DNSのメッセージサイズについて考える
～ランチのおともにDNS～

2013年11月28日
Internet Week 2013 ランチセミナー
株式会社日本レジストリサービス（JPRS）
森下 泰宏・堀 五月

講師自己紹介

- 森下 泰宏（もりした やすひろ）
  - 日本レジストリサービス（JPRS）広報宣伝室
  - 主な業務内容：技術広報担当として
    ドメイン名・DNSに関する技術情報をわかりやすく伝える
  - 最近の願いごと：平穏無事な7月が訪れますように・・・
- 堀 五月（ほり さつき）
  - 日本レジストリサービス（JPRS）システム部
  - 主な業務内容：システム・ネットワークエンジニアとして
    JPドメイン名の安定運用を支える
  - 最近の願いごと：1日が48時間になればいいのに・・・

昨年に引き続き、われわれ2名が担当します

本日の内容

「DNSのメッセージサイズについて考える」

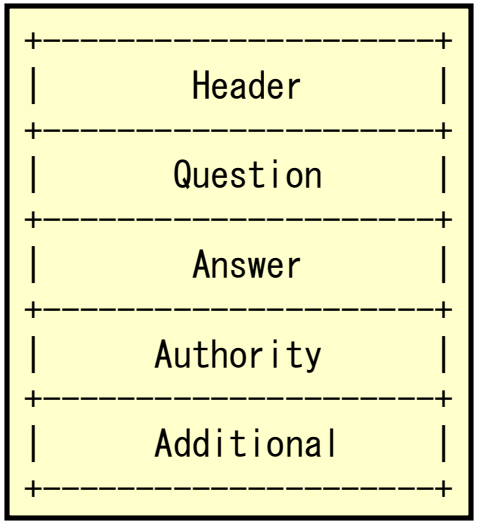
- メッセージサイズ決定の背景
- メッセージサイズの検討と標準化の歴史
- メッセージサイズに関する
  最新トピックスと検討状況
- 考察・まとめ

本日は特に「DNSのUDPメッセージサイズ」に着目します

メッセージサイズ決定の背景

基本的なDNS通信の流れ

- RFC 1034/1035（1987年）
  - UDPとTCPの双方を使用
  - 問い合わせと応答のフォーマットが同一
  - UDPメッセージサイズは512バイトまで
- RFC 1123（1989年）
  - 最初にUDPで問い合わせることが必須
    - ゾーン転送を除く
  - RFC 5966（2010年）で「問い合わせるべき」に緩和
    - ただし、TCPを先に使うには理由が必要
- 上記から導かれる基本的なDNS通信の流れ



DNSメッセージフォーマット
（RFC 1035より）

① 最初は常にUDPで問い合わせる
② 応答が512バイトを超えた場合、
TCPで同じ内容を再問い合わせする（TCPフォールバック）

「512バイトまで」の理由

- その由来はIP（IPv4）の仕様にまで遡る
- RFC 791（1981年）

> “All hosts must be prepared to accept datagrams of up to 576 octets (whether they arrive whole or in fragments). It is recommended that hosts only send datagrams larger than 576 octets if they have assurance that the destination is prepared to accept the larger datagrams.”

IPv4では576オクテット（バイト）までのデータグラム（ヘッダーを含むパケット）の受信を義務付けている

IPv4における「576バイト」の理由

- 前ページの続き（576が選ばれた理由）

> “The number 576 is selected to allow a reasonable sized data block to be transmitted in addition to the required header information. For example, this size allows a data block of 512 octets plus 64 header octets to fit in a datagram. The maximal internet header is 60 octets, and a typical internet header is 20 octets, allowing a margin for headers of higher level protocols.”

512バイトのデータブロックと
64バイトのヘッダーを想定

UDPメッセージ制限の理由

- RFC 1035（DNSの現在の仕様）には理由の記載なし
- RFC 883（1983年）に記述あり
  - DNSの最初の仕様

> “A non-reliable (i.e. best effort) method of transporting a message of up to 512 octets. Hence datagram messages are limited to 512 octets.”

非信頼型が512まで。従ってメッセージは512に制限

- RFC 791でTCPを「reliable communicationのためのプロトコル」と定義
  - つまり、「non-reliable method」はUDPを想定
- この値がRFC 1035に引き継がれた

512 + 8 + 20 < 576

- 結果として、

① メッセージサイズ（512バイト）
② UDPヘッダー（8バイト）
③ 一般的なIPヘッダー（20バイト）

の合計が576を超えない、つまりIPv4ネットワークにおいて1パケットで送受信可能なサイズが、DNSにおけるUDPメッセージサイズとして採用された

大事なポイント「1パケットで送受信可能」

メッセージサイズ決定のメリット

- メッセージサイズが512バイト以下である場合、やりとりが必ず1往復で完結する
  - 信頼性が高くない通信路でも実用的に使える
- 通信のコスト（負荷）を低く抑えられる
  - 名前解決に必要なコスト（負荷）を低くできる
- レイテンシー（問い合わせの送信から応答の受信までにかかる時間）を小さく抑えられる
  - 名前解決までの時間を短くできる

DNSがインターネットの急成長を支えられた理由の一つ

メッセージサイズ決定のデメリット

- メッセージサイズが512バイトを超える場合、送信側・受信側双方で追加の処理が必要になる
  - TCPフォールバック
- 最初からTCPを使うよりも更にコストが高くなる
  - TCPフォールバックのネゴシエーション処理
    ⇒ 名前解決にかかるコストがより高くなる
- 同じく、レイテンシーが更に大きくなる
  - TCPフォールバック完了までの通信時間
    ⇒ 名前解決にかかる時間がより長くなる

これらは後に「DNSの512の壁」と呼ばれることとなった

メッセージサイズの検討と標準化の歴史

基本仕様の決定（1981～1989年）

- RFC 791（IPv4の仕様）（1981年）
  - IPv4における「576バイト」
- RFC 882/883（DNSの最初の仕様）（1983年）
  - UDPにおける「512バイト制限」
  - RFC 1034/1035（現在の仕様）（1987年）も同様
- RFC 1123（インターネットホストの要求仕様）（1989年）
  - 「最初の問い合わせはUDPで」
  - TCPフォールバックの義務付け

DNSキャッシュポイズニングとDNSSECの標準化作業（1990～1997年）

- 1990年に書かれたS. Bellovin氏の論文をきっかけにDNSキャッシュポイズニングが関係者の間で問題視され、DNSSECの標準化開始
- Bellovin氏の論文（1990年執筆、1995年発表）
  - “Using the domain name system for system break-ins.”
    <https://www.cs.columbia.edu/~smb/papers/dnshack.pdf>
- より詳しい経緯は2009年のJPRSランチセミナー「桃栗三年柿八年、DNSSECは何年？」資料を参照
  <http://jprs.jp/tech/material/iw2009-lunch-L1-01.pdf>

DNSSECの標準化（1997年）

- RFC 2065（DNSSECの最初の仕様）（1997年）
- 仕様は標準化されたが・・・、

> “However, larger keys increase the size of the KEY and SIG RRs. This increases the chance of DNS UDP packet overflow and the possible necessity for using higher overhead TCP in responses.”

DNSSEC（鍵、署名）によりメッセージサイズが増大し、
それに伴うUDPパケットのオーバーフローと
TCPの高いオーバーヘッドが問題となった

UDPメッセージサイズの拡張の検討（1997年～1999年）

- 512バイトを超える場合に起こる問題をどう解決すべきか？
  - 処理の複雑化（TCPフォールバック）
  - コストの増加
  - レイテンシーの増大
- 考えられる二つの解決方法
  - 最初からTCPで送る（TCPフォールバックをやめる）
    - UDPによるメリットも失われる
  - UDPで送受信可能なメッセージサイズを拡張する
    - UDPによるメリットは維持される

こちらが採用された

最初の標準化作業（1997～1998年）

DNSSECの最初のRFC（RFC 2065）の著者

- D. Eastlake氏によって進められた（標準化には至らず）
  - Bigger Domain Name System UDP Replies
    <http://tools.ietf.org/html/draft-ietf-dnsind-udp-size-02>（1997～1998年）
- DNS問い合わせのRCODEを使用し、受け取り可能なUDPメッセージサイズをサーバーに伝達
  - 例：RCODE=4なら3200バイトまで
- 同時に、UDPメッセージサイズのデフォルトを、512バイトから1280バイトに拡張

RCODEは
DNS応答でのみ使用

> “1280 bytes of DNS data is chosen as the new default to provide a generous allowance for IP headers and still be within the highly prevalent approximately Ethernet size or larger MTU and buffering generally available today.”

Ethernetのサイズが基準（「1パケットで送受信」を重要視）

EDNS0（1999年）

- RFC 2671、著者：P. Vixie氏
  - 前述のI-Dのアイディア（RCODE流用）もVixie氏によるもの
- フラグビットやRCODEなど、UDPメッセージ（ペイロード）サイズの拡大以外のDNS機能拡張も含む
- 最大65535バイトのUDPメッセージを取り扱い可能
  - TCPメッセージと同じ大きさ
- 前述のI-D同様、Ethernetに言及

> “Choosing 1280 on an Ethernet connected requestor would be reasonable.”

Ethernet接続では1280バイトがリーズナブル

DNSSECbis（2005年）

- DNSSECの運用上の弱点などを改良、RFC 4033～4035として標準化
- 現行のDNSSECの基本仕様

権威DNSサーバーの仕様

> “A security-aware name server MUST support the EDNS0 ([RFC2671]) message size extension, MUST support a message size of at least 1220 octets, and SHOULD support a message size of 4000 octets.”

> “A security-aware resolver MUST support a message size of at least 1220 octets, SHOULD support a message size of 4000 octets.”

キャッシュDNSサーバーの仕様

EDNS0（=UDP）メッセージサイズとして
「少なくとも1220バイトをサポートしなければならず、
4000バイトをサポートすべき」と規定

改定版EDNS0（2013年）

- RFC 6891
  - EDNS0（RFC 2671）の改定版

> “Choosing between 1280 and 1410 bytes for IP (v4 or v6) over Ethernet would be reasonable.”

> “A good compromise may be the use of an EDNS maximum payload size of 4096 octets as a starting point.”

Ethernet接続におけるリーズナブルな値が、
1280バイトから1410バイトの間に改定

最大ペイロード（メッセージ）サイズの「良い妥協点」を
4096バイトと記載

まとめ：UDPメッセージサイズの変化

- 最初の基本仕様（RFC 883）（1983年）
  - 512バイトまでに制限
- EDNS0（RFC 2671）（1999年）
  - 制限緩和、1280バイトがリーズナブル
- DNSSECbis（RFC 4035）（2005年）
  - 1220バイト必須、4000バイトをサポートすべき
- EDNS0改定版（RFC 6891）（2013年）
  - 1280から1410バイトの間がリーズナブル
  - 最大メッセージサイズの良い妥協点は4096バイト

時代を経るにつれて徐々に大きくなっていった

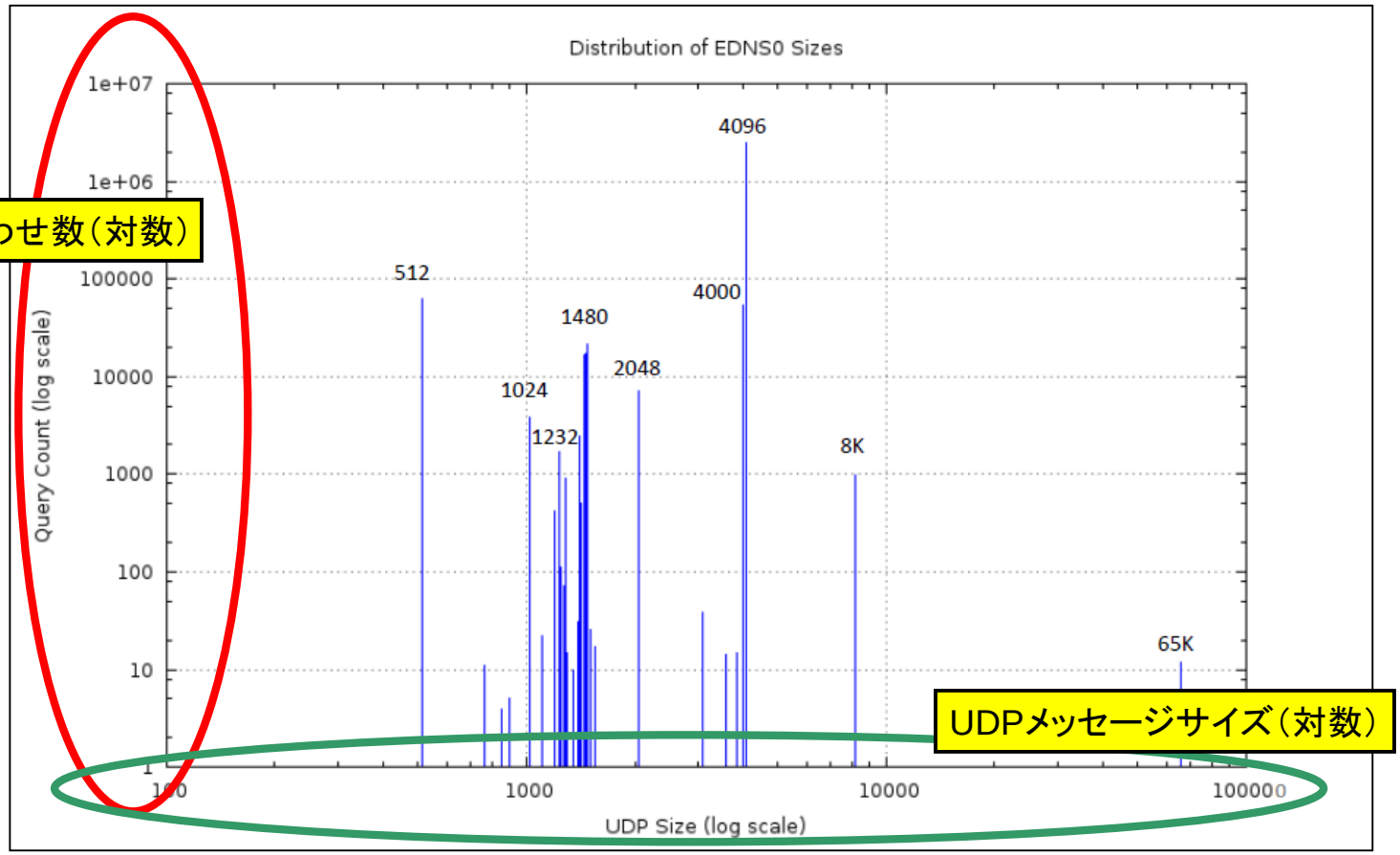
メッセージサイズに関する最新トピックスと検討状況

本日取り上げるトピックス

1. UDPメッセージサイズの現状
2. UDPメッセージサイズ増大による影響

1. UDPメッセージサイズの現状

- G. Huston氏による調査



<https://ripe67.ripe.net/presentations/112-2013-10-16-dns-protocol.pdf>より引用

UDPメッセージサイズの現状（続き）

- 観測された主な値
  - 512、1024、1232、1480、2048、4000、4096、8192、65535
- 数多くのサーバーが512よりも大きい値に
  - JP DNSサーバーにおける観測（統計情報）もこれを裏付け
    - 総IPアドレス数の70%程度が、512バイトより大きい応答を許容
    - 出典：JP DNS Update（Internet Week 2010）
      <https://www.nic.ad.jp/ja/materials/iw/2010/proceedings/d2/iw2010-d2-02.pdf>
- 4000と4096の数が多い
  - 4000：DNSSECにおいてサポートすべき値
  - 4096：BIND 9におけるデフォルト値

現状、数多くのDNSサーバーが4096をサポートしている

2. UDPメッセージサイズ増大による影響

本日取り上げる二つの話題

① データの増幅率が大きくなる

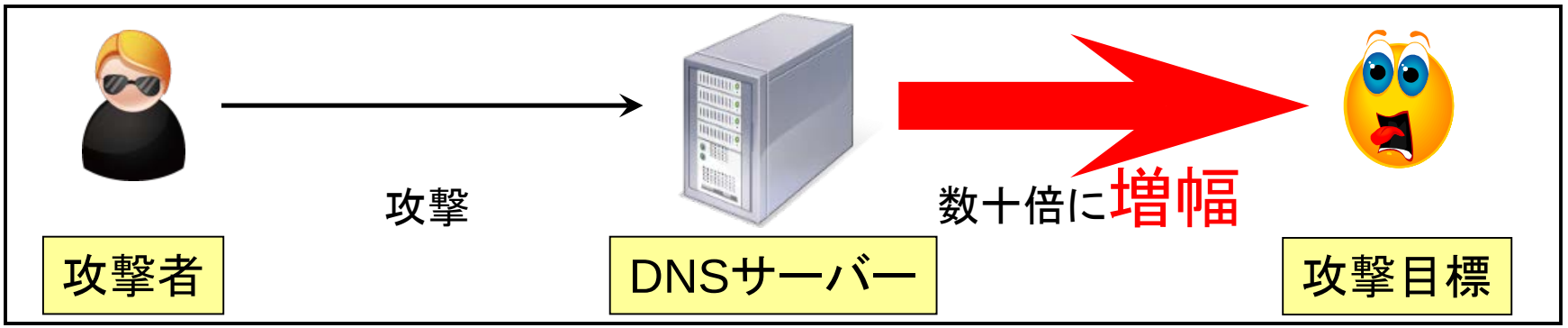
– DNSリフレクター攻撃のリスクが増大する

② 応答が1パケットに収まらなくなる

– IPフラグメンテーションが発生する

おさらい：DNSリフレクター攻撃の概要

- DNSが持つ特性を悪用



① 応答のサイズが問い合わせより大きい
② 主な通信がUDPである
③ 広く普及している

⇒

① リフレクターとして悪用可能
② 発信元の詐称が容易
③ 大規模な攻撃が可能

最近のトピックス：DNSリフレクター攻撃の進化（洗練）

- 従来からのTXTレコードやANYレコードに加え、多数のAレコードを攻撃に使用する事例が観測され始めている
- TXTレコードに対するフィルタリングやANY-to-TCPなどの対策を受けたものと考えられる

実例(現在は既に消されている)

```
$ dig ****.ru
...
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 9418
;; flags: qr rd ra; QUERY: 1, ANSWER: 238, AUTHORITY: 2, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
;; QUESTION SECTION:
;****.ru.                IN A

;; ANSWER SECTION:
****.ru.                3179 IN   A **.224.**.49
****.ru.                3179 IN   A **.224.**.136
****.ru.                3179 IN   A **.224.**.63
...

;; AUTHORITY SECTION:
****.ru.                345178 IN NS dns2.******.ru.
****.ru.                345178 IN NS dns1.******.ru.

;; Query time: 9 msec
;; SERVER: 127.0.0.1#53(127.0.0.1)
;; WHEN: Mon Oct 28 14:11:45 2013
;; MSG SIZE  rcvd: 3889
```

極めて多数の
Aレコード

応答の大きさを
4000バイトよりも
少しだけ小さく設定

この方法のポイント

- 攻撃の効率が高い
  - 応答が4000バイト以下のできるだけ大きな値になるように、Aレコードの数を調整可能
- 元ネタを作りやすい
  - Aレコードは大抵のDNSサービスにおいて登録可能
- 事前にフィルターしにくい
  - Aレコードは必要不可欠なレコードの一つ
- キャッシュの状況によらず同じ（大きな）応答を返す
  - ANYと異なる
    - djbdns（tinydns）では、任意の8つのAをランダムに選んで応答する（応答は常に512バイトを超えない）

IPフラグメンテーションの概要

- IPにおいて1回で送れる最大の単位（MTU）よりも大きなデータを送る場合に発生
  - 経路（Path）MTUより大きなUDPパケットの送信時に必ず発生
- IPの仕様上有害であることは古くから認識
  - Fragmentation Considered Harmful（1987年）
    <http://www.hpl.hp.com/techreports/Compaq-DEC/WRL-87-3.pdf>
- DNSのUDPメッセージサイズの検討においても、IPフラグメンテーションを防ぐ配慮がされていた
  - 基本仕様における「512 + 8 + 20 < 576」
  - EDNS0における「1280」「1280から1410の間」

IPフラグメンテーションとDNSのUDPメッセージサイズの関係

- EDNS0の著者（P. Vixie氏）も危険性を認識していた
  - 2013年9月9日のdns-operations MLの投稿

> “regrettably, the author of RFC 2671 knew the dangers and limitations of fragmented IP, but specified it anyway.” <https://lists.dns-oarc.net/pipermail/dns-operations/2013-September/010704.html>

「危険性と制限を知っていたが、それ（EDNS0）を策定した」

- しかし、DNSSECはIPフラグメンテーションの発生を前提として開発された（RFC 4035）

> “A security-aware resolver's IP layer MUST handle fragmented UDP packets correctly regardless of whether any such fragmented packets were received via IPv4 or IPv6.”

あえて言うなら「みんな悪い」

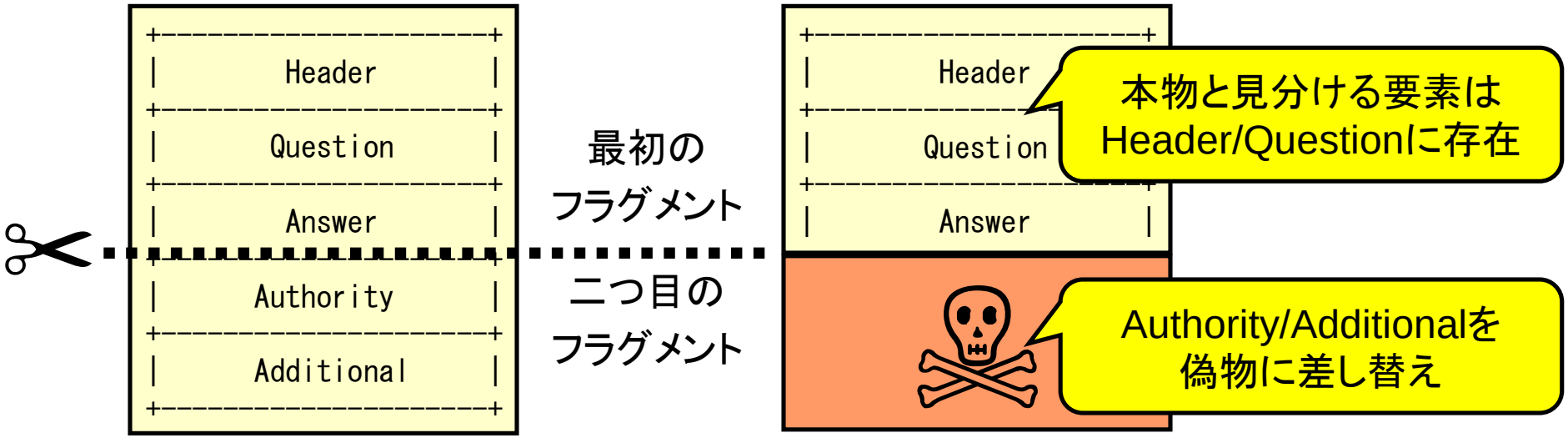
注：「誰が悪いのか」という話をしたいわけではありません

最近のトピックス：第一フラグメント便乗攻撃

- 原文名称：“1st-fragment piggybacking attacks”
  - IPフラグメンテーションの弱点を悪用した、新たなDNSキャッシュポイズニング攻撃手法
- 2012年5月17日にイスラエル・バル＝イラン大学のA. Herzberg教授とH. Shulman氏により発表された論文、“Fragmentation Considered Poisonous”で報告
  - この時点ではDNS関係者の間では大きな話題にはならず
- 2013年8月1日にIETF 87 saag（Security Area Advisory Group）の招待講演において発表
  - “DNS Cache-Poisoning: New Vulnerabilities and Implications, or: DNSSEC, the time has come!”
    <http://www.ietf.org/proceedings/87/slides/slides-87-saag-3.pdf>
  - 発表後、dns-operations MLで大きな話題に

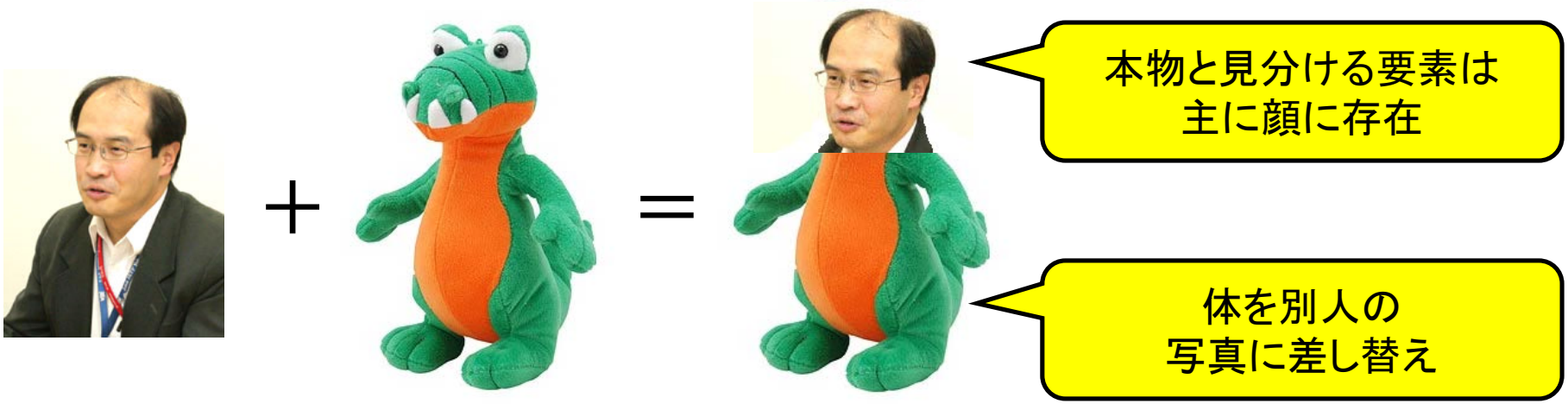
攻撃の概要

- 攻撃対象：フラグメントされたUDPによるDNS応答
- 応答の二つ目（以降）のフラグメントを偽物に差し替え
- DNS応答の同定に使える要素が、最初のフラグメントにしか存在しなくなることを悪用
  - IPアドレス（IPヘッダー）、ポート番号（UDPヘッダー）
  - 問い合わせID、問い合わせ名（DNSメッセージ）



攻撃の概要（続き）

- 顔写真に別人の写真をつなぎ合わせる（コラージュ）行為に相当
- 類似点
  - 同定の要素の多くが「顔」に存在している
  - 全体としてうまくつなぎ合わせる必要がある



ぬいぐるみ写真引用元: <http://pixabay.com/ja/> (Public Domain CC0)

この方法のポイント

- リアセンブリー（フラグメントの再構成）の際に使用するIdentificationフィールドの大きさが、IPv4では16ビットしかないために成立しうる
  カミンスキー型攻撃手法と同じ予測可能性
- IPv6では同フィールドの大きさは32ビットだが、外部から値が予測可能な実装が多く存在する（3.0.0以前のLinuxなど）
- 二つ目の偽フラグメントを一つ目のフラグメントよりも先に送り込む「待ちぶせ」攻撃が可能
  - IPの仕様により、順番が入れ替わっても再構成される
  - これによりキャッシュポイズニングの確率が向上
- チェックサムは防御の手段になり得ない
  - 「応答の内容」と「どこで切れるか」がわかっていれば、同じ値になるように偽造データの一部を細工可能

二つの攻撃手法（アタックベクター）

- フラグメントが発生する大きなDNS応答を狙う
  - Shulman氏の論文に書かれている方法
    - DNSSECに対応したドメイン名のDNSKEY RR
    - 登録済ドメイン名に長い名前のNSを登録
- IPフラグメンテーションを意図的に発生させる
  - 2013年10月のRIPE Meetingにおいて、CZ.NICのT. Hlavacek氏が発表した方法
    - IP fragmentation attack on DNS
      <https://ripe67.ripe.net/presentations/240-ipfragattack.pdf>
  - 偽のICMP PacketTooBigを送りつけてMTUが小さいと誤認させ、応答パケットをフラグメントさせる
  - RIPE Meetingにおいて同氏がPoCのデモを実施

考えうる対策例

- BCP38の適用
- DNSSECの適用
  - DoS攻撃（名前解決の妨害）は不可避
- IPv6におけるIdentificationのランダム化
  - IPv4ではランダム化のみでは不十分（16ビットしかない）
- IPフラグメンテーションのリスク低減
  - EDNS0のUDPメッセージサイズを小さく設定
    - TCPフォールバックの発生増加を考慮する必要あり
  - DNSKEYレコードをできるだけ小さく設定
    - JP DNSサーバーにおいて2011年に実施済
- ICMP type=3, code=4を無視
  - 意図的なフラグメント発生への対策
  - 本当にPath MTU Discovery（PMTUD）が必要な場合に困る

考えうる対策例（続き）

- DNS側における対策
  - 応答にランダムなRRを付与する
    - チェックサムの値をずらす
  - NSのキャッシュ方法を工夫する
    - Google Public DNSが採用
  - DNS Cookiesの導入
    - EDNS0（OPT RR）によりDNSパケットにクッキーを追加
    - D. Eastlake氏が2006～2008年に提案、未RFC化
      - Domain Name System (DNS) Cookies <http://tools.ietf.org/html/draft-eastlake-dnsext-cookies-03>
    - IETF 88 dnsop WGの議事録より
      - “Donald Eastlake announced revival of DNS cookies”

考察・まとめ

1. DNSのUDPメッセージサイズはいくつが適切なのか？

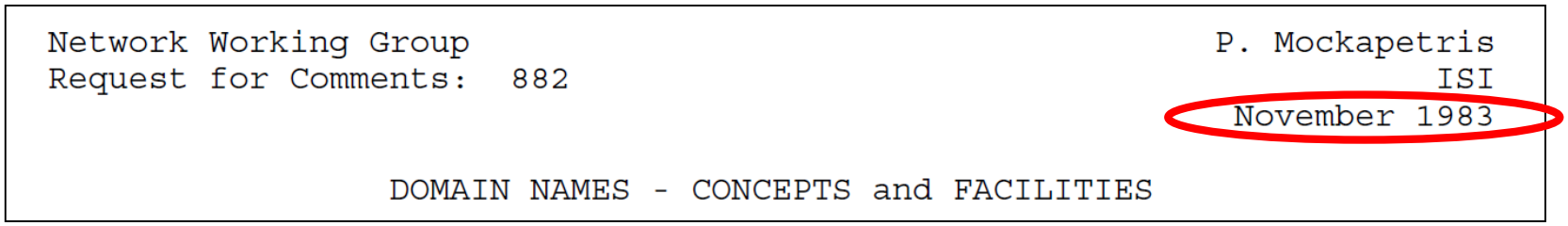
- 現状を考えた場合、
  4000（4096）バイトは少し大きすぎたのかもしれない
  - 頻発するDNSリフレクター攻撃
  - IPフラグメンテーションの悪用（第一フラグメント便乗攻撃）
  - IETFにおける、IPv6のIPフラグメンテーション廃止の動き
- しかし、512バイトは現状においてあまりにも小さい
  - DNSSEC、IPv6、SPF/DKIM、DANEなど
- 改定版EDNS0（RFC 6891）における「Ethernetにおけるリーズナブルな値」は一つの指標であると言える
  - 1280バイトから1410バイトまでの間

2. DNSはそもそもUDPでいいのか？

- UDPを使い続ける限界は従来から指摘されている
  - TCPに全面移行してWeb技術のノウハウを活かせばよい？
- 実際問題として、本当にTCPに移行できるのか？
  - 負荷・コストの問題
    - 特に、ルート・TLDサーバーや大規模なキャッシュDNSサーバーなどにおけるスケーラビリティの問題
  - レイテンシーの問題
    - UDPの場合よりも確実に増大
- TCPに移行する際に考慮が必要そうなこと（例）
  - DoS攻撃耐性の確保
    - DoS攻撃の方法もノウハウがたくさんある
  - UDPであることを前提としている既存のサービスへの影響
    - BGPを利用した広域のIP Anycast（RFC 3258）など

3. そして・・・これからのDNSは？

- DNSは今月、誕生から30周年を迎えました



Network Working Group
Request for Comments: 882

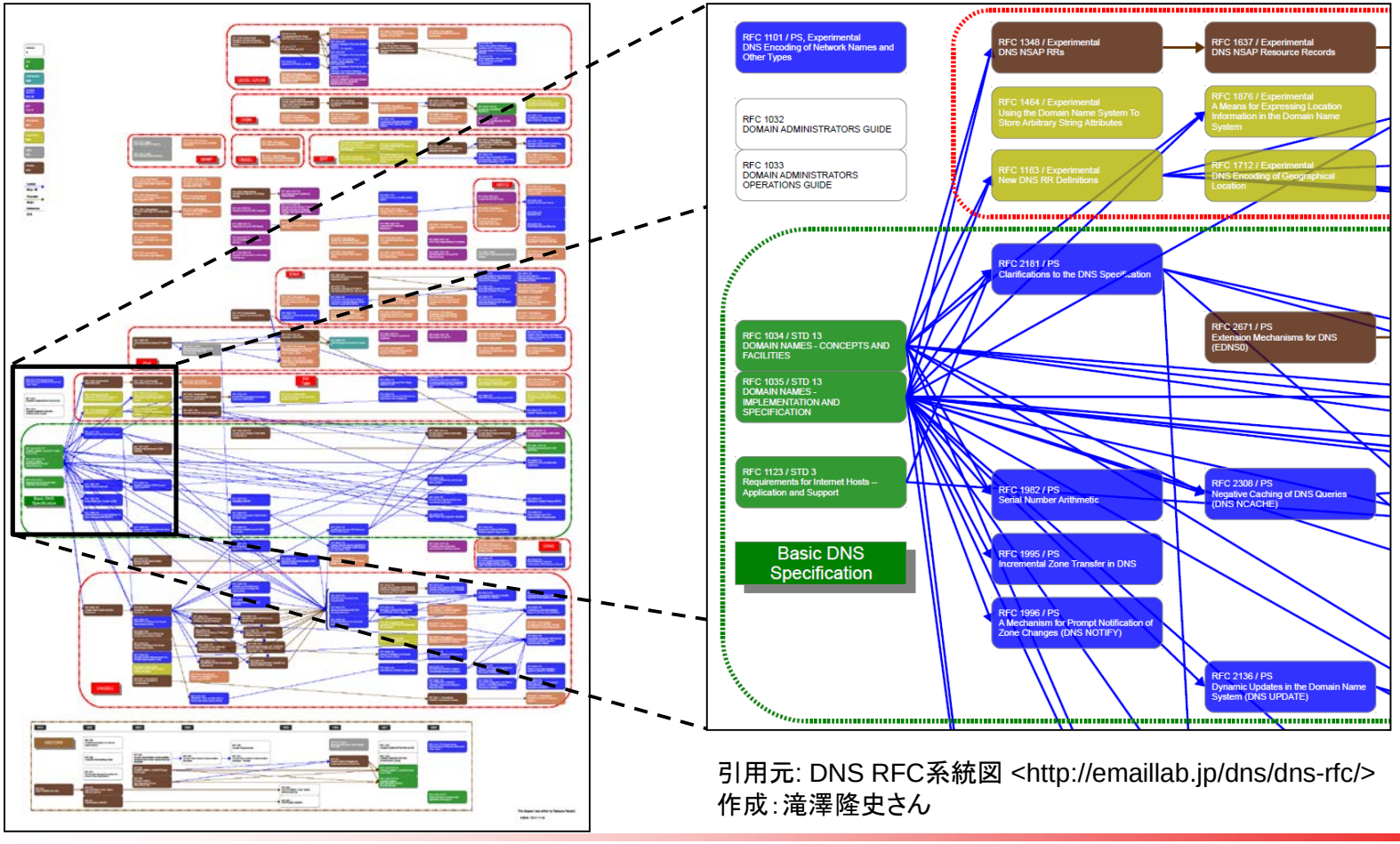
P. Mockapetris
ISI
November 1983

DOMAIN NAMES - CONCEPTS and FACILITIES

- インターネットにおいて大成功した技術の一つ
  - 広く普及
  - 長期にわたり基盤技術として利用
  - DNSを利用したさまざまな応用技術の存在
- しかし・・・・・

DNSの30年 = インターネットの30年

- あまりにも多くのClarification（明確化）とUpdate（更新）



引用元: DNS RFC系統図 <http://emaillab.jp/dns/dns-rfc/>
作成：滝澤隆史さん

これからも続く「戦いの道」

- そして、数多く存在する「運用でカバー」
- しかしそれは、30年にわたり皆でさまざまな工夫を重ね、DNSを維持・発展させてきた戦いの足跡であるとも言える
- 今回取り上げたDNSのUDPメッセージサイズの問題もまた、その典型的なものの一つ
  - RFCのこうした表記にもその片鱗が現れている
    - リーズナブルであろうEDNS0メッセージサイズ（RFC 6891）
    - TCPを先に使うに足る運用上の理由（RFC 5966）

そしてこれからも、その戦いは続いていく

Q & A