セカンダリDNSサービスについて
ゾーン転送を実装・改造してみた

坂口　俊文

DNS Summer Days 2016

2016/06/24

(発表後一部追記)

自己紹介

坂口　俊文

- 3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/

ゾーン転送(AXFR)

- プライマリDNSサーバとセカンダリDNSサーバ間で、ゾーンデータを同期するための方法
- ゾーンデータの同期は、rsync, scp等でも可能
- ゾーン転送は異なる組織やシステムの間で同期可能な方法
- セカンダリDNSサービスは、ゾーン転送でプライマリサーバからゾーンデータを取得
- 今回は増分ゾーン転送(IXFR)は扱わない

ゾーン転送(AXFR)

ゾーン転送時のプライマリ⇔セカンダリ間のやり取り

リクエスト

example.com.　IN　AXFR

レスポンス

example.com.　86400　IN SOA　(2016062401 …)
example.com.　86400　IN NS　ns01.example.com.
example.com.　86400　IN NS　ns02.example.com.
www.example.com.　86400　IN A　192.0.2.10

…

example.com.　86400　IN SOA　(2016062401 …)

実験

ゾーン転送は、最後にSOAレコードをセカンダリサーバへ送信して終了
もし、SOAレコードを送信せずに、延々とゾーン転送を続けると？

試しに実験してみる。

- ゾーン転送のプライマリサーバ側を実装
- セカンダリサーバとしてLinux + BINDを用意

⇒

メモリ使用量が増加し、OOM Killerで強制終了(Linux + BIND)

実験

メモリ使用量が急激に増加するようにプライマリサーバ側を改造

8GBメモリのセカンダリDNSサーバを用意し、LinuxのOOM Killerで終了するまでの時間を計測

プライマリサーバ
CPU: 4論理CPU
メモリ: 8GB
OS: Ubuntu 15.04

セカンダリサーバ
CPU: 2論理CPU
メモリ: 8GB
スワップ: なし
OS: CentOS 6.7 64bit

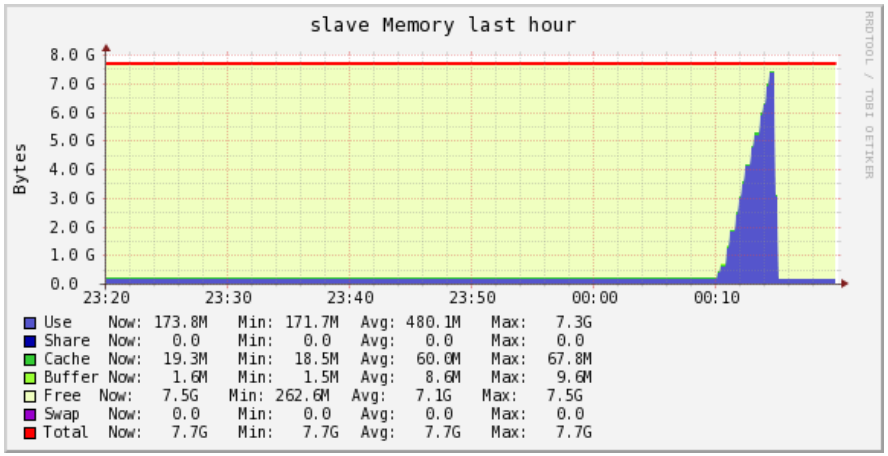
CPU: AMD FX-8120E
メモリ: 24GB
OS: Windows 8.1
VMware Workstation 10

BIND

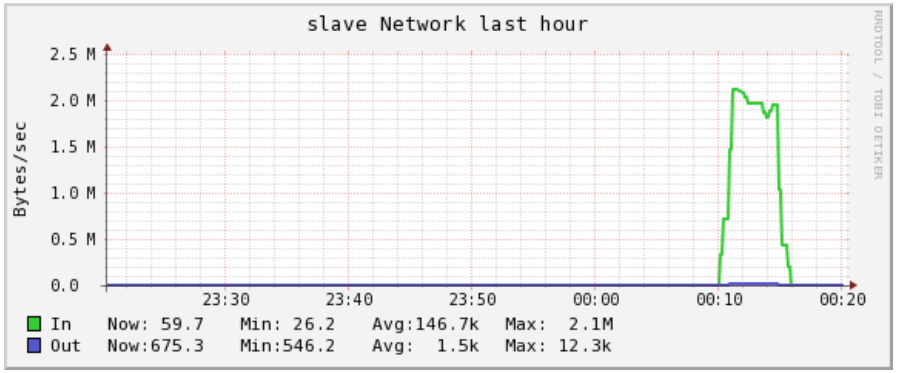
4分21秒でBINDが強制終了。

BIND 9.10.3

メモリ使用量(GB)



トラフィック(Bps)



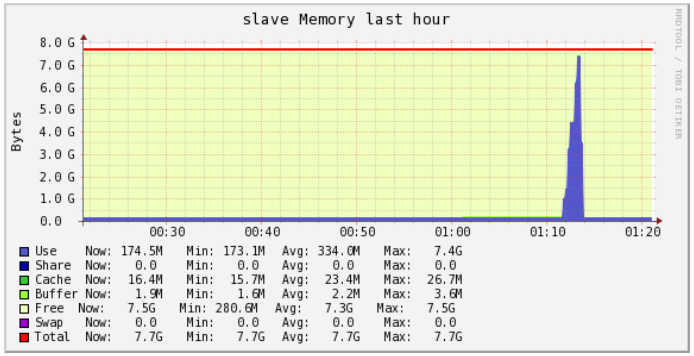
By Ganglia

knot DNS, PowerDNSの場合

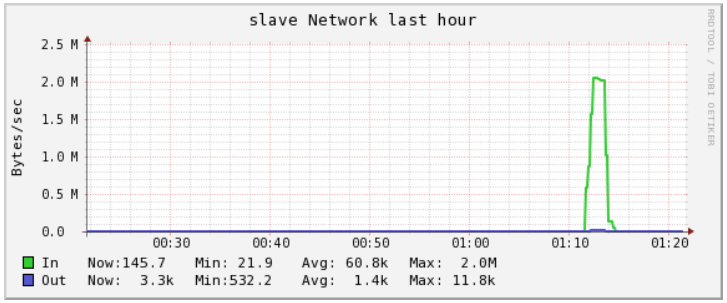
knot DNS(1分41秒), PowerDNS(2分55秒)も同じ。

knot 1.6.5

メモリ使用量(GB)

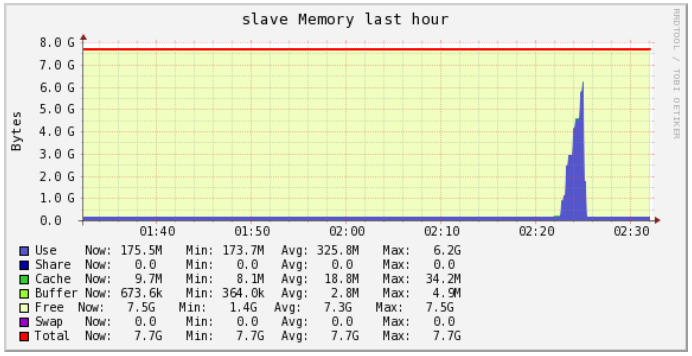


トラフィック(Bps)

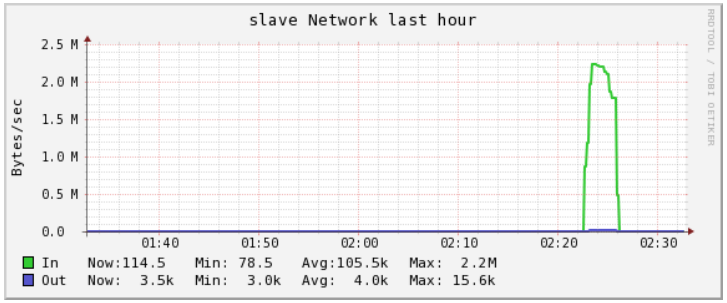


PowerDNS 3.6.8

メモリ使用量(GB)



トラフィック(Bps)



By Ganglia

NSDの場合

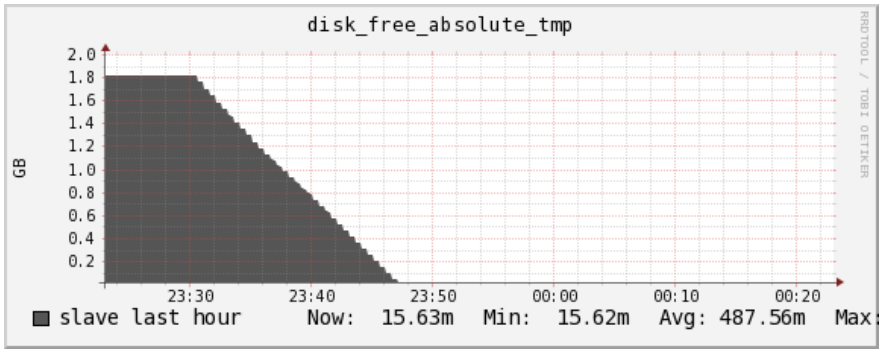
NSDではメモリの使用量は増加しない。

/tmp(xfrin: で指定したディレクトリ)に一時ファイルを作成するため、ディスクの使用量が増加。

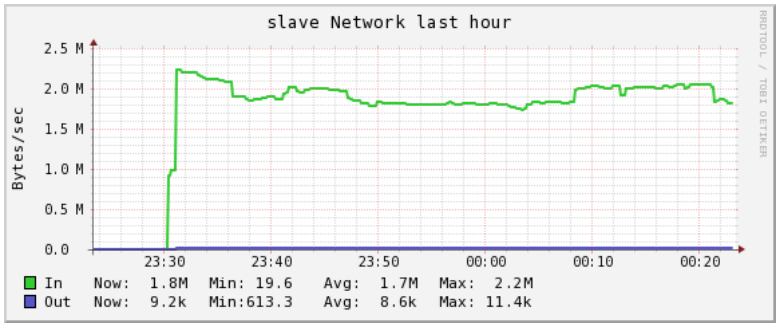
1.8GBの/tmpを16分44秒で使い切る。空きがなくなってもゾーン転送を停止しない。

NSD 4.1.5

/tmpの空き容量(GB)



トラフィック(Bps)



By Ganglia

サービス妨害可能か？

プライマリサーバをクラックできれば、これを利用してセカンダリサーバのDNSサービスを妨害可能。

しかし、プライマリサーバをクラックできた時点で、ゾーンデータを改竄することができるので、

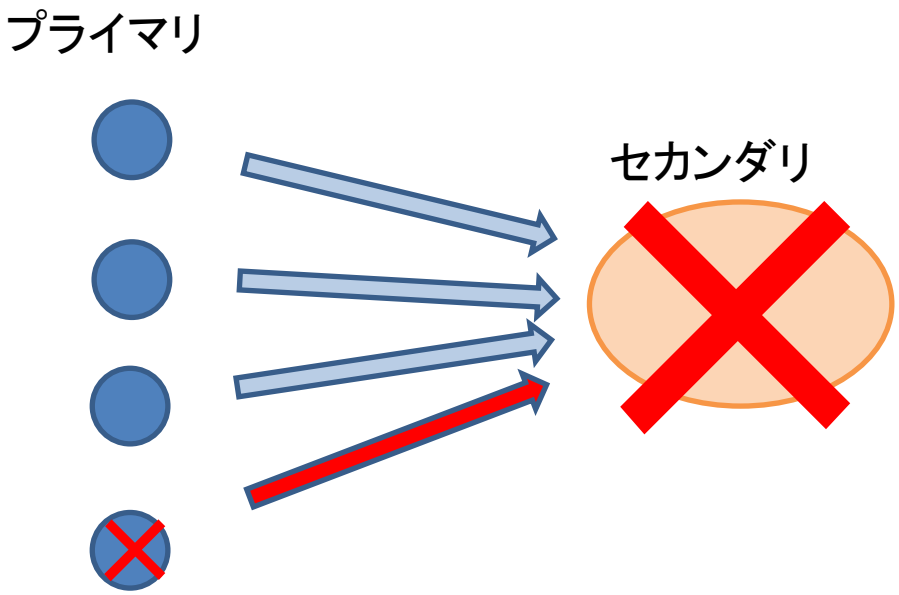
・偽のWebサイトへの誘導(Aレコードの操作)

・メールの覗き見(MXレコードの操作)

などが可能なため、心配する必要はなし。

セカンダリDNSサービス

- 多くのプライマリサーバに対してひとつのセカンダリサーバ。
- プライマリサーバは管理外
- ひとつのプライマリサーバからセカンダリDNSサービスを妨害可能



対策

1. **信頼できない管理外のサーバからゾーン転送を受けない。**
2. **ゾーン転送のデータサイズやレコード数を制限…する機能は、BIND, NSDなどにはない**。代替としてBINDの場合は、ゾーン転送時間を制限する。
   - max-transfer-time-in <min>; (default 120分)
3. メモリやディスク容量の空きを大きく
4. 同時に(concurrently)ゾーン転送可能な数を抑える。
   - transfers-in <n>; (default 10) サーバ全体
   - transfers-per-ns <n>; (default 2) NSあたり
5. トラフィックやリソースの監視

対策(BIND)

ゾーン転送のデータサイズを制限する機能を追加してみました。受信したデータのサイズが設定値よりも大きい場合は、ゾーン転送を中止します。

https://github.com/sischkg/xfer-limit/blob/master/README.jp.md

BIND

```
zone "example.com" {
   type slave;
   masters { 192.0.2.101; };
   file "example.com.db";
   max-transfer-size-in 2000000;
};
```

```
Jun 24 17:00:00 slave named[12944]: transfer of 'example.com/IN' from
192.0.2.101#53: Transfer status: bad zone
```

対策(NSD, knotDNS)

NSD

```
zone:
   name: "example.com“
   zonefile: "example.com.db“
   request-xfr: 192.0.2.101 NOKEY
   size-limit-xfr: 2000000
```

Jun 24 17:00:00 slave nsd[20391]: xfrd : transfered zone data was too large 2002280

knotDNS

```
example.com {
    file "example.com.db";
    xfr-in master0;
    xfr-in-limit 2000000;
    notify-in master0;
}
```

Jun 24 17:00:00 slave knot[6397]: error: [example.com] transfered data size is exceeded: size: 2001418, limit: 2000000
Jun 24 17:00:00 slave knot[6397]: error: [example.com] AXFR, incoming, 192.0.2.101@53: failed (failed)

対策(PowerDNS)

PowerDNS

```
mysql> select id from domains where name = 'example.com';
+----+
| id |
+----+
|  2 |
+----+

mysql> select * from domainmetadata;
+----+-----------+----------------+----------+
| id | domain_id | kind           | content  |
+----+-----------+----------------+----------+
| 2  | 2         | XFR-SIZE-LIMIT | 20000000 |
+----+-----------+----------------+----------+


Jun 24 17:00:00 slave pdns[16336]: Unable to AXFR zone 'example.com' from remote ' 192.0.2.101 '
(PDNSException): AXFR size is exceeded for example.com
```

履歴

2015/09/**	ゾーン転送の実装・実験
2015/10/02	BIND, NSDのパッチ作成
2015/10/04	knot DNSのパッチ作成
2015/10/05	IPAの脆弱性関連情報の届け出
2016/01/22	PowerDNSのパッチ作成
2016/04/**	IPA/JPCERTからBINDの開発者(ISC)へ連絡
2016/05/16	IPA/JPCERTを経由してISCから回答 本件は脆弱性ではなく、機能追加の要望として扱う
2016/06/06	IPAで脆弱性ではないと判断し、取扱い終了
2016/06上旬	IPA/JPCERTからNSD, knot DNS, PowerDNSの開発者へ連絡

最後に

プライマリ・セカンダリ間でゾーン転送を行うことは、一定の危険を伴うため、セカンダリDNSサービスを提供する際は、その対策が必要

例

1. プライマリサーバから非常に大きなデータを送りつけられる
2. SOAレコードのrefreshに非常に小さい値(1など)を設定される
   ([nsd-users] https://open.nlnetlabs.nl/pipermail/nsd-users/2016-June/002329.html)
3. ゾーン転送経由で、セカンダリサーバが停止する脆弱性
   (ゾーン転送で不正なAPLレコードを受け取るとnamedが停止
   https://jprs.jp/tech/security/2016-01-20-bind9-vuln-stringformat.html)

など