広告ドメインブロッキングを
やってみよう

やまぐちたかのり

Knot Resolver で

広告ドメインブロッキングを やってみよう

やまぐちたかのり

Knot Resolver

- CZ NIC によるキャッシュDNSサーバ
  - 開発元は Knot DNS と同じだが、あちらとはかなり発想が異なる
  - BIND や Unbound ともぜんぜん違う
- Lua 内蔵
  - 組み込み向けインタプリタ言語
  - 設定ファイルは Lua スクリプト
- たまにヤバい脆弱性が見つかる
- おもな採用実績: cloudflare (1.1.1.1)

Knot Resolver のインストール

- FreeBSD、Debian、Fedora、Arch Linux はパッケージあり
- それ以外はソースからビルド
  - 依存するライブラリがひじょーに多いので、すべて自力コンパイルするのはかなりめんどくさい
- 公式 Docker イメージあり
  - https://hub.docker.com/r/cznic/knot-resolver/

広告ブロックの前に

- まずはブロックしないキャッシュサーバとして動くようにする

```
net.listen({ '127.0.0.1', '::1' }, 53)                  -- listen するアドレス、ポート
user('kresd', 'kresd')                                  -- ユーザ、グループ
modules = { 'policy', 'view', }                         -- モジュール
cache.size = 100*MB                                     -- キャッシュサイズ
reorder_RR(true)                                        -- ラウンドロビン有効化
trust_anchors.file = 'root.key'                         -- DNSSEC 検証用 root KSK

view:addr('127.0.0.1', policy.all(policy.PASS))  -- アクセス制限
view:addr('::1', policy.all(policy.PASS))        -- (デフォルトではオープンレゾルバ)
view:addr('0.0.0.0/0', policy.all(policy.REFUSE))
view:addr('::/0', policy.all(policy.REFUSE))
```

起動

- ふつーに起動するとフォアグラウンドで動き続ける

- デーモン化するには systemd などのスーパバイザを使う必要あり
  - OS ごとに異なるので手順略
  - マニュアル見てね

- やっと広告ブロックの話に入れる…

RPZ

- Response Policy Zone
  - https://dnsrpz.info/
- 名前解決をブロックしたり応答を書き換えたりする設定をゾーンファイルの形式で記述するもの
  - ゾーンファイル形式なので、既存のゾーン編集ツールが使える
  - ゾーンファイル形式なので、ゾーン転送で複数ホストに配布できる

```
@               SOA     localhost.              root.localhost.     1 10800 3600 604800 3600
@               NS      localhost.
ad.example.com                  CNAME   .
*.ad.example.com                CNAME   .
```

- DNS をうまく利用したしくみ…のわけがない

Knot Resolver で RPZ

- RPZ はキャッシュサーバで必要になる機能
  - が、ゾーンファイルを読んだりゾーン転送するのは権威サーバの役割
  - キャッシュ専用ならそんな機能はないのがふつう

- なのに、Knot Resolver では RPZ が使える
  - 付属モジュール(zonefile.lua) の中で Knot DNS のゾーンファイル解釈用のライブラリ(libzscanner.so) を強引にロードしてます…
  - RPZ 仕様のすべてが実装されてるわけではないが、凝った使い方をしなければたいてい問題ない
  - ゾーン転送はさすがにできないけど

RPZ 用ゾーンファイルを作ろう

- 気に入らないドメインのリストを自分で作る
  - あるいは広告ドメインリストをどこかから拾ってくる

- 具体的なゾーンの書き方は以下を参照
  - https://dnsrpz.info/
  - https://www.ietf.org/archive/id/draft-ietf-dnsop-dns-rpz-00.txt

RPZ を設定しよう

- config ファイルに以下の設定を追加して完了

```
policy.add(policy.rpz(policy.DENY, 'rpz.zone'))
```

- 数が少なければ RPZ を使わずベタ書きでいいかも

```
policy.add(policy.suffix(policy.DENY,
    policy.todnames{ 'ad.example.com', 'adv.example.jp' }))
```

RPZ を更新しよう

- config に右の設定を追加

ドメインリストの取得から RPZ 形式への変換までを make_rpz.sh に書いておき、それを実行

RPZ がロード済みならいったん削除

作り直した RPZ をロード

…という処理を7日ごとに実行する

```
rpz = {}
function rpz.reload()
   local r = os.execute("sh make_rpz.sh")
   if r ~= 0 then
      log "[rpz] failed to exec make_rpz.sh"
      return nil
   end
   if rpz.policyid then
      policy.del(rpz.policyid)
   end
   r = policy.add(policy.rpz(policy.DENY, "rpz.zone"))
   if r then
      rpz.policyid = r.id
      log("[rpz] reloaded; id=%d", rpz.policyid)
   end
end
rpz.eventid = event.recurrent(7*day, rpz.reload)
```

- cron を使わず定期更新できる
  - 管理コンソールから rpz.reload() を実行することで手動更新も可

使ってみよう

- お手元の端末の DNS 設定を Knot Resolver が動いてるホストに向けてくださいませ

- あれ、スマホってどうやって DNS いじるの?

スマホの DNS

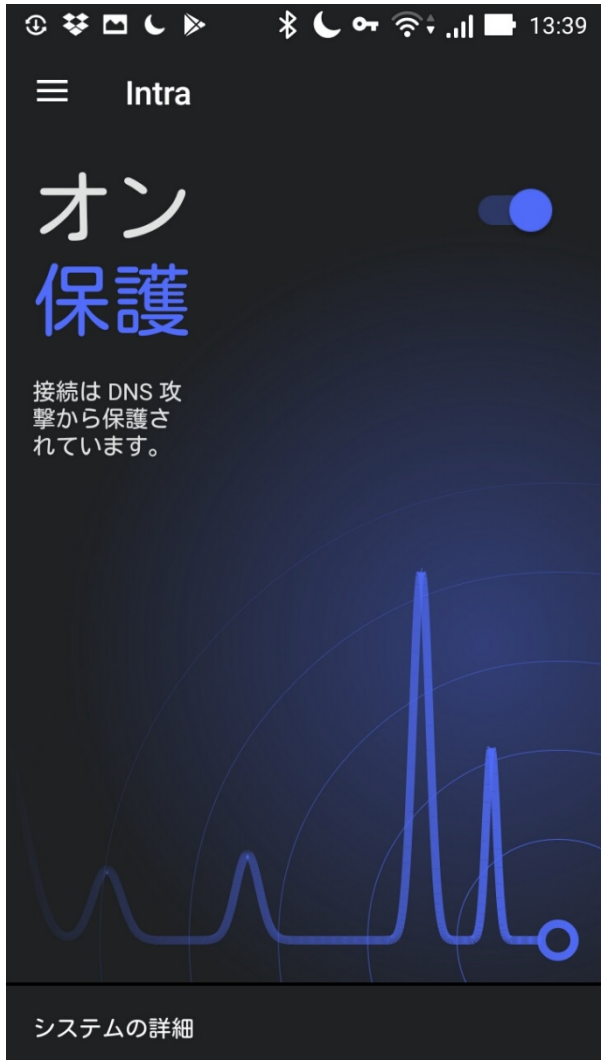
- wifi 接続のときは自由にいじれる

- が、LTE などキャリア回線利用時の DNS は自分で設定できない
  - キャリアが用意した DNS サーバが強制される

- せっかく作った広告ブロック DNS を使えない！

Intra

- Intra: Android 用 DNS over HTTPS クライアント
  - google だけでなく任意の DoH サーバを指定できる
- DNSCloak: iOS 用 DNSCrypt/DoH クライアント
  - 任意の DNSCrypt/DoH サーバを指定できるが手順がめちゃくちゃめんどくさい
  - 自前でドメインブラックリストを設定できるので、サーバ側でブロックしなくてもいいかも…
- DNS サーバは自由に変更できないが、DoH サーバなら自由に変更できる
- じゃあ DoH サーバを作ればいいんだ



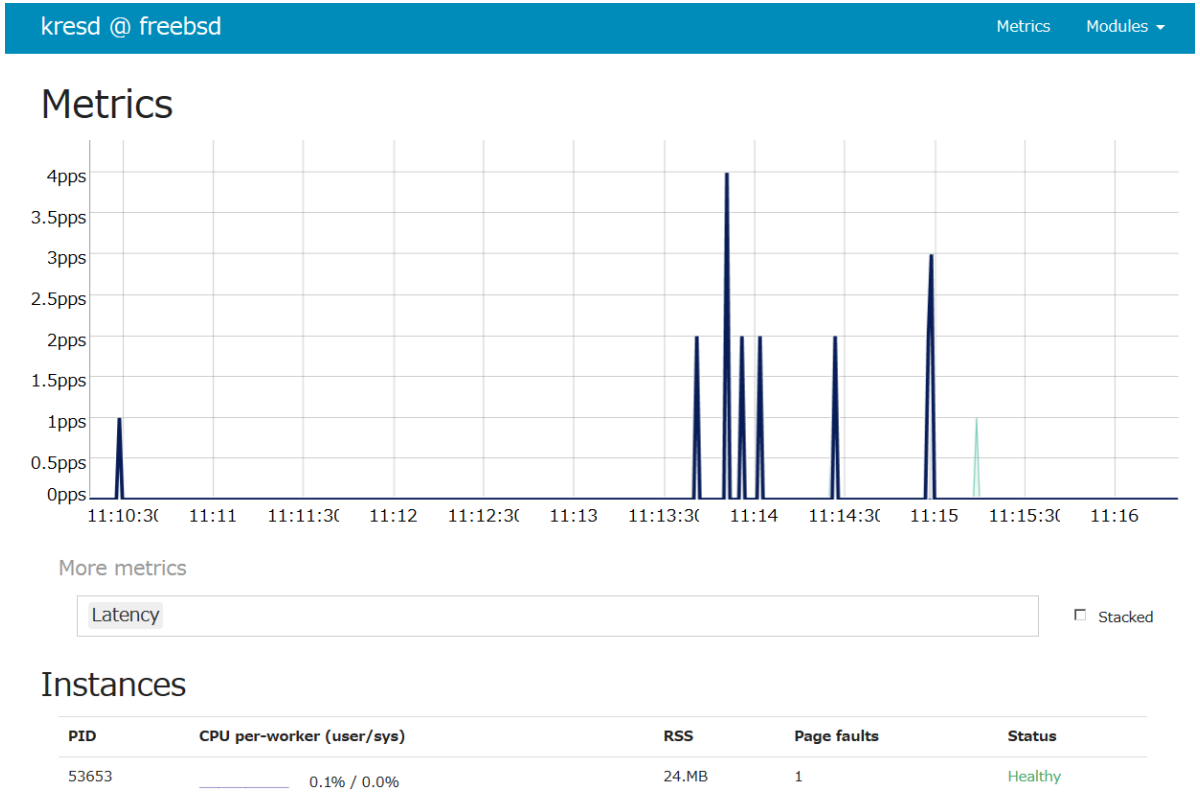
DNS over HTTPS サーバの構成

- ふつーはこうだよね

- でも…

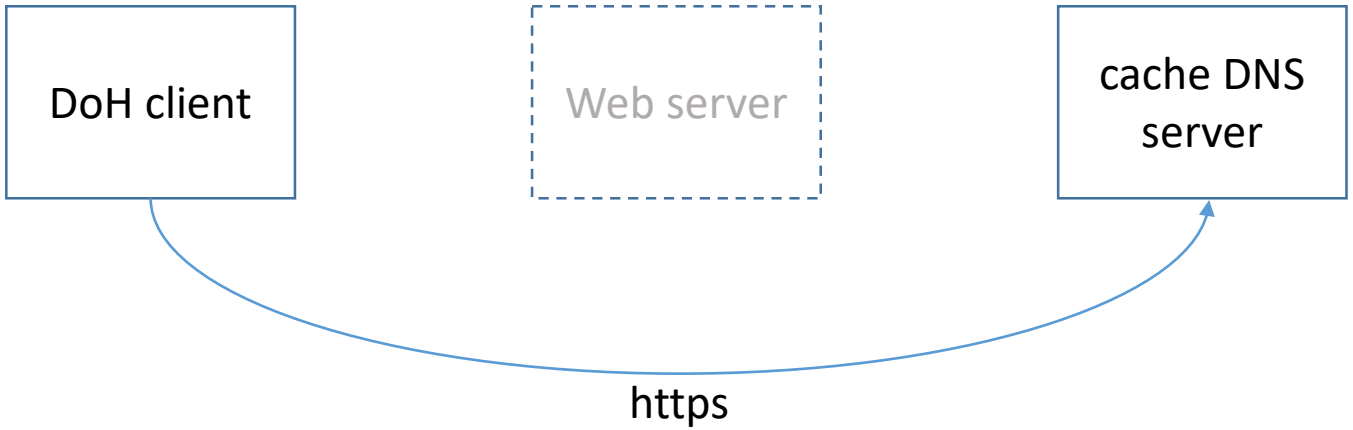
Knot Resolver で HTTP/2

- こいつ、キャッシュ DNS サーバなのに、HTTP/2 を喋れるんです…
  - kresd 本体に実装されているものではなく、付属の Lua モジュールの仕業
  - 稼働状況や統計情報の取得が本来想定されていた用途
  - prometheus から exporter なしで直接監視できるよ



Knot Resolver で DoH?

- HTTP を喋れるなら DoH もやらせればいいんじゃね?



- とはいえ、そんなモジュールは用意されてない

Knot Resolver で DoH

- ないので、自分で DoH モジュール作りました
  - https://dns.maya.st/doh.lua
- 使用上の注意
  - アクセス制限ありません
  - URI がバレると世界中から使えちゃいます
- cloudflare の中の人が DoH モジュールを本家に寄贈してくれるらしいので、そっちを待ったほうがいいかも
  - https://gitlab.labs.nic.cz/knot/knot-resolver/issues/280

なんということでしょう

びふぉー

あふたー

まとめ

- Knot Resolver = 柔軟に機能を拡張できるキャッシュ DNS サーバ
  - forwarder、アクセス制限、DNS cookie、DNSTAP、DNS64 など多くの機能がモジュールとして実装されている
  - 自作するのも難しくない
  - モジュールは Lua だけでなく、C や go でも書ける

- 単純に名前解決するだけでなく、ちょっと変わった機能がほしいのであれば、それを実現するための土台になるかも?
  - 逆にそういうことが必要ないのであれば、積極的に Knot Resolver を選ぶ意義はあんまりないかなぁ