InternetWeek 2008 DNS BoF

キャッシュポイズニング攻撃に対する権威サーバ側の対策

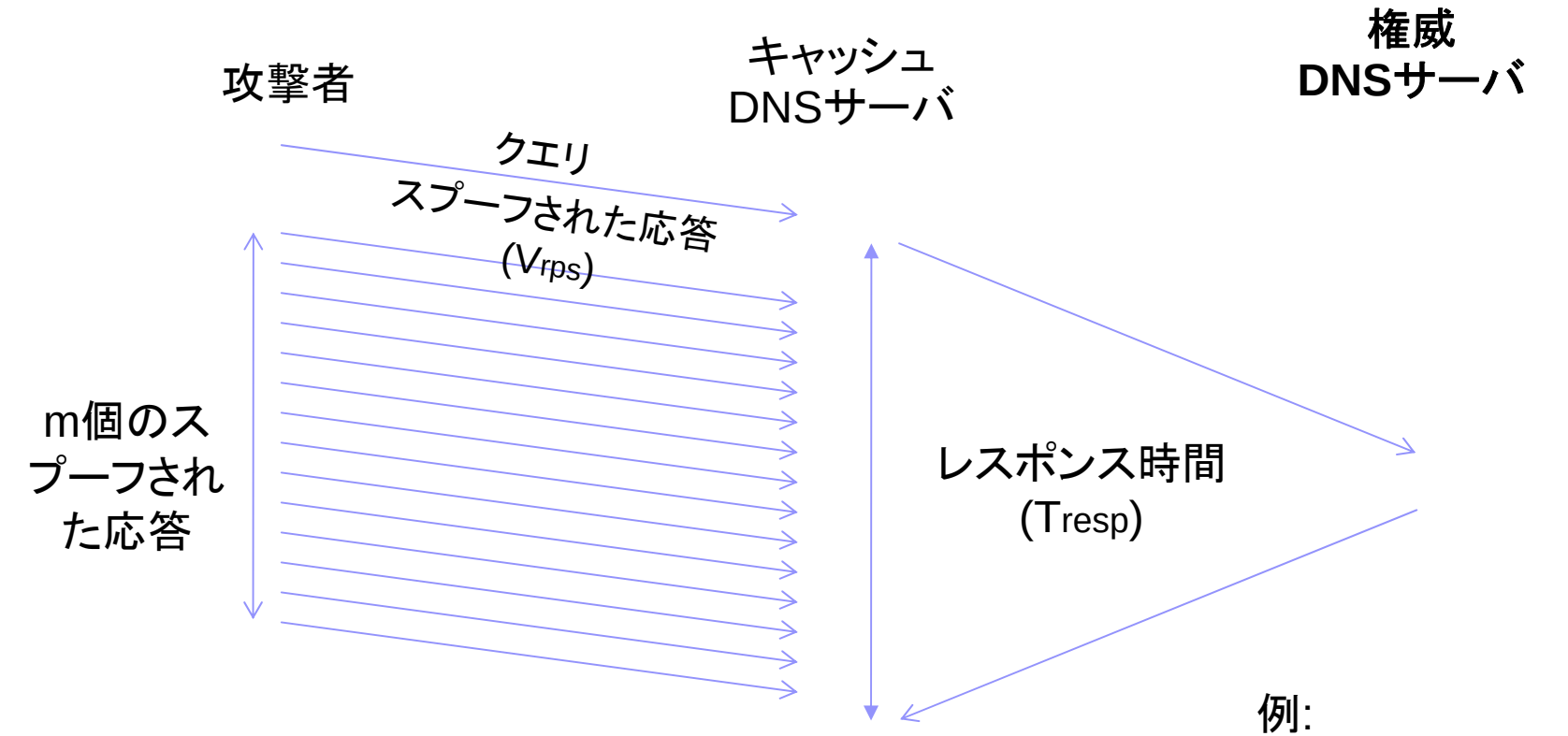
2008年11月25日

住商情報システム株式会社
森　拓也（tak@scs.co.jp）

背景

- キャッシュポイズニング脆弱性
  - いわゆるKaminskyアタックもその一つ
  - DNSの設計に起因
    - 根本的な「対策」は不可能
    - 確率を下げることが現状のワークアラウンド（一時的回避策）
      - 対策でもベストプラクティスでもない
  - 根本対策はDNSSEC?
  - 確率低減には?
    - キャッシュサーバ側の「対策」：ユーザー側の対策
    - 権威サーバ側の「対策」：コンテンツ提供者にできることは?

キャッシュ汚染確率（BIND 9.5.0P2）



Tresp：レスポンス時間
Vrps：スプーフ応答/秒
m = Tresp * Vrps

例:
Tresp：20 msec
Vrps：10000 スプーフ応答/秒
m = 20 * 10000 = 200

BIND 9.5.0-P2キャッシュ汚染確率

- 仮定:
  - N: 場合の数
    - N_{TXID}: トランザクションIDの数 (最大値:2^16)
    - N_{Ports}: 使うポート数 (最大値:63 * 2^10) (well-knownポートを除外)
  - $N=N_{TXID} * N_{Ports}$
  - N_{BIND}=2^16 * (63 * 2^10) → 約40億
- 等価な問題
  - N枚あるカードから一枚取る
  - 外れだったらそのカードは捨てる
  - m枚繰り返す
- 汚染確率
  - $P_{BIND}= 1-(1-P_1)(1-P_2)\ldots(1-P_m)$
    - $P_i = 1/(N_{BIND}-i+1)$ (i=2…m)←一枚ずつ減っていくので確率は大きくなっていく
  - $P_{BIND}= m/N_{BIND}$

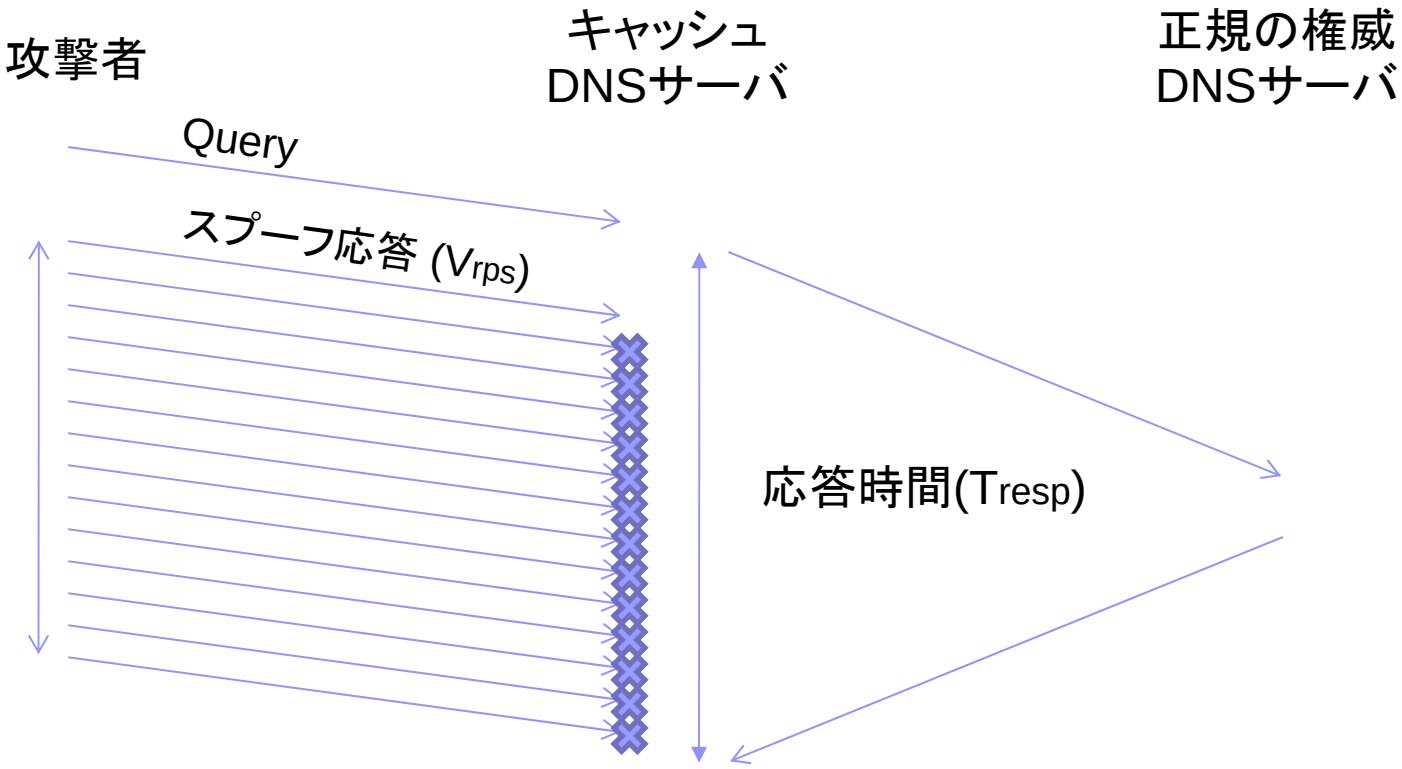
BIND 9.5.0 P2のキャッシュ汚染確率

- 有効なスプーフ応答の数
  - $m = T_{resp} * V_{rps}$
    - T_{resp}：正規の応答が返ってくるまでの時間
    - V_{rps}：秒間のスプーフ応答の数
- 計算例
  - m=200
    - T_{resp} : 20msec
    - V_{rps} : 10000 rps
- $P_{BIND}= m/N_{BIND}$=200/2^32→約5×10^{-6}

汚染確率は積算される

- 一回あたりの汚染確率は低くとも、繰り返すことにより汚染確率は積算されていく
  - p: 攻撃者からの一回の攻撃で汚染される確率
- N回の攻撃で汚染される積算確率
  - $P_N = 1 - (1 - p)^N = N * p$
- pを小さくすることも、Nを小さくすることも重要
  - Nを小さくする→攻撃を認識して対応を早く行う?
    - 対応って何ができる?
  - Nが大きければ、権威サーバ側で検出が容易に

ほかの実装の例：CNS 3.0.5.0



“Detect and Defend”機能により、応答のトランザクションIDが異なると、その応答は捨てて新規問い合わせを行う

「N」を大きくしないと、攻撃成功確率小

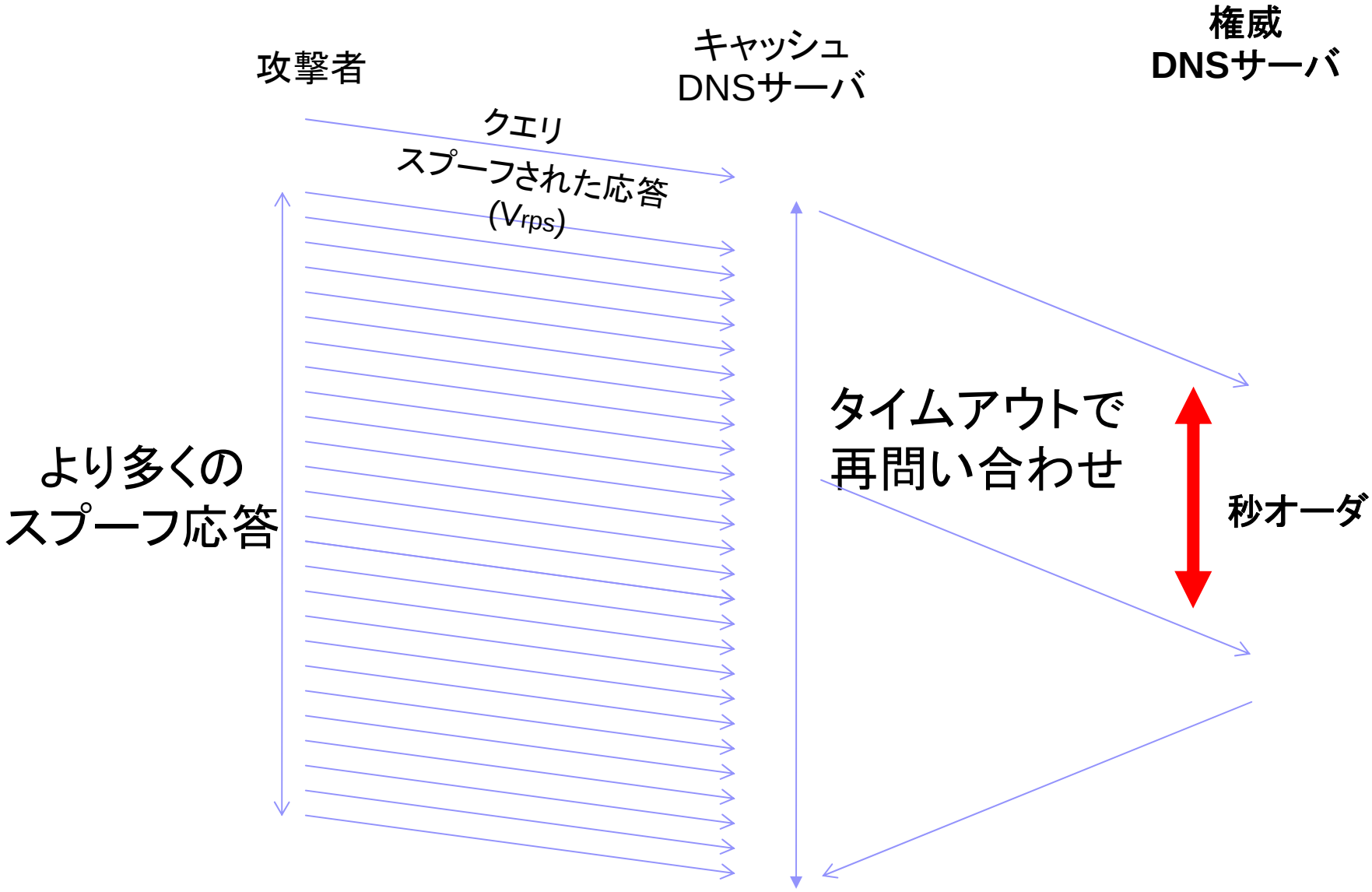
☆等価な問題

1. N枚あるカードから一枚取る
2. 外れの場合そのカードを戻す
3. m枚繰り返す

高可用性とresolverの動作

- DNSの仕組み自体、高可用性を考慮
  - キャッシュサーバ（フルリゾルバ）→権威サーバ
    - 複数のネームサーバをタイムアウトで渡り歩く
      - （一回目は）遅延が発生
      - 二回目以降はレスポンスタイムを覚えているので遅いサーバは使われない
  - 上記挙動は、キャッシュポイズニング攻撃に弱い
    - 実装によっては、複数サーバにクエリを出す物もある
    - タイムアウト時間も、固定ではなく過去の応答時間からダイナミックに決定する物もある

キャッシュ汚染確率（無応答時）

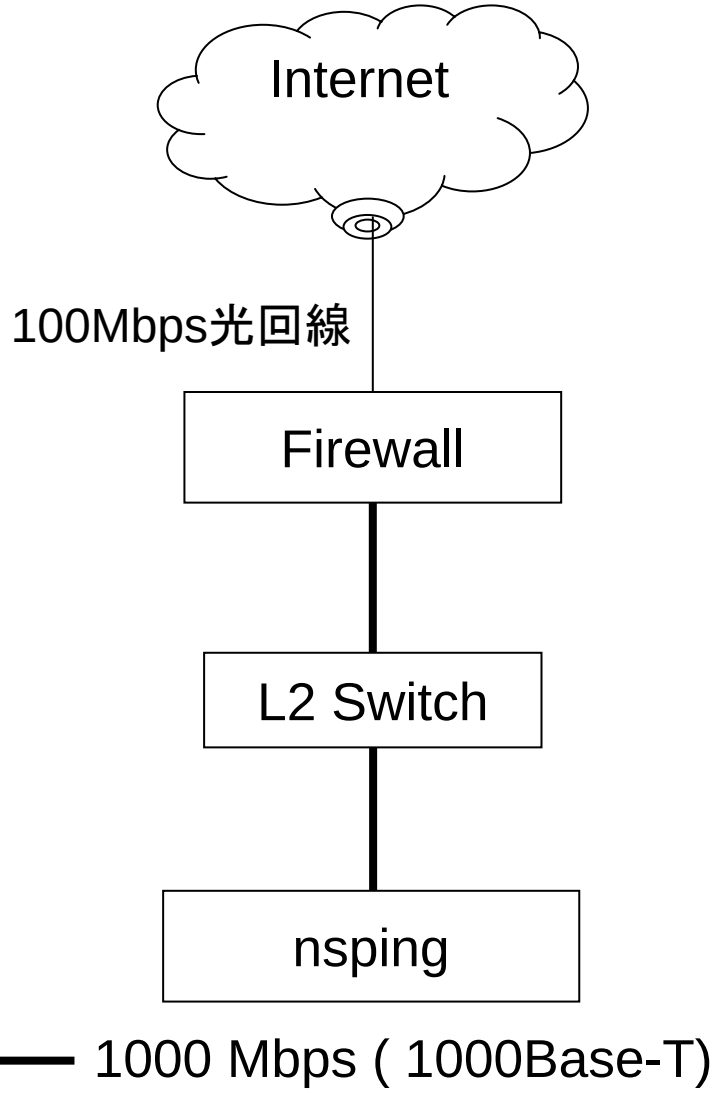


テスト対象

- 一番心配なのは金融系?
  - 特に銀行
- 対象
  - 銀行のWebページ
    - 全国銀行協会の会員Webページより抽出
      - http://www.zenginkyo.or.jp/
      - http://www.zenginkyo.or.jp/abstract/outline/organization/member_01.html
      - 会員
        - 正会員(127会員)
        - 銀行持株会社会員(3会員)
        - 準会員(63会員)

試験環境（金融系DNSサーバの応答調査）

- スペック
  - Cache server , Resperf
    - HP DL145
    - CPU: AMD Opteron 248
    - メモリ: 4GB
    - OS: RedHat EL5 / EL3 (32bit)
      - 2.6.18-92.1.17 / 2.4.21-47.0.1
  - テストツール
    - nsping 0.8



使用したツール
nsping

- Pingと同じようなことをDNSで行うツール
  - FreeBSDのport
  - Linuxでも、ライブラリ（**nsl, resolv**）をリンクすれば簡単に使用可能
- nsping [-z <zone> | -h <hostname>] –c <#> -p <port> -t <timeout> -a <local address> -P <local port> -T <type> <-r | -R, recurse?>
- 使用例
  - Nsping –h www.example.jp example.jp
- 注意点
  - 失敗（Missed)と表示される数は、実際に失敗した数と異なる
    - 失敗（Missed)-遅延（Lagged）-2 (-c 20の時)

テスト内容

- nspingを用い、DNSサーバへの負荷を考慮して、きわめて低いレートで応答を確認
  - 下記を3セット繰り返した
    - 5秒おきに10回問い合わせ
    - 10秒休憩
- 実際のコマンドは下記
  - nsping -c 10 -t 5.0 -h www.example.jp ns.example.jp

応答がなかったケース

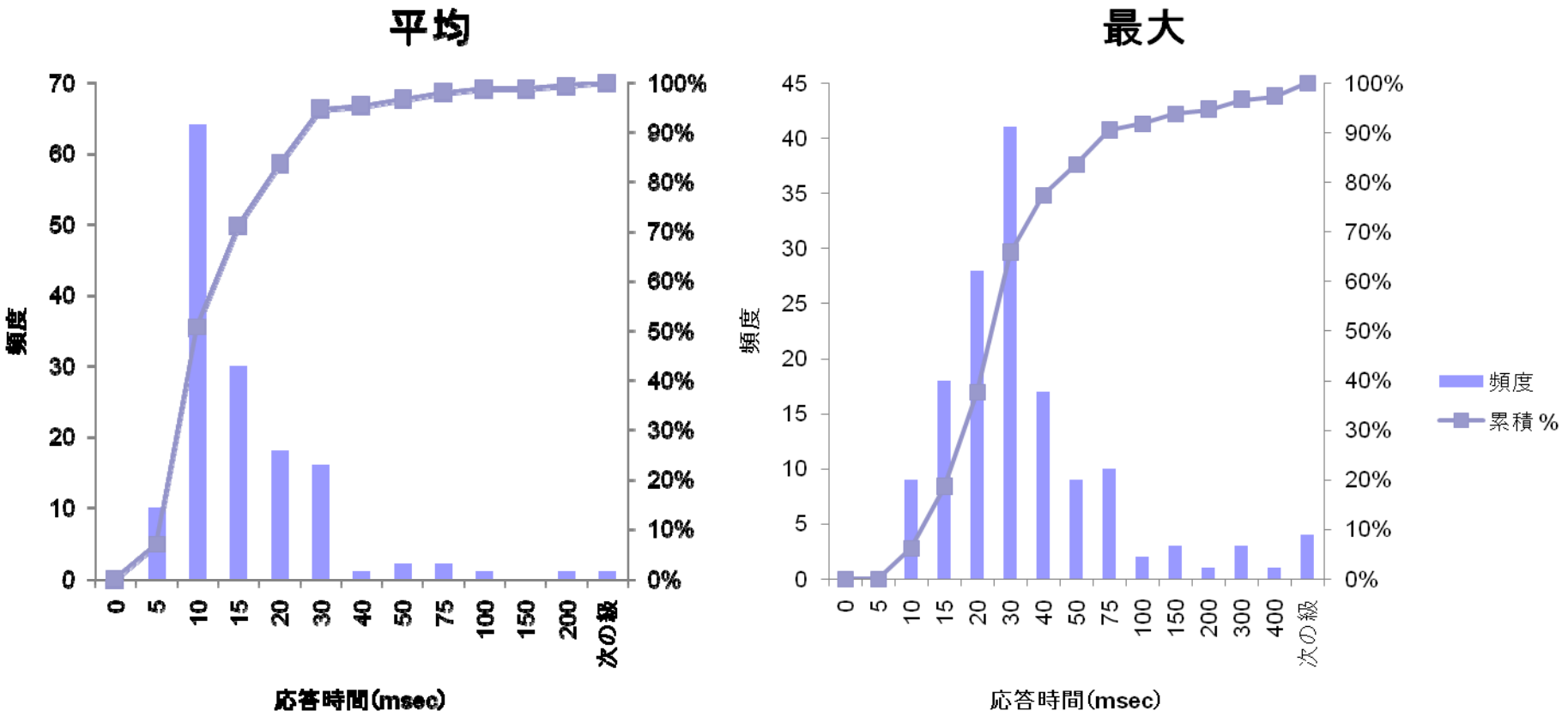
- 全Webサーバ数：146
- 全権威サーバ数：244（最大7行で共有）
- 応答がなかったケース（各権威サーバごと4回試行）
  - 9行（のべ12ケース）

【例】

```
1 www.example.jp:Total Sent: [  21 ] Total Received: [  18 ] Missed: [   3 ] Lagged [   0 ]
1 www.example.jp:Total Sent: [  21 ] Total Received: [  18 ] Missed: [   3 ] Lagged [   0 ]
2 www.example.jp:Total Sent: [  21 ] Total Received: [  17 ] Missed: [   4 ] Lagged [   0 ]
3 www.example.jp:Total Sent: [  21 ] Total Received: [  16 ] Missed: [   5 ] Lagged [   0 ]
5 www.example.jp:Total Sent: [  21 ] Total Received: [  12 ] Missed: [   9 ] Lagged [   2 ]
```

- 二倍応答が返ってきた：1

名前解決にかかった時間（平均/最大）



名前解決にかかった時間

- 応答時間の平均
  - 最頻値は、5-10msec
  - 応答全体の95%は30msec以内
- 応答時間の最大値
  - 最頻値は、20-30msec
  - 応答全体の8%は100msec以上
- 応答時間が100msecの場合、一回の問いあわせで1000qpsのスプーフ応答を出すと100発有効な攻撃が可能

高負荷時の挙動

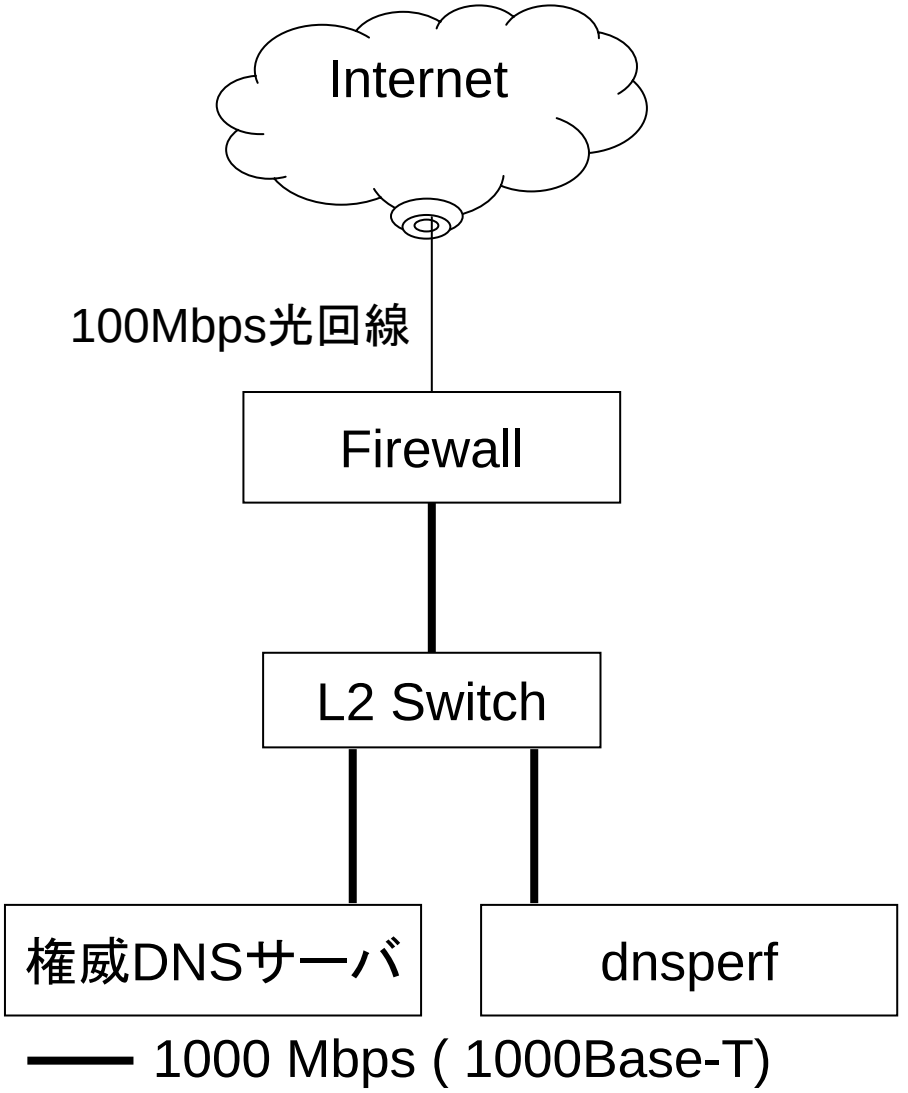
- 実験室環境で、高負荷時のサーバ挙動を観察
- 一般的に高負荷状態では:
  - 応答時間の増加
  - 無応答確率の増加
- 特に問題なのは無応答
  - 再問い合わせは秒のオーダ
    - リゾルバの実装に依存
  - (攻撃可能な)窓の大きさが劇的に広がる
- DoS攻撃との合わせ技で汚染確率を上げることが可能

高負荷時の権威サーバ挙動（BIND 9.5.0-P2）

- dnsperfで、負荷を発生
  - 同時「疑似」コネクション数（-q）を変化させて負荷を変化

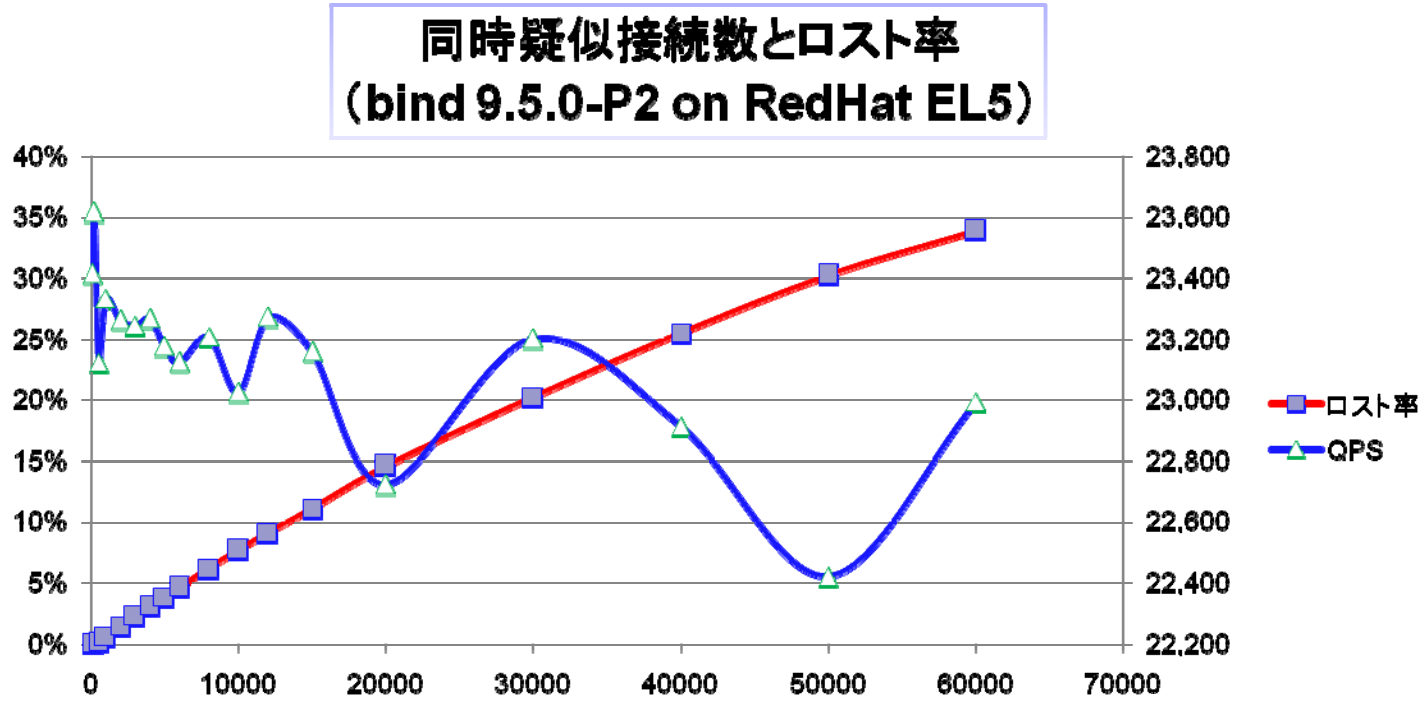
試験環境（高負荷時の権威サーバ挙動）

- スペック
  - Cache server , Resperf
    - HP DL145
    - CPU: AMD Opteron 248
    - メモリ: 4GB
    - OS: RedHat EL5 / EL3 (32bit)
      - 2.6.18-92.1.17 / 2.4.21-47.0.1
  - 権威DNSサーバ
    - Bind 9.5.0-P2
  - Nominum dnsperf
    - dnsperf 1.0.1.0



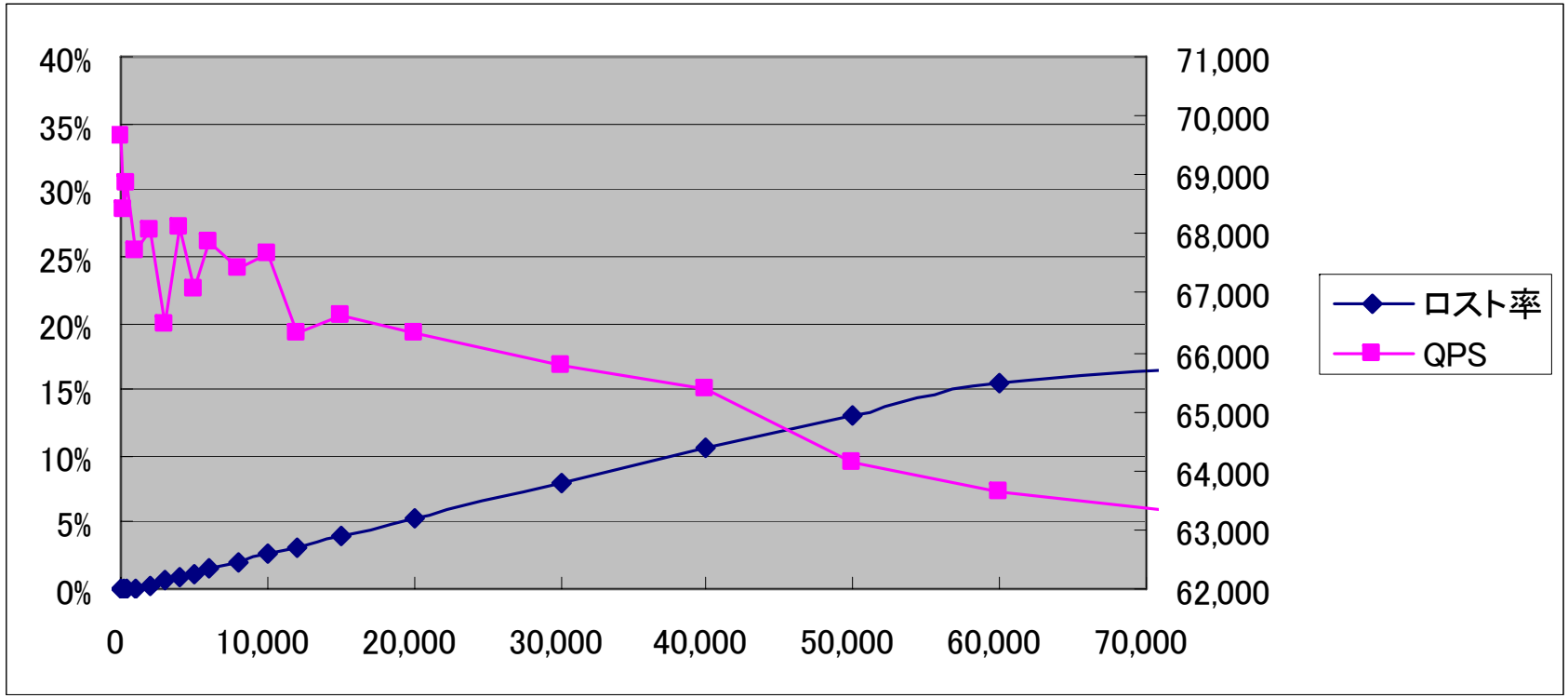
結果例

- 高負荷時には、ロストが多発
  - 別クライアントからのnspingの結果もほぼ同様



DoS攻撃下では、キャッシュ汚染リスク増大

別実装の例
（Nominum社ANS on RHEL3）



まとめ

- 権威サーバが頑丈ならば、権威サーバへの問い合わせ一回あたりのスプーフ応答の数を減らせる
  - 権威サーバ側で、攻撃の検出が容易に
  - 権威サーバへのDoS攻撃に対する対処もクリティカルなシステムでは必要
    - 意外と金融系等のDNSは…
    - RTTを短くするには、IPAnycastも有効か?
- キャッシュポイズニング攻撃への総合的な対策には、権威サーバの強化も必要か