Knot DNSを使ってみた
- DNS Summer Days 2014

NTTコミュニケーションズ株式会社
先端IPアーキテクチャセンタ
高田 美紀
2014/6/27

自己紹介

- 1993～ 株式会社NTTPCコミュニケーションズ
  - ISP (InfoSphere) サーバの運用
  - ホスティング (WebARENA) 立ち上げ～開発～運用
  - 主にDNS、メールシステム担当
- 2013/4～ NTTコミュニケーションズ株式会社
  - 先端IPアーキテクチャセンタ
    - ✓ R&D部門
    - ✓ DDoS対策技術、DNSまわりでの事業部サポート、対外活動
- 対外活動
  - dnsops.jp 幹事
  - ときどき JANOG meeting スタッフ、などなど
- エンジニア＋母親業の両輪で活動中

Knot DNS とは?

- CZ(チェコ共和国) NIC製のDNSサーバソフトウェア
  - 権威DNS専用
  - オープンソース
  - 高速、マルチスレッド、大部分はlock free
    - ✓ SMPで適切にスケールするようデザイン
    - ✓ userspace-rcuを利用してlockを減らしている
  - 主なDNSプロトコルをサポート
    - ✓ AXFR/IXFR, TSIG, EDNS0, NSID (RFC 5001)
    - ✓ DNSSEC with NSEC3 (automatic signing)
    - ✓ Response Rate Limiting
    - ✓ Dynamic DNS, オンラインでのzoneの追加/削除
  - そのほかおまけ的機能
    - ✓ Auto Forward/Reverse Zone
    - ✓ disable-any

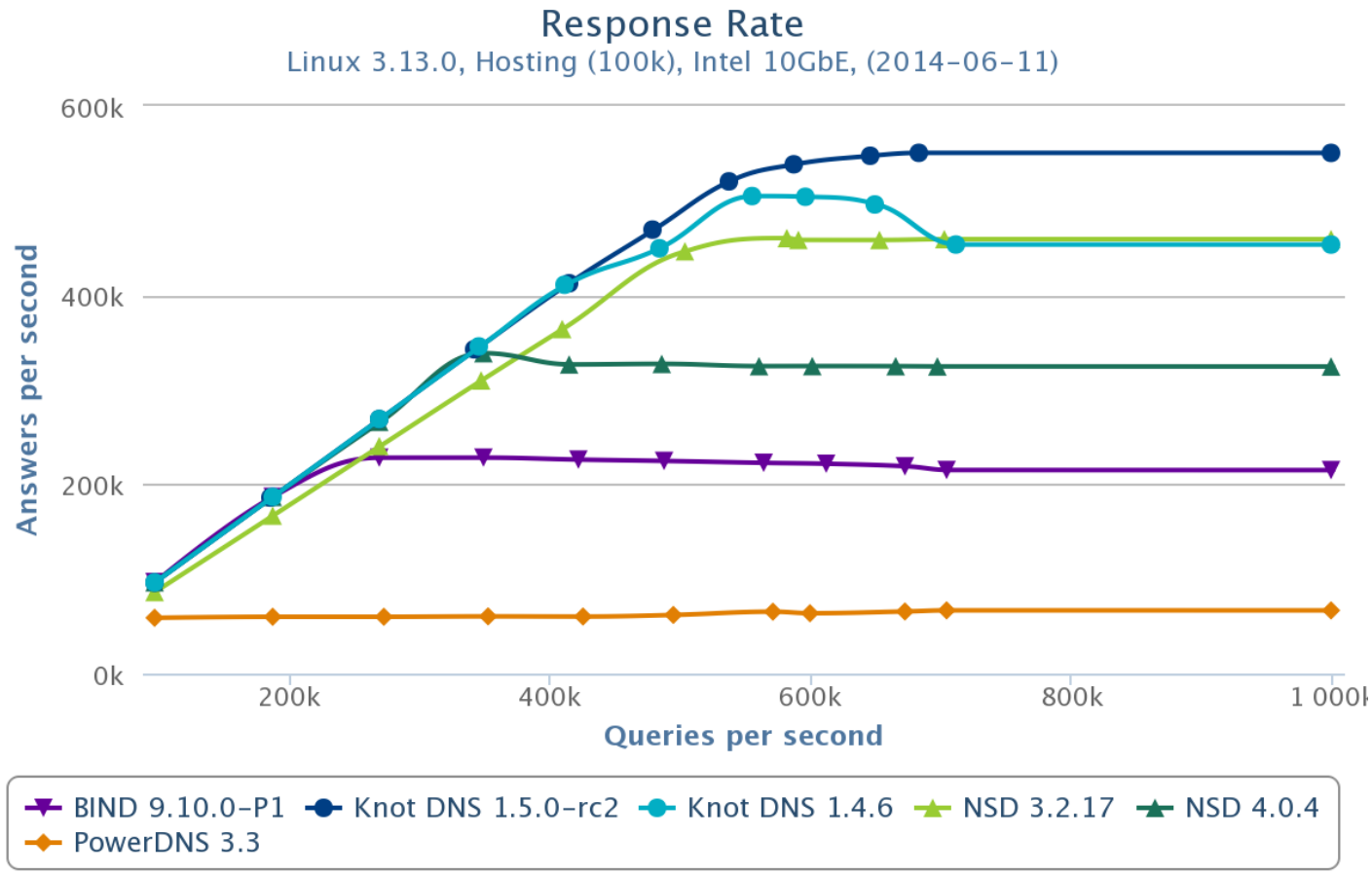
実績?

- Knot DNS update@2013 ENOG6/RIPE NCC Regional Meeting
  - http://www.enog.org/presentations/enog-6/210-KNOT-ENOG6-20131002-JT.pdf
  - .CZ, .DKのroot
  - いくつかのチェコ共和国のレジストラ/ホスティング事業者
  - ルートサーバでもテストした
    - ✓ ICANN(I), RIPE NCC(K)

高速らしいと聞いて: ベンチマーク

- 伊藤高一さんのベンチマーク@2012年
  - http://www.kkdlabs.jp/dns/benchmark-2012/ja/
    - ✓ BIND 9.9.1-P1 vs NSD 3.2.10 vs Knot DNS 1.0.6
  - qps: BINDの6倍以上、NSDの2.8倍程度、高速
  - Knot DNSのみlost発生せず=>まだ余力があったのでは
- Knot DNS本家のベンチマーク@2014年
  - https://www.knot-dns.cz/pages/benchmark.html
    - ✓ BIND 9.10.0-P1 vs Knot DNS 1.5.0-rc2,1.4.6 vs NSD 3.2.17,4.0.4 vs PowerDNS 3.3
  - qps: 次ページ
  - 起動時間: BINDと同じ～2倍程度、高速
  - メモリ使用量: Knot DNS 1.4.6ではBINDの2.5倍多く使用
    - ✓ 1.5.0-rc2ではBINDより少ない使用量
  - Dynamic DNS処理スピード
    - ✓ 100万RRの巨大ゾーンではBINDより3倍程度遅い

Benchmark: Response Rate/Hosting(100k)



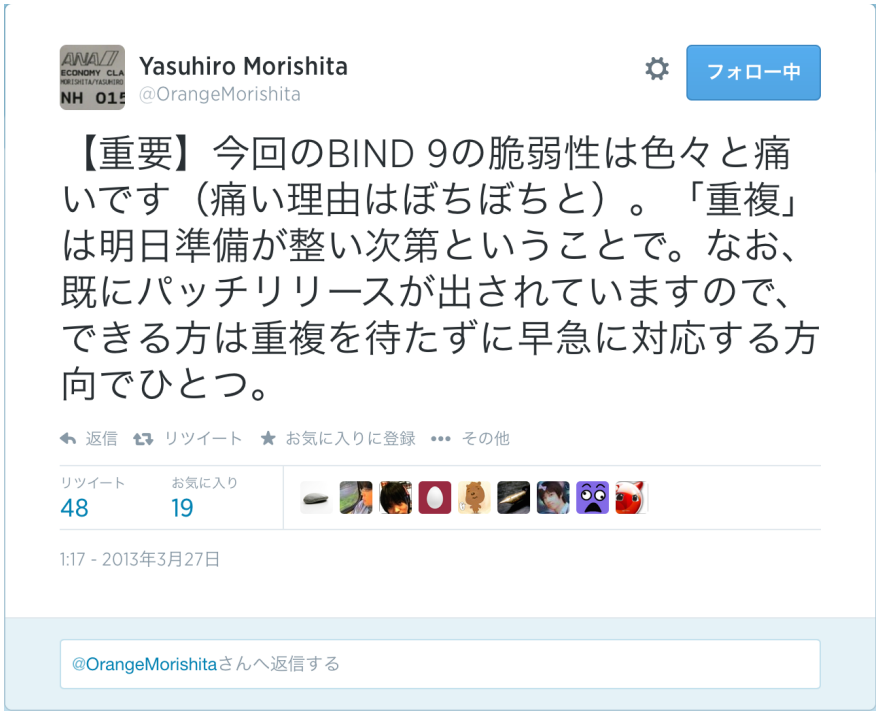
脱BIND

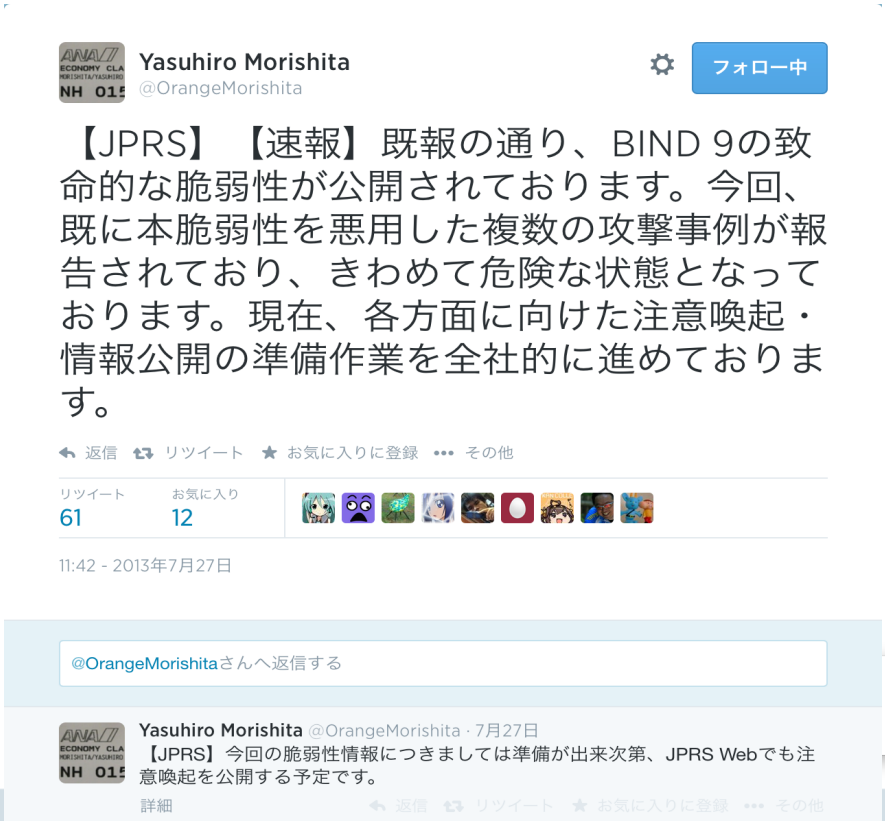
- DNSサーバソフトウェア == BIND?
- ここがいけてないよBINDさん
  - 権威DNS機能とキャッシュDNS機能が分離されてない
  - 機能が豊富すぎる
  - デフォルト設定がいけてない
    - ✓ メモリ使い切っちゃったりとか
  - 設定がわかりづらい
    - ✓ notify まわり、allow-なんとか
  - バージョンによって同じ設定で挙動が違う
  - やばい設定を書けてしまう
    - ✓ ソースポート固定とか
  - 巨大すぎる

■脆弱性の多さ

この支配からの卒業

- 「重複」から卒業しませんか !?
  - キャッシュDNS: Unbound
  - 権威DNS: NSD, Knot DNS, PowerDNS, etc..
- 参考: DNS実装ダイバーシティの話@IW 2012 DNS DAY
  - https://www.nic.ad.jp/ja/materials/iw/2012/proceedings/d2/





Knot DNS: 使い方

9

インストール

- tarballダウンロード
  - https://www.knot-dns.cz/pages/download.html
- cat README
  - yum なり apt-get なりでbuildに必要なパッケージを入れる
  - configure して make, make install

```
$ sudo apt-get install git-core libtool autoconf flex bison libssl-dev liburcu-dev
or
$ yum install libtool autoconf flex bison openssl-devel userspace-rcu-devel
$ ./configure && make
$ sudo make install
```

- Knot DNS用に knot ユーザ作成
- /var/run/knot, /var/lib/knot ディレクトリ作成、knotユーザで書き込めるようにしておく

```
$ useradd -g users knot
$ mkdir -p /var/run/knot /var/lib/knot
$ chown knot.users /var/run/knot /var/lib/knot
```

knotc

- knotdコントロール用ユーティリティ
  - rndc みたいなもの
  - knotc用の設定ファイルは *ない*
    - ✓ simple is best!
  - デフォルトではUNIXドメインソケット経由
- コマンド
  - stop, reload, refresh [zone]
  - flush
    - ✓ journalファイルをflushしてzoneに反映
  - status, zonestatus, memstats [zone]
  - checkconf, checkzone [zone]
    - ✓ knot.confをreloadする前に確認できる
  - signzone [zone]
    - ✓ DNSSECの署名動作

knotc

```
# /usr/local/sbin/knotc
Usage: knotc [parameters] <action>

Parameters:
 -c, --config <file>    Select configuration file.
 -s <server>            Remote UNIX socket/IP address (default /var/run/knot/knot.sock).
 -p <port>              Remote server port (only for IP).
 -y <[hmac:]name:key>   Use key specified on the command line.
 -k <file>              Use key file (as in config section 'keys').
 -f, --force            Force operation - override some checks.
 -v, --verbose          Verbose mode - additional runtime information.
 -V, --version          Print knot server version.
 -i, --interactive      Interactive mode (do not daemonize).
 -h, --help             Print help and usage.

Actions:
 stop                   Stop server.
 reload                 Reload configuration and changed zones.
 refresh [zone]         Refresh slave zone (all if not specified). Flag '-f' forces retransfer.
 flush                  Flush journal and update zone files.
 status                 Check if server is running.
 zonestatus             Show status of configured zones.
 checkconf              Check current server configuration.
 checkzone [zone]       Check zone (all if not specified).
 memstats [zone]        Estimate memory use for zone (all if not specified).
 signzone [zone]        Sign all zones with available DNSSEC keys.
```

knotd.conf (システム的なとこ)

```
system {
  rundir "/var/run/knot";
  user knot.users;
  max-udp-payload 1220;
}

interfaces {
  all_ipv4 {
    address 0.0.0.0;
  }
  all_ipv6 {
    address [::];
  }
}

log {
  syslog { any info; }
}
```

権限分離ユーザで読み書きできることが必要
pidファイル、control用ソケットファイルなど

権限分離ユーザ

UDPのペイロードサイズ上限

Listenするインタフェースの設定
書かないとListenしない(!)
IPv4とIPv6を別々に書く必要がある

名前 (all_ipv4とか) は何でもok
名前に対する address は1行のみok

ログの出し方設定。syslog, stdout, stderr など

knotd.conf (マスターサーバ)

```
remotes {
  ns2_ipv4 {
    address 192.0.2.53@53;
  }
  ns2_ipv6 {
    address 2001:db8::53@53;
  }
}

zones {
  storage "/var/lib/knot";
  example.jp {
    file ”example.jp";
    xfr-out ns2_ipv4, ns2_ipv6;
    notify-out ns2_ipv4, ns2_ipv6;
  }
}
```

スレーブサーバのIPアドレス、ポートの定義
192.0.2.0/24 などサブネット単位でもok

ゾーンの設定

ワーキングディレクトリ

ゾーン定義、ゾーンファイルの場所

このアドレスからのゾーン転送のみ許可する

このアドレスにnotifyを送信する

knotd.conf (スレーブサーバ)

```
remotes {
  ns1_ipv4 {
    address 192.0.2.53@53;
  }
  ns1_ipv6 {
    address 2001:db8::53@53;
  }
}

zones {
  storage "/var/lib/knot";
  example.jp {
    file ”example.jp";
    xfr-in ns1_ipv4, ns1_ipv6;
    notify-in ns1_ipv4, ns1_ipv6;
  }
}
```

マスターサーバのIPアドレス、ポートの定義
192.0.2.0/24 などサブネット単位でもok

ゾーンの設定

ワーキングディレクトリ

ゾーン定義

ゾーンファイル。storageディレクトリ配下

このアドレスに対しゾーン転送を要求する

このアドレスからのnotifyのみ受け取る

Automatic forward/reverse records

- 指定したアドレス帯のA/AAAA/PTRを自動生成
  - BINDでいう$GENERATEみたいなの
  - (forward|reverse) <prefix> <ttl> <address>/<netblock>
- Knot DNS 1.5以降
- 設定例:

```
zones{ example. {
    file "ns_soa_only_zone";
    query_module {
      synth_record "forward dynamic- 86400 2001:db8::/32";
      synth_record "forward dynamic- 86400 198.51.100.0/25";
    }
  }
  100.51.198.in-addr.arpa {
    file "ns_soa_only_zone";
    query_module {
      synth_record "reverse dynamic- example. 86400 198.51.100.0/25";
    }
  }
  8.b.d.0.1.0.0.2.ip6.arpa {
    file "ns_soa_only_zone";
    query_module {
      synth_record "reverse dynamic- example. 86400 2001:db8::/32";
    }
  } }
```

Automatic forward/reverse records

■ 実行例:

```
$ kdig -x 2001:db8:dead:beef::53 @::1 +norec
;; ->>HEADER<<- opcode: QUERY; status: NOERROR; id: 5688
;; Flags: qr aa; QUERY: 1; ANSWER: 1; AUTHORITY: 0; ADDITIONAL: 0

;; QUESTION SECTION:
;; 3.5.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.f.e.e.b.d.a.e.d.8.b.d.0.1.0.0.2.ip6.arpa.
        IN      PTR

;; ANSWER SECTION:
3.5.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.f.e.e.b.d.a.e.d.8.b.d.0.1.0.0.2.ip6.arpa. 86400
        IN      PTR     dynamic-2001-0db8-dead-
beef-0000-0000-0000-0053.example.

;; Received 159 B
;; Time 2014-06-24 14:11:48 JST
;; From ::1@53(UDP) in 0.0 ms
```

disable-any

- ANYで問い合わせられたら、空のレスポンスにTCをつけて返す
  - 権威DNSサーバを狙ったDNS reflection attackの踏み台になりづらくなる
  - デフォルトではdisable

```
zones {
  disable-any on;
          :          :
}
```

```
zones {
  example.jp {
    file ”example.jp";
          :          :
    disable-any on;
  }
}
```

起動

```
$ sudo /usr/local/sbin/knotd -d
$ sudo tail /var/log/messages
Jun 24 16:09:00 ns1 knot[3974]: Knot DNS 1.5.0-rc2 starting.
Jun 24 16:09:00 ns1 knot[3974]: Binding to interface 0.0.0.0@53.
Jun 24 16:09:00 ns1 knot[3974]: Binding to interface ::@53.
Jun 24 16:09:00 ns1 knot[3974]: Configured 2 interfaces and 4 zones.
Jun 24 16:09:00 ns1 knot[3974]: Changing group id to '100'.
Jun 24 16:09:00 ns1 knot[3974]: Changing user id to '521'.
Jun 24 16:09:00 ns1 knot[3974]: PID stored in '/var/run/knot/knot.pid'
Jun 24 16:09:00 ns1 knot[3974]: Changed directory to /.
Jun 24 16:09:00 ns1 knot[3974]: Loading zones...
Jun 24 16:09:00 ns1 knot[3974]: Zone '100.51.198.in-addr.arpa.' will be loaded (serial 0)
Jun 24 16:09:00 ns1 knot[3974]: Zone 'example.jp.' will be loaded (serial 0)
Jun 24 16:09:00 ns1 knot[3974]: Zone 'example.' will be loaded (serial 0)
Jun 24 16:09:00 ns1 knot[3974]: Zone '8.b.d.0.1.0.0.2.ip6.arpa.' will be loaded (serial 0)
Jun 24 16:09:00 ns1 knot[3974]: Starting server...
Jun 24 16:09:00 ns1 knot[3974]: Zone '100.51.198.in-addr.arpa.' loaded (0 -> 2014062301).
Jun 24 16:09:00 ns1 knot[3974]: Zone 'example.jp.' loaded (0 -> 2014062301).
Jun 24 16:09:00 ns1 knot[3974]: Zone 'example.' loaded (0 -> 2014062301).
Jun 24 16:09:00 ns1 knot[3974]: Zone '8.b.d.0.1.0.0.2.ip6.arpa.' loaded (0 -> 2014062301).
Jun 24 16:09:00 ns1 knot[3974]: Server started as a daemon, PID = 3974
Jun 24 16:09:00 ns1 knot[3974]: Binding remote control interface to '/var/run/knot/
knot.sock' .
```

BINDからの移行

- 違うとこ
  - viewがない
  - 設定ファイル形式
  - ログの形式

viewは悪と考えましょう
BINDと心中したいですか?

NW構成を工夫する等して
viewを駆逐しましょう

- 同じとこ
  - ゾーンファイル形式はBINDと同じ
  - Dynamic DNSのインタフェースは同じ

まとめ

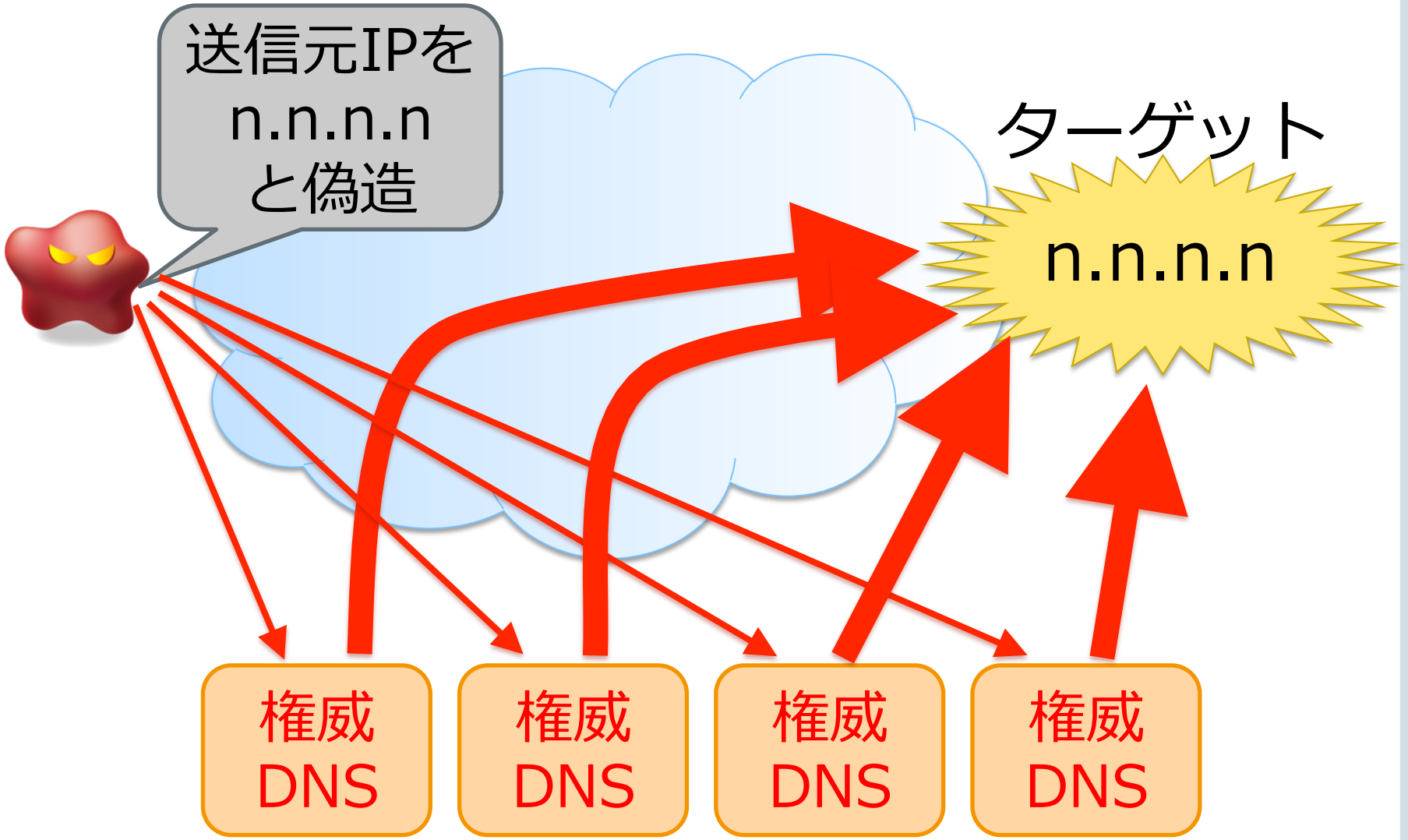
- Knot DNSのいいとこ紹介しました
  - わかりやすい
  - シンプル
  - 速い
  - 新しい設計
    - ✓ 競合を少なくした、とか
- BINDとさよならしよう
  - 選択肢はいくつもある
  - 権威/キャッシュの分離
  - viewも無くしてシンプルな世界に
- 「重複」に怯えない暮らしを!

おまけ: 権威DNSサーバでのDDoS対策

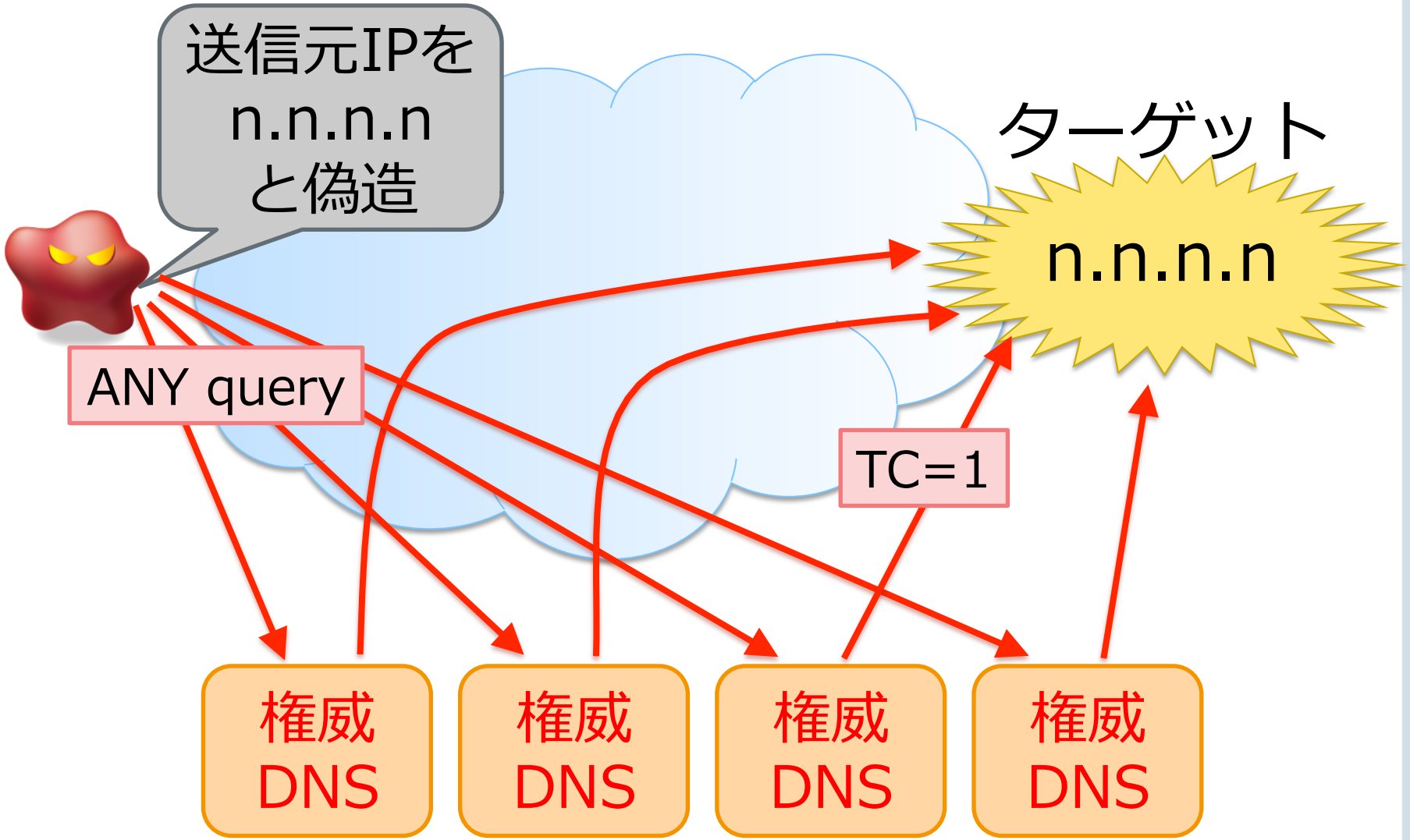
- 権威DNSサーバソフトウェアのDDoS対策機能として必要なもの
  - 大きな返答をsource IPへ返さない
    - ✓ 詐称されてるかもしれないから
    - ✓ TCPへのフォールバック (TC=1)
      - ✓ disable-any
  - 同じ返答を短時間にたくさん返さない
    - ✓ レートリミット
      - ✓ Response Rate Limiting

- これらは必須の機能となってきます
- しかし、ここまでやったとしても。。
  - 世界中のbotから1つずつ、別々の権威DNSサーバにqueryを送られると。。。

権威DNSサーバを使ったDDoSの仕組み



ANYへの対策: disable-any



その他RRの対策: Response Rate Limiting