブサーバの発見

エンドユーザはどのリカーシブサーバを利用するか任意に決定可能

- (ベンダによる指定)
- 手動
- DHCP/DHCPv6 による discovery/自動設定
  - Do53 しか設定できない
  - DoH を DHCP/DHCPv6 に載せる議論が IETF で行われている

TLP:CLEAR

TLP:CLEAR

リカーシブサーバの動作 - プライミング

https://www.internic.net/domain/named.root

8109

リカーシブサーバはどうやってルートサーバにアクセスする?

1. 持っている **root hints** ファイルの中からランダムにクエリし，最新の hints ファイルをもらってくる
2. 自身の持つ hints ファイルを更新する

```
drill NS . @202.12.27.33 -o rd -b 1232

dig NS . @202.12.27.33 +norec
```


TLP:CLEAR

TLP:CLEAR

ルートゾーンと冗長化

非中央集権管理

- 12 の組織，13 のインスタンスによって管理

Anycast

- ある IP アドレスを複数のサーバに割り当て，ユーザはそのうちネットワーク的に一番近いものにアクセスする

https://root-servers.org

TLP:CLEAR

リカーシブサーバの動作 - root zone

```
❯ drill mail1.jj1lfc.dev. @202.12.27.33 -o rd
;; ->>HEADER<<- opcode: QUERY, rcode: NOERROR, id: 39515
;; flags: qr ; QUERY: 1, ANSWER: 0, AUTHORITY: 5, ADDITIONAL: 10
;; QUESTION SECTION:
;; mail1.jj1lfc.dev.          IN        A

;; ANSWER SECTION:

;; AUTHORITY SECTION:
dev.      172800  IN        NS        ns-tld3.charlestonroadregistry.com.
dev.      172800  IN        NS        ns-tld1.charlestonroadregistry.com.
dev.      172800  IN        NS        ns-tld2.charlestonroadregistry.com.
dev.      172800  IN        NS        ns-tld5.charlestonroadregistry.com.
dev.      172800  IN        NS        ns-tld4.charlestonroadregistry.com.
```

```
;; ADDITIONAL SECTION:
ns-tld1.charlestonroadregistry.com. 172800  IN   A      216.239.32.105
ns-tld2.charlestonroadregistry.com. 172800  IN   A      216.239.34.105
ns-tld3.charlestonroadregistry.com. 172800  IN   A      216.239.36.105
ns-tld4.charlestonroadregistry.com. 172800  IN   A      216.239.38.105
ns-tld5.charlestonroadregistry.com. 172800  IN   A      216.239.60.105
ns-tld1.charlestonroadregistry.com. 172800  IN   AAAA   2001:4860:4802:32::69
ns-tld2.charlestonroadregistry.com. 172800  IN   AAAA   2001:4860:4802:34::69
ns-tld3.charlestonroadregistry.com. 172800  IN   AAAA   2001:4860:4802:36::69
ns-tld4.charlestonroadregistry.com. 172800  IN   AAAA   2001:4860:4802:38::69
ns-tld5.charlestonroadregistry.com. 172800  IN   AAAA   2001:4860:4805::69

;; Query time: 14 msec
;; SERVER: 202.12.27.33
;; WHEN: Wed Mar  3 23:49:03 2021
;; MSG SIZE  rcvd: 389
```

リカーシブサーバの動作 - dev zone

```
❯ drill A mail1.jj1lfc.dev. @216.239.32.105 -o rd
;; ->>HEADER<<- opcode: QUERY, rcode: NOERROR, id: 24457
;; flags: qr ; QUERY: 1, ANSWER: 0, AUTHORITY: 2, ADDITIONAL: 4
;; QUESTION SECTION:
;; mail1.jj1lfc.dev.     IN      A

;; ANSWER SECTION:

;; AUTHORITY SECTION:
jj1lfc.dev.      10800   IN      NS      ns1.jj1lfc.dev.
jj1lfc.dev.      10800   IN      NS      ns2.jj1lfc.dev.

;; ADDITIONAL SECTION:
ns1.jj1lfc.dev. 3600  IN      A       165.22.250.24
ns1.jj1lfc.dev. 3600  IN      AAAA 2400:6180:0:d1::776:1001
ns2.jj1lfc.dev. 3600  IN      A       45.76.100.215
ns2.jj1lfc.dev. 3600  IN      AAAA 2401:c080:1000:4c9f:5400:3ff:fe2f:a3b

;; Query time: 49 msec
;; SERVER: 216.239.32.105
;; WHEN: Wed Mar  3 23:53:22 2021
;; MSG SIZE  rcvd: 157
```

glue record

TLP:CLEAR

リカーシブサーバの動作 - jj1lfc.dev zone

```
❯ drill A mail1.jj1lfc.dev. @165.22.250.24 -o rd
;; ->>HEADER<<- opcode: QUERY, rcode: NOERROR, id: 61989
;; flags: qr aa ; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 0
;; QUESTION SECTION:
;; mail1.jj1lfc.dev.        IN      A

;; ANSWER SECTION:
mail1.jj1lfc.dev.           3600 IN      A       128.199.228.253

;; AUTHORITY SECTION:

;; ADDITIONAL SECTION:

;; Query time: 118 msec
;; SERVER: 165.22.250.24
;; WHEN: Wed Mar  3 23:56:13 2021
;; MSG SIZE  rcvd: 49
```


TLP:CLEAR

TLP:CLEAR

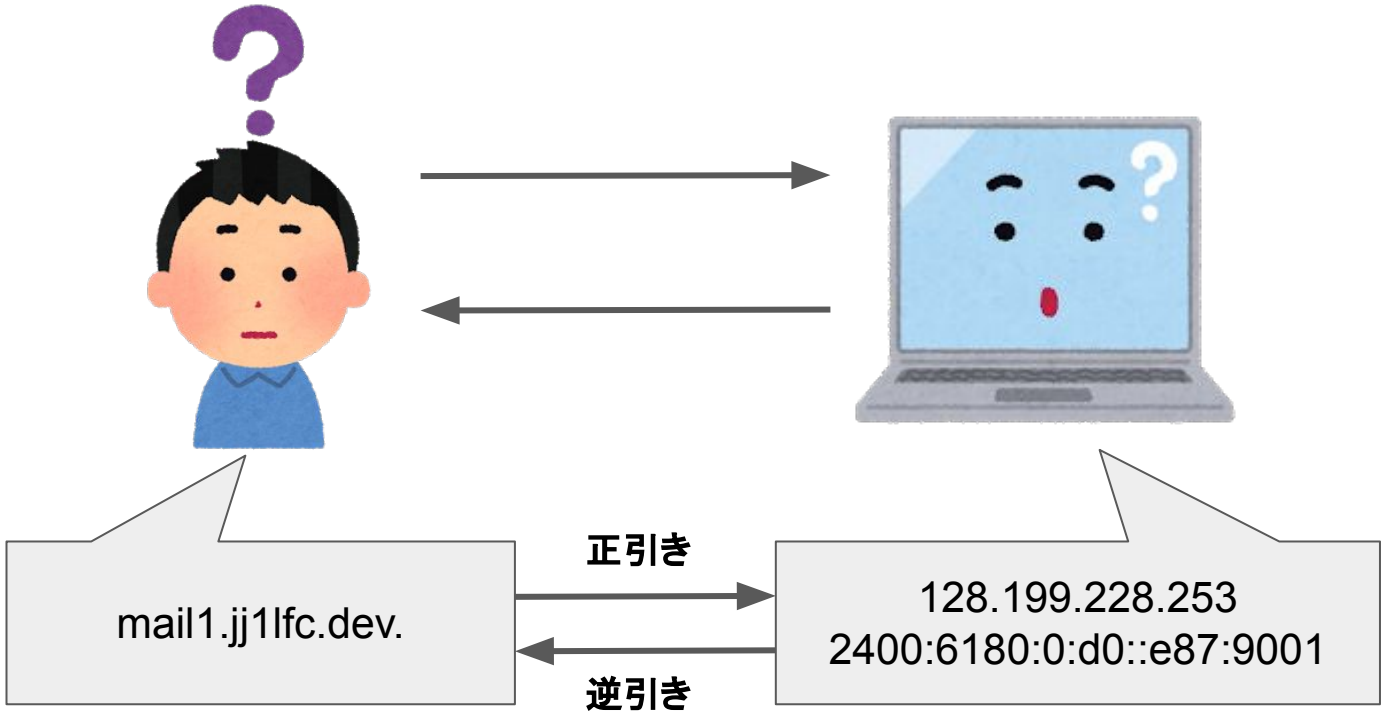
名前解決 1

まとめ

- 名前解決の関係者
  - 「DNS サーバ」
- 応答の種類
- 権威サーバの 5 種類の応答
- フルリゾルバになって考える
  - dig, drill コマンド
  - 再帰検索要求と反復検索
  - プライミング
  - ルートゾーン

TLP:CLEAR

逆引き


mail1.jj1lfc.dev.
正引き
逆引き
128.199.228.253
2400:6180:0:d0::e87:9001

TLP:CLEAR

PTR レコード

1035, 2181

253.228.199.128.**in-addr.arpa.** 3600 IN PTR mail1.jj1lfc.dev.
mail1.jj1lfc.dev. 3600 IN A 128.199.228.253

(1.0.0.9.7.8.e.0.0.0.0.0.0.0.0.0.0.0.d.0.0.0.0.0.0.0.8.1.6.0.0.4.2.**ip6.arpa.** 3600 IN PTR mail1.jj1lfc.dev.)
mail1.jj1lfc.dev. 3600 IN AAAA 2400:6180:0:d0::e87:9001

TLP:CLEAR

TLP:CLEAR

有名な OSS 実装

- BIND

リカーシブサーバ

- Unbound
- PowerDNS recurser
- knot resolver

コンテンツサーバ

- NSD
- PowerDNS
- knot

TLP:CLEAR

TLP:CLEAR

DNS 関連の Security issues (一部)

Lame delegation とゾーン乗っ取り
サブドメインテイクオーバー
レジストラアカウントの乗っ取り
DNS amp attack
Cache poisoning (カミンスキー，NXNS，SADDNS 等)
オープンリゾルバ
権威・キャッシュ共存
プライバシ
BIND

TLP:CLEAR

TLP:CLEAR

名前解決 2

まとめ

- 逆引き
  - PTR レコード
  - in-addr.arpa.
  - ip6.arpa.
- 有名な OSS 実装
- Security issues

TLP:CLEAR

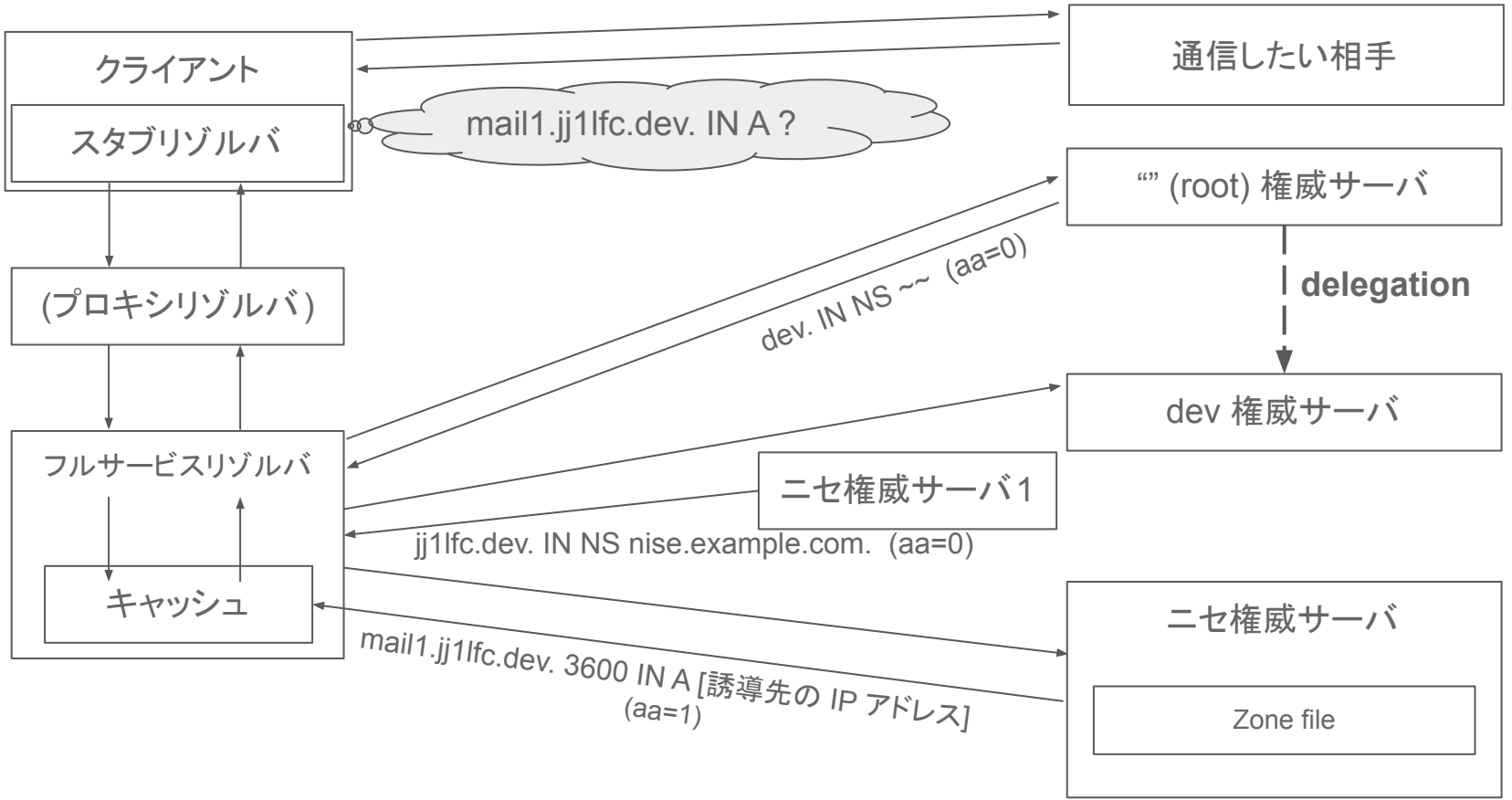
TLP:CLEAR

DNSSEC

Cache Poisoning と DNS Security Extension の概略の一部

TLP:CLEAR

Cache Poisoning (例)



TLP:CLEAR

DNSSEC

2535, 4033

非対称鍵暗号による電子署名を用いて権威サーバ間で PKI を形成

提供するもの

- 応答データの権威の認証
- ゾーンデータの完全性保証
  - RR の不存在証明を含む

提供しないもの

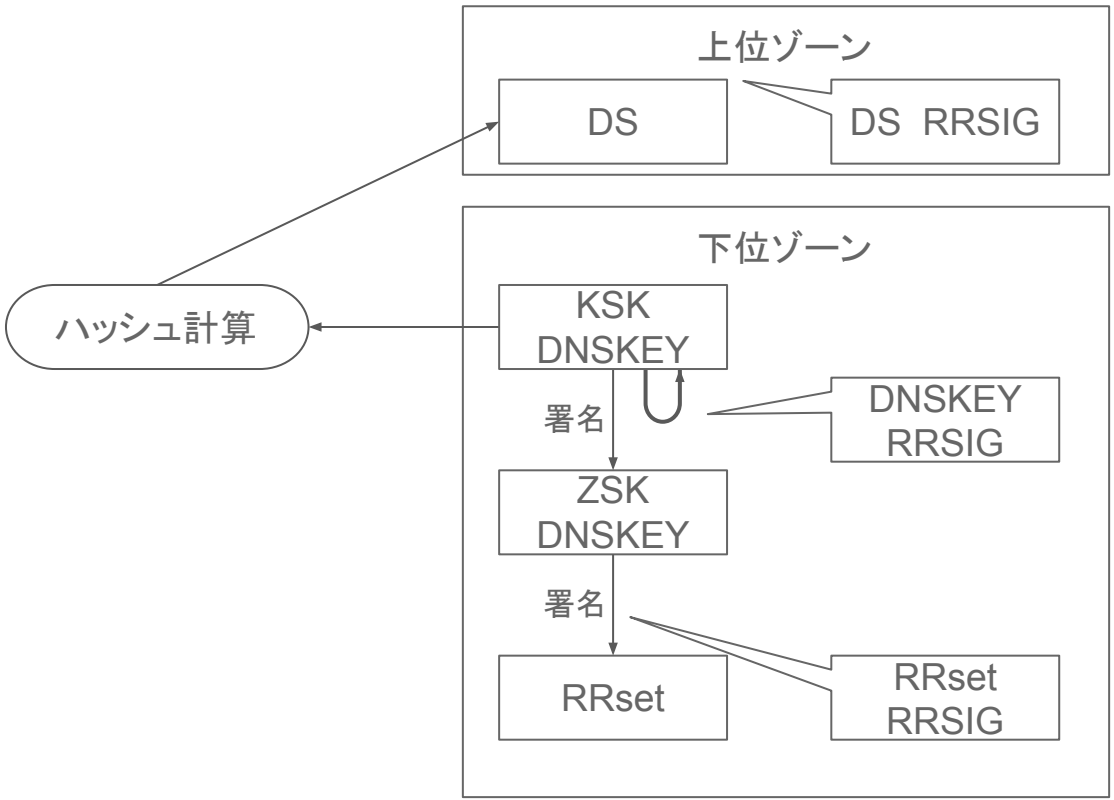
- 機密性，アクセスコントロール等の発信元による区別
- DoS 耐性
- zone transfer, dynamic update 等のセキュリティ　　など

TLP:CLEAR

TLP:CLEAR

トラストチェーン (概略)

2535, 4033



This structure provides:
- Integrity of the records
- Authenticity of the answer

TLP:CLEAR

TLP:CLEAR

DNSSEC の RR Type (一部)

4035, 5155

DS: KSK 公開鍵のハッシュ (親ゾーンで提供)

DNSKEY: KSK/ZSK 公開鍵

RRSIG: RRset の ZSK (もしくは KSK) による署名

NSEC/NSEC3/NSEC3PARAM:
不存在証明

TLP:CLEAR

TLP:CLEAR

DS レコードの構成

```
jj1lfc.dev.   3600    IN   DS 46088 13 2 677A2C5B54D969FAFE52～
jj1lfc.dev.   3600    IN   DS 46088 13 4 5358212B291650170CAD～
```

- 親ゾーンが持つ，子ゾーンへのポインタ
  - 鍵タグ: 鍵の識別子
  - アルゴリズム番号: 非対称鍵暗号アルゴリズム
  - ハッシュアルゴリズム番号: 公開鍵のハッシュアルゴリズム
  - KSK 公開鍵のハッシュ: 子ゾーンへのポインタとなる

※DS レコードは唯一の親ゾーンにのみ存在するレコード

```
dig jj1lfc.dev. DS @ns-tld1.charlestonroadregistry.com. +norec
```


TLP:CLEAR

TLP:CLEAR

DNSKEY レコードの構成

```
jj1lfc.dev.   600     IN   DNSKEY   256 3 13 dreQ6CTxkRhYZUw3～
jj1lfc.dev.   600     IN   DNSKEY   257 3 13 smtx2KIcNDXSIdM6g～
```

- KSK と ZSK の公開鍵
  - フラグ: ZSK は 256, KSK は 257
  - プロトコル番号: DNSSEC は 3
  - アルゴリズム番号: 非対称鍵暗号アルゴリズム
  - 公開鍵: Base64 でひねる

※DNSKEY レコードはゾーン頂点のみに存在

```
dig jj1lfc.dev. DNSKEY @ns1.jj1lfc.dev. +norec
```


TLP:CLEAR

TLP:CLEAR

RRSIG レコードの構成

SFC KEIO UNIVERSITY

```
mail1.jj1lfc.dev.  3600  IN  RRSIG  A 13 3 3600 20210610012234 20210526235234 10196 jj1lfc.dev.
B5FPI3B7NG0s0Gmdmw10lJjK0D6UyOwSg4gv8pzsBLHA2toEuDbn6+dgDmKhudxEjQ4ztkUU1y2I3ud1
UUDS9Q==
```

- RRSIG 以外の RRset に対する署名
  - 対象 type:　　署名対象 RRset の type
  - アルゴリズム番号:　　非対称鍵暗号のアルゴリズム
  - ラベル数:　　署名対象ドメイン名のラベル数
  - 対象 TTL:　　署名対象の TTL
  - 署名期限:　　署名自体の有効期限 (UTC)
  - 署名開始:　　署名自体の発効日時 (UTC)
  - 鍵タグ:　　署名に使った鍵タグ (参照する DNSKEY)

TLP:CLEAR

TLP:CLEAR

RRSIG レコードの構成 2

SFC KEIO UNIVERSITY

```
mail1.jj1lfc.dev.  3600  IN  RRSIG  A 13 3 3600 20210610012234 20210526235234 10196 jj1lfc.dev.
B5FPI3B7NG0s0Gmdmw10lJjK0D6UyOwSg4gv8pzsBLHA2toEuDbn6+dgDmKhudxEjQ4ztkUU1y2I3ud1
UUDS9Q==
```

- RRSIG 以外の RRset に対する署名
  - ゾーン名:　　署名対象が存在するゾーン頂点
  - 署名:　　署名本体 (RRset ごとに一つの署名)

```
dig mail1.jj1lfc.dev. A @ns1.jj1lfc.dev. +dnssec
```


TLP:CLEAR

TLP:CLEAR

NSEC による不存在証明

ゾーンの完全性証明には，存在しない RR が存在しないことの証明も必要

→NSEC レコードに以下を格納し署名

- アルファベット順に次のドメイン名へのポインタ
- 次のドメイン名に存在する RR type

不存在のドメイン名への問い合わせには前後 1 つずつの NSEC と NSEC の RRSIG を返すことで，不存在を証明する

TLP:CLEAR

TLP:CLEAR

NSEC による不存在証明

ゾーンの完全性証明には，存在しない RR が存在しないことの証明も必要

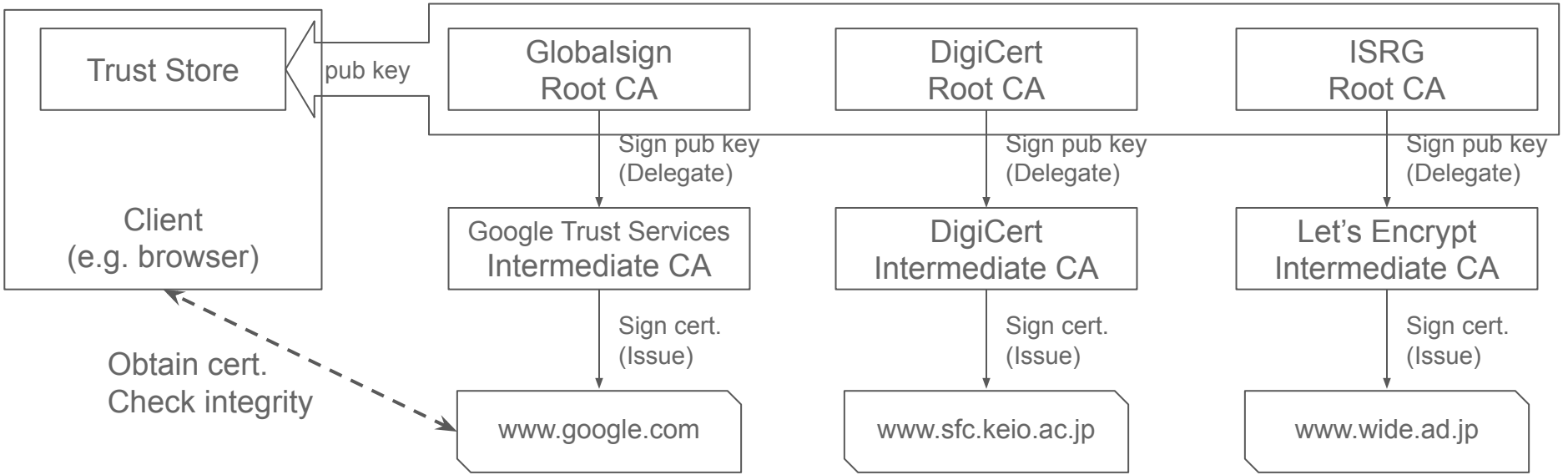
```
a.example.com. IN A ~
c.example.com. IN A ~
e.example.com. IN A ~
```

```
a.example.com. IN NSEC c.example.com. A RRSIG NSEC
c.example.com. IN NSEC e.example.com. A RRSIG NSEC
e.example.com. IN NSEC a.example.com. A RRSIG NSEC
```


TLP:CLEAR

TLP:CLEAR

参考: CA と X.509 を利用した PKI 例



WebPKI の例. 実際は Revocation repository などが絡んでくるためもっと複雑.

TLP:CLEAR

TLP:CLEAR

DNSSEC

まとめ

- Cache poisoning
- DNSSEC
  - 提供するもの・しないもの
- トラストチェーンの構造
- 使われる RR type

TLP:CLEAR

TLP:CLEAR

DNS Privacy

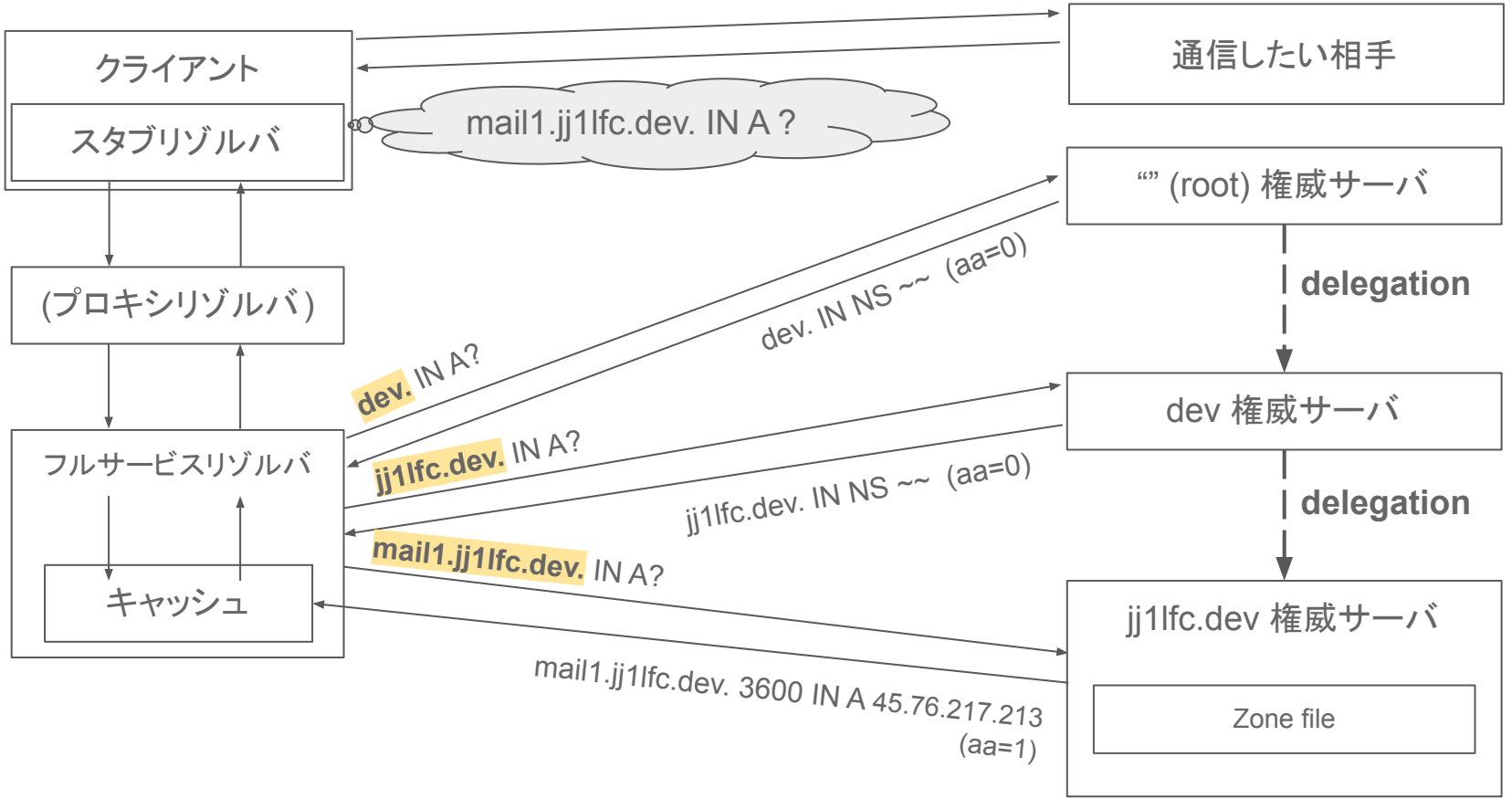
QNAME minimisation と DoHoge など

TLP:CLEAR

QNAME minimisation

7816

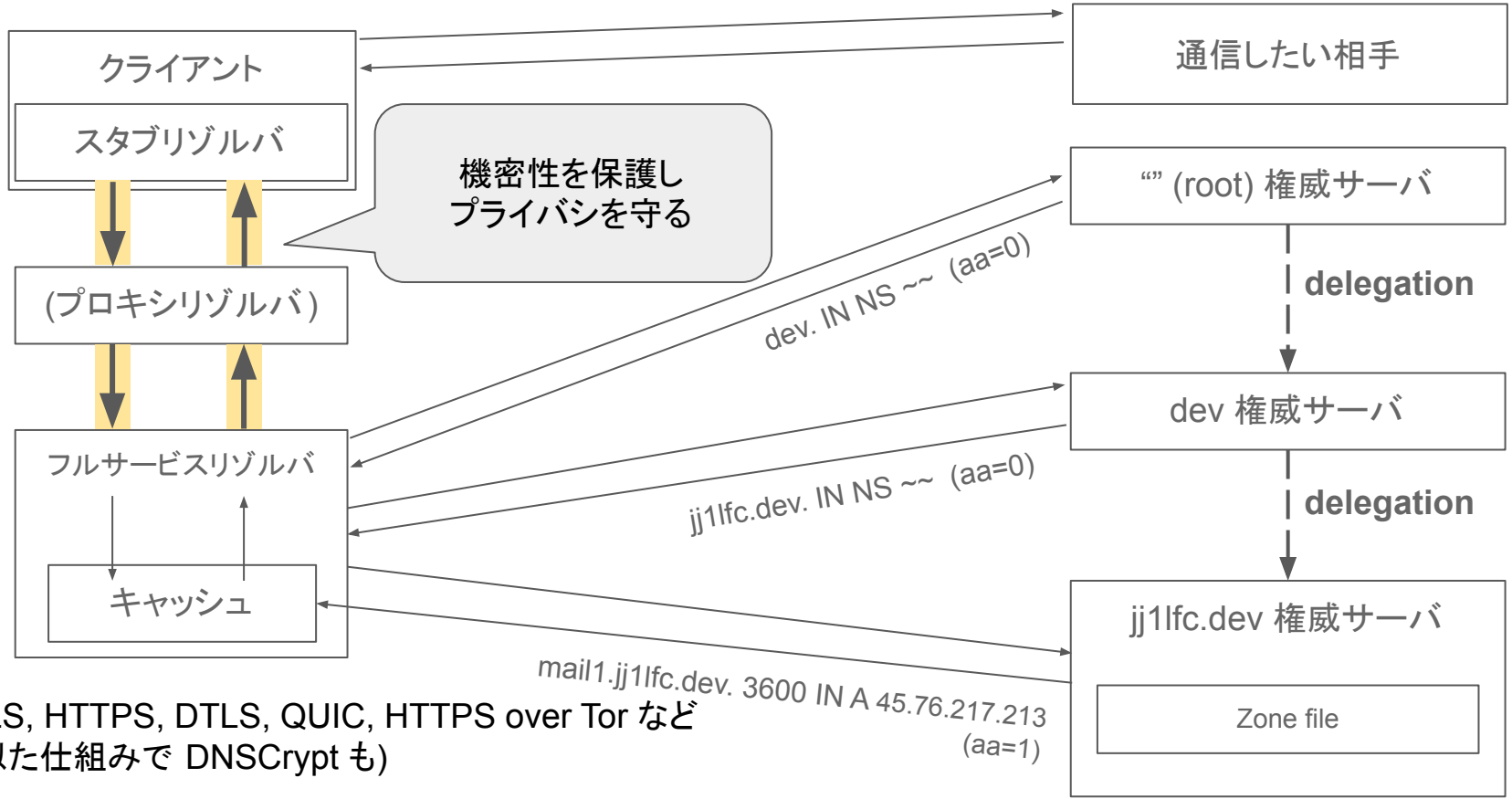
KEIO UNIVERSITY



TLP:CLEAR

DoT, DoH など

7858 等



TLS, HTTPS, DTLS, QUIC, HTTPS over Tor など
(似た仕組みで DNSCrypt も)

TLP:CLEAR

TLP:CLEAR

DNS Privacy

まとめ

- QNAME minimisation
- DoT, DoH など

TLP:CLEAR

参考文献

- 渡邉結衣, 佐藤新太, 藤原和典. DNS がよくわかる教科書. 2018.
- 滝澤隆史. DNS の RFC の歩き方. 2012, available from <https://dnsops.jp/event/20120831/DNS-RFC-PRIMER-2.pdf>, accessed 2021-03-04.
- D.J.Bernstein. Notes on the Domain Name System, available from <https://cr.yp.to/djbdns/notes.html>, accessed 2021-03-04.
- ICANN. Beginner's Guide to Participating In ICANN. 2013, available from <https://www.icann.org/en/system/files/files/participating-08nov13-en.pdf>, accessed 2021-03-19.
- T.Suzuki. 浸透言うな!. 2011, available from <https://www.e-ontap.com/dns/propagation/>, accessed 2021-03-21.
- man dig
- man drill
- (各ページ記載の IETF RFC)

TLP:CLEAR

付録 1: ドメイン名を登録したら

難しい DNS の上で生きるために
最低限考えてほしいこと

TLP:CLEAR

TLP:CLEAR

ドメイン名を登録する前にやること

1. 本当にドメイン名登録が必要か考える
    a. ドロップキャッチを考えるとドメイン名失効なんてできませんよね
2. レジストラを選ぶ
    a. 値段だけに騙されていませんか?
        i. ずっと契約するんだから 1 年目だけ安くても大差ない
    b. 責任もって運用してくれますか?
        i. レジストラアカウントはちゃんと守ってもらえますか?
        ii. サポートは嘘つきませんか?
3. 権威サーバを建てる
    a. レジストラの権威サーバにこだわる必要はない
        i. クラウド権威サーバは RFC に従っていますか?
    b. 自分でたててもいいんですよ

TLP:CLEAR

TLP:CLEAR

ドメイン名を登録したらやってほしいこと

1. DNSSEC 署名
   a. なりすまされることを防ぎましょう (これだけで完璧と思わない)
2. (メールを受け取らないなら) null MX 設定
   a. `example.com. [TTL] IN MX 0 .`
3. (メールを使っても使わなくても) SPF と DMARC の設定
   a. メールを使うなら
      i. それぞれ適した設定を
   b. メールを使わないなら
      i. `example.com. [TTL] IN TXT “v=spf1 -all”`
      ii. `(_dmarc.example.com. [TTL] IN TXT “v=DMARC1; p=reject”)`

TLP:CLEAR

TLP:CLEAR

ドメイン名を登録したら

まとめ

- 登録する前に
  - 本当に必要?
  - レジストラを選ぶ
  - 権威サーバを建てる
- 登録したら
  - DNSSEC 署名
  - (null MX の設定)
  - SPF/DMARC の設定

TLP:CLEAR

TLP:CLEAR

付録 2:「浸透言うな」って?

なぜ殴られるのか
言葉狩りじゃない「浸透言うな」

TLP:CLEAR

TLP:CLEAR

「DNS 浸透しない」

浸透，伝播，反映，有効化，propagation…

- 「反映まで最大 72 時間かかります」
- 「DNS 浸透しないので Wordpress が見られない」
  - 「DNS を 8.8.8.8 にしたら浸透した!」
  - 「Shift + Ctrl + R 押したら浸透した!」
- DNS Propagation Checker

※殴っているわけではなく，なんで殴る人がいるかの説明です

TLP:CLEAR

TLP:CLEAR

「浸透」という仕組みはそもそもない

- 本スライドで説明済み (P22-25, 29-33)
- 全部 pull 型の動作
  - 強いていえば zone transfer での notify が push に似ている
  - 通常はすぐ transfer され冪等性が保たれる
  - transfer されていないなら異常なのでのんきに待ってはいけない

TLP:CLEAR

TLP:CLEAR

ではみんな何を待っているのか

おそらく

- クラウド権威サーバ事業者のレコード更新実施待ち
  - 仕事が遅いのは「DNS の仕様です」?
- 自分の使うキャッシュサーバの TTL 切れ待ち
  - どこに問い合わせているかも知らずに「浸透待ち」
- 権威サーバ移行がちゃんとできていない
  - 手順を踏まない危険な作業
- (幽霊ドメイン名脆弱性) (権威キャッシュ共存サーバの実装不備)

などなど (そもそも DNS のせいじゃないことも)

TLP:CLEAR

TLP:CLEAR

あるべきリソースレコードの変更と確認

- 削除/更新時には TTL を前もって短くする
- 追加時には negative cache TTL を考慮する

- 確認時にはリカーシブサーバに頼らず手で権威サーバに dig る

TLP:CLEAR

TLP:CLEAR

本来あるべき権威サーバの移行方法

1. 移行元・委任元サーバの NS レコード TTL を短くする
2. DNSSEC 秘密鍵を安全にコピーし署名する
   a. 秘密鍵が取り出せないなら DNSSEC を一度やめる
3. 移行先の権威サーバで権威応答を開始する
   a. 移行先では移行先のホスト名で NS レコードを書く
   b. TTL はもとの長さで
4. 移行元・委任元サーバの NS レコードを移行先ホスト名にする
   a. 同時に委任元の TTL をもどす
5. 移行元の権威サーバで権威応答をやめる
   a. 2 で DNSSEC 署名をやめたなら再開する

TLP:CLEAR

TLP:CLEAR

本来あるべき権威サーバの移行方法 2

権威サーバと同時に他のものも移行する場合

- レジストラの移管
  - 同時にするのは避けましょう
  - 非協力的な事業者 (non-cooperating DNS operators) が多い
- 他サービス (Web, mail, etc…) のサーバ移行
  - 同時にするのは避けましょう
  - 1 の前に移行先でサービス開始し，5 の後に移行元でサービス終了する

TLP:CLEAR

TLP:CLEAR

浸透を待っていると

- DNS をミリも理解していないことがバレる
  - DNS を理解している人が誰一人いない会社が運用していることがバレる
- サービスのダウンタイムが増えてユーザに影響する
- 一部の共用権威サーバを使っているとゾーンが乗っ取られる?
  (ツイートしたら怒られる理由)
  - NS レコードのホスト名によっては共用サーバじゃなくてもありえる
- 怒られる

TLP:CLEAR

TLP:CLEAR

「浸透言うな」って?

まとめ

- そんな仕組みはない
- 「浸透しない」理由がある
  - 異常は放置しない
- RR の変更と確認
- 権威サーバの移行方法
- 浸透待ちしていると

TLP:CLEAR