第 XVI 部

DNS extension and operation environment

第 16 部
DNS extension and operation environment

第 1 章　DNS WG

DNS WG は、主に DNS のプロトコルに起因した問題や、DNS の運用上発生する問題について、その解決法や改善策の提案に関する研究を行う。具体的には、

- DNS に対する攻撃の調査と、それを防ぐための手段の研究
- DNSSEC もしくはそれに取って代わる新たなプロトコルの考案や、運用と普及に対する障害を解決するための研究
- DNS の運用もしくは実装上発生する問題点の調査と、その解決方法の研究

などを行う。

現在の主なトピックとして、DNS に対する攻撃について、その脅威および対策の研究が挙げられる。

本報告書では、2004 年度に WIDE 合宿で行った DNS に対する攻撃実験の結果および、DNS のセキュリティ技術として考案されている DNSSEC の要不要に関する考察を述べる。

第 2 章　WIDE 合宿における DNS 攻撃実験
—Monkey in the middle attack 実験報告—

概要

インターネットの普及にともない DNS の重要度は増しているが、現在の DNS プロトコルには、いくつかの攻撃対象となりうる部分がある。そのうち、DNS データを改竄する攻撃では、利用者を別サイトに誘導することができ、容易に Phishing 型詐欺を行うことができるため、金銭的利益を得る目的の攻撃が行われる可能性がある。

本実験では、盗聴可能なネットワークに対し、中間者攻撃を行い、DNS データの改竄ができることを示した。

2.1 DNS への攻撃実験

DNS はインターネットにおける唯一で不可欠な分散データベースであり、インターネットの普及にともない重要度を増している。ところが、現在の DNS プロトコルには、いくつかの攻撃対象となりうる部分があり、RFC3833 Threat Analysis of the Domain Name System にまとめられている。DNS への攻撃のうち、サーバへの DoS 型攻撃では、機会の損失を与えることはできるが、それ以上の被害を与えることは困難である。DNS データを改竄する攻撃では、利用者を別サイトに誘導することができ、容易に Phishing 型詐欺を行うことができるため、金銭的利益を得る目的の攻撃が行われる可能性がある。

また、共有ネットワーク型の無線 LAN が普及しているが、このネットワークではユーザが出すパケットを容易に盗聴できるため、ユーザとサーバの間のデータを改竄する中間者攻撃を行いやすい。そこで今回は、中間者攻撃（Monkey-in-the-middle attack）を行い、DNS データの改竄ができることを示した。

具体的には、無線ネットワークでパケットを監視し、適度に偽の名前解決応答を正しい応答の前に返すという攻撃プログラムを試作し、影響を観察する実験を行った。その結果、ほとんどのユーザを騙しサイトへ誘導することができた。

2.2 節にて実験の目的を定義し、2.3 節にて開発したソフトウェアの説明、2.4 節にて実験環境を紹介する。2.5 節にて実験手順を示し、2.6 節に結果を分析する。また、2.7 節にて考察を行い、2.8 節に今後の課題をまとめる。

2.2 実験の目的

本実験は、DNS に対する現状の脅威を明らかにし、DNS の運用者ならびに開発者に対して、改善策の提言ならびに貢献を行うことを最終的な目的とする。

まず、WIDE 研究者に、DNS による名前解決の脆弱性と弊害の大きさを実感させ、攻撃の影響と防御方法に関する議論と研究を促進する。また、WIDE 合宿の無線 LAN 接続環境は大規模なイベント（IETF、InternetWeek、N+I など）で提供される無線 LAN

環境やホットスポットと同等の環境であり、WIDE 合宿環境で検証できればホットスポットなどでの問題点を指摘できる。

そのために、影響を受けた OS の種類、人数、被害の状況などの情報を収集する。さらに、より効率的で致命的な悪影響を与える攻撃方法や、被験者が攻撃を受け、別サイトに誘導されたことに気づかないようにする方法などについて、意見の募集を行う。

2.3 攻撃に用いたソフトウェア

攻撃ソフトウェアは、pcap ライブラリ経由で BPF や Raw ソケットを用いて共有ネットワークを流れる DNS 問い合わせパケットを監視し、問い合わせを解釈し、攻撃応答パケットを生成し、それをネットワークに送る。

今回は主に Linux で動作する Raw ソケットを用いた uso800d と、主に BSD 系の OS で動作する BPF を用いた dnsattack という 2 つのツールを試作した。また、従来から dnshijack/dnshijacker というツールが出回っているが動作原理は同じである。

2.3.1 uso800d

pcap ライブラリを用いて DNS 問い合わせパケットを拾い、生成した攻撃応答を raw ソケットを用いてネットワークに出力している。Linux にて動作する。

2.3.2 dnsattack

pcap ライブラリを用いて DNS 問い合わせパケットを拾い、生成した攻撃応答を BPF を用いてネットワークに出力している。FreeBSD、NetBSD、MacOSX で動作確認を行っている。

2.4 実験環境

本節では、実験を行った環境について述べる。今回の実験は WIDE 合宿にて行った。WIDE 合宿は 200 人以上の WIDE 研究者が集まる場であり、多くの被験者を得ることができる。なお、実験は前述の通り無線 LAN のネットワークセグメントにて行った。

2.4.1 トポロジ

今回の実験環境となる WIDE 合宿ネットワークの無線セグメントは、1 つの Layer-3 ネットワークセグメントにて構成された。すなわち、ユーザはすべて同一セグメントに存在している。また、WIDE 合宿地において DHCP にて指示されるリゾルバ DNS サーバは、無線 LAN セグメントとは別のネットワークに存在している。

前述の通り、無線 LAN は 1 つのネットワークセグメントで構成されている。しかし、複数の会議場もしくは部屋にて無線 LAN を提供するため、無線 LAN 基地局は複数台設置され、ローミング接続を利用してサービスが提供された。

そのため、実際には複数の無線 LAN 基地局をスイッチで接続した形態となっている。また、ユーザに 802.11a 方式と 802.11b 方式という、2 つの無線 LAN 方式を提供した。さらに、広い部屋、複数の部屋をサポートするため、1 つの方式でも複数のチャンネルを用いて無線ネットワーク間のローミング接続を提供している。

無線 LAN にて盗聴型攻撃を行う場合、1 つの無線 LAN 基地局にアソシエーションして盗聴するだけでは、その無線 LAN 基地局にアソシエーションしているクライアントの通信のみしか盗聴することができない。

とくに、最も広い Plenary 部屋では、複数のチャンネルを用いて複数の場所に無線 LAN 基地局を設置しているため、攻撃するためにも複数のホストが必要となる。

2.4.2 使用機器

今回の実験のために、次の機器を使用した。

- 802.11a
  攻撃マシン NotePC（Linux）
  計測兼誘導先マシン miniPC（Linux）
  誘導先で提供したサービス http、ssh、telnet、pop3、imap、smtp、ftp
- 802.11b
  攻撃マシン 1 MacOSX（11b – channel 6）
  攻撃マシン 2 FreeBSD（11b – channel 1）
  計測兼誘導先マシン FreeBSD（11b – channel 1）
  誘導先で提供したサービス http、ssh

また、使用する無線 LAN カードによって、クライアントから無線 LAN 基地局へのパケットを盗聴できる場合とできない場合があった。とくに 802.11a の無線 LAN カードにて盗聴できない場合が多く発生した。

これは、Linux や FreeBSD の無線 LAN カードのドライバに依存することがわかった。そのため、今

回の実験では、無線 LAN カードドライバに手を加え、クライアントから無線 LAN 基地局への通信も盗聴できるよう改造したドライバを利用した。

2.5 実験

攻撃の効果を段階を経て見極めるため、今回は 3 段階にわけて実験を行った。

まず、DNS WG の BoF の時間に、実験を行うことを周知した上で、DNS WG BoF 参加者を対象に実験を行った。これを第 1 回実験とする。その結果を踏まえ、第 2 回実験として、Plenary 部屋にて被験者を増やし実験を行った。この際は、開始時間を告げずに、Plenary 開催中のどこかの時間帯にて 15 分間攻撃を行うことをあらかじめ告知した。さらに、第 3 回実験として、何も周知せずに Plenary 部屋にて実験を行った。

それぞれの実験について述べる。

2.5.1 第 1 回実験（BoF）

この実験では、BoF に参加した DNS 有識者を対象とした。BoF 部屋の中で実験したため、部屋にいる約 40 名の人を攻撃の対象とした。実験開始をアナウンスし、すべての IPv4 アドレス問い合わせ（A 問い合わせ）を 203.178.136.57（www.wide.ad.jp）に、IPv6 アドレス問い合わせ（AAAA 問い合わせ）を ::1 に誘導した。なお、2 つの攻撃プログラムのうち、uso800d を用いて 802.11a の攻撃を行い、dnsattack を用いて 802.11b の攻撃を行った。

その結果、全参加者約 40 中 6 名を除き、85%を誘導することができた。誘導されなかった 6 名は、違う部屋に設置されている無線 LAN 基地局を利用していたか、Proxy を利用していた人であることがわかった。誘導されなかった 6 人の環境を次に示す。

被験者 1 Windows XP、802.11a（atheros）
被験者 2 Windows XP、802.11b、using Web Proxy
被験者 3 Windows XP、802.11b
被験者 4 Linux（RedHat-9.0）、802.11b
被験者 5 Windows 2000、802.11a
被験者 6 Windows XP + VMware（NetBSD）、802.11b

この実験の結果、攻撃によって多くの人を誘導できることがわかった。次に、この攻撃方法のスケーラビリティを確認するため、より広い範囲を対象として実験を行うこととした。

2.5.2 第 2 回実験（Plenary）

第 2 回実験は、ほぼ参加者全員が集まる Plenary の時間帯を選び、参加者全員を被験者として行った。参加者には実験開始時間を明確に告げずに、15 分間実験を行った。

また、誘導先となるホストを用意し、Web サーバ、ssh サーバなどを動作させ、誘導されたことを被験者にわかるようにした。さらに、別のマシンで無線セグメントのパケットを tcpdump にて記録し、本来の DNS 応答よりも攻撃ツールが素早く偽応答を送れているか検証した。

実験環境では、802.11a 1 チャンネルと、802.11b 3 チャンネルの無線セグメントが使用されていたが、そのうちの 802.11a と 802.11b のうちの 2 チャンネル分を攻撃した。今回は、すべての A 問い合わせに対して誘導先マシンの IP アドレスを答えた。

この実験でも、第 1 回実験と同じく、uso800d を用いて 802.11a の攻撃を行い、dnsattack を用いて 802.11b の攻撃を行った。

なおこの実験においては、攻撃ホストの台数の制限から、すべての無線 LAN 基地局にアソシエーションすることができなかった。そのため、参加者全員を被験者とすることができなかった。

すなわち、この攻撃を効率的に行うためには、無線 LAN 基地局 1 台に対して 1 台の攻撃ホストが必要であることがわかった。この実験結果の詳細については、2.6 節にて述べる。

2.5.3 第 3 回実験（Closing）

第 3 回実験は、とくに実施のアナウンスをせずに、ゲリラ的に行った。合宿最終日の Closing にて、ほぼ参加者全員が集まっている環境にて実施した。今までの実験と同じように、uso800d を用いて 802.11a の攻撃を行い、dnsattack を用いて 802.11b の攻撃を行った。

今回の実験では、次の名前の問い合わせに対してのみ嘘の応答を行うよう、攻撃ツールを設定した。

- www.asahi.com
- www.yahoo.co.jp
- www.zakzak.co.jp
- www.2ch.net
- slashdot.jp

誘導先の偽サーバには Web サーバを動かし、実在するニュースサイトに類似したデザインのページが

表示されるよう設定した。

この実験の結果、参加者に対して、詐欺目的での誘導の怖さを体験してもらうことができた。

2.6 第 2 回実験の詳細

本節では、被験者数が最大で、パケットキャプチャを行った第 2 回実験の詳細結果について述べる。

第 2 回実験においては、802.11b では、1、6、11 チャンネルの 3 波のうち、1 チャンネルと 6 チャンネルに属しているクライアントを攻撃した。また、802.11a では、61 チャンネルに属しているクライアントを攻撃した。

そのため、すべての参加者を被験者とできたわけではなく、攻撃ホストがアソシエーションしていた無線 LAN 基地局にアソシエーションしていたユーザのみを被験者として実験を行った。

2.6.1 攻撃による影響

802.11a へアソシエーションしていた人への攻撃結果を、表 2.1 に示す。これは、誘導先の偽ホストにて tcpdump を行い、コネクションを張ろうと試みてきたホストのソース IP アドレスを集計した。なお、誘導されたクライアントがコネクションを試みたプロトコルの内訳を表 2.2 に示す。

表 2.1.　偽ホストに誘導された被験者の統計

被験者	65
一度でも誘導された人	60

表 2.2.　偽ホストに誘導されたプロトコルの内訳

http	55
ssh	12
telnet	1
pop3	10
imap	1
smtp	3

表 2.3.　偽ホストに誘導された被験者の統計（802.11b）

一度でも誘導された被験者数	53
http サーバに誘導された被験者数	51
ssh サーバに誘導された被験者数	11
1 チャンネル以外の誘導された被験者数	17
1 チャンネルの被験者数	52
1 チャンネルで誘導された被験者数	34
1 チャンネルで誘導されなかった被験者数	18

これらの結果から、92.3%の被験者が一度は誘導されたことがわかる。

802.11b では、攻撃は 2 つのチャンネルを対象としたが、機材の都合でパケットダンプを取ることができたのは 1 チャンネルのみであるため、1 チャンネルでの分析を詳細に行った。802.11b において偽ホストに誘導された被験者数を表 2.3 に示す。被験者数はユニークな IP アドレス数で数えている。

これらの結果から、802.11b 1 チャンネルでは 65.3%の被験者が一度は誘導されたことがわかる。

2.6.2 DNS パケットの分析

次に、攻撃ホストとは別のホストにて tcpdump コマンドによるパケットキャプチャを行い、本来の DNS サーバの返答と、攻撃ホストによる攻撃の返答の時間差の分析を行った。

802.11a における結果を図 2.1 に、802.11b における結果を図 2.2 に示す。

この図において、横軸は問い合わせパケットの時刻、縦軸は問い合わせに対して攻撃ホストが答えた時刻から、本来の DNS サーバからの応答が得られた時刻を差し引いたものである。すなわち、0 より下に位置する点は、攻撃ホストの応答が本来の DNS サーバよりも早かった場合を示す。逆に 0 より上に

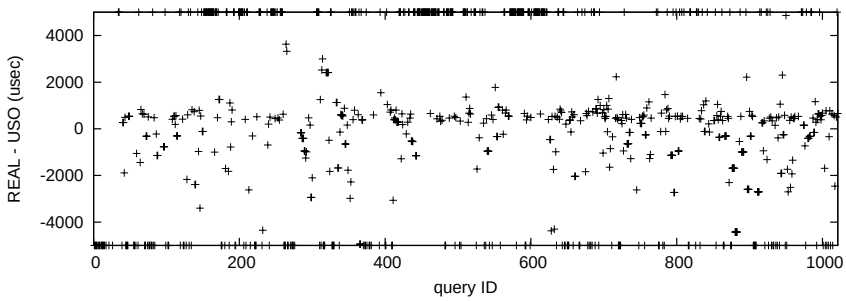

図 2.1.　DNS 応答の分析 802.11a

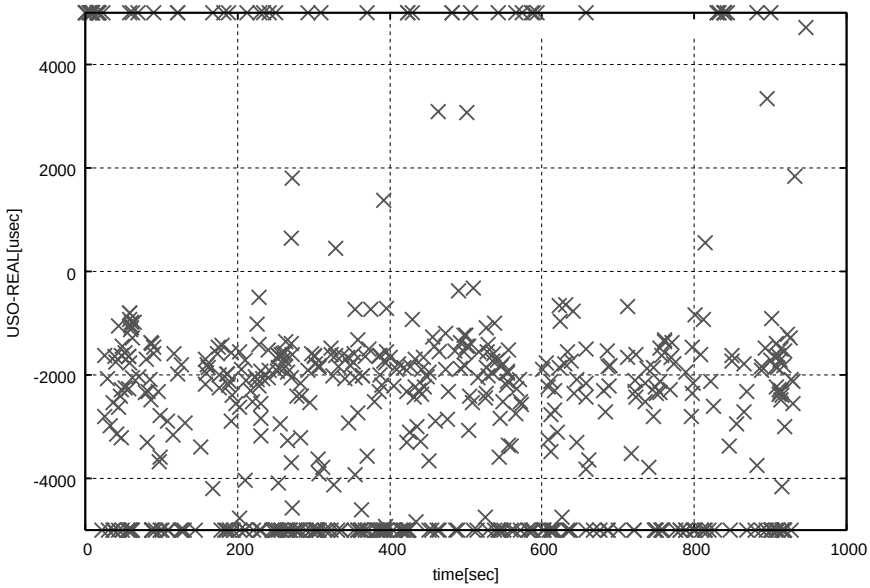

図 2.2. DNS 応答の分析 802.11b channel 1

表 2.4. DNS 応答の内訳 802.11a

観測された問い合わせ数	1021
攻撃ホストのほうが早く応答した場合	654
本来の DNS サーバのほうが早く応答した場合	330
応答が観測されなかった場合	37

表 2.5. DNS 応答の内訳 802.11b channel 1

観測された問い合わせ数	599
攻撃ホストのほうが早く応答した場合	535
本来の DNS サーバのほうが早く応答した場合	64
本来の応答が観測されなかった場合	6
攻撃ホストの応答が観測されなかった場合	53

位置する点は、本来の DNS サーバの応答のほうが早かった場合を示す。

また、−5000 usec の横軸に張り付いている点は、攻撃ホストからの応答しか観測できなかった場合を示し、5000 usec の横軸に張り付いている点は、本来の DNS サーバからの応答しか観測できなかった場合を示す。

表 2.4、表 2.5 に、内訳を示す。

2.6.3 被験者へのアンケート

今回の実験についてのアンケートを実施し、38 人からアンケートを回収した。質問項目は以下のとおりである。

影響がなかった被験者 7 名のうち、3 名は実験中に DNS 検索をともなう新しい接続を行っていなかったというコメントがあった。

表 2.6. アンケート結果

ホスト数			39
誘導された	総数		32
	OS 別	WindowsXP/2000	20
		MacOSX	4
		NetBSD/FreeBSD	5
		Linux	3
	DNS サーバ設定	localhost	2
		DHCP	30
影響なし	総数		7
	OS 別	WindowsXP/2000	3
		MacOSX	1
		NetBSD/FreeBSD	2
		Linux	1
	DNS サーバ設定	localhost	1
		DHCP	6

1. 実験にて名前解決が騙されましたか？
2. 1. にて騙されたと答えた方は、どんなアプリケーションが騙されましたか？
3. 1. にて騙されなかったと答えた方は、リゾルバにどの DNS サーバを指定されていましたか？（IP address）
4. お使いになっている環境を差し支えない範囲で教えて下さい。（OS、無線 LAN）
5. その他、コメント、不思議な挙動、要望などありましたらご記入をお願いいたします。

アンケート結果を表 2.6 に示す。

2.7 実験の考察

本節では、第 2 回の実験結果をもとに実験結果の考察を行う。

まず、図 2.1、図 2.2 に示した DNS 応答の分析に関して、同一無線 LAN 基地局にアソシエーションしているホストでの tcpdump による結果なので、無線の電波状況による取りこぼしが発生していることが考えられ、誘導先ホストでの記録と合わせて結果を出さないと、正確な結果とはならないと考えられる。また、無線 LAN の基地局はブリッジとして動作しているが、別の無線チャンネルのパケットが漏れてきている場合があるという報告があり、すべてのチャンネル・イーサネットセグメントで記録をとり、つきあわせないと正確な分析はできない。

11a の場合、パケットダンプからは応答の 65%において攻撃が成功しているが、誘導先サーバにおける観測結果から見ると、90%以上を誘導できたという結果となった。11b 1 チャンネルの場合、パケットダンプからは応答の 90%で攻撃が成功しているが、誘導先サーバにおける観測結果から見ると、65%の被験者を誘導できたという結果となった。

本実験にて誘導することができなかったユーザが存在したのは、次の理由があると考えられる。

- 無線 LAN 基地局の問題
  無線 LAN 基地局の数だけ攻撃ホストを準備することができなかった。また、攻撃に利用した一部の OS では、確実にねらった無線基地局にアソシエーションさせることが難しかった。
- 問い合わせパケットが見えない
  OS の無線 LAN カードドライバの特性により、クライアントから基地局へのパケットを盗聴できないものが存在した。
- 本来の DNS サーバより速く攻撃応答を返せなかった場合
  本来の DNS サーバにキャッシュされているデータの場合、十分速く応答を返せるため、それよりも速く誘導応答を生成できない場合があった。
- IPv6 トランスポートでの問い合わせに非対応
  今回使用したツールが IPv6 トランスポートに非対応だったため、IPv6 による名前問い合わせを利用していたユーザを誘導することができなかった。

また表 2.6 より、OS による明らかな差異はない。ただし、UNIX 系 OS を使用していて、DNS 問い合わせに IPv6 トランスポートを使用している場合や AAAA を検索している場合は今回の実験の対象としていないため、誘導することができていない。

以上の考察から、広範囲な無線 LAN ネットワークにおいて、すべてのユーザを誘導するのは困難であるが、一部のユーザを一度でも誘導することは容易であることがわかった。

2.8 今後の課題

これらの実験結果をふまえ、データからのさらなる分析結果を導き出すことが今後の課題である。また、この攻撃を防ぐための有効な手法も同時に検討していく。

第 3 章　DNSSEC 要不要に関する議論

3.1 はじめに

IETF において長期にわたりプロトコル的検討が行われている DNSSEC ではあるが、そのインターネット実環境への導入は遅々として進んでいない。また、DNSSEC deployment を考える際に、その導入コストに見合う効果が得られるのかという基本的な問いに対して、技術コミュニティの中でもコンセンサスが得られていない。この状況を鑑み、ここでは、DNS WG における要不要論を対比させる形で内容の改訂・増補を行い、今後同様の議論が発生した際の足掛かりとなるべく、議論の整理を試みる。

3.2 議論

3.2.1 コスト対効果の議論

不要論：

DNS の response が正当であることが判明した際にも、最終的にはその先のアプリケーションレベルでのセキュリティも必要となることがほとんどである（https、ssh の host/user 鍵による認証など）。また、DNS の偽 response による攻撃を受けた場合も、その後の認証が完遂されれば致命的な事態が防げることも多い。この状況において、deployment のための労力をかけてまで DNSSEC を普及させることが本当に必要であるか？

対応する必要論：

本来セキュリティは 1 ヶ所だけ固めれば十分であるとは言えない。DNS のセキュリティは DNS のレベルで解決されるのが望ましい。https などによる解決策は対症療法的であり、社会インフラとして定着した DNS には、自己のレベルにおいて integrity 検証機能を有するセキュリティ構造が必要である。また、PKI ではドメイン名空間とは独立の信頼構造を構築するが、DNSSEC では、上位レジストリ・ドメイン名登録者間の DNS 委任構造と信頼構造が一致するため、信頼構造の構築手順がより簡潔となる。

DNS サーバ実装の DNSSEC 対応については、DNS サーバとして最も share を持つ BIND が対応したことから考えて、かなりの比率のサーバが潜在的に DNSSEC 機能を備える状態にまでは普及するだろう。時間と共にサーババージョンは自然に置き換わっていくため、このレベルまでの deployment コストは低いと考えられる。ただし、サーバ管理者が実際に設定を行うことについては別の問題となる。

root から TLD（top level domain）[1] の階層において、DNSSEC 化を行うことは登録規模が小さく、登録者（TLD）の DNS に関する専門性が高いため、十分に現実的である。

この phase への到達時期を予測することに難しさはあるものの、DNSSEC Deployment Working Group[2] など、幾つかの関連団体においても deployment に関する議論が本格化されつつあり、数年以内に root から多数もしくは殆んどの TLD への DNSSEC 的委任が達成され[3]、引き続き一部の TLD が opt-in 形式により、ドメイン名登録者に対して DNSSEC 対応したゾーン委任を開始する[4] ことが期待される。

この際、TLD レジストリシステムへの DNSSEC 機能の導入は、比較的低コストで実現できるので、TLD が DNSSEC 対応した委任を開始することは十分に可能である。

root、TLD が DNSSEC に対応されれば、有志のドメイン名登録者が、DNSSEC 運用を行うことは容易となる。このような経験を蓄積していくことで、DNSSEC は定着していくと考えられる。

不要論からのコメント：

DNS のセキュリティは DNS で行うというのも、全体としては多少状況が改善されるにすぎない。この改善度合いに対するコスト対効果の検証を行うべきであり、要はサーバ管理者に移行へのモチベーションを与えられるかである。ただし、運用できる環境があるならば、試行してみることは検討に値する。

必要論からのコメント：

潜在的な危機をどう捉えるかは個人差の問題があり、定量的な評価を行いがたい。しかし、世の中の状況を鑑みるに、将来大規模なサイバーテロのようなものが現実に行われる可能性を考えることは杞憂ではないだろう。インフラが準備されていなければ、問題発生時に迅速な対応は行われずに、より大きな被害を生むことになる。たとえば、DNSSEC を用いない DNS に対しては、相手に気づかれずに Phishing 詐欺などの攻撃を効果的に行うことができる。DNSSEC の本格導入によりこの例をはじめとする DNS に対する攻撃が減少することが考えられ、攻撃者のオプションを減らすという側面において、この改善は無駄でない。このような意義を認められるサーバ管理者[5] にとっては、移行するモチベーションはあるだろう。これは、他ユーザに何らかの責任を負う可能性のある組織体にとって、セキュリティ対応の一貫として自己の DNS サーバにおける対応を完了しておくということが、ビジネス上の責任論的な観点から重要な意味を持つためである。また、この際、個人的目的で運用されているようなドメイン名の管理者を含む、すべてのネームサーバ管理者の移行が必要なわけではない。

1　ここでは gTLD と ccTLD の双方を含む.
2　http://www.sdl.sri.com/other/dnssec/
3　すべての gTLD とローカルポリシーによって対応した ccTLD.
4　この段階は、DNSSEC 関連ワークショップ “Building a Road Map for DNSSEC Deployment”（NANOG 31 Meeting 併設で開催）において分類された 4 段階の deployment phase: Isolated — 数個の署名済みゾーンが存在する状態　Sparse — 多くの署名済みゾーンがばらばらに存在する状態　Dense — 多くの署名済みゾーンが密集して存在する状態　Complete — すべてのゾーンが署名済みになった状態の “Isolated” と “Sparse” の中間に該当すると考えられる.
5　特に root、TLD、ISP、政府組織、あるいは e-commerce を行っている企業のネームサーバ管理者を想定する.

3.2.2 鍵の配布に関する議論

不要論：

DNSSEC が有効となるためには、root zone 以下の対象となるすべての zone が DNSSEC 的に secure であることに加え、検証する側ではその最上位 zone（理想的には root、あるいはその他の trusted zone）の公開鍵を持つことが必要となる。

(i) 前者に関連し、「secure な zone」の現実的な範囲として想定するものが広すぎれば非現実的であり、狭すぎれば役に立たないものとなる。

(ii) 後者に関して、最上位 zone の鍵を配る方法として何を想定するか？ その方法は現実的であるか？（不要論（3.2.3）にも関連）

対応する必要論：

(i) TLD 直下のドメイン名が DNSSEC 的に委任されている状況を選択できることを当面の目標とする（この際の前提として、root zone から TLDs への secure な委任については、近い将来に実現されると考える）。この際、ゾーン以下すべてのドメイン名が DNSSEC 的に委任されることは必須でない。deployment 的観点からは、secure なドメイン名がクライアント側で検証できる状況がまず必要であり、3 階層の安全性を確保[6]できれば十分であるような場合も数多く存在する。

(ii) サーバ証明書+https 程度の信頼性で要求を満足するとみなす。具体的には下記のようなモデルとなる。

- 現行のサーバ証明書を用いた https での配布（これに加えて新聞などの公共媒体に signature を公開）
- IANA → RIR → LIR → user という経路による鍵の配布
- IANA → TLD → user（ドメイン名登録インターフェースは secure かつ信頼できるものとみなす）
- 一般的ユーザは、OS のデフォルト状態を信頼する
- 鍵更新については、

  Microsoft:　Windows Update
  Apple:　Software Update
  FreeBSD:　FreeBSD Update
  など

不要論からのコメント：

(i) 3 階層だけで十分であるかについては判断がつかない。

(ii) 最上位 zone 鍵の配布方法案については現実的である。

必要論からのコメント：

(i) 商用で利用されるドメイン名などは短いものが好まれる傾向があるため、3 階層でも一定の効果が見込めるとの考えだが、定量的根拠はない。この議論を続けるためにはリサーチが必要となる。いずれにせよ、root–TLD–TLD 直下ドメインまでの階層が DNSSEC 化されれば、あとはドメイン名運用者が DNSSEC 対応の範囲を決定すればよい。

3.2.3 クライアントに関する議論

不要論：

DNSSEC の validation を行うためには、スタブリゾルバが signature の検証をする必要がある。一方で、validation の処理はかなり複雑な上に負荷が高く、現状ではクライアント側で BIND 9 を導入する以外の方法はない。スタブリゾルバ側への deployment scenario としてはどのような方法を想定しているのか？ それは現実的なものか？

対応する必要論：

(i) の 2 階層の安全性が達成できている段階で、大手 ISP のキャッシュサーバが root 公開鍵を導入すれば、ISP 傘下ユーザはその TLD のドメインに対して、DNSSEC 的検証を行うことができる。ここで DNSSEC が認知され、有用と判断されるようであれば、各スタブリゾルバ実装について、DNSSEC への対応を行うモチベーションが与えられる。結局は OS ベンダが対応しないとスタブリゾルバへの普及は見込めないので、このような有効性の検証が必要だろう。ただし、より効果的な展開のためには、クライアントアプリケーションから簡単に利用できる上位互換性を持った検証ライブラリルーチンが過渡的に必要となるかもしれない。

不要論からのコメント：

この場合、スタブリゾルバと「大手 ISP のキャッ

6　root, TLD のゾーンが DNSSEC 的に secure な状態で、TLD 直下のある委任先ゾーンが DNSSEC 対応した場合を指す.

シュサーバ」の間はsecureであると仮定することになる。これは、実験サービスとして成り立つかもしれないが、現実的に意味があるかについては下記の理由により疑わしい。

a)「大手ISP」の配下にある公衆ネットワーク（例：インターネットカフェ） 内での攻撃は防げない
b)「大手ISP」までがsecureなら、「大手ISP」から先における攻撃は比較的困難なので、DNSSEC非対応でも実用上は問題とならない

必要論からのコメント：

a) については、DNSSEC対応したキャッシュサーバに加え、

(1) スタブリゾルバ−キャッシュサーバ間でTSIGを使う
(2) キャッシュサーバでなんらかのユーザ認証を行い通信路を暗号化する

のいずれかが解決策として考えつくところではあるが、決定的な案というものは現段階で存在しない。

3.2.4 NSECレコードに関する議論

この件に関してはDNS WGでは詳細に議論されていないものの、DNSSEC deploymentに少なからず影響を与えるものと考えられるため、著者の見解をまとめておく。

NSECレコード問題：

DNSSECでは、検索対象名を前後に挟むように連続している名前のゾーンにおける存在を証明することで間接的に対象名の不在を証明するが、この際に使用するNSECレコードを順に辿れば、ドメイン名などのゾーン内すべてのレコードが芋蔓式に入手できるという状況を指す。

NSECレコード問題は存在しないという論拠：

DNSのレコードは公開情報であるからゾーン内のドメイン名が検索できることに何らの問題も認められない。したがって、NSEC2、NSEC3などの検討は時間の浪費に過ぎない。

NSECレコード問題は存在するという論拠：

現実問題として、多くのTLDはNSECレコード問題を自らのゾーンをDNSSEC化する際の致命的問題と認識している。1つにはプライバシ的な問題があり、また1つにはDoS的な問題がある。NSECを導入すれば、検索者はゾーン内の1つのドメイン名を知っているだけで、すべてのドメイン名を知ることができる。これによりいわゆる名簿作りが可能となり、SPAMメールやダイレクト商法などの活動が容易となることは明白である。TLDゾーンにおけるドメイン名を、一般の企業とその顧客名というアナロジーで捉え直すと、ドメイン名リストは顧客名簿に類似する。個人情報保護法的観点で、個人情報ではない個人データが何らかのルールに基づき集合体となった際には個人情報とみなされるように、公開情報のドメイン名の集合体であるドメイン名リストは非公開情報とみなすことができる。社会インフラとなったDNSを上流で管理する責務を持つTLDとしては、これを公開情報とみなし、自ゾーンにおいてSPAM活動を助長する状況を容認することはできない。したがって、DNSSEC deployment促進のためには、ドメイン名リストを作成することが不可能であるようなNSECに代わるアルゴリズムを導入することが必須条件である。

3.3 今後の展開

DNSSECの要不要は議論のみでなく実証すべき段階に差し掛かっていると考えられる。ここにおいては、rootゾーンにおけるDNSSEC導入の動きに連動し、TLDにおいてもDNSSEC化を図り、システムパフォーマンス、その他DNSSEC導入障壁となり得る問題についての洗い出しとその評価方法を定めた後に、一定期間の検証を行う必要がある。この際の検証ポイントとして、上述のNSECレコードによるDNSレコードの芋蔓式取得に関する問題も留意すべきだろう。この実証段階において、各検証ポイントが実運用レベルの要件をクリアできるとすれば、その時点でDNSSECの本格導入を検討すべきと考えられる。