

# 権威DNSサーバを作ってみよう

ゼロから始めるファジング生活

坂口 俊文

2018/06/27

# 自己紹介

坂口 俊文

- 4年前までは、ISPのメール・DNS...サーバの管理者
- 現在はとあるクラウドサービスのサポート
- 本日も個人参加
- Twitter: @siskrn
- GitHub: <https://github.com/sischkg/>

## はじめに

本日は、最近開発・利用しているDNSフルリゾルバ専用のファジングツールについて発表します。

春のJANOG Interimの応募しようとしたが、間に合いませんでした。

# 動機

これまで

- 公開されたDNSサーバの脆弱性を検証
- フルリゾルバの脆弱性を検証する場合は、特殊な応答を返す権威サーバを作成

課題

- 権威サーバの作成を効率化
- DNSSEC validation有効時のみ影響のある脆弱性にも対応

この課題の対応を検討中に、応答の自動生成によるフルリゾルバ専用のファジングツールの作成を思いつく

# ファジングとは

- バグや未知の脆弱性を検出するセキュリティテスト
- 問題が起きそうな様々な細工をしたデータを送り、検査対象に異常な動作が起きないかどうかを検査
- 問題を起こしそうなデータは、ファジングツールを使うと自動的に作成することができる

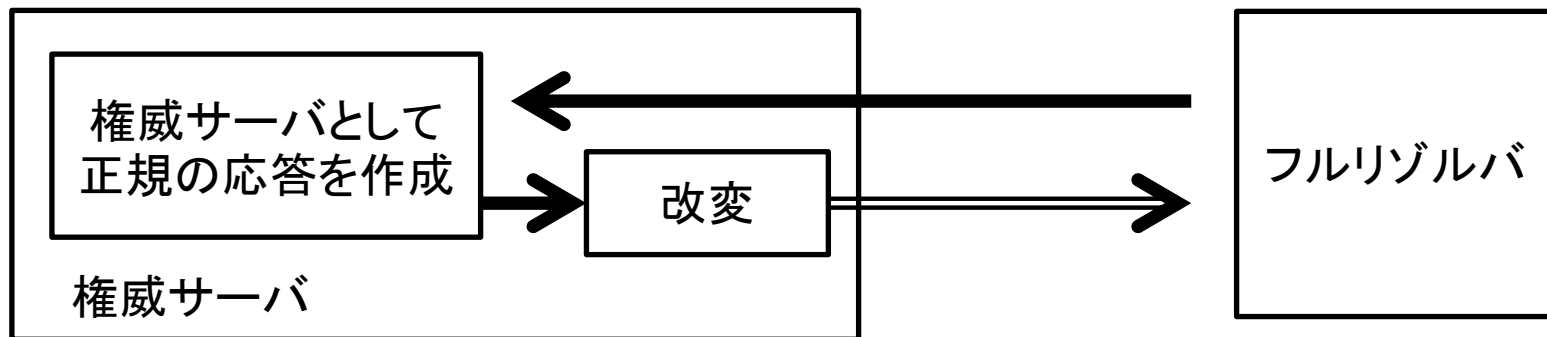
## 詳細

IPA : 情報セキュリティ : ファジング : FAQ

[https://www.ipa.go.jp/security/vuln/fuzz\\_faq.html#001](https://www.ipa.go.jp/security/vuln/fuzz_faq.html#001)

# ファジングツールの概要

- 最初に権威サーバとして正規の応答を作成し、それを改変してからフルリゾルバへ送信する
- 特殊な応答は乱数をもとに作成
- フルリゾルバが異常終了 (assert/segmentation fault)する不具合を探す
- 毒入れなどは検知が困難なため、対象外とする



# 権威サーバの実装

- DNSSECに対応 (RRSetの署名/NSEC/NSEC3)
  - 完全な権威サーバを実装しない
    - ゾーンはひとつのみ
    - 以下の機能は未実装
      - ワイルドカード
      - NSEC3 Opt-out
      - TSIG
- など

# ファジングの実装

過去の(BINDなどの)脆弱性を参考に実装

- RRの追加・変更・削除
  - RRを追加
    - CNAME/DNAME
      - CVE-2017-3137(<https://kb.isc.org/article/AA-01466/0/>)  
A response packet can cause a resolver to terminate when processing an answer containing a CNAME or DNAME
    - DNSSEC関連 (DNSKEY/RRSIG/DS/NSEC/NSEC3)
      - DNSSEC validation有効なときの脆弱性があったため
      - CVE-2017-15908(<http://blog.trendmicro.co.jp/archives/16583>)  
"Linux PC に対する DoS 攻撃が可能な「systemd」の脆弱性について解説"
    - TSIG, TKEYなどのRR
      - CVE-2016-9131(<https://www.fortinet.com/blog/threat-research/analysis-of-isc-bind-tkey-query-response-handling-dos-cve-2016-9131.html>)  
Analysis of ISC BIND TKEY Query Response Handling DoS (CVE-2016-9131)
  - ドメイン名を乱数から自動生成



# ファジングの実装

## – RRのクラスを変更

- IN ⇒ CH/HS/NONE/ANY

CVE-2015-8000(<https://kb.isc.org/article/AA-01317>)

Responses with a malformed class attribute can trigger an assertion failure in db.c

## – 乱数をもとに作成したTTL/TYPE/RDATA

未実装のRRに対応

## – OPT RR

- ペイロードサイズ、拡張RCODEを乱数から生成

CVE-2016-2848(<https://kb.isc.org/article/AA-01317>)

A packet with malformed options can trigger an assertion ...

- NSID, ECS, COOKIE, TCP-KEEPALIVEを追加
- 乱数をもとに作成したOPTION

# ファジングの実装

- RCODEの変更
- 署名
  - ファジングで追加・変更したRRSetも署名
  - OPT, TSIG, RRSIGなども署名
- 各セクションでのRRの順序変更
- DNSメッセージの変更
  - バイナリデータの追加・変更・削除

# ファジングの実装

クライアント (スタブリゾルバ)

- 問い合わせの作成
  - QNAME  
権威サーバ側のゾーンデータをもとに生成
  - QTYPE, QCLASS  
乱数から生成
  - OPT  
ペイロードサイズを乱数から生成
- 問い合わせを一定間隔でフルリゾルバへ送信  
フルリゾルバが異常終了するまで放置

# 例1

```
$ dig @127.0.0.1 www.example.com a +dnssec +nored
;; Warning: Message parser reports malformed message packet.

; <<>> DiG 9.9.4-RedHat-9.9.4-51.el7_4.2 <<>> @127.0.0.1 www.example.com a +dnssec +nored
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 42115
;; flags: qr aa ad; QUERY: 1, ANSWER: 5, AUTHORITY: 1, ADDITIONAL: 1
;; WARNING: Message has 3319 extra bytes at end

;; QUESTION SECTION:
www.example.com.                IN      A

;; ANSWER SECTION:
www.example.com.                10      IN      A      192.168.0.101
www.example.com.                10      IN      A      192.168.0.102
www.example.com.                961138880 HS     DNAME   \028\159fc\167,\191\2005\181p\186KG\198m\209\226\014\200\236\152\21
5I\209\030\018\133\169\192\152\197` \000*\135\171i\207` \159\192\027\234.\251\158\174\135\014\180\015}knk\152\209\173
\183\235{\196e_\144\240S\199\008\196\\\161\137v\167\005\021V\012\162\138\027\159\245\138\138\015\9\198\2004\012.\1
50>\018\144\236\139|\220\131\211\133\2357\005\)\012n\133\008\2397\223\237-c\194\159\154\204\013y\227K\139t\183\150q
\020\153\196.\031\167\151\212\219|\197\171w\2513\180{\168\012P\178z\173y1p\246V\168\232\227\000=\212i\|\001\176\21
5\253v\003U\2417\138.www.example.com.
www.example.com.                10      IN      RRSIG  A 13 3 10 20180613124222 20180603124222 59425 example.com. 3aJpbgbW
8Xf2tIBdMi3MvjKVc8NCmxwNxFk+qhab9yV3/AQcDMO+pVIJ Mpa9Ivt78N4aCzXD0cNvvAx0u6WGsw==
www.example.com.                961138880 HS     RRSIG  DNAME 13 3 961138880 20180613124222 20180603124222 59425 example.co
m. 3CvHapBL5vYyeFGSbeQkBaq6owXbkKIN2vEGZGnlBshJjIeP2uIdClQNB qdYEPghafzJFCwOAEbWixJFJjnT+yQ==
```

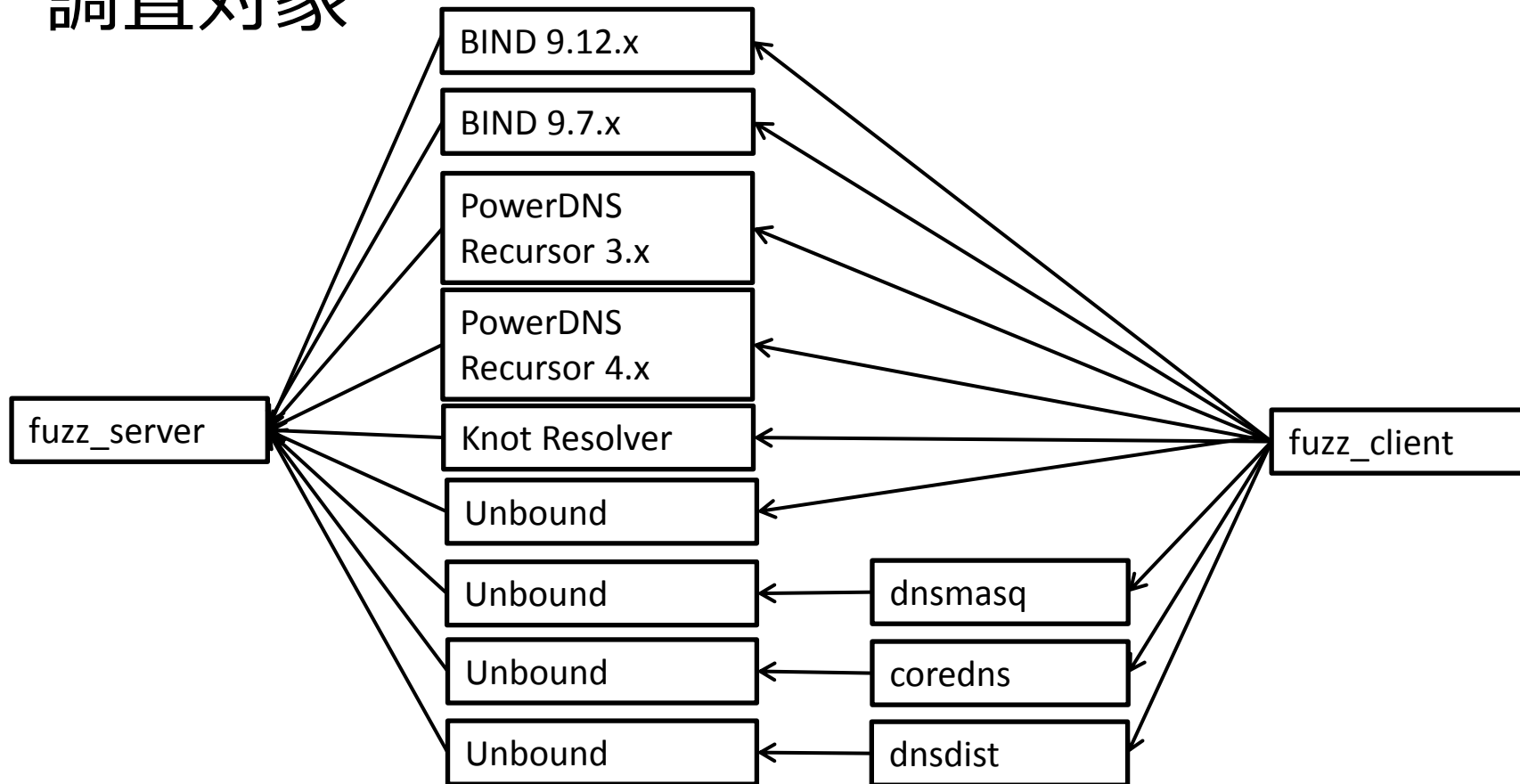
## 例2

```
$ dig @127.0.0.1 www.example.com a +dnssec +noredc
;; Warning: Message parser reports malformed message packet.
;; Truncated, retrying in TCP mode.
;; Got bad packet: bad compression pointer
8746 bytes
32 fb 84 20 00 01 00 03 00 04 00 01 03 77 77 77          2.....www
07 65 78 61 6d 70 6c 65 03 63 6f 6d 00 00 01 00      .example.com....
01 c0 0c 00 01 00 01 00 00 00 0a 00 04 c0 a8 00      .....
65 c0 0c 00 01 00 01 00 00 00 0a 00 04 c0 a8 00      e.....
56 c0 0c 00 2e 00 01 00 00 00 0a 00 5f 00 01 0d      f....._...
03 00 00 00 0a 5b 21 11 2e 5b 13 e2 2e e8 21 07      .....[!..[....!.
65 78 61 6d 70 6c 65 03 63 6f 6d 00 a8 60 6c 7f      example.com.`l.
5f 7c d7 4a a7 5c 83 f4 ee d9 62 29 71 42 1d ce      _|.J.\....b)qB..
46 89 8a c6 c0 b7 4a f3 0f 9c 86 91 0f 91 31 5d      F.....J.....1]
65 1a 64 96 55 fe 54 c1 25 4f 3c d6 f3 02 3a f6      e.d.U.T.%0<....:
08 df d6 25 37 ed c6 23 26 48 cf 19 c0 0c 00 2b      ...%7..#&H.....+
00 04 6c 1f 77 ed 00 24 0b 69 05 02 81 31 31 c1      ..l.w..$.i...11.
7b 47 e8 96 8a c1 2c b5 17 ce 0f 20 df be 26 47      {G.....,&G
0c 2d b5 64 ee 45 2f 6a 03 43 99 84 c0 0c 00 0b      .-.d.E/j.C.....
00 fe 44 97 5d 51 00 04 7b f5 aa 0f f4 5d 64 e4      ..D.]Q..{....]d.
64 8c 13 01 c4 5f 5b d7 40 85 35 00 9d 60 83 ad      d...._[.@.5...`..
0f 10 72 24 24 82 63 f3 45 bf 34 98 c6 e7 1f 42      ..r$.c.E.4....B
c4 62 02 03 09 22 63 02 2c 19 44 22 40 92 26 b6      .b...".c.,.D"@.&.
e1 30 02 c3 62 e0 d7 78 cf 2f a2 28 30 aa 57 44      .0..b..x./.(0.WD
ae 84 bc 69 3b be d3 21 b3 65 70 c2 96 22 04 80      ...i;...!.ep.."...
40 4c ef 8c 81 c8 e0 86 48 f4 20 2b 3b 02 05 c0      @L.....H..+;...
```

# 調査対象

- フルリゾルバ
  - BIND 9.12.x, 9.7.x(CentOS 6.x RPM)
  - Unbound
  - PowerDNS 3.x, 4.x
  - Knot Resolver
- ついでに
  - dnsmasq
  - dnsmasq
  - dnsmasq
  - coredns

# 調査対象



# 脆弱性発見後の対応

原因となる応答の調査

- パケットキャプチャしながらファジング
- デバッガ上で再度ファジングして解析
- クラッシュ時のメッセージやデバッガの出力からフルリゾルバのソースを探索

検証コードや再現手順を開発元へ報告



# 結果

- PowerDNS Security Advisory 2017-08: Crafted CNAME answer can cause a denial of service
  - <https://doc.powerdns.com/recursor/security-advisories/powerdns-advisory-2017-08.html>
- Knot Resolver: fix CVE-2018-1110: denial of service triggered by malformed DNS messages (2件の問題)
  - <https://lists.nic.cz/pipermail/knot-resolver-announce/2018/000000.html>
  - <https://gitlab.labs.nic.cz/knot/knot-resolver/issues/334>
  - <https://gitlab.labs.nic.cz/knot/knot-resolver/issues/335>
- Knot-Resolver 2.3.0 crashes in module/stats.  
libknot(knot-dns)の"knot\_dname\_to\_str memory overflow"(起因)
  - <https://gitlab.labs.nic.cz/knot/knot-dns/raw/v2.6.7/NEWS>
  - <https://gitlab.labs.nic.cz/knot/knot-resolver/issues/354>

# 最後に

- ファジングツールにより4件?脆弱性を発見できた。
- 特にQuad9(9.9.9.9)、Cloudflare(1.1.1.1)で利用されているPowerDNS Recursor/Knot Resolverの脆弱性を発見できた意義は大きい
- 過去の(BIND)の脆弱性に類似したものが、他のフルリゾルバでも発見できた(BIND:CVE-2015-8000 →PowerDNS Security Advisory 2017-08)