

DNS設定例の紹介

【権威DNSサーバ編】

DNSOPS.JP 高嶋隆一
aka 酔っ払い.jp

ちょっとだけ自己紹介

昔

- 通信事業者で運用,設計
- ドメインレジストリで同上



ちょっと前

- ネットワーク機器ベンダ



最近

- クラウド向けの仮想ネットワークスタックを売ってます



昔とった杵柄でがんばります!

Agenda

- ✓ 本セッションの目的
- ✓ named.conf を観察してみる
 - options {} 編
 - logging {} 編
 - zone {} 編
 - その他共通設定編
- ✓ おまけ
- ✓ Questions ?

本セッションの目的

対象者

- 権威DNSサーバの管理者を任されたものの、「どうも自信が持てないなー」と考えているあなた

狙い

- 実在するドメインの権威DNSサーバ設定を元に、ちょっとした注意点や tips を共有
- こんなもんでいいのかなー、と自信を持って頂く

本セッションの目的 cont.

解説対象

権威DNSサーバ (= Authoritative DNS server)

キャッシュDNSサーバ

今回の範囲

ゾーン情報

その他の設定情報

本セッションの目的 cont.

前提条件

権威DNSサーバとキャッシュDNSサーバを BIND9 の view 機能などを使ってがんばってひとつのサーバで同居させることも不可能ではありませんが、

ややこしいのでおすすめしません。

今回の設定も権威DNSサーバ、キャッシュDNSサーバの同居はしない前提ですおすすめしていきます。

本セッションの目的 cont.

登壇者が BIND9 しか触っていない為、設定例は BIND9 ベースのものになります orz



本セッションの目的 cont.

というわけで

named.conf の話をします。

named.conf を観察してみる

今回の観察対象

ns1.dnsops.jp

dnsops.jp, dnssec.jp の master

urquell.xn--n8j1c913r6j1b.jp

= urquell.酔っ払い.jp

酔っ払い.jp の master だったりいくつか slave をもっていたり

named.conf を観察してみる

options {} 編

ns1.dnsops.jp の場合

```
options
{
    directory "/var/named"; // the default
    dump-file   "data/cache_dump.db";
    statistics-file "data/named_stats.txt";
    memstatistics-file "data/named_mem_stats.txt" ;
};
```

BIND9 設定ファイルの親パスと、
rndc (stats|dumpdb) の出力先くらいしか設定してませんでした。
こんなんでも充分です。

options {} 配下の便利設定

```
transfers-in [Integer];
```

並行してゾーン転送を slave として“受ける”数。デフォルトが 10 と少ないです。大量の slave ゾーンを抱える ISP さんの権威 DNSサーバでは増やしたほうがいいです。

```
transfers-out [Integer];
```

同様に master “送る”数。おなじくデフォルトが 10。master でたくさんゾーンをもっていて、一括して内容を変更するようなケースでは増やしたほうがいいかも。

in ほどシビアではないとおもいます。

options {} 配下の便利設定

```
max-transfer-time-in [Integer];
```

ゾーン転送を打ち切る時間[分]。

デフォルトが 120分と長いので、もうちょっと短かくしましょう。これも ISP の slave などでも大量のゾーン転送を受ける場合に、不良セッションに占領されるのを防ぐためのオプションです。

```
tcp-clients [Integer];
```

TCP接続でのクエリの同時接続数。

同時接続数で 100 なのでそのままでも問題ないとおもいますが、気持ち増やしたほうがいいかも？

options {} 配下の便利設定

```
masterfile-format TYPE;
```

ゾーンファイルの記述方式。
raw と text があり、
master でのデフォルトは text
slave でのデフォルトは raw
とややこしい。

大きなゾーンファイルや大量のゾーンを抱える環境では raw のほうが起動時間を減らせるかもしれません。

BIND9.10 からは map も増えたらしい。。。

master-file-format map;

<https://kb.isc.org/article/AA-01120/0/Using-the-map-zone-file-format-in-BIND-9.10.html>

memory mapped file を使う、らしい

raw よりずっと速い、らしい

異なる BIND のバージョンで使ってはいけない、らしい。
ちょっと version up のときとかめんどろ？

named.conf を観察してみる

logging {} 編

urquell.酔っ払い.jp の場合

割とシンプル。
rndc trace で吐かれる debug の場所以外は一箇所に書く形。

```
logging
{
    channel default_debug {
        file "data/named.run";
        severity dynamic;
    };

    channel default_channel {
        file "/var/log/named.log" size 10M versions 10;
        print-time yes;
    };
    category default { default_channel; };
};
```

10世代までログを残し、ひとつひとつのログファイルは 10MBまで。

ns1.dnsops.jp の場合

```
logging
{
    channel default_debug {
        file "data/named.run";
        severity dynamic;
        print-category yes;
        print-severity yes;
        print-time yes;
    };

    channel default_channel {
        file "/var/log/named.log" size 10M versions 10;
        severity dynamic;
        print-category yes;
        print-severity yes;
        print-time yes;
    };
};
```

ここまでは大体同じ

ns1.dnsops.jp の場合 cont.

```
category queries { default_debug; };  
  
category update-security { default_channel; };  
category default { default_channel; };  
category general { default_channel; };  
category database { default_channel; };  
category security { default_channel; };  
category config { default_channel; };  
category resolver { default_channel; };  
category notify { default_channel; };  
category client { default_channel; };  
category unmatched { default_channel; };  
category network { default_channel; };  
category update { default_channel; };  
category query-errors { default_channel; };  
category dispatch { default_channel; };  
category dnssec { default_channel; };  
category delegation-only { default_channel; };  
category edns-disabled { default_channel; };
```

やたらカテゴリ定義してますが、クエリ関連を debug に入れた他は、ほとんど default_channel。

notify は次頁のゾーン転送用チャネルにだしてもいいかも…

ns1.dnsops.jp の場合 cont.

```
channel xfer_channel {  
    file "/var/log/named-xfer.log" size 10M versions 10;  
    severity dynamic;  
    print-category yes;  
    print-severity yes;  
    print-time yes;  
};  
category xfer-in { xfer_channel; };  
category xfer-out { xfer_channel; };
```

ゾーン転送に関するログは別のファイルに出力するようになってます。
大量の slave を抱えるようなサーバでは別にしたほうがよいかも。

named-xfer.log はこんな感じ

```
11-Jun-2014 00:00:02.680 xfer-out: info: client 183.181.160.83#43401 (dnssec.jp) :  
view external: transfer of 'dnssec.jp/IN' : AXFR-style IXFR started  
11-Jun-2014 00:00:02.691 xfer-out: info: client 183.181.160.83#43401 (dnssec.jp) :  
view external: transfer of 'dnssec.jp/IN' : AXFR-style IXFR ended  
11-Jun-2014 00:00:03.174 xfer-out: info: client 183.181.160.83#51925 (dnsops.jp) :  
view external: transfer of 'dnsops.jp/IN' : AXFR-style IXFR started  
11-Jun-2014 00:00:03.174 xfer-out: info: client 183.181.160.83#51925 (dnsops.jp) :  
view external: transfer of 'dnsops.jp/IN' : AXFR-style IXFR ended  
21-Jun-2014 00:00:02.437 xfer-out: info: client 183.181.160.83#49666 (dnssec.jp) :  
view external: transfer of 'dnssec.jp/IN' : AXFR-style IXFR started  
21-Jun-2014 00:00:02.463 xfer-out: info: client 183.181.160.83#49666 (dnssec.jp) :  
view external: transfer of 'dnssec.jp/IN' : AXFR-style IXFR ended  
21-Jun-2014 00:00:02.837 xfer-out: info: client 183.181.160.83#48734 (dnsops.jp) :  
view external: transfer of 'dnsops.jp/IN' : AXFR-style IXFR started  
21-Jun-2014 00:00:02.837 xfer-out: info: client 183.181.160.83#48734 (dnsops.jp) :  
view external: transfer of 'dnsops.jp/IN' : AXFR-style IXFR ended
```

logging {} 配下の便利設定

```
channel lame_channel {  
    file "/var/log/named-lame.log" size 10M versions 10;  
    severity dynamic;  
    print-category yes;  
    print-severity yes;  
    print-time yes;  
};  
category lame-servers { lame_channel; };
```

お客さんゾーンの slave を大量に抱えていて、lame なゾーンが多い場合には別だしにしたほうがよいかも… channel 名を null にしてログに出さないのもアリ。

print-category yes にしておけば、こんなログの小分けもあとからやりやすいです。

named.conf を観察してみる

zone {} 編

ns1.dnsops.jp の場合

allow-transfer で slave サーバにのみゾーン転送を許可している以外は特殊設定ナシ。

notify もデフォルト yes なので抜いてもいいかも…

権威DNSサーバなので zone "." も localhost 関連の zone もありません。

```
zone "dnsops.jp" {
    type master;
    file "dnsops.jp.signed";
    allow-transfer { 183.181.160.83; };
    notify yes;
};
zone "dnssec.jp" {
    type master;
    file "dnssec.jp.signed";
    allow-transfer { 183.181.160.83; };
    notify yes;
};
```

urquell.酔っ払い.jp の場合

大体 ns1.dnsops.jp と一緒だが、allow-transfer に localhost と自身の IP Address も記述している

```
zone "hanya-n.org" {
    type master;
    file "hanya-n.org";
    allow-transfer { localhost; 49.212.57.196; 49.212.140.172; };
};
zone "xn--n8j1c913r6j1b.jp" {
    type master;
    file "xn--n8j1c913r6j1b.jp.signed";
    allow-transfer { localhost; };
};
zone "example.net" {
    type slave;
    masters { 49.212.57.196; };
    file "slaves/example.net";
    allow-transfer { localhost; 49.212.57.196; 49.212.140.172; };
};
```

slave も allow-transfer の設定をしている

urquell.酔っ払い.jp の場合 cont.

master の allow-transfer

基本的には slave のアドレスだけを書けばよい。
が、手動でゾーン転送して確認したい場合など用に記述。

secondary の allow-transfer

基本的には none でかまわない。
が、master 同様に手動でゾーン転送して確認したい場合など用に記述。

named.conf を観察してみる

その他の共通設定編

rndc 関連

```
key "rndc-key" {  
    algorithm hmac-md5;  
    secret "XXXXXXXXXXXXXXXXXXXXXXXXX==";  
};  
  
controls {  
    inet 127.0.0.1 port 953  
        allow { 127.0.0.1; } keys { "rndc-key"; };  
};
```

rndc 制御は必要最低限に許可。

必要なら個別に listen する address と、アクセスを許可する address を足しましょう。

rndc 関連 cont.

が、最近は
rndc-confgen -a
するだけでいいらしい！

どうも前述の書き方は BIND8 からの歴史的経緯
を私がひきづってるだけらしいです。。。

共通設定全般

```
match-clients      { any; };  
match-destinations { any; };  
  
recursion no;
```

大事なのは“recursion no”！
権威DNSサーバとしての機能のみを持たせ、キャッシュDNSサーバとして動作させない為、不要です。

他の二行はデフォルト値なのでいらないかも…

関連する operation ツール

named-checkconf

```
named-checkconf [-h] [-v] [-j] [-t directory] {filename} [-p] [-x] [-z]
```

named.conf の場所を指定して起動すれば OK。

文法チェックをしてくれるので、変更したら rndc (reload|reconfig) する前に確認しましょう。

なお、ゾーン情報変更後の確認は named-checkzone です。

dig によるゾーン転送

```
dig axfr @192.168.0.1 example.jp
```

手動で zone 転送が確認できるので、allow-transfer が正しく設定されているかなどを確認するのに利用

あとはシリアルの上げ忘れでシリアルは同じだけどゾーンファイルの中味が違う場合の確認など。。。

raw 形式のゾーンファイルの見方

```
named-checkzone -D -f raw example.jp /var/named/example.jp.zone.raw
```

raw で記述されているファイルの中味を確認したいときに。
前述の dig で axfr でもいいかも。

BIND9 ARM (Administrator Reference Manual)

BIND 9.10 ARM	ftp://ftp.isc.org/isc/bind9/cur/9.10/doc/arm/
BIND 9.9 ARM	ftp://ftp.isc.org/isc/bind9/cur/9.9/doc/arm/
BIND 9.8 ARM	ftp://ftp.isc.org/isc/bind9/cur/9.8/doc/arm/
BIND 9.6-ESV ARM	ftp://ftp.isc.org/isc/bind9/cur/9.6/doc/arm/

named.conf の設定のリファレンスです。

デフォルト値なども記述されていますので、なにかあったら自分が使っているバージョンの ARM を確認しましょう。

おまけ

BIND9 の Smart signing を用いた DNSSECのお手軽運用

実は、酔っ払い.jp 結構むかしから DNSSEC 対応してます。

```
Domain Information: [ドメイン情報]
[ドメイン名]                酔っ払い.JP
[Domain Name]                XN--N8J1C913R6J1B.JP

[登録者名]                  酔っ払い協議会
[Registrant]                 GUILD OF DRUNKS

[Name Server]                urquell.xn--n8j1c913r6j1b.jp
[Signing Key]                40756 8 1 (
                              872F8B4148E3AB1BFD8BCC45F9454819
                              CE667B96 )
[Signing Key]                40756 8 2 (
                              FC9F449CA7769A38A9028BE1E6220C8C
                              A98E7D34027D5755C526A7C40E75FBF5 )

[登録年月日]                2011/01/13
[有効期限]                  2015/01/31
[状態]                       Active
[最終更新]                  2014/02/01 01:05:15 (JST)
```

BIND9 の Smart signing を用いた DNSSECのお手軽運用 cont.

```
http://酔っ払い.jp/material/20110205-nisoc/yoppara.pdf  
http://酔っ払い.jp/material/20110205-nisoc/CreateZSK.sh.txt  
http://酔っ払い.jp/material/20110205-nisoc/SignZone.sh.txt  
http://酔っ払い.jp/material/20110205-nisoc/crontab.txt
```

酔っ払い.jp で利用している BIND9 の Smart signing を用いた DNSSECの運用と、とっつきづらい鍵のライフサイクルについて説明した資料がありますのでご興味があるかたはどうぞ。

dnsops.jp も同じ方法で運用中です。

Questions ?
