Knot DNS の紹介

Internet Week 2018 DNS DAY

IIJ 山口崇徳

InternetWeek 2012 DNS DAY

	ノ)	
3) DNS実装ダイバーシティの話		
code diversityの概況	伊藤　高一(株式会社ブロードバンドタワー)	163KB
Unboundの紹介	高田　美紀(株式会社エヌ・ティ・ティピー・シーコミュニケーションズ)[講演者が高橋 央から変更となりました]	119KB
djbdnsの紹介	市川 剛(株式会社データホテル)	328KB
100万ゾーンを管理するDNSの運用 ～Nominum社が開発しているANSの運用を含め～	井上　昌之(さくらインターネット株式会社)	957KB
NSDの紹介	滝澤　隆史(株式会社ハートビーツ)	907KB
PowerDNSの紹介	松田　顕(Dozens株式会社)	730KB
BIND10の紹介	神戸　直樹(株式会社日本レジストリサービス)	161KB

(敬称略)

イベントカレ

WHOIS

DNS Summer Day 2016

プログラム

BINDからの卒業 10:00 - 12:40

近年より注目を集めているBIND以外のDNS実装についての特徴、魅力などを
共有し、BINDを辞められないという悩みを議論するセッション。

時間	タイトル	発表者	資料
10:00-	Opening(開催挨拶、会場諸注意)	石田 慶樹 / 日本DNSオペレーターズグループ代表幹事	資料
10:15-	はじめに 実装紹介: Unbound	島村 充　/ 株式会社インターネットイニシアティブ	資料 Unbound資料
10:40-	実装紹介: PowerDNS	大野 公善 / 株式会社デージーネット OSS研究室	資料
11:10-	小休憩		
11:20-	実装紹介: NSD	山口 崇徳 / 株式会社インターネットイニシアティブ	資料
11:50-	質疑応答	ALL	
12:10-	BINDを辞められない理由 Q&A	ALL	資料

Knot DNS とは

- CZ NIC による権威DNSサーバ実装
  - https://www.knot-dns.cz/
  - キャッシュ機能なし
    - 同じく CZ NIC による Knot Resolver をご利用くださいませ
  - GPL
  - 最新版 2.7.4

- 採用実績
  - .cz
  - K.root-servers.net (の一部)
  - それほど多いわけではない

主な機能

- NSD でもできること
  - RRL (response rate limting)
- BIND にできて NSD ではできないこと
  - automatic DNSSEC signing
  - dynamic update
  - DNSTAP
  - GeoIP
- BIND でもできないこと
  - online DNSSEC signing (クエリを受けてから動的に署名)
  - 正引き、逆引きの自動生成

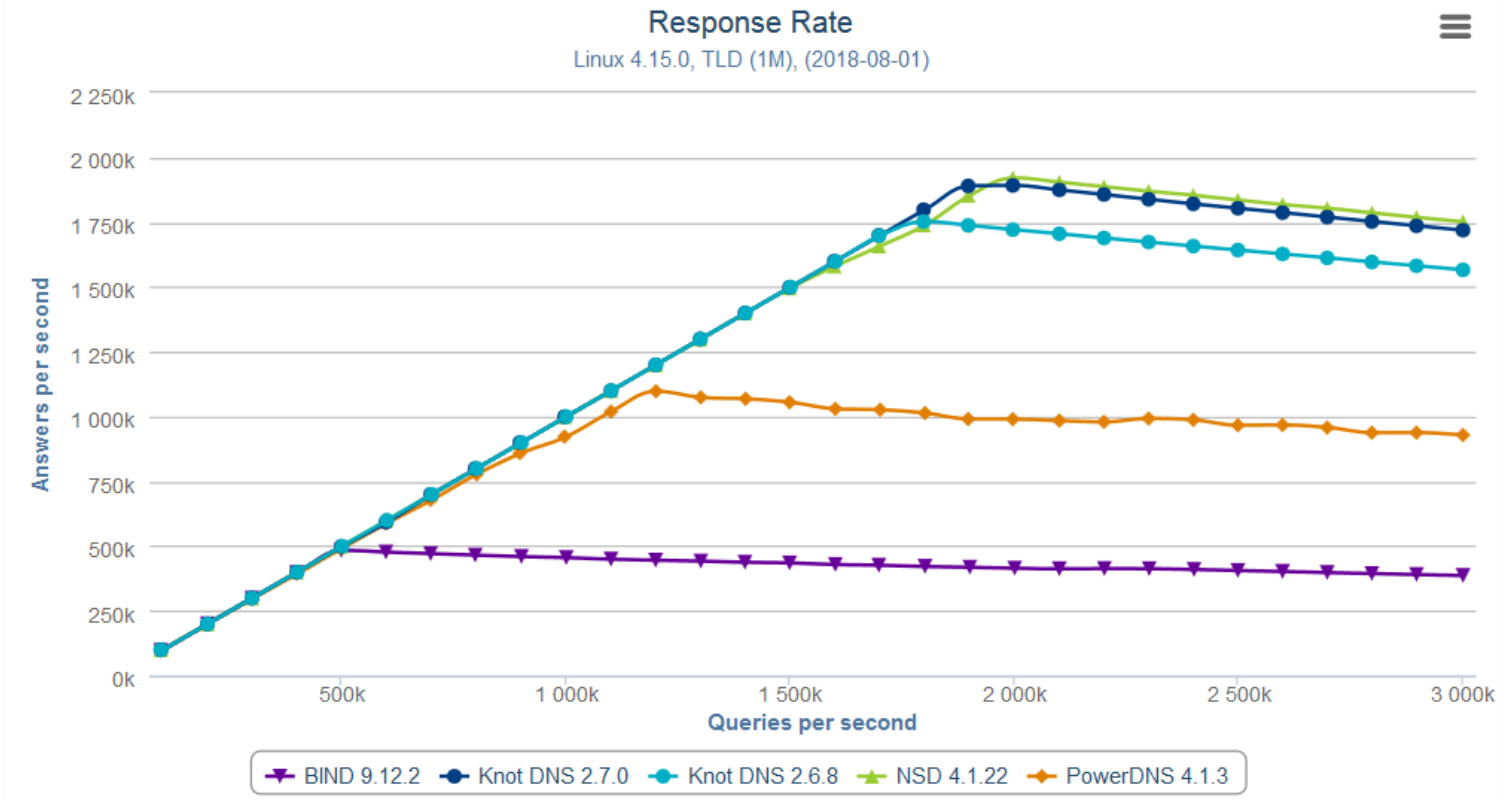
セキュリティ

- 現時点で CVE として採番されたものは2件
  - CVE-2016-6171 巨大ゾーン転送によるリソース枯渇
    - DNS Summer Day 2016 で坂口さんが報告したもの
  - CVE-2017-11104 TSIG 認証バイパス

- 未発見の脆弱性がある可能性もないではないが、BIND に比べればずっと安全なんじゃないかな(たぶん)

パフォーマンス

- CZ NIC のベンチマークでは、NSD とほぼ同等の性能



https://www.knot-dns.cz/benchmark/

サポート

- 有償サポートあり
  - ブロンズ (5000 EUR/year) からプラチナ (50000 EUR/year) まで
  - プラチナだとオンサイトサポートもあるみたいなんですが、日本まで来てくれるんですかね…?
  - https://www.knot-dns.cz/support/
- 国内 SIer の対応はまったく期待できない状況かと…

設定

- YAML ファイルと LMDB データベースの2通り
  - 同時に使えるわけではない
  - 相互変換可能(knotc conf-import/conf-export)
- 動的な設定変更

```
# knotc conf-begin
# knotc conf-set parameter value
# knotc conf-commit
```

  - YAML の場合、再起動すると元の値に戻る
  - LMDB の場合、再起動後も永続的に有効

knotc

- 制御コマンド
  - BIND における rndc のようなもの
    - ネットワーク越しには使えないので、BIND8 の ndc のようなもの、の方が正しいかも
- 設定変更
- ゾーンへの操作
  - ゾーン転送、notify
  - レコード編集
  - DNSSEC 署名
  - 統計情報取得
- など

view (split-brain DNS)

- BIND から離れられない最大の理由?
- Knot では最近になって実現できるようになりました
- geoip モジュール(v2.7.0)
  - 問い合わせ元サブネット/地域により同じレコードに対して異なる応答を返す
  - edns-client-subnet を有効にしていると、ECS 詐称により内部のホスト情報が漏れる可能性があるので注意
  - https://en.blog.nic.cz/2018/10/16/geoip-in-knot-dns-2-7/
- queryacl モジュール(v2.7.3)
  - src address / target interface によりゾーン単位でアクセス制限する
  - プライベートアドレスの逆引きゾーンなどに利用

権威とキャッシュの共存

- BIND を捨てられない理由その2
- Knot でできちゃいます…
- 自身が権威を持ってないゾーンへの問い合わせを別サーバに forward させることができる
- できるからといって推奨はしない
  - アクセス制限できない
  - やはり分離が正道

```
remote:
  - id: recursive
    address: [10.0.0.53, 10.0.1.53]
mod-dnsproxy:
  - id: default
    remote: recursive
    fallback: on
template:
  - id: default
    global-module: mod-dnsproxy/default
```

DNSSEC

- DNSSEC はめんどくさい、難しいと思ってませんか
- Knot なら超ラクチンです

DNSSECを1分でおさらい

- RRに署名して、それを検証することで正しさを確認

- 権威サーバにおけるDNSSECの運用
  - 署名鍵(DNSKEY)の更新(ロールオーバー)
    - KSK、ZSKの2種類
    - KSKは対応する鍵ハッシュ(DS)を上位ゾーンに登録
  - 署名の更新
    - ゾーンを編集した後
    - 署名の有効期間が過ぎる前

DNSSEC を始める前に

- SOA のシリアル番号は Knot が勝手に更新してくれる
  - デフォルトでは更新のたびに +1 される
  - 既存のシリアルが YYYYMMDDnn の形式だと、そのうち13月50日みたいな日付っぽくなってしまう
- 必要に応じてシリアル番号を変更しておきましょう
  - 詳しくは RFC1982
  - serial-policy: dateserial に設定すると、YYYYMMDDnn のままでもいける
    - が、1日100回以上更新があると破綻する
- 見た目の問題だけなので、気にならなければそのままでも不都合はありません

Automatic DNSSEC signing

- policy セクションでDNSSEC の運用ポリシーを定義
- zone セクションで dnssec-signing:on にして、ポリシーを指定
  - すべてデフォルトならポリシーは省略できる
  - が、デフォのアルゴリズムは .jp で未対応の ECDSAP256SHA256 なので…
- knotc reload すると、適切な鍵が自動で生成され、適切な署名が自動で付与される

```
policy:
  - id: rsa
    algorithm: RSASHA256
    nsec3: on

zone:
  - domain: example.jp
    dnssec-signing: on
    dnssec-policy: rsa
```

DS 登録

- レジストラに DS 鍵を登録する
  - 登録すべき値は以下のコマンドで知ることができる

```
# keymgr example.jp ds
```

  - 自動で CDS (child DS) レコードが追加されるので、それを調べてもよい

```
# kdig example.jp cds
```

    - 親が CDS 非対応なのに CDS が追加されるのが気持ち悪ければ、載せないよう設定変更することも可能
- 親ゾーンに DS レコードが掲載されたら、Knot にそれを教えてやる

```
# knotc zone-ksk-submitted example.jp
```

ゾーン編集

- 鍵ロールオーバーや署名有効期間更新のため、Knot がゾーンを書き換えることがある
- そのタイミングで人間が手作業でゾーン更新すると不整合が起きる
- ので、Knot によるゾーン自動更新の一時停止/再開を人間が指示する必要がある
  - knotc zone-freeze/zone-thaw
  - BIND の rndc freeze/thaw と同じ
- Knot が自動更新したゾーンはコメントなどが失われているので手作業で編集するのはわかりづらいかも…
  - 解決策あります (後で)

ゾーン署名

- ゾーン編集後の署名
- 署名有効期間更新のための定期再署名

- いずれも自動でおこなわれるので人間は何もしなくてよい

鍵ロールオーバー

- KSK はデフォルトでは無期限で、ロールオーバーされない
  - つまり、何もしなくてよい
  - ルート KSK も8年使ったんだし、そんなにしょっちゅう変えなくていいよね
  - 期限を設定した場合、DS 更新だけ人力、それ以外は自動
    - DS 登録の準備ができたらログにメッセージが出るので、それを監視して作業する
    - 親ゾーンが CDS に対応していれば、DS 更新も自動でできる
- ZSK は完全自動ロールオーバーなので何もしなくてもよい

- 人間がロールオーバー作業をする必要はない

アルゴリズムロールオーバー

- 署名アルゴリズムを変更する場合、特殊な手順が必要になる

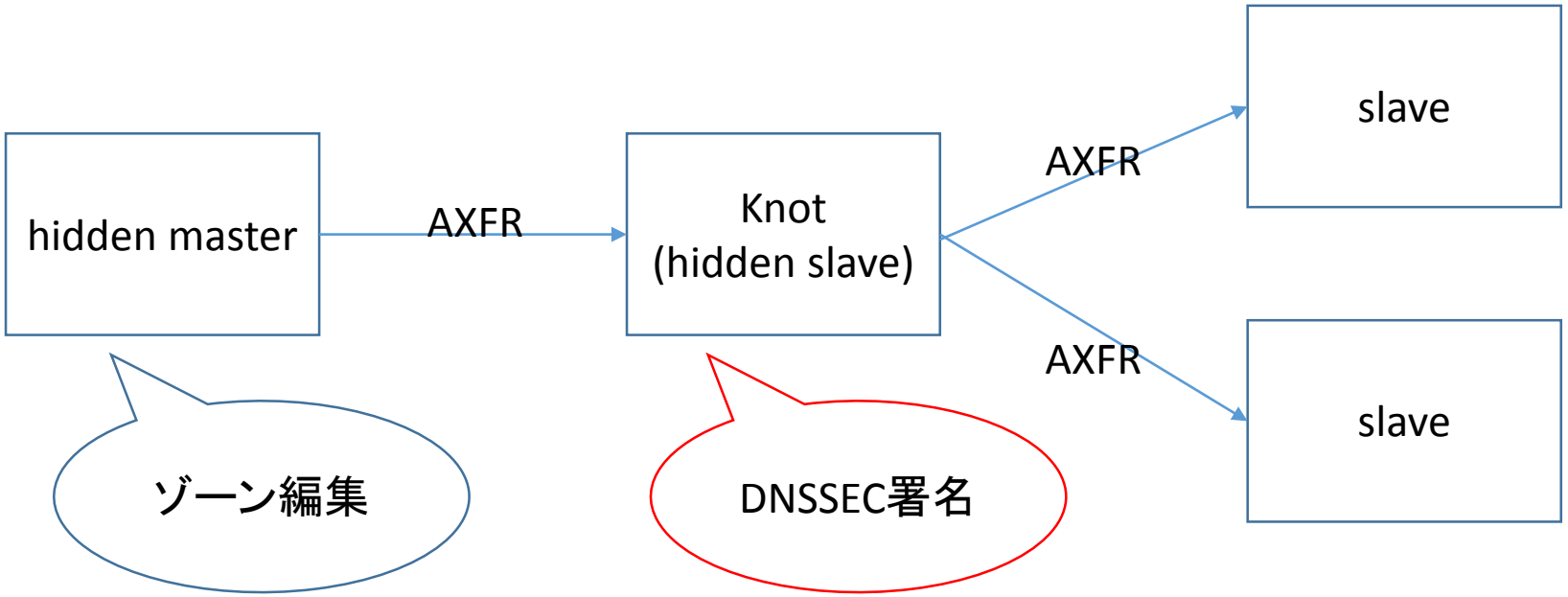
- が、ポリシー設定のアルゴリズムを変更するだけで適切な手順で自動ロールオーバーしてくれる
  - DS 更新だけ手作業が必要なので、ログを見張っておきましょう

自動ゾーン更新の問題点

- automatic signing では、ゾーンファイルを Knot が直接書き換える
  - ゾーンファイル中のコメントなどが失われ、可読性が下がる
  - ゾーンの履歴管理がしにくくなる
  - knotc zone-freeze/zone-thaw を忘れると不整合が起きることがある
- BIND の自動署名も同じ問題

- 一気に解決しちゃいましょう

on-slave signing



- Knot は slave でも署名ができる
- master で普段どおり編集して未署名ゾーンを Knot に転送するだけ
- Knot は署名したゾーンをさらに公開サーバに転送

on-slave signing の設定

- 以下ぜんぶまとめて設定するだけ
  - master からゾーン転送する設定
  - slave からゾーン転送要求を受ける設定
  - DNSSEC 署名する設定
- ゾーン編集用の master と署名用の Knot は、リソースがほとんど必要ないので同一ホストに同居していて問題ない

```
remote:
  - id: master
    address: 127.0.0.1
  - id: slave
    address: [192.0.2.1, 192.0.2.2]
acl:
  - id: master_notify
    address: 127.0.0.1
    action: notify
  - id: slave_xfr
    address: [192.0.2.1, 192.0.2.2]
    action: transfer
zone:
  - domain: example.jp
    dnssec-signing: on
    master: master
    notify: slave
    acl: [master_notify, slave_xfr]
```

Knot の DNSSEC まとめ

- DS 登録だけ手作業
- それ以外の一切の作業はすべて自動でやってくれるので人間は何もしなくてよい
  - いちおう完全手動でやることもできるけど…
- ゾーン編集はちょっとだけ手順が煩雑になる
  - slave 署名で解決できる

- 念のため、BIND でも大半の作業は自動化できます
  - なのにみんな手作業でやって難しいと嘆き、手作業でやって障害を起こす

dynamic update (RFC2136)

- 標準化されたゾーン更新 API
  - Knot によるゾーン書き換えと衝突せずにゾーン編集するもうひとつの手段
  - BIND でも昔から使えますが、あんまり使ってる人はいないんじゃないかな…

- でも、最近になって便利に使える用途ができたんですよ

Let's Encrypt で dns-01 認証

- dns-01: ACME (Let's Encrypt で使われる証明書取得プロトコル)で定義される認証方式のひとつ
  - _acme-challenge.example.com/IN/TXT に認証トークンを記述
  - 認証方式は dns-01 以外にもいくつかあるが、ワイルドカード証明書の取得は dns-01 しか使えない
- certbot の dns-01 認証は、素のままではインタラクティブな操作必須
  - せっかくの ACME なのに自動化できない
  - 自動化プラグインを入れればよい
    - 既存のプラグインは大手 DNS ホスティング事業者の API を叩くものがほとんどだけど…
    - RFC2136 プラグインなるものが！

Knot で SSL 証明書取得自動化

- Knot で dynamic update を許可
- あとは certbot に rfc2136 plugin をインストールしてドキュメントどおりやればおk
  - https://certbot-dns-rfc2136.readthedocs.io/en/stable/
- lego という ACME client も rfc2136 に対応してます
  - https://github.com/xenolf/lego

```
key:
  - id: acme
  # keymgr -t acme の出力をコピペ
acl:
  - id: acme
    key: acme
    action: update
zone:
  - domain: example.jp
    acl: acme
```

まとめ

- Knot DNS は
  - BIND に迫る豊富な機能で BIND よりずっとよい性能
  - NSD よりずっと多機能で NSD に迫る高性能
  - とくに DNSSEC の簡単さは感動もの
- BIND から乗り換えたいんだけど NSD では機能が足りない、と悩んでいる場合にはぜひ検討を