

これでいいのかTTL

短いDNS TTLのリスクを考える

2007年1月26日

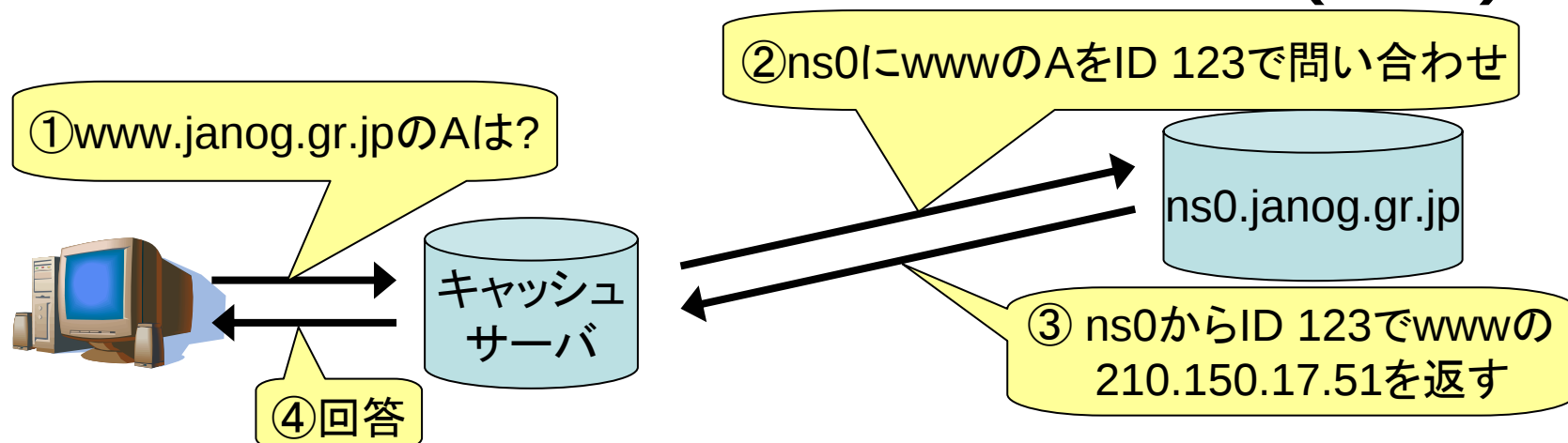
民田雅人

株式会社日本レジストリサービス
JANOG 19@沖縄県那覇市

DNSプロトコルのおさらい(1/2)

- 問合せと応答の単純な往復
 - この名前のIPアドレスは?
⇒ IPアドレスはXXXXだよ
 - トランスポートは主にUDP
 - 条件によってTCPになることもある
 - 問合せパケット クエリ名+ID+etc...
 - 応答パケット クエリ名+ID+回答+etc...
- ID: 識別のための16bitの値

DNSプロトコルのおさらい(2/2)



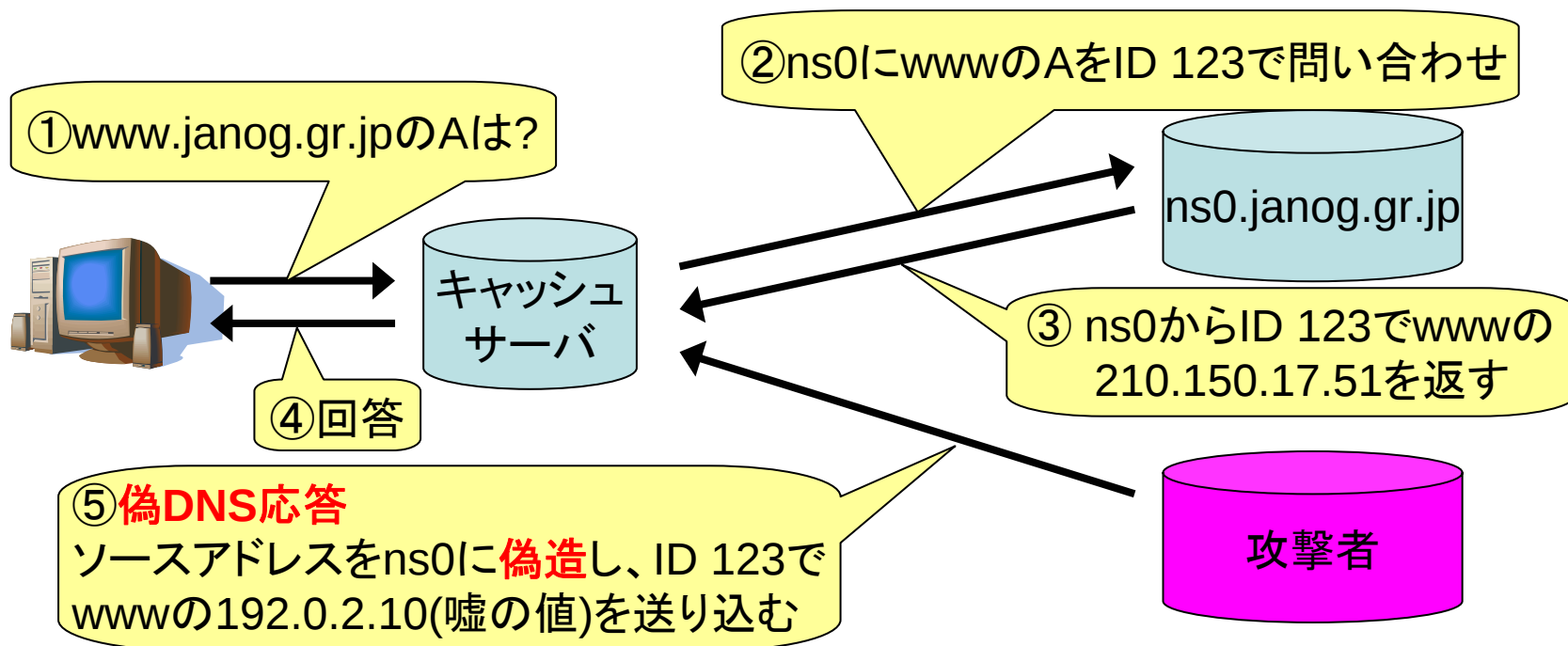
- キャッシュサーバは、応答パケットを、ソースアドレス、クエリ名、IDで識別
 - IDが違えば別の応答と判断
 - クライアントとキャッシュサーバ間(①と④)も同様

毒入れ攻撃とは

- DNS Cache Poisoning攻撃
 - DNSキャッシュサーバに
嘘の情報をキャッシュさせる
- 毒入れされたDNSサーバを使うユーザは、
別のサイトに誘導されてしまう
 - Phishingの可能性!
- 攻撃手法は二通り
 1. 正規の応答パケットに嘘の情報を仕込む
 2. 嘘の応答を正規の応答より先に返す

今回の話題！

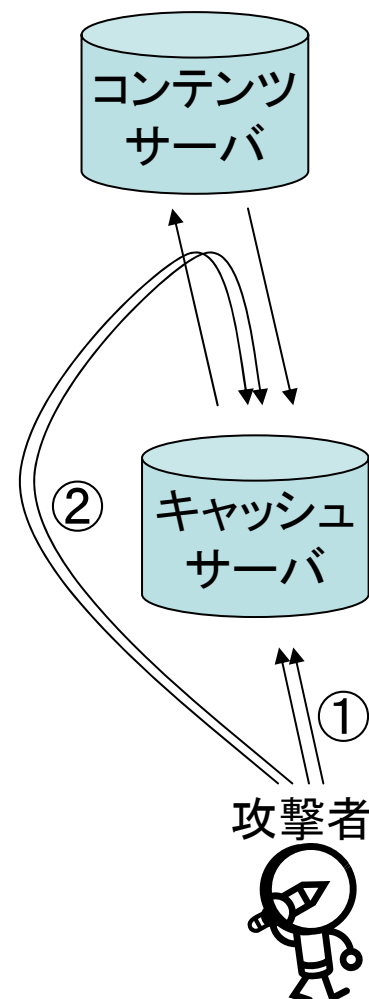
キャッシュサーバへの毒入れ



- ③より先に⑤の偽DNS応答が送り込まれると、キャッシュサーバは嘘情報をキャッシュする
 - ④で嘘をクライアントに送る
- クライアントPCは、偽のサイトへ誘導される

毒入れ攻撃の古典的手法

1. オープンなキャッシュサーバに大量のリクエストを出す
2. 同じサーバに対して、偽装したDNS応答パケットを、IDをランダムに変えながら送る
 - クエリ名とソースアドレスは自明
3. IDが正規応答と一致すれば攻撃が成功



DNSプロトコルの脆弱性

- CERT VU#457875で警告
 - 2002年11月初出
 - 参考 : <http://www.kb.cert.org/vuls/id/457875>
- ソースアドレスを偽装しやすい
 - DNSはUDPを主に使う
- IDは16bitしかない
 - 意外に当たりやすい
 - いわゆる「誕生日のパラドックス」を利用した「誕生日攻撃による毒入れ」ができてしまう

参考:「誕生日のパラドックス」

- ある場所に23人いた「この中で同じ誕生日の人がいると思う人」の問いに対して
 - ○×のどちらに答えるべきか?
 - を答えたほうが当たる可能性が高い
⇒23人いると、確率が0.5を超える
- ここでの「パラドックス」は「矛盾」ではなく、「直感的な感覚と合わない」という意味
- 「誕生日攻撃」 - Birthday Attack
 - 誕生日のパラドックスを利用した攻撃手法

誕生日攻撃による毒入れ

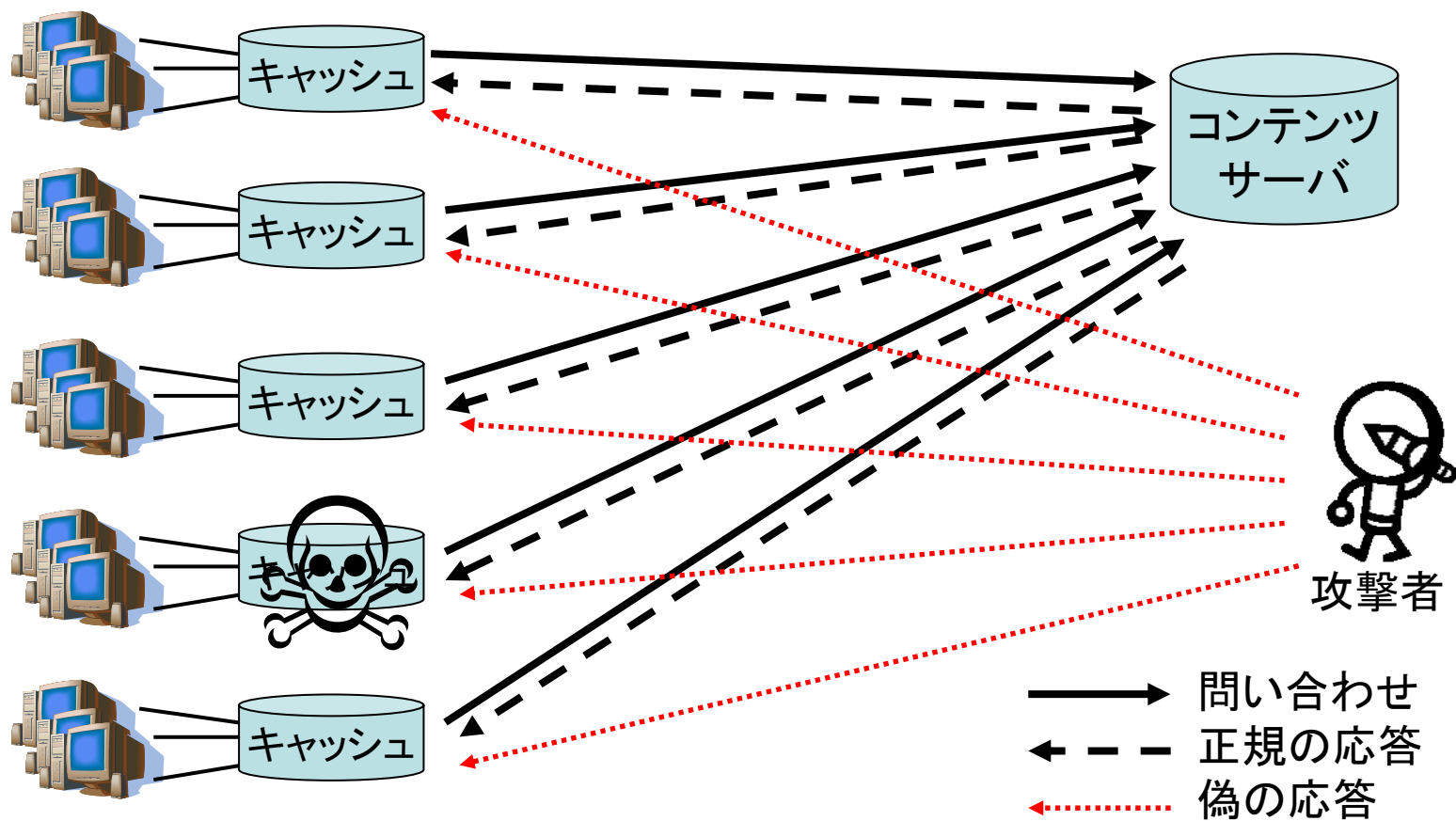
- 各DNS RR(リソースレコード)のTTLが短いとキャッシュサーバの問合せ頻度は高くなる
 - TTLが86400秒 → 1日に**1**回 問い合わせ
 - TTLが30秒 → 1日に**2880**回 問い合わせ
- 気長に嘘のDNS応答をキャッシュサーバに送り続ければ、**そのうち**当たる
 - 「そのうち」が**意外に短い**

攻撃のストーリー

- アクセスが多くTTLが短いドメイン名を狙う
 - どのキャッシュサーバでもかまわない
 - 誰が使っていようとかまわない
 - ユーザは多いほどよい
 - 気が付かれないようにこっそりと攻撃したい
- 同時に複数のキャッシュサーバへ
嘘の応答を定期的に送り続ける

⇒時間の問題で、どこかのキャッシュサーバへの
毒入れが成功する

攻撃の様子



あるキャッシュサーバへ 毒入れが成功する確率

$$P_s = \frac{R \times W}{N \times Port \times ID}$$

R: 攻撃対象1台あたりに送るパケット量(pps)

W: 攻撃可能な時間(Query⇒AnswerのRTT)

N: 攻撃対象レコードを保持するコンテンツサーバの数

Port: Query portの数(BINDの場合固定なので1)

ID: DNSのID (16bit = 65536)

どこかのキャッシュサーバへ 毒入れが成功する確率

$$P_V = 1 - (1 - P_S)^V$$
$$= 1 - \left(1 - \frac{R \times W}{N \times Port \times ID} \right)^V$$

V: 攻撃対象のキャッシュサーバの総数

毒入れが1回でも成功する確率

$$P_A = 1 - (1 - P_V)^A$$

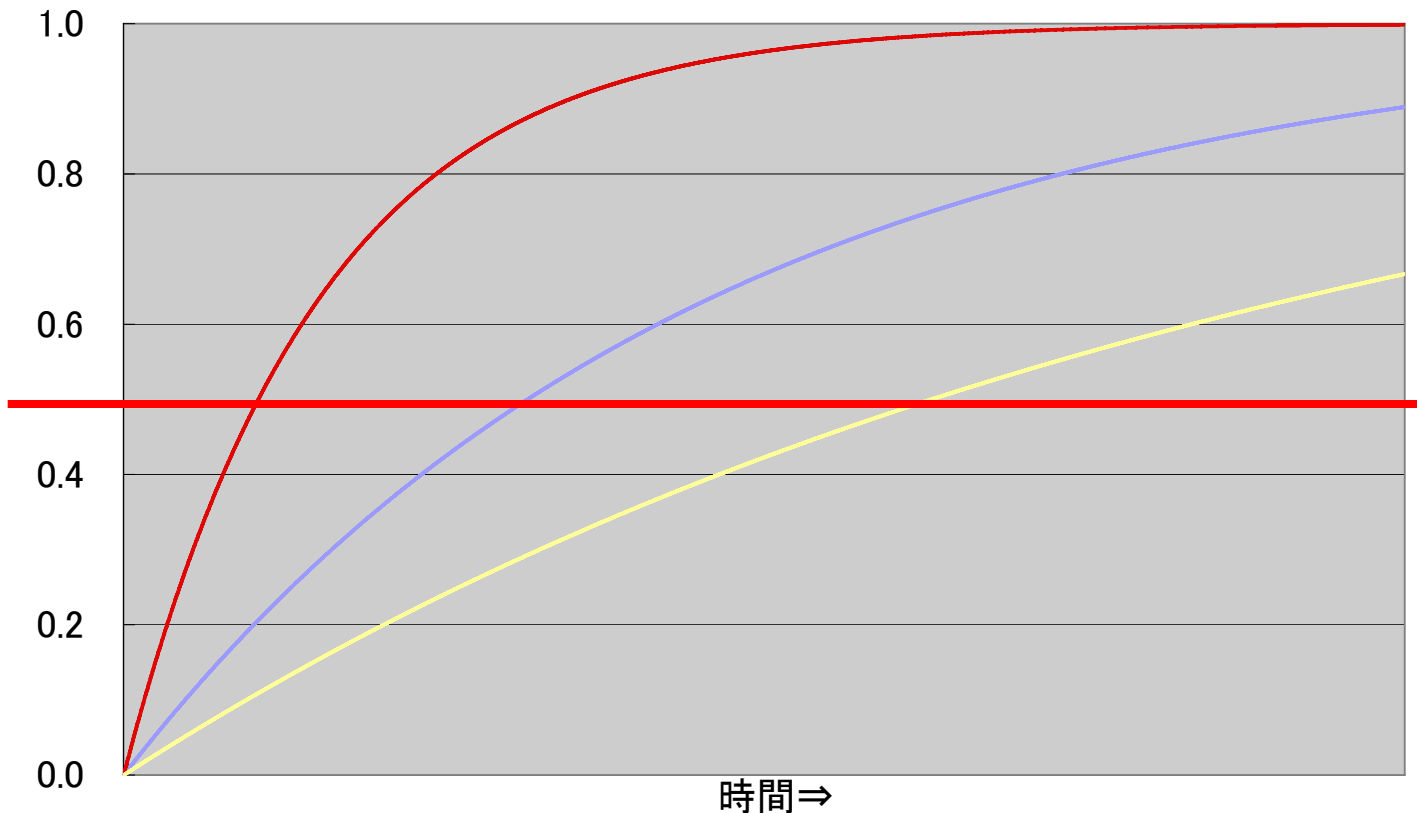
$$= 1 - \left[1 - \left\{ 1 - \left(1 - \frac{R \times W}{N \times Port \times ID} \right)^V \right\} \right]^{\frac{T}{TTL}}$$

$$= 1 - \left(1 - \frac{R \times W}{N \times Port \times ID} \right)^{\frac{V \times T}{TTL}}$$

A: 攻撃数(=T/TTL)

T: 攻撃時間 TTL: DNSレコードのTTL

毒入れが1回でも成功する確率の 時系列変化



確率が0.5を超えると毒入れが成功しても不思議ではない

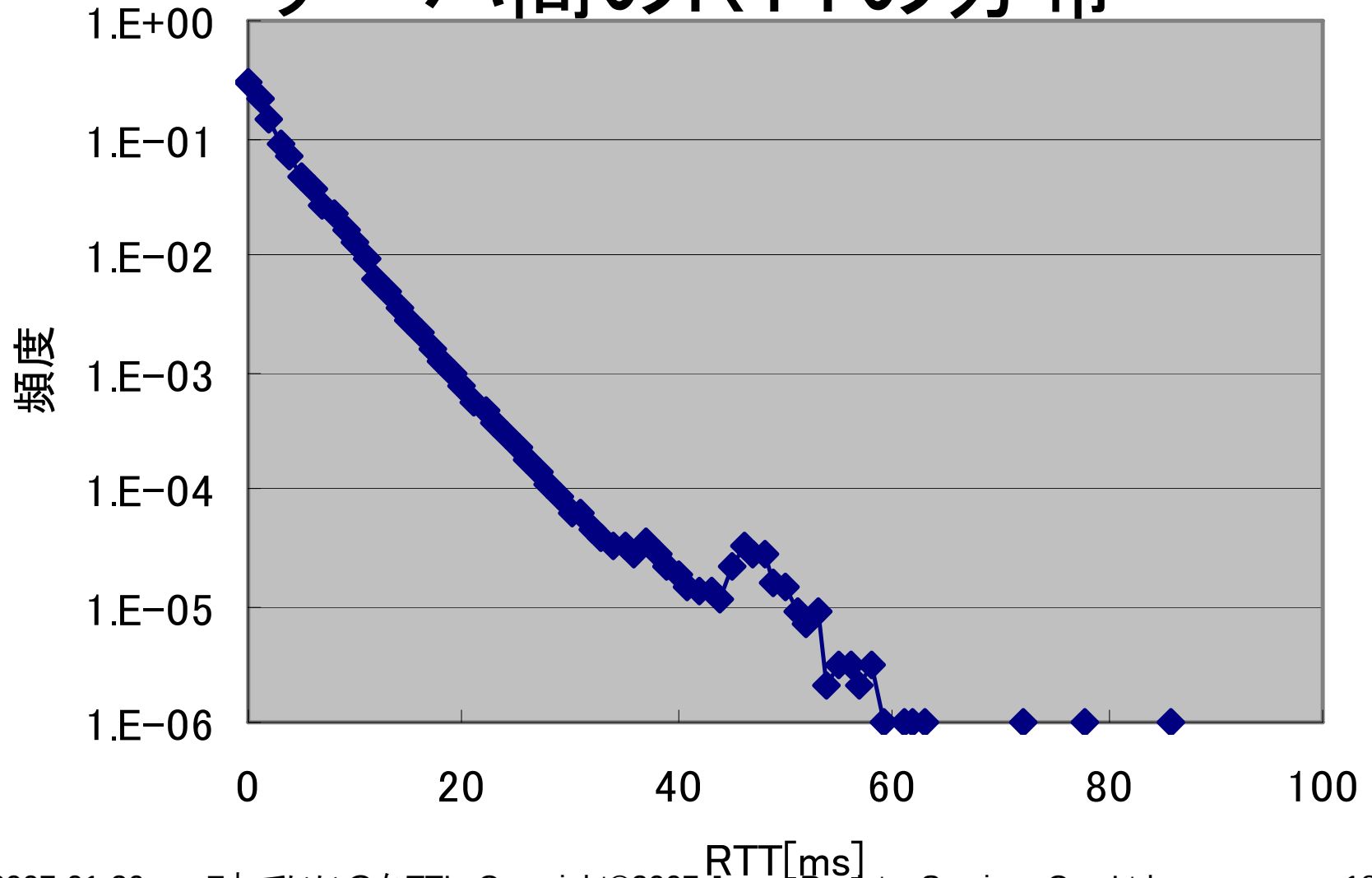
www.janog.gr.jpの例

- TTL **86400秒**
- Authoritative Server 1台
- 1台あたりの攻撃レート 10pps
- 攻撃対象のキャッシュサーバ 1000台
- RTT(とりあえず10msと仮定) 10ms
- 確率0.5を超えるまでの時間 **約15ヶ月**

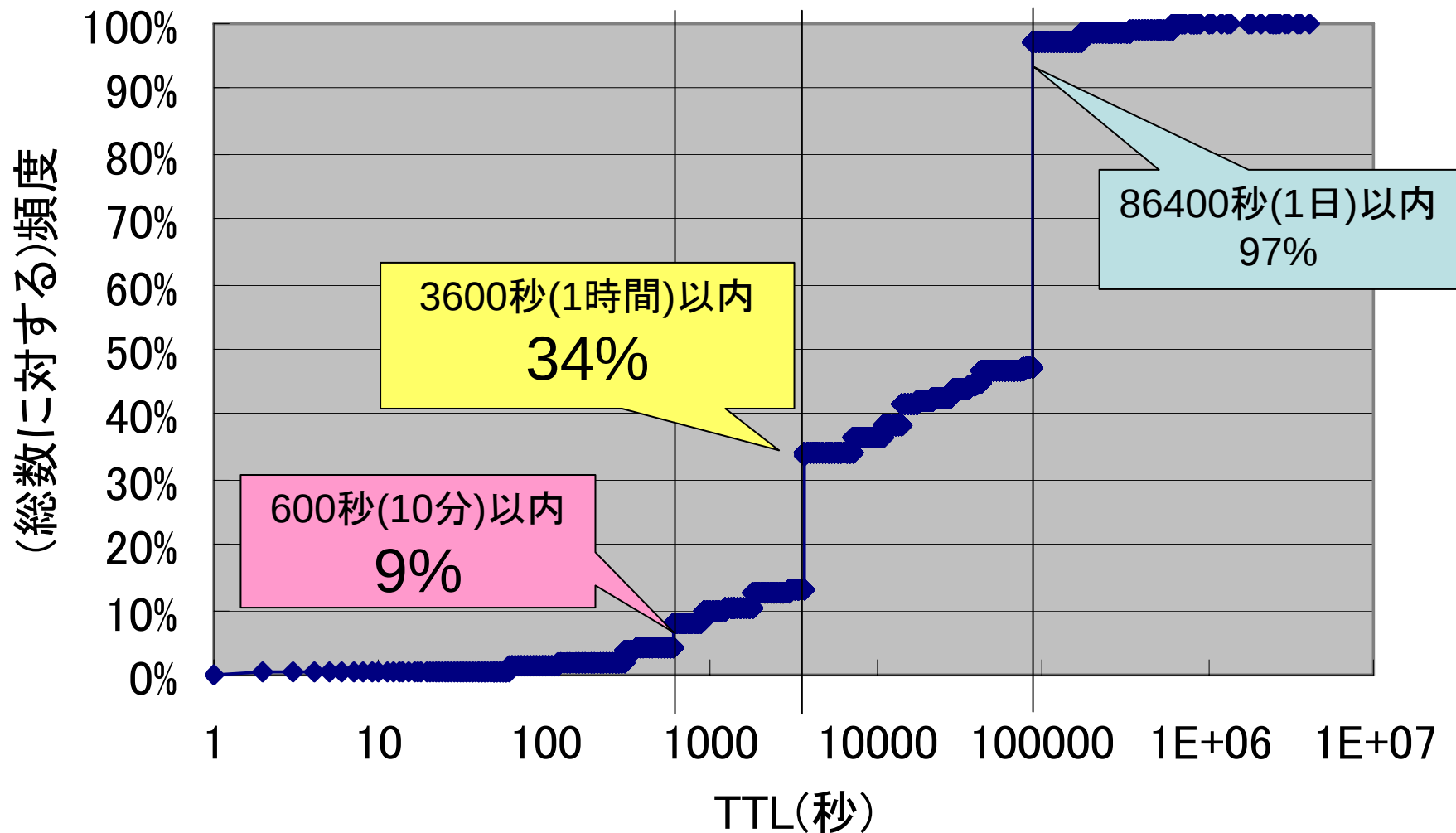
www.janog.gr.jpで 仮にTTLを30秒にすると...

- TTL **30秒**
- Authoritative Server 1台
- 1台あたりの攻撃レート 10pps
- 攻撃対象のキャッシュサーバ 1000台
- RTT(とりあえず10msと仮定) 10ms
- 確率0.5を超えるまでの時間 **約3.8時間**
 - 攻撃に必要な総帯域 約6.5Mbit/sec

キャッシュ⇔コンテンツ サーバ間のRTTの分布



DNSレコードのTTLの分布



著名サイトってどうよ？

ドメイン名	TTL	NS数	確率0.5超までの時間
???????.com	86400	3	3.8年
?????.jp	600	2	7日
www.???.net	300	2	76時間
www.???????.co.jp	60	4	30時間
www.???????.com	60	2	16時間
www.???????.co.jp	300	7	12日
www.?????.jp	30	2	8時間
www.???????.ne.jp	86400	2	2.5年
www.???????.com	300	2	76時間
www?.????.or..jp	20	8	21時間

10ppsで1000台のキャッシュサーバを同時に攻撃
RTTは10msで計算

スタッフとかスポンサーとか

ドメイン名	TTL	NS	確率0.5超までの時間
?????.jp	86400	3	3.8年
www.??????.co.jp	300	5	8日
www.??????.net	3600	3	57日
www.???.ad.jp	20	8	21時間
www.???.ne.jp	300	2	76時間
www.??????.co.jp	60	4	30時間
www.?????.co.jp	300	2	76時間
???.?????????????.org	3600	3	57日

計算条件は前スライドと同じ

CDN屋さんの気持ち

(Contents Distribution Network)

- TTLが短い代表選手
- 落ちてたからユーザさんを逃しました
ではいけない
「キャッシュなんかせずに、
毎回必ず聞き直しに来い」
「いつでもお前(SrcIPアドレス)に
最適なIPアドレスを教えてやるぜ」

問題点のまとめ

- TTLが短いと、
低レートでも攻撃が成功する確率が高くなる
 - 低レートの攻撃は検出しにくい
 - 攻撃が高レートになれば、
より短時間で攻撃が成立する可能性がある

それでもあなたは
短いTTLで運用しますか？

Q and A



おわりに

DNSコンテンツ設定側で 毒入れ確率を減らすために

- 極端に短いTTLは可能な限り避ける
- TTLを短くしなければならない場合、
ネームサーバの数を多くする
 - ただし、大きな効果は期待できない
 - 他のパラメータの調整は
コンテンツ設定側では不可能

問題の本質

- IDが16bit しかないDNSプロトコルの問題
 - もしIDが32bitやそれ以上であれば攻撃困難
- キャッシュサーバのクエリポートが固定
 - クエリポート固定の実装
BIND系全て
WindowsのDNSサービス etc...
 - クエリポートが動的に変わる実装(攻撃困難)
dnscache (djbdns)
PowerDNS recursor

根本解決のために

- Ingress Filterの導入 (BCP38 RFC2827)
 - インターネットに対して、ソースアドレスに嘘をついたパケットを送出できなくする。
(さまざまな攻撃を防止できるようになる)
- DNSSECの導入
 - DNSの応答パケットが正式のものか虚偽のものかを判断でき、虚偽の応答を捨てることができる

参考文献とIETFの動き

- draft-ietf-dnsext-forgery-resilience-00.txt
 - 特定(1台)のキャッシュサーバを攻撃した場合について記述してある
- キャッシュサーバのクエリポートは、動的に変わるように実装するべきであるという方向へ

Acknowledgements

This presentation is included the research activities founded by National Institute of Information and Communications Technology (NICT).

Special Thanks to
Information Sharing Platform Lab. NTT
and Staff of JANOG 19