博士論文 平成 16 年度 (2004 年度)

大規模かつグローバルな インフラストラクチャの 運用基盤分析に関する研究

2005 年 2 月 16 日

慶應義塾大学大学院 政策・メディア研究科

関谷 勇司

Copyright ©2004, 2005 by Yuji Sekiya

Year 2004 DISSERTATION

A Research into Analysis of Operational Status on Large Scale and Global Infrastructure

Yuji Sekiya

Graduate School of Media and Governance
Keio University
5322 Endo Fujisawa 252-8520 JAPAN

Submitted in partial fulfillment of the requirements for the degree of Doctor of Media and Governance

Copyright ©2004, 2005 by Yuji Sekiya

概要

本研究では，大規模かつグローバルなインフラストラクチャの運用基盤分析に関する手法を提案し，実践することによって検証した．本研究が定める大規模かつグローバルなインフラストラクチャとは，世界規模のサービス範囲を有し，サービス拠点が自律分散，協調してひとつのサービス空間を構成するようなインフラである．

現在では，インターネットに代表されるような大規模かつグローバルなインフラストラクチャが身の回りに増加している．例として，位置情報システムの GPS や世界規模の携帯システムである GSM，インターネット上で展開される Web や DNS といったサービスをあげることができる．

このような自律分散型の情報通信インフラストラクチャは，急速に社会基盤として確立し，その信頼性や健全な運用性への極めて厳しい要求が生まれている．一方、このようなインフラストラクチャは，大規模に分散し，かつ，自律的な機能が相互に作用しながら運用されているために，その全体の運用状態を把握することは困難であった．現状では，システムとしての障害が発生すると，その発生の事象を元に障害を修復するという手法でシステムは復旧される．このようなリアクティブな障害復旧のプロセスは，比較的変化の小さい静的な運用環境では，障害箇所の追求を確定的に実現できるため効率的である．しかし，上記の例のような大規模な自律分散システムは動的に運用規模や運用状態が変化することが多いために，障害箇所の追求の時間も増加する．

本研究では，大規模かつグローバルなインフラストラクチャの事例としてとして DNS を取りあげ，DNS の運用基盤分析を行うためのモデル定義と手法の確立を行った．さらに，本研究の手法を適用することにより，DNS の運用状態を分析することが可能であることを検証した．その結果，大規模かつグローバルなインフラストラクチャの運用基盤分析を行う際に有用である分析モデルを確立することができた．

本研究では，まず大規模かつグローバルなインフラストラクチャというものを定義した．その中において，インフラストラクチャに求められる必要条件を明確にした． その要件とは，(1) サービスの公共性，(2) サービスの信頼性・耐障害性，(3) サービス拠点の識別性である．

次に，これらの必要条件をもとに，運用基盤分析を行うモデルを定義した．このモデルは，(1) ユーザの視点からのサービス分析，(2) サービスの完全性分析，(3) サービス識別情報の分析の 3 点から構成される．

この分析モデルに基づいて，DNS の運用基盤分析モデルを作成した．その手法とは，(1)DNS 到達性の調査，(2)DNS 委譲ツリーの調査，(3)DNS サーバ識別情報の調査である．(1) の調査においては，世界各地からの DNS サーバの応答時間ならびに応答が得られる割合に関する調査を行った．(2) の調査では，DNS の委譲ツリーの完全性を調査することによって，DNS の正確性を調査した．(3) の調査では，DNS の個体識別情報ならびに運用情報を調査することによって，平常時における DNS サーバ毎のサービス範囲を明らかにした．

これら調査の結果，DNS の運用基盤分析において個々の DNS サーバを単体を計測するのではなく，DNS 全体をひとつのシステムとして分析することができた．これによって，本研究にて提案したモデルの有用性を確認することができた．

システムの障害をあらかじめ予測するためには，稼働中のインフラストラクチャ全体を恒常的に把握する手法が必要である．本研究では，大規模かつグローバルなインフラストラクチャの恒常的ならびに定量的な分析手法を提案し，その手法を用いて実際のシステムの分析を行い，評価した．これにより，本論文の定義に当てはまる大規模かつグローバルなインフラストラクチャであれば，運用基盤の分析を行うことが可能となった．その結果，大規模かつグローバルなインフラストラクチャのプロアクティブな信頼性と安全性を確立することに貢献できるようになった．

以上の結果により，本研究において大規模かつグローバルなインフラストラクチャの運用基盤分析に有用なモデルを確立することができた．

Abstract

In this dissertation, a method to analyze the operation of a large scale and global infrastructure is proposed and evaluated. The DNS is taken as an example of such infrastructure, and the model together with its method in analyzing the operation of DNS is defined. Furthermore, using the method, its application to analyze the operation status of DNS is confirmed. As a result, this dissertation achieved to propose a generic model to analyze the operation of a large scale and global infrastructure.

Large scale and global infrastructures, as represented by the Internet, have been increasing nowadays. Examples of such infrastructures are Global Positioning System(GPS), Global System for Mobile Communication(GSM), World Wide Web, and Domain Name System(DNS).

Such information-communication infrastructures are usually distributed autonomous systems, and they instantly became one of the necessities in our society. Therefore, strict requirements and demands on reliability and stability need to be fulfilled by each infrastructures. However, it is difficult to know the overall status due to the fact that they are operated autonomously. As a result, current troubleshooting takes reactive approach from the type of system failure. This approach is effective in the operation environment where change in the system operation occurs rarely. On the other hand, large-scale, global infrastructure mentioned earlier takes frequent change in scale and status, resulting in requiring longer period of time to discover the cause of failure.

In this research, DNS is used as one of the examples of large-scale, global infrastructure. Definition of model and establishment of techniques for analyzing operation platform of DNS is introduced. In addition, effectiveness of the techniques introduced in analyzing operation status of DNS is validated. Effective analysis model for analyzing large-scale, global infrastructure's operation platform is established as a result of the research.

The dissertation first describes the definition of a large scale and global infrastructure, and clarifies the requirements towards those infrastructures. The requirements are defines as; (1) fair and public service, (2) service reliability, and (3) identification of the service point.

The dissertation second describes the model to analyze the operation of a large scale and global infrastructure. The model is constructed from the following elements; (1) analysis of the service from user's point of view, (2) analysis of the service integrity, and (3) analysis of the information in service operation.

Analysis model for DNS operation platform is established based on the general analysis model for large-scale, global infrastructure introduced. The technique consists of following methods: (1) examination of DNS reachability (2) examination of DNS delegation tree (3) examination of identity of DNS servers For (1), time required for receiving

reply is measured from different places in the world and the rate of receiving reply is also examined. In (2), integrity of DNS delegation tree is examined for DNS accuracy. In (3), service range for each DNS server is indicated by examining identification and operation information of DNS.

It is required to be aware of overall infrastructure regularly, in order to predict upcoming system failure. This research proposes analysis techniques for quantitative and constant examination of large-scale, global infrastructure and evaluation is taken by analyzing the systems in operation using the introduced technique. It is proved to be effective in analyzing the operation platform of large-scale, global infrastructure defined in the dissertation. As a result, the contribution can be made to proactively avoid system failure and provide system reliability and security using the proposed analysis model.

The dissertation as a result achieved to provide a method to efficiently analyze the operation of a large scale and global infrastructure.

目 次

図 目 次

表 目 次

1章　はじめに

本章では，本研究の前提となる背景を説明し，本研究の目的，本論文の構成について述べる．

1.1 節　本研究の背景

従来のインフラストラクチャは，電気や水道といった，生活に密着して地域毎に提供されるインフラストラクチャであった．しかし，現在の世界には，旧来の枠に収まらない，大規模でグローバルなインフラストラクチャが数多く存在している．

その例としてあげられるのが，世界規模の携帯電話システムである Global System for Mobile Communications(GSM) システム [1] や，衛星を利用した Global Positioning System (GPS)[2], またインターネットである．これらのシステムは，地球規模のサービスを提供しているインフラストラクチャである．

インフラストラクチャ[1]は，提供されるサービスの性質によって，いくつかの種類に分類される．まず，水道やガス，電気といった，いわゆる人間が生活していくのに必要不可欠な生活インフラがある．さらに，道路や鉄道といった，流通にとって不可欠なインフラも存在する．さらに，電話やインターネットといった，通信のためのインフラも，現代の生活には欠かせない重要なインフラである．つまり，インフラとは人々の生活に必要なものであり，人間が人間として生活していくうえで必要となるサービスである．

一方，大規模かつグローバルなインフラとは，ある地域のみに範囲を限定せずに，世界規模において，普遍的にサービスを提供しているインフラである．サービスの提供範囲が広がれば，必然的にサービスの利用者も増大する．

このようなインフラでは，従来のような地域毎に提供されるようなサービスと異なり，広い範囲にて数多くのユーザに対してサービスが提供される．そのため，サービスを提供するための設備が分散し，複数存在している場合が多い．

たとえば GPS の場合には，個々の衛星は単独で動作しており，GPS 信号の受信者側が複数の衛星から受信した信号をまとめることによって，位置の特定を行っている．また，インターネットの場合には，基本的にホスト同士が 1 対 1 の通信を行っている．この際，通信経路の中間に位置する中継ルータは，その通信の内容に対して関知していない．つまり，個々のホストや中継ルータが自律的に，かつ協調して動作することで通信が行われている．

一方，サービス拠点が分散することによって，サービスの運用状態を監視，分析することが難しくなる．これは監視する範囲が大きくなり，なおかつ分散された複数のサービス拠点を監視し

[1]以下インフラと略す

なければならないからである．この際，障害の検知だけではなく，ある拠点にのみ負荷が集中することなく，適切にサービスが提供されているかを監視ならびに分析することも必要である．

1.2 節　本研究の目的

私たちの身の回りには，従来のインフラに加えて，大規模でグローバルなインフラが出現している．そしてそのサービス範囲も，従来のインフラに比べ，大規模なものとなっている．

大規模でグローバルなインフラが整備されることにより，私たちの生活はより便利になり，それらは生活に欠かせないものとなっている．しかしその一方で，従来のインフラには無かった問題点も発生する．それは，サービス拠点が複数に分散することによる管理コストの増大や，負荷の効率的な分散といった問題である．

例えば，インターネットの場合には，サービスが無数に分散して存在しているため，それらすべてを一元的に監視することは不可能である． さらに，地球規模のインフラであるがゆえに，障害が発生した場合，何が原因でどこに障害が発生しているのか，特定するのが困難となる場合もある． また，あるサービスの定常的な運用状態を普段から把握していなければ，サービスに異常が発生していることを検知するのさえ困難な場合もある． そのため，大規模かつグローバルなインフラを効率的に監視，分析するための手法が必要となる．

そこで本研究では，大規模かつグローバルなインフラの運用基盤を分析するための手法を提案し，実践を経て手法を確立することを目的とした．

具体的には，インターネットにおける最も重要なサービスのひとつであり，大規模かつグローバルなインフラである DNS をとりあげ，本研究にて提案する手法に従い，DNS の運用状態を効率的に監視するためのシステムを作成した．このシステムを利用して DNS 運用状態を分析できることを実証し，本手法が大規模かつグローバルなインフラの運用基盤分析に有効であることを実証することを目的とした．

1.3 節　本論文の構成

本論文では，まず 1 章において，本研究の背景と目的について述べる．

次に，2 章にて，大規模かつグローバルなインフラに関して論じる．この章にて，大規模かつグローバルなインフラの必要条件と，その運用基盤を分析するためのモデルに関する定義を行う．

そして，3 章にて本研究の課題事例とする DNS をとりあげ，その運用基盤に関して述べる．さらに，本研究にて提唱した手法をもとに，DNS の運用基盤分析を行うモデルを定義する．

また，4 章において，本研究の基礎となった研究や関連研究について述べる．

その後，5 章において，DNS の運用基盤分析を行うシステム構築について述べ，6 章において，システムによる分析結果を示す．

最後に，7 章にて結論をまとめる．

2章　インフラの運用基盤分析

本章では，本研究の主題となる大規模かつグローバルなインフラについて述べる．また，インフラに求められる要件をもとに，インフラの運用基盤分析に必要な要件について定義する．この定義をもとに，本研究にて提唱する分析手法を述べる．

2.1 節　インフラとは

インフラはインフラストラクチャー（infrastructure）の略で，語源は「下部構造」という意味であり，これが発展して「生活や経済活動の基盤として整備される施設」を意味するようになった．

1 章にて述べた通り，従来のインフラとは違った，大規模かつグローバルなインフラが出現している．大規模かつグローバルなインフラと従来のインフラとの最大の違いは，そのサービス提供範囲と提供形態にある． 大規模かつグローバルなインフラの場合，サービス提供範囲はある範囲に限定されず，世界規模にて提供される．また，世界規模にて提供するために，サービス提供拠点は分散して存在しており，それらが協調して機能することによって，世界規模のサービス形態となる．

しかし，サービスの範囲や規模が違うにせよ，インフラとして機能するために求められる必要条件は変わらない． なぜならば，インフラとは，その規模に関わらず，人間が文化的に生活するにあたって必要不可欠なものであるという性格を持っているからである．従来のインフラがインフラとして機能するための必要条件は，大規模かつグローバルなインフラにとっても，満たすべき最低限の必要条件となり得る．

生活に必要なインフラとしては，上下水道，電気，ガス，電話といったものがあげられる．また，経済活動に必要なインフラとしては，道路，鉄道，航空機といったものもインフラとしてとらえられる．さらに，広い意味のインフラとしては，学校や公園といったものも社会インフラとして考えられる．いずれのインフラの場合も,「社会で共有する」という性格を持っている．

また，生活インフラとして提供されている，上下水道や電気，ガス，電話といったものは，安定して定常的にサービスが提供されることが期待されている．これは，これらインフラのサービス提供が断たれた場合，生活にとって不便なばかりでなく，人間の生命に関わる問題が発生するからである．例えば，地震等の災害によって生活インフラの提供が断たれた場合，地震そのものの被害に加えて，生活インフラの欠如による，二次的な人的被害が発生する場合もある．よって，

インフラは定常的に安定して供給されることが望まれているものである．

さらに，万が一サービスに障害が発生した場合には，早期に復旧することが望まれる．サービス復旧を行う際，まず，どこの地点において障害が発生しているのかを調査することが復旧への第一歩である．従来のインフラの場合には，サービスを受けられていないユーザから，サービス拠点までのどの位置にて障害が発生しているのかを調査する．しかし，大規模かつグローバルなインフラの場合，サービス拠点が複数存在するため，まずサービス拠点を特定することが必要となる．つまり，サービスを受けられていないユーザが，通常どのサービス拠点に属していたのかを識別する必要がある．これによって，他のサービス拠点に振り替える等の対策が可能となる．

以上の議論をふまえ，本研究では，インフラがインフラとして機能するための必要条件を，次の 3 つと定義する．

(1) サービスの公共性

インフラによるサービスが，サービスを受ける権利のある人に対して，適切にかつ公平に提供されることである．つまり，特定の人にのみ提供されるサービスではなく，公共的な性格をもって提供されることが必要となる．

(2) サービスの信頼性・耐障害性

サービスが恒常的に安定して提供され，万が一インフラの一部に障害が発生した場合にも，サービスに与える影響を最小限に抑えるよう設計されていることが必要となる．

(3) サービス拠点の識別性

従来のインフラの場合には，サービスの保守性とも言える．障害箇所をより早く発見するために，平常時から，サービス拠点によるサービスの範囲を把握できる仕組みが必要とされる．

例として，電気やガスといった生活インフラストラクチャに対して，上記の必要条件をあてはめてみる．(1) の条件は，対価を支払った，権利のある人に対して公平に提供されているため，満たされている．(2) の条件は，24 時間体制でサービス運用監視が行われ，障害時にもすばやく復旧作業が行われるため，満たされている．(3) の条件は，サービス提供者がサービスする範囲があらかじめ定義されており，障害箇所を特定することができるので，満たされている．

2.2 節　インフラとしての GPS

本節では，大規模かつグローバルなインフラである，GPS の運用に関して述べる．

GPS は，人工衛星を利用する汎地球的な測位システムである．GPS は，従来のロラン C[3] に代わる航法システムとして米国国防総省を中心に 1970 年代初頭より開発され，1993 年米国運輸省に正式運用開始宣言を通達したことにより，民間への利用が正式に開始された．この時点で，インフラとして成立したと言える．

GPS は，宇宙空間に配置される数個の人工衛星および衛星を制御する地上局，測位を行う GPS 受信機から構成されている．

GPS 受信機は，衛星からの信号を取得し測位を行う．その信号は軌道データ，時刻補正データ，全衛星の配置データである．これらのデータは，衛星が大気の影響で減速するなどの関係上，あらかじめ算出することができない．従って，地上からの観測によって作成する必要がある．

軌道データなどを作成し，GPS 衛星の運用や制御，監視を行う地上の設備は米国国防総省によって運用されている．文献 [4] にある通り，次に述べる設備が GPS 運用基盤となっている．

- モニター局

  Ascension Island, Cape Canaveral, Colorado Springs, Diego Garcia, Kwajalein, Hawaii の計 6ヶ所に存在する．各衛星からの電波を受信し，衛星軌道の変化や時計の変化について監視する．

- 主制御局

  Colorado Springs に存在する．モニター局からの情報を集め，GPS 全体のサービス監視と調整を行う．

- アップロード局

  Ascension Island, Cape Canaveral, Diego Garcia, Kwajalein の 4ヶ所に位置する．主制御局からの修正情報を各衛星に送信する設備である．

- 時刻修正情報局

  米国海軍天文台の約 50 台のセシウム時計ならびに 12 台の水素メーザー原子時計を利用し，時計情報の修正を行う．

GPS を運用するために，以上の監視設備の運用コストに加え，代替衛星の打ち上げなどを含む GPS 関連予算として，およそ年間 3 億 4700 万ドルが費やされている．

GPS を，前節にて述べたインフラの必要条件に照らし合わせてみると，まずサービスの公共性は，地球規模で確保されている．GPS 衛星からの電波を受信できる位置にいる人であれば，誰でもサービスを利用できる．また，サービスの信頼性・耐障害性という点においては，GPS の地球上を周回する軌道パターンは 6 種類あり，それぞれ 1 軌道毎に 4 台の衛星が飛んでいる．つまり，24 台の衛星にてサービスが提供されており，衛星に障害が発生した場合の予備衛星も常に数基打ち上げられている．さらに，サービス拠点の識別性に関しては，GPS モニター局にて各衛星の場所を把握しており，各衛星の電波が届く範囲を推定できるようになっている．

2.3 節　インフラとしてのインターネット

インターネットは現代において，最も重要な，大規模かつグローバルなインフラのひとつである．生活や経済活動に必要な情報の多くが，インターネットを通じて提供されている．

個人レベルにおいても，普段当たり前のように使っている電子メールや Web ブラウザ，Voice over IP(VoIP) といったものは，コミュニケーションの道具として，また情報収集の道具として，生活に必要なものとなっている．

また，企業内においても，遠隔地からの社内ネットワークへの通信，グループ内でのファイル共有，業務書類のやりとりや内線電話通信といったものが，どんどんインターネットの上にて実現されつつある．

さらに，インターネットをデータ転送のための基盤として利用し，その上にて様々なインフラスを構築しようとしている事例も存在する．例えば，現在の交通システムである Intelligent Transport Systems (ITS)[5][6] をインターネット上にて構築しようとする研究 [7] も行われている．

インターネットは，ひとつの回線を複数人の通信で共有することができる仕組みであり，回線と回線を結ぶ中継点 (ルータ) も，共有して利用されている．インターネットの通信モデルは，端点と端点による通信 (End-to-End モデル) である．これは，端点に位置するホスト同士が，お互いに通信相手を指定し合い通信するモデルである．ホストとホストの通信路の中間に位置するルータは，データの中身を関知せず，通信の宛先だけを見てデータ転送を行う．通信の中間に存在する複数のルータがバケツリレー方式でデータを転送することによって，ホストからホストへの通信が成立する．

また，ルータはルータ同士で回線の接続状況に関する情報をやりとりし，最適な通信経路を選択している．

このように，インターネットは，自律的にデータ転送を行うホストやルータが，結合して協調して動作することによって成立している．

そこで，インターネットが，3.1 にて定義したインフラの必要条件を満たしているか考察する．

まず，サービスの公共性に関しては，インターネットに接続する権利や機会を有する人にとっては，接続することによって同一のサービスを受けることができる．これは，インターネットがひとつのネットワークとして形成されており，かつ End-to-End な通信モデルであるため，基本的に接続する地域等に左右されることなく，同一のサービスを受けることが可能である．

また，サービスの信頼性・耐障害性に関しては，ある回線やルータに障害が発生した場合にも，ルータ同士が情報交換し合うことによって，代替となる経路を発見することができる．

しかし，サービス拠点の識別性に関しては，従来のインフラよりも難しいと言える．これは，インターネットが自律分散協調的に動作するものであり，従来のインフラのように，あらかじめサービス範囲が特定されていたり，GPS のようにどこか一点にて運用基盤の監視を行っているものではないからである．障害が発生した場合には，ルータ同士が交換する経路情報を調査したり，通信経路をたどっていくツールを利用して，障害箇所を特定することとなる．

最後の条件をより完璧に満たすことが，インターネットがよりインフラとしての地位を確立するためには必要である．

2.4 節　インフラとしての DNS

本節では，インターネットにおいて DNS が果たす役割について述べる． また，DNS をインフラとして見た場合に，DNS がインフラの必要条件を満たすかどうか考察を行う．

2.4.1 節　DNS の果たす役割

インターネットは，社会生活における情報インフラとして，日常社会に浸透している．これはインターネット白書 [8] 等の，利用者数や利用用途のデータを見ても明らかである．インターネットが情報インフラとしての重要性を増すことにより，インターネットを支える基盤技術である DNS も，重要な役割を担うこととなる．

また，3.1 節にて述べた通り，DNS は集中データベースではなく，名前空間を委譲することによって成立している分散データベースである．インターネット上に無数に存在する DNS サーバ

が連携しあうことによって，ひとつのデータベースを形成している．すなわち，DNS は世界規模に展開されている，グローバルなデータベースである．

さらに，3.1 節にて述べた通り，メールやウェブといった，インターネットにて一般的に利用されているサービスは，DNS が正常に動作しなければ，通信ノードを特定することができず，通信を行うことができない．すなわち，DNS はインターネット上における通信をを支えるインフラとなっている． インターネット上における多くのサービスを正常に運用するためには，DNS を正常に運用することが必要不可欠である． つまり，DNS は，インターネットという通信インフラを成立させている，インフラのためのインフラであると言える．

2.4.2 節　次世代インターネットと DNS

現在のインターネットの大部分は，Internet Protocol Version 4(IPv4)[9] と呼ばれるプロトコルによって運用されている．しかし，IPv4 は 20 年ほど前に制定された規格であり，インターネットの普及と拡大に伴って，様々な問題が発生している．そこで，IPv4 に続く次世代のインターネットプロトコルとして，Internet Protocol Version 6(IPv6)[10] が制定され，利用され始めている．

IPv4 にて発生している問題点とは，主に規模性の問題である．インターネットがインフラストラクチャとして利用され始めると，様々なものがインターネットにつながり，通信を行う．この際，通信を行うためには識別子となる IP アドレスが必要となり，IP アドレスが枯渇し始めた．そのため，IPv4 においては 32bit であった IP アドレス空間を，IPv6 では 128bit に拡張することによって解決した．

IPv6 では，IP アドレス空間が増大したことによって，識別子である IP アドレスの表現方法が 10 進数から 16 進数に変更された．これによって，IP アドレスによって通信相手を直接指定することは困難となり，通信相手指定のために，DNS による名前解決を利用することが必然となる．さらに，IPv6 では，従来の計算機のみならず，様々な機器がネットワークに接続されることが想定されている．そのため，インターネットに接続され，IPv6 アドレスを持つ通信ノードの数が，現在の IPv4 に比べて飛躍的に増加する．それにともない，DNS に登録されるホスト名も増加する．すなわち，IPv6 アドレスをもつノードが増加し，DNS に登録されるホストが増加すると，DNS への名前解決問い合わせも増加する．

また，IPv6 のアドレス割り当ては，可能な限りネットワークの構成に従った割り当てとする [11][12] ことが決められた．この割り当てによって，経路情報の削減と自律分散的なアドレス割

り当てが可能となった．このアドレス割り当てモデルをもとに，アドレス使用者の情報管理システムである whois[13] を階層的に構成する研究を行った．IPv6 の場合，すべてのアドレス割り当てがインターネットレジストリによって行われるわけではないため，あるアドレス空間の利用者を調べる際に，レジストリへの問い合わせのみではわからない場合が多い．そのため，アドレス利用者情報も階層的に管理するモデルを提案した．これは論文 [14] にて述べられている． この結果，何か障害が発生した場合，アドレス使用者情報をより詳細に得ることが可能となるシステムを提案できた．この際設計したシステムが，本研究にて行った DNS の運用基盤分析に生かされている．この研究の詳細に関しては，4.1 にて述べる．

次世代インターネットの発達によって，インターネットがよりインフラとして活用されるようになると，DNS への問い合わせが増大する．それと同時に，インフラに求められる要件である，耐障害性やサービスの公平性といった要素が，インターネットをささえる基盤技術である DNS にも求められるようになる．

DNS をインフラスとしてとらえた場合，DNS はどのユーザからの名前解決要求に対しても，公平に応答を返す．インターネットに接続しているユーザであれば，だれでも名前解決のサービスを受けることができる． したがって，サービスの公共性は満たされている．

また，同一の名前空間を保持する DNS サーバを複数台用意しておくことによって，耐障害性を確保している．

さらに，サービス拠点の識別性に関しては，再帰 DNS サーバの場合，ユーザが自分で利用する DNS サーバを指定するため，利用している DNS サーバの識別が可能である．権威 DNS サーバの場合は，ユーザが直接問い合わせを行う機会は少なく，通常再帰 DNS サーバ経由で問い合わせを行う．従って，基本的にユーザがサービス拠点を意識することはない．

同一の名前空間を持つ権威 DNS サーバが複数台あり，その中の 1 台に障害が発生した場合には，DNS サーバを特定する必要がある．名前空間の委譲先として複数の DNS サーバが指定されている場合には，それら DNS サーバに対して 1 台ずつ調査を行うことで，障害地点を発見できる．一方，3.2 節にて述べるエニーキャストを利用している場合には，5.3 節にて述べる手法にて DNS サーバを識別できる．

よって，DNS は 3 つの条件を満たしていると言える．

2.5 節　本研究の分析モデル

本節では，本研究が対象とする，大規模かつグローバルなインフラを定義し，その運用基盤分析を行うためのモデルを定義する．

2.5.1 節　大規模かつグローバルなインフラ

1.1 節にて述べた通り，大規模かつグローバルなインフラとは，従来のインフラのサービス範囲を超えたインフラである．従来のインフラのように，サービスの範囲がある地域のみに限定されているのではなく，全世界規模において，複数のサービス拠点から普遍的にサービスされているインフラを意味する．

具体的に本研究にて想定する大規模かつグローバルなインフラとは，次の特徴を満たす，各種情報通信を支える土台となるようなインフラを意味する．

(1) 世界規模のサービス範囲

サービス範囲がある特定の地域に限定されず，全世界規模にて提供される．

(2) サービスの均質性

同一のサービス範囲内であれば，どのサービス拠点においても，同じサービスが提供される．

(3) サービス拠点の自律分散性

サービス拠点が中央集権的に存在するのではなく，分散して存在している．また，サービス拠点が互いに協調し，役割を分担することによって，一つのサービスを形成している．

特に，本研究にて提案する運用基盤分析モデルは，(3) の特徴を強く持つ大規模かつグローバルなインフラの運用基盤分析に対して有効である．それは，インターネットに代表されるような情報通信インフラであり，大規模なインフラを形成する場合に欠かせない要素となる．サービスの規模が大きくなればなるほど，一極集中による管理が困難となり，また耐障害性の面からも，一極集中型の管理は好ましくない．そこで，サービス拠点や管理範囲を分割し，分散することで規模性や耐障害性を確保することができる．このように構築されるインフラは増加しており，GSM 携帯電話や，インターネット上にて展開されるサービスの多くはこの特徴を有している． つまり，サービス拠点が有機的に，自律分散して結合し，単一のサービス空間を形成しているような情報通信インフラが，本研究が対象とする大規模かつグローバルなインフラである．

2.5.2 節　運用基盤分析モデル

これまで述べた通り，GPS やインターネット，DNS はインフラとしての必要条件を満たしている．すなわち，2.1 節にて定義した必要条件が満たされていれば，インフラとして機能している．

つまり，インフラがインフラとして機能しているかどうか分析するということは，これら必要条件が満たされているかどうか分析することである．それも一時的に条件が満たされているのではなく，定常的に満たされている必要がある．

したがって，これらの必要条件がどれほど満たされているか，定量的に分析する手法があれば，インフラの運用基盤を分析することができる．

そこで本研究では，インフラの必要条件を満たしているか調査するために，図 2.1 のモデルを提案する．このモデルは，2.1 節にて述べたインフラとしての必要条件を分析するために必要な調査をモデル化したものである．

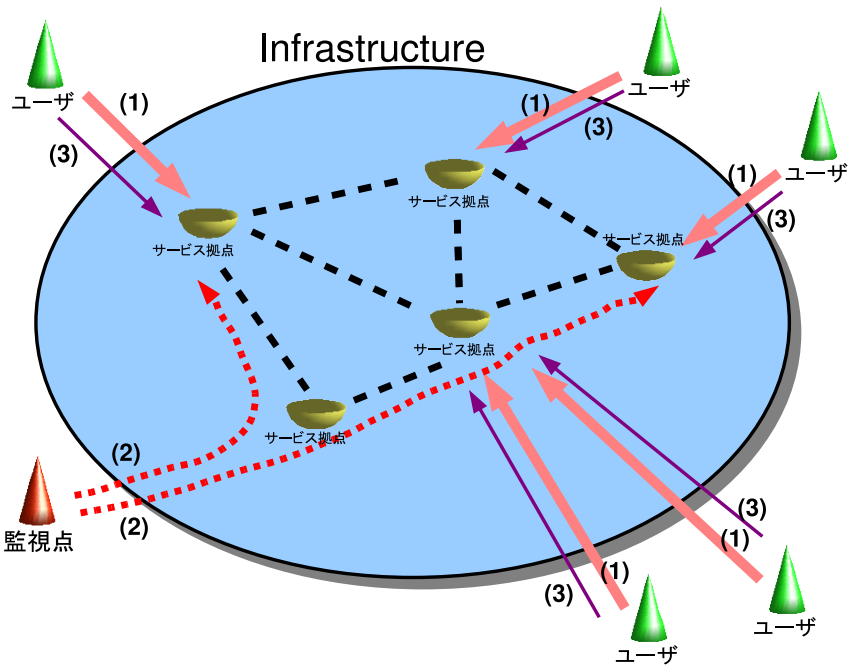


図 2.1: 本研究にて提案する分析モデル

図について説明を加える．

(1) ユーザの視点からのサービス調査

サービスの公平性を確認をするためには，ユーザの視点からのサービス調査が必要である．どのユーザも均一なサービスを受けることができているか，複数地点ら，通常のユーザと同じようにサービスを利用することによってサービスの公平性を確認する．対象が大規模かつグローバルなインフラの場合，複数のサービス複数にまたがる程度に複数のユーザからの計測が必要である．

(2) サービスの完全性調査

サービスの信頼性・耐障害性を確保するためには，インフラ全体の完全性が保たれている必要がある．そのため，定期的にサービスの完全性を調査する必要がある．大規模かつグローバルなインフラの場合には，サービス拠点同士が連携し合って，全体としてグローバルなサービスを提供できているかどうか調査する．

(3) サービス識別情報の調査

インフラに障害等が発生した場合，どの地点において発生しているのか迅速に検知する必要がある．サービス拠点が単一なインフラの場合は，ユーザからサービス拠点までの経路をたどることで障害箇所が発見できる．しかし，大規模かつグローバルなインフラの場合は，どのサービス拠点で障害が発生しているかを検知する必要がある．このため，平常時におけるそれぞれのサービス拠点のサービス範囲を把握しておく必要がある．

以上の 3 つの観点から調査，分析を行うことが，本研究にて提唱する運用基盤分析モデルである．

2.6 節　　本研究のアプローチ

本節では，前節にて定義した運用分析モデルをもとに，本研究の進め方について述べる．

本研究では，まず 2.1 節にてインフラについて論じた． そして，インフラがインフラとして成立するための必要条件を定義し，従来のインフラがそれを満たしていることを述べた．さらに，インフラの必要条件をふまえて，2.5 節にて大規模かつグローバルなインフラの運用分析を行うためのモデル作成行った．

これからの本研究のアプローチとして，まず，このモデルが運用基盤分析に有用であることを実証するため，大規模かつグローバルなインフラである DNS を事例課題として，運用基盤分析モデルを作成する．

次に，このモデルに従って分析システムを構築し，実際に分析を行う．分析結果をもとに，DNS の運用基盤が定量的に分析できているかを考察する．これによって，本研究にて提案した運用基盤分析モデルの有用性を確認し，モデルを確立する．

3章　DNS の運用と運用基盤分析

本章では，大規模かつグローバルなインフラの事例として DNS を取り上げる．DNS を課題事例として，現在の DNS の運用状況とその問題点について述べる．さらに，本研究が提唱する分析手法に基づき，DNS の運用基盤分析を行う場合の分析モデルについて述べる．

3.1 節　DNS とは

本節では，本研究の分析手法の対象となる Domain Name System (DNS)[15] について述べる．

インターネットは，自律分散協調的に動作するシステムとして成立している．データはパケットと呼ばれる小さな単位に分割され，宛先に配送される．その際，通信先までの専用の回線が用意されるわけではなく，共用されている回線をいくつも経由して，宛先までパケットが配送される．この配送路の中間にあり，パケットの宛先に応じて適切にパケットを配送するのが，ルータと呼ばれる装置である． このルータも，インターネット内の全てのルータが中央集権的に管理されているわけではなく，個々の管理母体毎に分散して管理されているものが，協調して動作することによって，インターネットという大規模でグローバルなネットワークを構築している．

インターネットにおいては，通信相手は IP アドレスと呼ばれる識別子にて識別される．しかし，IP アドレスは 10 進数もしくは 16 進数の数字の羅列であり，利用者にとって利用しやすい識別子ではない．そこで，現在のインターネットにおいては，個々のホストもしくはサービスに対して，人間が識別しやすい名前を割り当て，これを通信時に IP アドレスに変換する仕組みが提供されている． この仕組みを DNS と呼ぶ．インターネットにおけるサービスのほとんどは，この DNS を用いて通信相手を特定している．

通常のインターネットの通信においては，通信開始時に DNS を利用して名前を IP アドレスに変換し，通信相手を特定する．また逆に，あるホストから通信要求が来た場合，その IP アドレスを名前に変換し，通信相手を認識する．これらの変換を名前解決と呼ぶ．この名前解決を利用することによって，ユーザは各ホストの識別子を意識することなく，通信を行うことができる．

DNS は，大規模分散データベースとして設計され，構築されている．数多くのサーバが名前空間を分割して保持することによって，ひとつの名前空間データベースを分散的に形成している．つまり，1 台のサーバにて情報を集中管理する一極集中型のデータベースではなく，複数の DNS サーバが協調して動作することによって，ひとつのデータベースが形成されている．

DNS の名前空間は，一方向の木構造のデータベースになっており，サーバからサーバへ名前空

間の一部を委譲する [16] ことによって形成されている．データ空間は，ゾーンという単位によって管理されており，図 3.1 に示すとおり，ゾーンはその一部を他のゾーンとして分割し，管理権限を分割することができる．これを「ゾーンの委譲」と言う．ゾーンの委譲を繰り返すことにより，木構造のデータ空間が構成され，分散データベースが形成される．

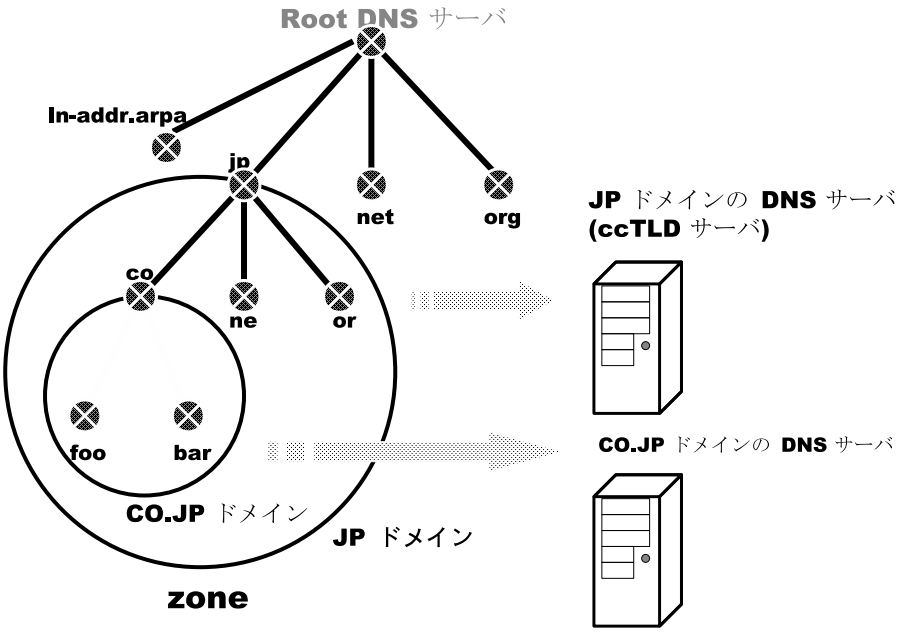


図 3.1: DNS のデータ空間

この際，同一の名前空間を担当する DNS サーバを複数台指定することが可能である．そのため，あるゾーンを保持する DNS サーバに何らかの障害が発生したとしても，同じゾーンを持つ DNS サーバを複数台設置しておくことによって，名前解決サービスに致命的な影響を与えることなく，運用が可能となる．

図 3.2 に，www.u-tokyo.ac.jp の名前解決を行う場合の動作例を示す． ユーザが DNS を利用する際には，ユーザからの問い合わせを受け付けた DNS サーバが，代理となって名前解決を行う．そして，その結果のみをユーザに伝えるという仕組みになっている．この際ユーザが利用する，図中の「身近な DNS サーバ」のことを，再帰 DNS サーバ (Recursive DNS Server) と呼ぶ．また，実際に名前空間を保持し，からの問い合わせに答える DNS サーバを権威 DNS サーバ (Authoritative DNS Server) と呼ぶ [15].

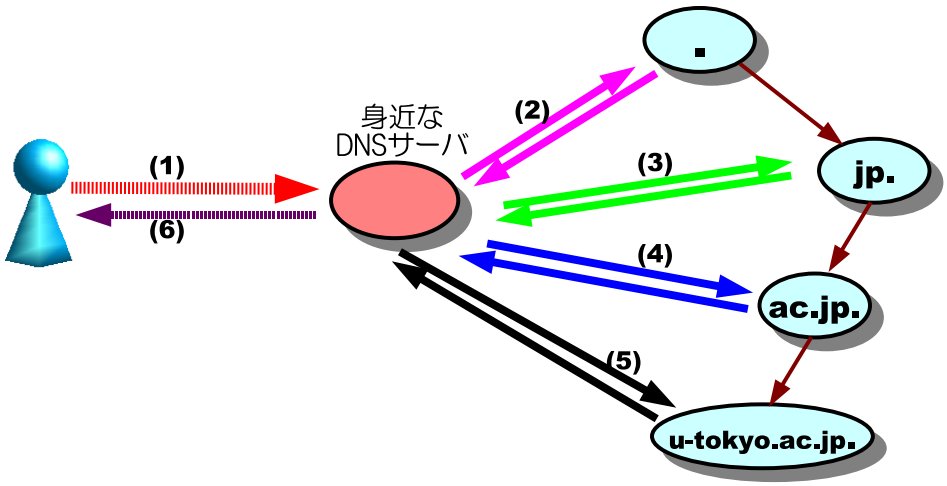


図 3.2: DNS の動作

3.1.1 節　TLD サーバ

DNS の名前空間において，DNS 木構造データベースの頂点となるゾーンである,「.」ゾーンを保持している DNS サーバを，ルート DNS サーバと呼ぶ．このルート DNS サーバは，名前解決においてデータを検索する際の起点となっている．

また，.com や .net といった，世界中において広く利用される名前空間を，Generic Top Level Domain (gTLD) と呼び，この名前空間を保持する DNS サーバを，gTLD サーバと呼ぶ．また，jp や kr，uk といった国や地域に属する名前空間を Country Code Top Level Domain (ccTLD) と呼び，この名前空間を保持する DNS サーバを，ccTLD サーバと呼ぶ．

これらのルート DNS サーバや gTLD DNS サーバ，ccTLD DNS サーバといった，Top Level Domain (TLD) ゾーンを持つ DNS サーバは，DNS の中においても非常に重要な役割を果たしている．特に，ルート DNS サーバは，すべての名前解決の起点となるサーバであるため，ルート DNS サーバに障害が発生すれば，DNS 全体のサービスが停止してしまう可能性がある．そこで，耐障害性と冗長性を確保するために，ルート DNS サーバは現在，世界中にて複数台運用されている．表 3.1 に，2004 年 8 月現在の，ルート DNS サーバの運用地点を示す．

表 3.1: ルート DNS サーバの運用地点

Root DNS Server	location
a.root-servers.net	Dulles, US
b.root-servers.net	Marina Del Rey, US
c.root-servers.net	Herndon; Los Angeles; New York City; Chicago, US
d.root-servers.net	College Park MD, US
e.root-servers.net	Mountain View CA, US
f.root-servers.net	Ottawa; Palo Alto; San Jose CA; New York City; San Francisco; Madrid; Hong Kong; Los Angeles; Rome; Auckland; Sao Paulo; Beijing; Seoul; Moscow; Taipei; Dubai; Paris; Singapore; Brisbane; Toronto; Monterrey; Lisbon; Johannesburg;Tel Aviv
g.root-servers.net	Vienna VA, US
h.root-servers.net	Aberdeen MD, US
i.root-servers.net	Stockholm; Helsinki; Milan; London; Geneva; Amsterdam; Oslo; Bangkok; Hong Kong; Brussels; Frankfurt
j.root-servers.net	Dulles VA; Mountain View CA; Seattle WA; Amsterdam; Atlanta GA; Los Angeles CA; Miami; Stockholm; London; Tokyo; Seoul; Singapore; Sterling VA
k.root-servers.net	London, UK; Amsterdam, NL; Frankfurt, DE; Athens, GR; Doha, QA
l.root-servers.net	Los Angeles CA, US
m.root-servers.net	Tokyo, JP; Seoul, KR

なお，ひとつのルート DNS サーバについて複数の運用拠点が明記されているものは，エニーキャストと呼ばれる技術を用いて運用されているルート DNS サーバである．エニーキャストの詳細に関しては，3.2 節にて述べる．

3.2 節　エニーキャスト DNS

本節では，DNS のエニーキャスト技術について述べる．DNS におけるエニーキャストとは，同一のデータ空間を持つ複数台の DNS サーバがインターネット上に存在しており，それが DNS のプロトコル上も，1 台の DNS サーバであるがごとく振る舞わせるようにする仕組みである．

これは，文献 [17] にあるように，インターネットの経路制御において，同じアドレス空間を複数の地点から広告することによって実現している．すなわち，名前解決要求が発生する地点から，ネットワーク的に最も近い位置にあるエニーキャスト DNS サーバにて，名前解決が行われる．

つまり，ユーザから見れば 1 台の DNS サーバに見えるが，実は同じ DNS サーバがインター

ネット上に複数台存在しており，どの DNS サーバからサービスの提供を受けているのかは，通常のユーザには判別できない．

エニーキャストの動作概念を図 3.3 に示す．この図は，例としてルートゾーン (.) を保持するルートネームサーバのひとつである m.root-servers.net という DNS サーバが，インターネット上に 3 台存在する場合を示した．

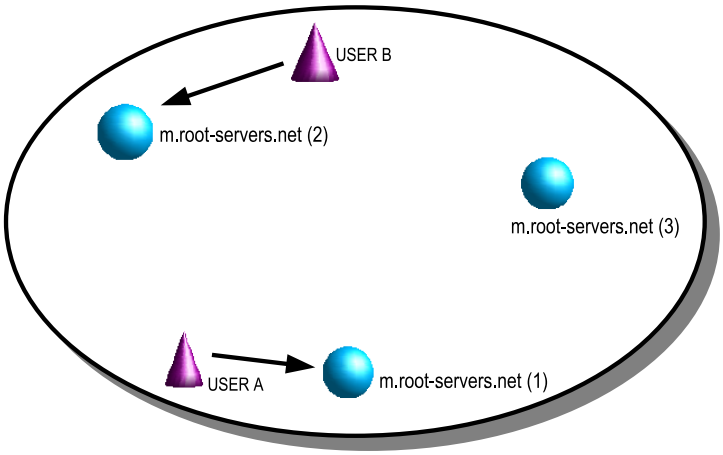


図 3.3: エニーキャストの仕組み

USER A がルートゾーン (.) の名前解決を行おうとした場合，ネットワークの経路制御的に最も近い m.root-servers.net(1) に対して問い合わせ要求が届く．そして，m.root-servers.net(1) にてその名前解決要求が処理され，返答が行われる．また，USER B が同じくルートゾーンの名前解決を行おうとした場合には，ネットワーク的に最も近い m.root-servers.net(2) にて名前解決要求が処理される．この際，m.root-servers.net (1) と (2) による差異は何もなく，ユーザは実際にはどのエニーキャスト DNS サーバから応答が得られたのか，認識することはできない．

このエニーキャスト DNS サーバは，現在ルート DNS サーバや ccTLD サーバといった，DNS データベースの根幹を担うような DNS サーバにて採用されている．エニーキャストの利点として，以下の点があげられる．

- 負荷分散

  エニーキャスト技術を用いることで，DNS サーバに送られる名前解決要求を分散させることが可能となる．これは，頻繁に問い合わせが行われる名前空間を保持しているルート DNS サーバや gTLD DNS サーバにとって有効である．

- サービス妨害攻撃の回避

ある DNS サーバに対して複数地点から攻撃が行われた際，その DNS サーバがエニーキャスト DNS サーバであれば，受ける攻撃を分散させることが可能となる．また，ある地点から集中的に攻撃された場合には，その地点から最も近いエニーキャスト DNS サーバのみに影響を抑えることができる．

- 耐障害性

  エニーキャスト DNS サーバに障害が発生し，サービスを提供できなくなった場合には，そのエニーキャスト DNS サーバ拠点からインターネットへの経路広告を止めることで，自動的に他のエニーキャスト DNS サーバへ名前解決要求を振り替えることが可能となる．

このような利点があるため，重要な DNS サーバにはエニーキャスト技術が用いられる機会が増えている．

3.3 節　DNS のセキュリティ

2.4 節にて述べたとおり，DNS をインフラとしてとらえた場合，耐障害性が要件として求められる．この際，インターネットの中で動作するインフラであるがゆえに，DNS の耐障害性にとって無視できない要件がある．

それは，インターネットの世界で頻繁に行われている，サービスに対する妨害攻撃である．インターネットにおいては，正常なサービス提供を妨げる目的で様々な攻撃が行われている．攻撃され被害を被ったと，CERT/CC[18] に対して連絡があったものについては，CERT/CC Incident Notes[19] にまとめられている． 攻撃の目的は，愉快犯であったり，特定の目的を持って行われたりと様々である．

DNS も例外ではなく，過去に様々な攻撃を受けている．DNS をインフラとしてとらえた場合，こういった攻撃に関する議論が必要である． そこで本章では，DNS の正常な動作を脅かす攻撃やセキュリティの問題に関して述べる．

3.3.1 節　DNS に対する攻撃

2 章にて述べた通り，DNS は各種サービスの基盤技術である．DNS が社会インフラとして機能するためには，耐障害性に優れ，サービスが恒常的に提供される必要がある．

しかし，現状の DNS の運用においては，DNS に対する様々な攻撃が存在しており，過去にも被害が報告されている．代表的な攻撃として，DNS サーバのなりすまし (Spoofing) や，DNS サーバの乗っ取り，さらに DNS サーバに対するサービス停止攻撃 (DoS Attack) といったものがあげられる．これらの攻撃による被害も報告されており，CERT 発行の文章や，NANOG[20] にて行われる議論の中に発見することができる．

DNS に対する攻撃は，攻撃の目的や手法によって，次に述べる種類に分類することができる．

- サービス妨害攻撃

  DNS が提供する，名前解決というサービスを妨害する目的で行われる攻撃である．ある名前解決を妨害することによって，ユーザがその名前に対するサービスを受けられないようにすることを目的とする場合が多い．例として，ユーザがある Web サーバに通信を行うことを妨害したい場合を図 3.4 に示す． この図に示すとおり，通信を妨げたい Web サーバの名前を保持している DNS サーバを攻撃し，名前解決を妨げることによって，結果としてユーザは Web サーバに通信することができなくなる．つまり，直接 Web サーバを攻撃することなく，Web サーバへの通信を妨害することができる．

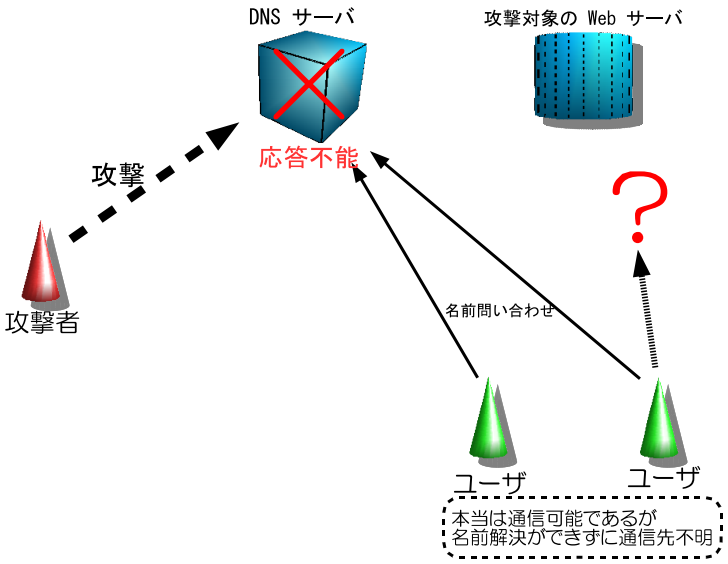


図 3.4: DNS サービス妨害による攻撃

この攻撃の具体的な手法としては，次のものがあげられる．

1. サーバに対する物理的攻撃

   DNS サーバに対して，機器の破壊行為等の物理的な攻撃を行う方法である．この攻撃を行うためには，攻撃者が，攻撃したい DNS サーバの物理的な所在地を知っており，かつその場所に侵入できることが前提となる．この攻撃は証拠が残ってしまいがちなため，実際に実行される可能性は低いと考えられる．しかし，ルート DNS サーバの運用基準を定めた RFC2870[21] には，物理的なセキュリティの確保という項目も盛り込まれており，無視できない要素となっている．

2. ネットワーク経由での遠隔攻撃

   DNS サーバに対する攻撃で最も多いのが，この手法による攻撃である．ネットワークを利用して，攻撃対象の DNS サーバに大量のパケットを送信することによって，DNS サーバに対する正常な通信を妨害するという手法である．この攻撃手法は Denial of Service(DoS) 攻撃と呼ばれる．

   攻撃者が特定の IP アドレスを利用して攻撃している場合には，フィルタ等の手法によってその IP アドレスからの通信を禁止することができる．しかし，攻撃者が送信元 IP アドレスを偽装して攻撃するか，もしくは複数の地点から同時に攻撃を行った場合は，攻撃元の IP アドレスで通信を禁止することが困難となる．この攻撃手法は Distributed Denial of Service(DDoS) 攻撃 [22] と呼ばれ，防ぐことが困難な攻撃である．図 3.5 に，DoS 攻撃と DDoS 攻撃の概念を示す．

   過去に，ルート DNS サーバに DDoS 攻撃が行われたことがある．これは資料 [23] に述べられている．2002 年の 10 月 22 日 (U.S. 東海岸時間) に，13 台のルート DNS サーバに対して，DDoS 攻撃が行われた．多数の発信元 IP アドレスから，ICMP echo request が大量に送信された．その結果，いくつかのルート DNS サーバは，処理のための負荷が増大し，正常なサービスに影響を及ぼした．

- 詐称攻撃

  DNS の提供する名前解決サービスにおいて，偽の名前解決結果を返すことによって，ユーザの通信を攻撃者が意図した方向に導くという攻撃である．図 3.6 に示す通り，DNS からの応答を偽ることによって，Web やメールといった，すべてのサービスを攻撃者の意図した方向に誘導することが可能となる．この攻撃は，ユーザが騙されたことに気づきにくく，正常な通信が行えないうえに，偽装された Web サイトによって詐欺行為に合うといった，二次的被害に発展しやすい攻撃である．

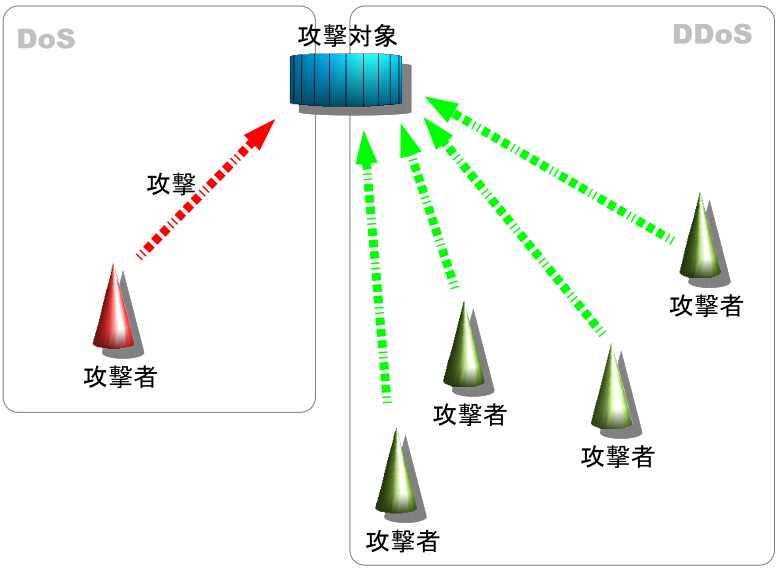


図 3.5: DoS 攻撃と DDoS 攻撃

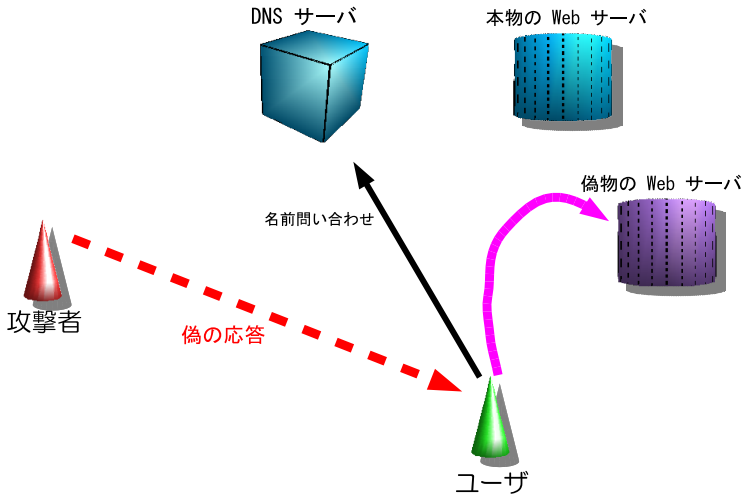


図 3.6: 詐称攻撃によるユーザの誘導

この攻撃の具体的な手法としては，次のものがあげられる．

1. DNS サーバ乗っ取り

DNS サーバの乗っ取りは，DNS サーバのセキュリティホールを利用して行われる．過去に明からになった DNS サーバのセキュリティホール [24] を利用すると，その DNS サーバが動作しているホスト全体の管理権限を搾取することができる． これによって，その DNS サーバが保持している名前空間を自由に書き換えることが可能となり，偽の応答を返すことが可能となる．したがって，DNS サーバを乗っ取ることは，詐称攻撃を行うための最適な手段となり得る．

2. キャッシュ汚染 (Cache Poisoning)

   再帰 DNS サーバは，一度解決した名前を一定時間保持している．保持期間中に再問い合わせがあった場合，権威 DNS サーバに再問い合わせを行うことなく，手元に保持している情報を返答する．これを DNS のキャッシュ機構 [15] と呼ぶ．このキャッシュ機構によって，DNS は再問い合わせの回数を減らすことができ，規模性と耐障害性を向上させることに成功している．キャッシュ機構の有効性については，論文 [25] にて述べられている．

   攻撃者は，再帰 DNS サーバが持つキャッシュに偽のデータを覚えさせることによって，その再帰 DNS サーバを利用しているユーザを騙すことができる． これは，再帰 DNS サーバからの問い合わせに対して，権威 DNS サーバが返答する際に，Additional Section[26] という部分に偽のデータ入れて返答することによって可能となる．

   本来，Additional Section に入っているデータは，名前解決の参考として使われるべきであり，無条件にキャッシュすべきではない．しかし，この攻撃が知られる以前の一部の DNS サーバ実装には，Additional Section に入っているデータをそのままキャッシュし，ユーザからの問い合わせ応答に利用していた実装があった．現在の DNS サーバ実装には，この攻撃を防ぐために，Additional Section に入っているデータを検証してからキャッシュする，という機構が取り入れられている．

   過去に，DNS サーバのこの挙動を利用した攻撃も行われ，CERT Coordination Center から勧告 [27][28] が出されている．

3. なりすまし攻撃

   DNS サーバのなりすまし攻撃は，図 3.7 に示す 2 種類の場合が存在する．なりすまし (I) の場合は，ユーザが DNS サーバに対して名前解決を要求した際に，第三者が発信元 IP アドレスを偽造して，偽の応答パケットを送信することによって，ユーザを騙す攻撃である．なりすまし (II) の場合は，DNS サーバ同士の通信において，(I) の場合と同様に第三者によって偽の回答が行われる場合である．

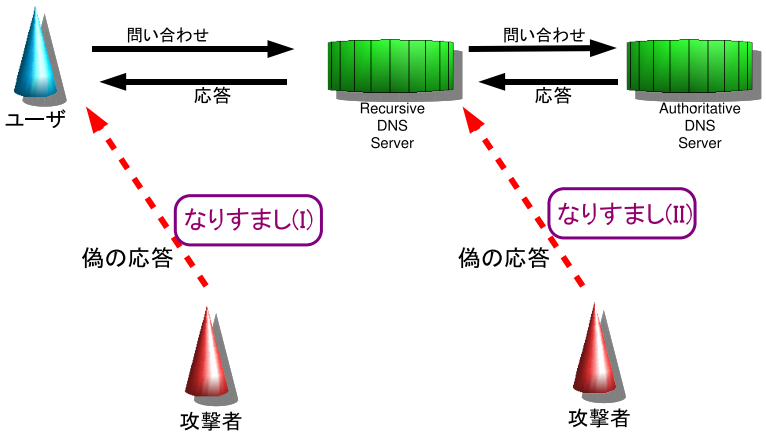


図 3.7: なりすまし攻撃

この攻撃手法のさらなる詳細に関しては，関連研究の 4.2 章にて述べる．

3.3.2 節　攻撃への対策

前節にて，DNS への主要な攻撃について述べた．それらの主要な攻撃に対しては，セキュリティ問題を解決するために新技術が考案され，実験されている．そこで本節では，攻撃を防御するための方法について論ずる．

まず，物理的な攻撃に対する防御策は，DNS サーバ設置場所のセキュリティを向上させることである．次に，ネットワークを利用したサービス妨害攻撃であるが，これは通常の名前解決要求と区別しにくいため，完全に防御することが難しい攻撃である．DoS の場合には，その送信元 IP アドレスからの通信を遮断することによって解決できる場合がある．しかし，DDoS の場合には，遮断することが難しい． そのため，対処療法として，3.2 節にて述べたエニーキャストの導入があげられる． エニーキャストによって攻撃対象となる DNS サーバを分散させておくことによって，攻撃によって受ける被害も分散させるという方法である．しかし，分散拠点の設置場所によっては，効果的に攻撃を分散させることができず，被害が集中してしまう場合もある． そのため，エニーキャストによる防御策を有効に機能させるためには，多地点からの到達性を考慮

した分散拠点の選定が必要となる．

さらに，サーバ乗っ取り攻撃に対しては，DNS サーバに関するセキュリティ勧告に注意し，問題のある実装を利用しないようにすることが基本的な防御策である．このためには，運用している DNS サーバのバージョン情報を管理することが必要となる． また，キャッシュ汚染攻撃に関しても，キャッシュ汚染を引き起こさないよう改良された DNS サーバ実装を用いることで防御できる．

最後に，なりすまし攻撃の防御に関しては，DNSSEC[29] と呼ばれる技術の導入が検討されている．これは DNS サーバが送信するデータの起源を保証する技術である．図 3.8 に示すように，上位の名前空間から下位の名前空間に対してその正当性を保証するための署名を行い，それを連鎖させることによって名前空間全体の正当性を保証する技術である．

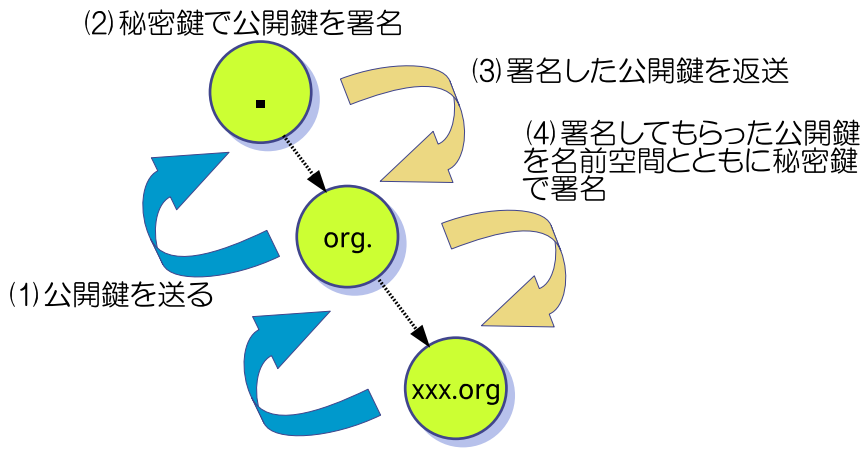


図 3.8: DNSSEC の仕組み

3.3.3 節　DNSSEC

なりすまし攻撃を防ぎ，DNS のセキュリティを確保するために，DNSSEC[29]，TSIG[30]，SIG(0)[31]，TKEY[32] といった技術が考案され，導入が検討されている． 以下に，これら新技

術について延べ，実際の DNS サーバ運用におけるそれぞれの適用範囲と，セキュリティを向上させるための条件について述べる．

- DNSSEC

  DNSSEC は，公開鍵暗号方式を用いてデータ起源の認証を行う技術である．まず，あるゾーンに対して秘密鍵と公開鍵の鍵対を作成する．公開鍵は KEY RR(Resource Record) にて配布する．そして，ゾーンに存在する各 RR に対して秘密鍵にて署名を行い，SIG RR に署名結果を記述する．従って，ひとつの RR に対して，対応する SIG RR がひとつずつ存在することとなる．また，公開鍵と署名結果を RR としてゾーン内に入れて公開することにより，既存の DNS の技術を利用して配布することが可能となっている．

  名前を引くクライアントは，公開されている署名と公開鍵とを用いて認証を行い，データの起源を確認する．この際，配布されている公開鍵の正当性を保証するために，公開鍵自体もデータの起源が保証されている必要がある．すなわち，公開鍵自体が上位のゾーンの秘密鍵を使って署名されている必要がある．図 3.8 に示すとおり，上位のゾーンから署名を連鎖させることによって，DNS の木構造全体のデータ起源を保証する．しかし，DNSSEC は基本的にデータの起源の正当性を保証する技術であり，DNS サーバ間，もしくは DNS サーバとユーザとの間のデータトランザクションの正確性を保証する技術ではない．これを実現するには，TSIG や SIG(0) といった技術を併用する．

- TSIG

  TSIG は，秘密共有鍵を用いて，DNS サーバとクライアントとの間で行われるトランザクションを認証する技術である．サーバとクライアントにて秘密鍵を共有することで，あらかじめ認証されたクライアント以外からのサーバの利用を制限したり，メッセージの完全性の認証が可能となる．また，サーバ間で TSIG を適用することにより，ゾーン転送の認証やアクセス制限を行うことが可能となる．ただし，認証に秘密共有鍵を用いているため，あらかじめ信頼できる方法によって鍵を共有しておく必要がある．例えば，フロッピーディスクなどを用いて鍵を交換する方法があげられる．この鍵交換を，ネットワークを通じて自動的に行う方法として，後述する TKEY という技術が存在する．

- SIG(0)

  SIG(0) も TSIG と同じく，DNS サーバとクライアント間，もしくはサーバ間のトランザクションの認証を提供する技術である．しかし，TSIG が秘密共有鍵を利用するのに対して，こちらは公開鍵暗号方式を採用している．SIG(0) RR によって配布される公開鍵を用

いて，トランザクションの認証を行う．あらかじめ鍵を共有することなく，サーバクライアント間の通信における完全性が認証される．その一方，公開鍵暗号方式であるため，TSIG に比べて処理の負荷が増大する．

- TKEY

  TKEY は鍵の自動交換を行なうための技術である．TSIG と併用して用いることで，DNS サーバとクライアントとの間で，秘密鍵を自動的に共有したり削除したりすることが可能となる．

3.3.4 節　DNSSEC の導入

本節では，前節にて述べた DNS セキュリティ確保の新技術が，攻撃の防御に対してどのように機能するかを述べる．

まず，DNS サーバが乗っ取られた場合について考える．DNSSEC によって名前空間が署名されている場合には，DNS サーバ乗っ取りによって名前空間を書き換えたとしても，名前空間を再署名しない限り，その正当性を証明することができない．つまり，DNSSEC が導入されていれば，データの改竄を検知することができる．この際，運用上注意すべき点がある．それは，DNSSEC の秘密鍵の管理である． もし秘密鍵を DNS サーバ上に保管しておいた場合には，DNS サーバが乗っ取られると，当然秘密鍵も漏洩する．すると，漏洩した秘密鍵を使用し，改竄した名前空間を再署名することができる．したがって，秘密鍵は DNS サーバ以外の部記憶装置に保管しておくべきである．

次に，DNS サーバのなりすましを防ぐ方法について述べる．図 3.7 におけるなりすまし (I) を防ぐには，DNS サーバとリゾルバクライアントの間にて，TSIG もしくは SIG(0) を利用する．これによって，偽の回答を判別することが可能となる．なりすまし (II) の場合は，DNSSEC を用いることにより防ぐことが可能である．例え DNS サーバのなりすましを行ったとしても，偽の回答の場合には，データ起源の認証に失敗するためである．

最後に，サービス妨害攻撃の防御について述べる．サービス停止攻撃は，DNSSEC では基本的に防ぐことができない．DNSSEC はユーザのためにデータの起源を保証する技術であるため，ユーザからの不正な名前解決要求を防ぐことはできないからである．この防御は，3.3.2 節で延べた通り，エニーキャストによる対処療法を利用する．

以上述べた通り，サービス妨害攻撃以外の代表的な攻撃手法に対しては，DNSSEC と TSIG

を組み合わせることによって，防御もしくは被害を最小限に抑えることが可能である．

しかし，DNSSEC の導入には，解決しなければならない課題が存在する．前述の通り，DNSSEC は名前空間の委譲に従って，署名を連鎖させて正当性を保証する技術である．そのため，名前空間の委譲が正確に行われていなければ，そこで署名の連鎖ができなくなり，そこから下位に位置する名前空間すべての正当性を保証することができなくなる．つまり，名前空間を正常に運用することが，DNSSEC 導入の前提条件となる．

3.3.5 節　DNS のセキュリティ向上にむけて

DNS 全体のセキュリティを向上させるためには，3.3.1 節にて述べたとおり，現在の DNS の運用において脅威となっている代表的な攻撃を理解し，その対策を行うことが必要である．

また，3.3.2 節にて述べた通り，攻撃への対策としては以下のものがあげられる．

- DNS サーバのバージョン管理
- エニーキャストによる分散化
- DNSSEC の導入

3.3.4 節に述べた通り，DNSSEC や TSIG といった新技術の導入は攻撃の防御に有効である．しかし，現状の DNS に対して DNSSEC をそのまま導入することは困難であると考えられる．前述した通り，DNSSEC という技術は，上位のゾーンから署名を連鎖させることによってデータの起源を保証している．すなわち，DNSSEC を有効に機能させるためには，ルート DNS サーバから末端の DNS サーバまで，ゾーン委譲が正しく行われている必要がある．しかし，残念ながら現状の DNS は必ずしも正確な委譲が行われているとは限らない．まとめると，現状の DNS にとってセキュリティを確保するためには，次にあげる運用要件を満たすことが必要である．

1. DNS サーバのバージョン管理
2. エニーキャストの有効性の検証
3. 名前空間の委譲の正確性

3.4 節　DNS における運用基盤分析

本節では，2.5 節にて提案した運用基盤分析モデルと，3 節にて述べた DNS の運用状況をふまえ，DNS の運用基盤分析を行うためのモデルを作成する．

DNS は，大規模かつグローバルなインフラである．DNS のサービス範囲はインターネット全域，すなわち世界規模のサービス範囲となる． また，DNS サーバはインターネット上に数万台の規模で存在し，それぞれがサービス拠点となって協調動作することで，DNS という分散データベースを形成している．

したがって，一つ一つの DNS サーバに対して，要件の調査を散発的に行うだけでは，運用基盤の分析として不十分である． なぜならば，大規模かつグローバルなインフラである DNS の場合には，サービス拠点が複数存在し，それを不特定多数のユーザが共有して使うことになるため，個々の DNS サーバだけを分析しても全体を把握することはできない．

すなわち，個々の DNS サーバに対して分析を行っていたのでは，インフラとしての DNS 全体の分析はできない．4 章にて紹介したする関連研究は，DNS サーバ単体，もしくは特定の DNS サーバ群の監視や計測を行うものでり，複数地点からの偏りのない分析や，ユーザの視点からの分析が難しい．

一般的に，インフラが大規模かつグローバルになればなるほど，一般的にその運用状態を把握するための監視や分析は難しくなる．さらに，サービスのための各設備が，自律分散協調的に動作する場合には，サービス対象の識別が困難となる．

DNS の場合で考えると，DNS サーバの台数が増えれば増えるだけ，管理コストは増大する．また，データベースの一部に障害が発生し，名前解決に失敗した場合，DNS サーバの障害によるものなのか，もしくはデータの不整合によるものなのか，規模が大きくなるにつれ，判断することが難しくなる．また，グローバルなサービスであるが故に，あるユーザがいる地点から正常に利用できているが，あるユーザがいる地点からは利用ができない，といった障害も発生する．

さらに，DNS の負荷分散やセキュリティ向上のために，3.2 節で述べた，エニーキャスト技術が導入されはじめている．運用状況の把握のためには，どのユーザがどの DNS サーバから名前解決サービスを受けているのか識別する必要がある．つまり，DNS が大規模かつグローバルなインフラとして拡大すればするほど，DNS 全体の運用状況の分析は困難となる．

3.5 節　DNS の運用基盤分析モデル

前節の議論をふまえ，次の観点から DNS の分析モデルを構築する．

まず，どのユーザも等しく通信を行う可能性のある DNS サーバ群への到達性を世界各地から測定することによって，ユーザの視点からのサービス分析を行う．これによって，ユーザに対して提供されるサービスの公平性を測定することができる．これを DNS 到達性調査システムとする．

次に，DNS のデータベースの完全性をチェックするために，DNS の委譲情報を調査する． これによって 3.3 節にて述べた，DNS のセキュリティ向上のための調査と，DNS データベースが正常に構築されているか調査することができる．これは，サービスの完全性分析にあたる．これを DNS 委譲情報調査システムとする．

最後に，DNS の運用情報収集と DNS サーバの個体識別を行うことによって，サービス運用情報の分析を行う．これは，DNS サーバ単位に運用者ならびに運用組織の情報を収集したデータベースを作成し，さらに DNS サーバ個体を識別する手法を利用して，エニーキャスト DNS サーバの識別を行うシステムである．このシステムを DNS 運用情報識別システムとする．

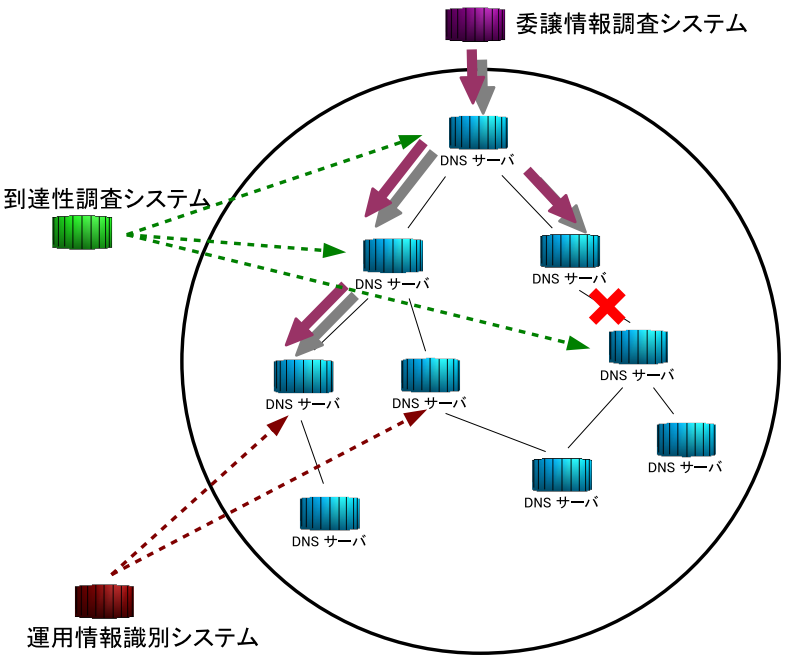


図 3.9: DNS 運用基盤分析システムの概要

- DNS 到達性調査システム

  ユーザの視点からのサービス分析を行うためのシステムである． 複数の地点から，通常のユーザと同様に，計測対象となる DNS サーバに対して名前解決要求を送信し，応答を得られるまでの時間ならびに応答の内容を計測する．この際計測対象とする DNS サーバは，DNS において公共性が高く重要なサーバ群を対象とする．本研究では，ルート DNS サーバと ccTLD DNS サーバを用いた．

- DNS 委譲情報調査システム

  DNS の完全性と正確性を調査するために構築したシステムである．3.3 節にても述べたとおり，DNS にとって委譲の正確性を確保することは，DNS のセキュリティにとっても重要なことであり，完全性を維持するための要件である．DNS の木構造データベースを，ルート DNS サーバから辿っていくことによって，DNS の完全性を調査するシステムである．

- DNS 運用情報識別システム

  障害時の対応を迅速に行ったり，分散拠点による負荷分散サービスが有効に機能しているかどうかを調査するシステムである．DNS サーバに実装されている，個体識別の機能を利用して，調査を行う．

4章　本研究の基盤となる研究

本章では，本研究の基礎となった関連研究について述べる．本章にて述べられる研究は，本研究の手法を確立するにあたって，その一部となっている研究であり，他の組織もしくはプロジェクトと連携して行ったものである．

4.1 節　アドレス配布ならびに管理に関する研究

本節では，関連研究として行った IPv6 アドレス割り当てとアドレス管理情報の連携について述べる．

IPv4 では，IP アドレスは基本的にインターネットレジストリから割り当てを受ける．インターネットレジストリは，規模によって Regional Internet Registry(RIR)，National Internet Registry(NIR)，Local Internet Registry(LIR) に分類される．RIR は，大きな地域に対してアドレス割り当てを行う組織であり，RIPE[33]，APNIC[34], ARIN[35]，AFRINIC[36]，LANIC[37] などが存在する．NIR は国単位でアドレスを割り当てる組織であり，LIR は小規模なユーザに対してアドレスを割り当てる組織を意味する．LIR は一般的には Internet Service Provider(ISP) である場合が多い．

IPv6 においては，IP アドレスはネットワークトポロジに従って割り当てられることが推奨されている [11]．現在の IPv6 では，上位 64bit をネットワークアドレス，下位 64bit をホストアドレスとして用いる構造 [12] となっている．大規模な組織や，ISP のようにさらにユーザにアドレスブロックの割り当てを行う組織は，35bit，もしくはそれ以上の大きなアドレスブロックの割り当てを RIR から受ける．大きなアドレスブロックの割り当てに関しては，RIR から割り当てを受けるため，IPv4 と同じである．従って，アドレス利用者情報も RIR の whois データベースに登録される．

一方，IPv6 アドレスブロックの割り当てを希望する ISP 以外の通常の組織は，48bit プレフィクス長の割り当てを受けることが推奨されている [38],[39]． 従って，ユーザである組織が割り当てられるアドレスブロックは，通常 48bit のプレフィクス長であり， LIR である ISP から割り当てをうける．

図 4.1 に示すように，48bit というプレフィクス長のアドレスブロックは，割り当てを受けた組織内において，16bit のネットワークアドレス部が利用できる． これは，65536 サブネットを構成できるアドレス空間である．

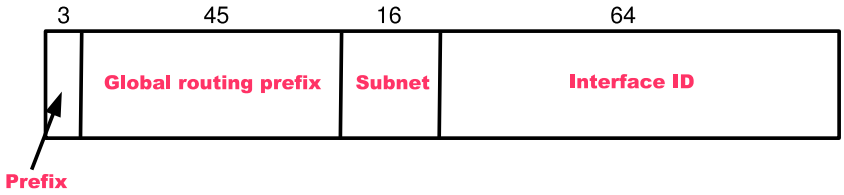


図 4.1: IPv6 のアドレス構造

この際，LIR が割り当てたアドレスブロックの割り当て情報を，RIR のデータベースに登録する義務は生じない [11]．そのため，48bit プレフィクス長のアドレスブロック単位で，whois データベースを参照することができなくなる．ルーティングの障害や，セキュリティ問題が発生した場合に，IPv4 ならば，whois の情報を用いてアドレス利用者を特定することができる．しかし，IPv6 の場合には，35bit や 32bit といった，大きなアドレスブロックに関する whois 情報しか参照できない．これでは障害の解決に支障を来たすため，この研究では，MARS[14] というレジストリデータベースを設計し，提案した．

MARS の概要を図 4.2 に示す．図中の RP は， Registry Point の略である． 図中の木構造は，アドレスブロックの改組式に対する割り当てを示す．また，A:B:E::1 等の表記は，割り当てられた IPv6 アドレスを表す．

RP が存在する場所においては，その組織にて MARS が動作しており，割り当てられたアドレスブロックに関する情報が登録されていることを示す．アドレスブロック割り当てを行った上位組織の MARS データベースには，割り当てを受けたアドレスブロックと割当先の MARS データベースに関する情報を登録する．

ユーザがアドレスブロック使用者情報の参照をする場合には，MARS の割り当て情報をもとに木構造データベースをたどることで，目的の情報を持った MARS にたどり着き，情報を参照することができる．

アドレスブロック割り当てとともに，利用者情報を登録した MARS データベースを連携させることによって，48bit プレフィクス長アドレスブロックの利用情報も検索できるようにしたシステムである．IPv6 の場合には，上位の接続 ISP を変更すると，割り当てられる IPv6 アドレスブロックも変更となる．この際，MARS システムを利用すると，アドレスブロック割り当てと利用者情報を同時に更新することができ，不整合が発生しないという利点がある．

この研究にて，アドレスブロック情報，すなわち「利用される資源」と，利用者情報，すなわち「管理情報」を連携させるというモデルを提案した．このモデルは，本研究の運用基盤分析モデルにも採り入れられている．

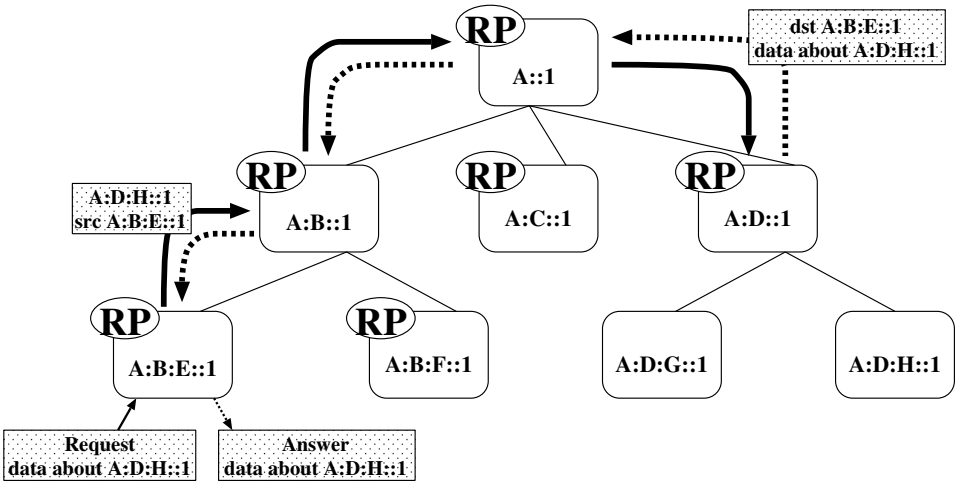


図 4.2: MARS システム

4.2 節　DNS への攻撃に関する研究

3.3 節にて述べた通り，DNS に対する攻撃は度々繰り返されている．特にルート DNS サーバや gTLD DNS サーバといった，広く利用される DNS サーバに対して攻撃が行われた場合，その影響範囲も大きくなる．そこで本節では，DNS に対する詐称攻撃の影響に関する研究について述べる．

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．それは文献 Threat Analysis of the Domain Name System[40] にて述べられている．

この関連研究では，ユーザへの被害が大きいと思われる詐称攻撃に関して，詐称攻撃の成功率を実験にて検証し，DNS の攻撃への耐性と，ユーザへの影響を計測した．その結果，現在の DNS に対しては，ある特定の状況下において，容易に詐称攻撃が可能であることが判明した．

その特定の状況とは，広く普及している無線 LAN 環境である．無線 LAN 環境下では，ユーザが出すパケットを攻撃者が容易に盗聴できるため，ユーザと DNS サーバの間のデータを改竄する，中間者攻撃 (Monkey-in-the-middle attack) を行いやすい．

一方，無線 LAN ネットワークは，ネットワークに関する標準化会議である，Internet Engineering Task Force(IETF)[41] の会場や，喫茶店や駅に普及し始めている，公衆無線 LAN サービスにおいても提供されている．特に公衆無線 LAN サービスは，ユーザにとってネットワークを気軽に使えるシステムであるため，様々な場所に普及し始めている．

すなわち，無線 LAN は攻撃者にとって非常に盗聴しやすい環境であるため，公衆無線 LAN

サービスの利用においては，常に攻撃にさらされる可能性がある．

そのため，この関連研究では，通常の無線 LAN ネットワーク環境と，通常の DNS サーバという環境下において詐称攻撃実験を行った．これは，詐称攻撃のための特別な実験環境を用意するのではなく，あくまでも通常の無線 LAN 環境にてどこまで詐称できるかを検証した．

攻撃に用いたソフトウェア

実験に用いた攻撃ソフトウェアは，この実験のために作成したものである．Linux OS にて動作する USO800d[42] というソフトウェアと，BSD 系 OS にて動作する dnsattack[43] という 2 種類のソフトウェアにて攻撃を行った．

どちらのソフトウェアも，BPF や Raw ソケットを用いてネットワークを流れる DNS 問い合わせパケットを盗聴し，偽応答を生成して返答するという仕組みである．

実験環境

この実験は，WIDE プロジェクト [44] に参加している人が一同に会する，WIDE 合宿と呼ばれる会議にて行った．参加人数は約 200 名程度である．

WIDE 合宿においては，ネットワークは無線 LAN にて提供されており，複数の無線 LAN 基地局をローミングする形で，会議場に接続性を提供していた．また，参加しているユーザには，802.11a 方式 と 802.11b 方式 [45] という，2 つの無線 LAN 方式が提供されていた．

無線 LAN にて盗聴型詐称攻撃を行う場合，一つの無線 LAN 基地局にアソシエーションして盗聴するだけでは，同一無線 LAN 基地局にアソシエーションしているユーザの通信のみしか盗聴できない．従って，複数の攻撃ホストを準備し，複数の無線 LAN 基地局にアソシエーションして攻撃を行った．

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

詐称実験

実験では，参加者には実験開始時間を明確に告げずに，約 15 分間詐称攻撃を行った．

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気づかせるようにした．計測は，詐称攻撃を行うホストとは別のホストにて，tcpdump を用いて行った．特に，一番利用者が多いと思われる Web に関しては，誘導先に本物に似せた Web ページ (図 4.3) を用意しておくことで，ユーザに詐称されることの危険性を実感してもらった．

図 4.3: 偽装 Web サーバ例

実験結果

802.11a にアソシエーションしていた人への攻撃結果を，表 4.1 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結果である． なお，騙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 4.2 に示す.

これらの結果から，92.3% の被験者が一度は騙されたことがわかる．

表 4.1: 偽ホストに誘導された被験者の統計 (802.11a)

被験者	65
一度でも騙された人	60

表 4.2: 誘導されたサービスの内訳 (802.11a)

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

また，802.11b において騙された被験者数を表 4.3 に示す．被験者数は 802.11a の場合と同じく，ホストのソース IP アドレスにて数えている．

これらの結果から，802.11b では 65.3% の被験者が一度は騙されたことがわかる．

表 4.3: 偽ホストに誘導された被験者の統計 (802.11b)

一度でも騙された被験者数	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 チャンネルにいない騙された被験者数	17
1 チャンネルの被験 IP アドレス数	52
1 チャンネルで騙されなかった被験者数	18
1 チャンネルで騙された被験者数	34

DNS クエリの分析

次に，攻撃ホストとは別のホストにて tcpdump を行い，本物の DNS サーバが返答するクエリと，攻撃ホストが返答する偽のクエリの時間差に関して分析を行った．

802.11a における結果を図 4.4 に示す．

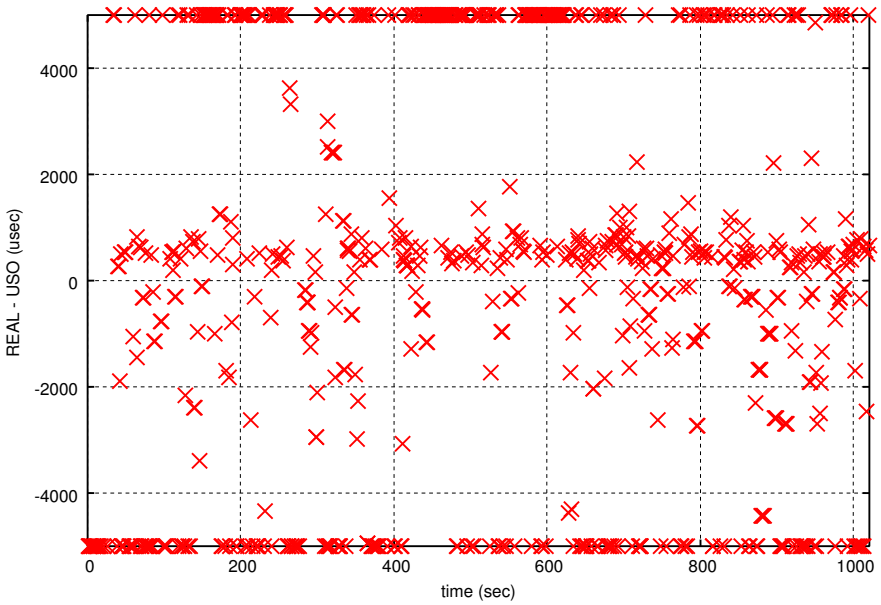


図 4.4: DNS クエリの分析 (802.11a)

この図において，横軸は tcpdump にて観測した DNS 問い合わせならびに応答クエリの組を，問い合わせクエリの時順に並べていったものである．縦軸は，問い合わせに対して攻撃ホストが答えた時間から，本物の DNS サーバからの返答が得られた時間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，攻撃ホストが本物の DNS サーバよりも先に応答した場合を示す．逆に 0 より上に位置する点は，本物の DNS サーバ方攻撃ホストよりも先に応答した場合を示す．

また，上の X 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，下の X 軸に張り付いている点は，本物の 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 4.4 に，内訳を示す.

表 4.4: DNS クエリの内訳 (802.11a)

観測された問い合わせ数	1021
攻撃ホストの方が先に応答した場合	654
本物の DNS サーバの方が先に応答した場合	330
応答が観測されなかった場合	37

また，802.11b における結果を図 4.5 に，内訳を 4.5 に示す.

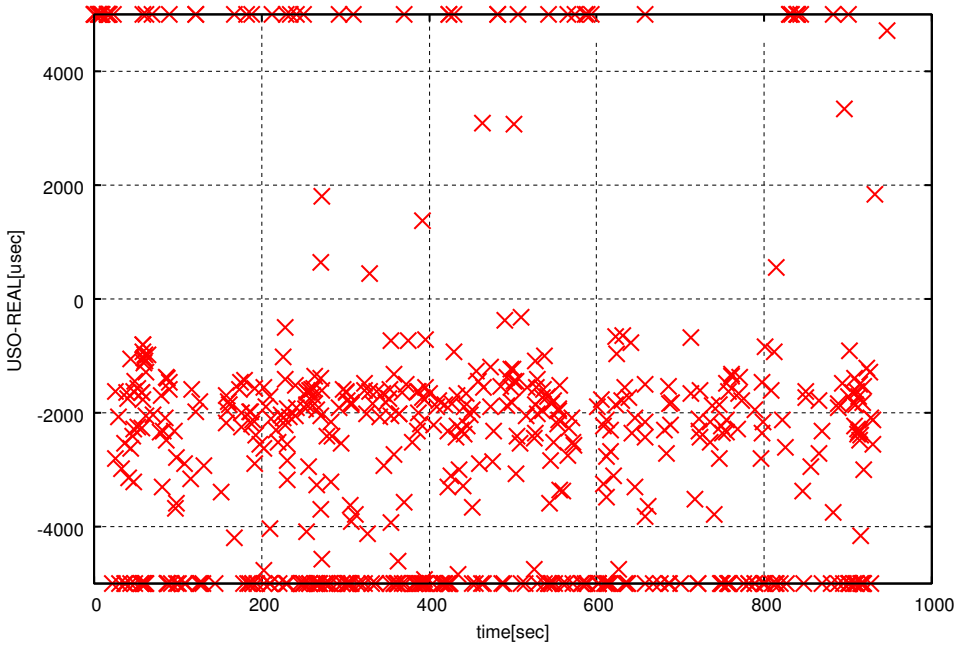


図 4.5: DNS クエリの分析 (802.11b)

表 4.5: DNS クエリの内訳 (802.11b)

観測された問い合わせ数	599
攻撃ホストの方が先に応答した場合	535
本物の DNS サーバの方が先に応答した場合	64
本物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

これらの実験結果から，無線 LAN ネットワークにおいて，ユーザに対して容易に詐称攻撃を

行うことが可能であるとわかった．また，誘導された先に本物に似せたサービスが提供されていた場合，ユーザが騙されていることに気づかずにさらに被害を拡大させる可能性があることもわかった．

この関連研究において，ユーザがインフラとしての DNS に如何に依存しているかということと，インフラとしての DNS の脆弱性を確認することができた．この脆弱性を防ぐためには，3.3.3 節にて述べた DNSSEC 等の技術を導入する必要がある．そしてそのためには，正確な DNS サーバの委譲関係を保つ必要がある．

4.3 節　DNS サーバの応答分析に関する研究

本節では，DNS サーバへの名前解決要求ならびに DNS サーバからの名前解決応答に関する，計測や分析を行った関連研究について述べる．

4.3.1 節　DNS Root/gTLD Performance Measurement

NeTraMet[46] というツールを用いた，ルート DNS サーバと gTLD サーバへの到達性計測 [47] が，CAIDA[48] という組織によって行われている．このデータ収集に参加し，慶應義塾大学と東京大学の 2 地点において NeTraMet によるルート DNS サーバへのトラフィック計測を行った．

NeTraMet とは

NeTraMet は，RFC2720[49]，RFC2721[50]，RFC2722[51]，RFC2723[52]，RFC2724[53] にて標準化された，Real Time Flow Measurement(RTFM) に従って作成されている．フロー観測の記述言語として Simple Ruleset Language(SRL) を用いており，柔軟なフロー計測を可能としている．

NeTraMet パッケージは，実際にフロー計測を行う NeTraMet コマンドと，計測結果を SNMP にて取り出して記録する NeMaC コマンドの 2 つのコマンドから構成され，Linux，FreeBSD，NetBSD といった一般的なフリー UNIX 系 OS の上で動作する．計測を行うためには，NeTraMet が動作するホストにて次の条件が必要となる．

1. 計測対象となるトラフィックが，NeTraMet を動作させるインタフェースにて観測できる．

2. 特権 (ルート権限) にて NeTraMet を動作させることができる.

NeTraMet は実トラフィックを監視して計測するという静的な計測ツールであるため，スイッチングハブにおけるポートミラー設定やトラフィック分岐装置等によってトラフィックを複製し，監視できる環境が必要となる.

NeTraMet パッケージは，http://www2.auckland.ac.nz/net/NeTraMet/から入手可能である.

NeTraMet による計測

この研究は，NeTraMet を利用してルート DNS サーバ群への名前解決要求ならびに応答クエリの計測を行い，固定点における長期的な静的計測を行うことで，ルート DNS サーバへの到達性を分析することを目的として行った.

NeTraMet の開発元である CAIDA においても，ルート DNS サーバ群ならびに gTLD DNS サーバ群へのクエリの計測を行っている．この計測は CU Boulder，University Auckland，University of California San Diego(UCSD) の 3 地点で行われている．計測結果は論文 [47] に示されている．また，CAIDA による最新の計測結果は，http://www.caida.org/cgi-bin/dns_perf/main.pl にて確認できる.

DNS のインフラとしての公平性，確実性を検証するため，WIDE Project においても，DNS データベースの起点となる ルート DNS への到達性を観測した．設置地点は，慶應義塾大学と東京大学の 2 地点である．慶應義塾大学においては，慶應義塾大学湘南藤沢キャンパスとそのネットワークの上流となる WIDE Project との中間に位置するスイッチにおいてポートミラー設定を行い，NeTraMet にて計測した．東京大学においては，東京大学とその上流となる学術情報ネットワーク (SINET)[54] との間に光分岐装置を設置し，NeTraMet にて計測した．これによって，慶應義塾大学においては，湘南藤沢キャンパスにて発生するルート DNS サーバへのクエリを計測し，東京大学においては東京大学全学部から発生するルート DNS サーバへのクエリを計測した.

計測するために利用したホストの仕様は，次の通りである．慶應義塾大学では，CPU に Pentium III 800Mhz，メモリ 256MB を搭載したホストに，1000base-SX インタフェースを用いて計測を行った．東京大学では，Xeon 3GHz 2 個，メモリ 1GB を搭載したホストに，同じく 1000base-SX インタフェースを用いて計測を行った．なお，NeTraMet パッケージはどちらも Ver 5.1b2 を使用した．これは，Ver 4 以前の NeTraMet では，DNS クエリの RTT を正確に測定することができないというバグがあったためである.

計測対象は A.root-servers.net から M.root-servers.net までの 13 個の Root DNS サーバである．計測に利用した SRL を図 4.6 に示す．

この SRL 設定によって，各 Root DNS サーバへの問い合わせならびに応答を記録し，それぞれの数や応答を得られるまでにかかった Round Trip Time(RTT) を記録した．この計測結果は，NeMaC によって 5 分単位で記録された．

この計測は，2003 年 9 月から試験的に開始され，2004 年 1 月から 2 地点において本格的に開始された．

計測結果

NeTraMet による測定結果の例として，図 4.7 に 2004 年 8 月 7 日の慶應義塾大学における測定結果を示す．

また，図 4.8 に，2004 年 12 月 31 日の慶應義塾大学における測定結果を示す．

縦軸が RTT(ms) もしくはルート DNS サーバに向けて出された名前解決要求のクエリ数を示し，横軸が時間を示している．グラフ中の「＋」の点が RTT を示し,「×」の点がクエリ数を示す．

どちらの日においても，m.root-servers.net に対する RTT が最小となっている．これは m.root-servers.net が WIDE プロジェクトによって運営されており，計測地点である慶應義塾大学から近い場所に位置しているためである．

注目すべきなのは，i.root-servers.net に対する結果である．8 月の結果と 12 月の結果で RTT が大きく異なっている．これは，i.root-servers.net が分散拠点を東京に設置し，エニーキャストを利用したサービスを開始したためである．このため，8 月の時点では海外にある i.root-servers.net のホストに送られていたクエリが，東京の分散拠点にて処理されるようになったため，RTT が小さくなったと考えられる．

さらに，東京大学における計測結果を図 4.9 に示す．

慶應義塾大学に比べ，東京大学の方が観測されるクエリの全体数が多い．そのため，他のルート DNS サーバよりも，m.root-servers.net に多量のクエリが送信されていることがはっきりと判別できる．これら WIDE プロジェクトにおける NeTraMet の計測結果は，http://dnstap.nc.u-tokyo.ac.jp/NeTraMet/にて公開されている．

NeTraMet による計測の限界

NeTraMet を利用することにより，計測地点におけるルート DNS の利用状況を分析することができた．しかし，NeTraMet による計測の問題点も明らかになった．

- NeTraMet 設置ならびに運用コスト

  NeTraMet を設置し運用するためには，ポートミラーやトラフィック分岐装置等によってトラフィックを複製し，計測する必要があるため，それ相応の設置コストならびに運用コストが必要となる．従って，数多くの地点で計測を行うのは困難である．

- エニーキャストへの対応

  多くの分散拠点によってエニーキャスティングが行われた場合，RTT やクエリ数の変化によって，クエリが処理される分散拠点の変化をつかむことが可能な場合もある．しかし，明確にどの分散拠点にて処理されているのかを把握することは難しい．

これは，NeTraMet 固有の問題ではなく，定点において静的に計測を行う場合に共通に発生する問題である．

```
define DNS = 53;
define PP_UDP_DNS       =  11;
define PP_TCP           = 192;
define PP_OK_SYNACK     =   1;  #   ->SYN,  <-SYN+ACK pairs
define PP_OK_SYNRST     =   2;  #   ->SYN,  <-SYN+RST pairs
define PP_OK_MULTI      =   8;  #   ->DATA, <-ACK for more than one packet
define PP_OK_SINGLE     =  16;  #   ->DATA, <-ACK 'lone' packet
define PP_OK_INGROUP    =  32;  #   ->DATA, <-ACK single packet in a group

define A_ROOT = 198.41.0.4/32;       #  Verisign, Dulles, Va
define B_ROOT = 192.228.79.201/32;   #  ISI, Marina del Ray, Ca
define C_ROOT = 192.33.4.12/32;      #  Cogent, Herndon, Va
define D_ROOT = 128.8.10.90/32;      #  U Maryland, Md
define E_ROOT = 192.203.230.10/32;   #  NASA Ames, Ca
define F_ROOT = 192.5.5.241/32;      #  ISC, Palo Alto, Ca
define G_ROOT = 192.112.36.4/32;     #  DoD NIC, Vienna, Va
define H_ROOT = 128.63.2.53/32;      #  ARL, Abdereen, Md
define I_ROOT = 192.36.148.17/32;    #  KTH, Stockholm
define J_ROOT = 192.58.128.30/32;    #  Verisign, Dulles, Va
define K_ROOT = 193.0.14.129/32;     #  RIPE NCC, Amsterdam
define L_ROOT = 198.32.64.12/32;     #  IANA, Los Angeles, Ca
define M_ROOT = 202.12.27.33/32;     #  WIDE, Tokyo
define B_ROOT_OLD = 128.9.0.107/32;      #  ISI, USC, Ca
define J_ROOT_OLD = 198.41.0.10/32;      #  NSI, Herndon, Va

define TestDestAddress =
   if DestPeerAddress == A_ROOT
      { store FlowKind :=  1\;  store FlowClass := 0\; }
   else if DestPeerAddress == B_ROOT || DestPeerAddress == B_ROOT_OLD
      { store FlowKind :=  2\;  store FlowClass := 0\; }
   else if DestPeerAddress == C_ROOT
      { store FlowKind :=  3\;  store FlowClass := 0\; }
   else if DestPeerAddress == D_ROOT
      { store FlowKind :=  4\;  store FlowClass := 0\; }
   else if DestPeerAddress == E_ROOT
      { store FlowKind :=  5\;  store FlowClass := 0\; }
   else if DestPeerAddress == F_ROOT
      { store FlowKind :=  6\;  store FlowClass := 0\; }
   else if DestPeerAddress == G_ROOT
      { store FlowKind :=  7\;  store FlowClass := 0\; }
   else if DestPeerAddress == H_ROOT
      { store FlowKind :=  8\;  store FlowClass := 0\; }
   else if DestPeerAddress == I_ROOT
      { store FlowKind :=  9\;  store FlowClass := 0\; }
   else if DestPeerAddress == J_ROOT || DestPeerAddress == J_ROOT_OLD
      { store FlowKind := 10\;  store FlowClass := 0\; }
   else if DestPeerAddress == K_ROOT
      { store FlowKind := 11\;  store FlowClass := 0\; }
   else if DestPeerAddress == L_ROOT
      { store FlowKind := 12\;  store FlowClass := 0\; }
   else if DestPeerAddress == M_ROOT
      { store FlowKind := 13\;  store FlowClass := 0\; }

optimise 3;
   if SourcePeerType == IPv4 save;
   else ignore;
   if SourceTransType == UDP save;
   else ignore;
   TestDestAddress;  # Sets FlowKind
   if FlowKind == 0 nomatch;
   else {
      if DestTransAddress == DNS save;  # Avoid 'match on non_DNS flow' msg
      else ignore;
      save ToTurnaroundTime = 120.11.0!0 & 4.2.10!7000;
      count;
      }

set dns_root_wide;
statistics;
format
  FlowRuleSet FlowIndex FirstTime SourcePeerType SourceTransType
  "  " FlowKind FlowClass
  "  " ToPDUs FromPDUs
  "  " ToLostPDUs FromLostPDUs
  "  (" ToTurnaroundTime
  ")";
```

図 4.6: SRL 設定

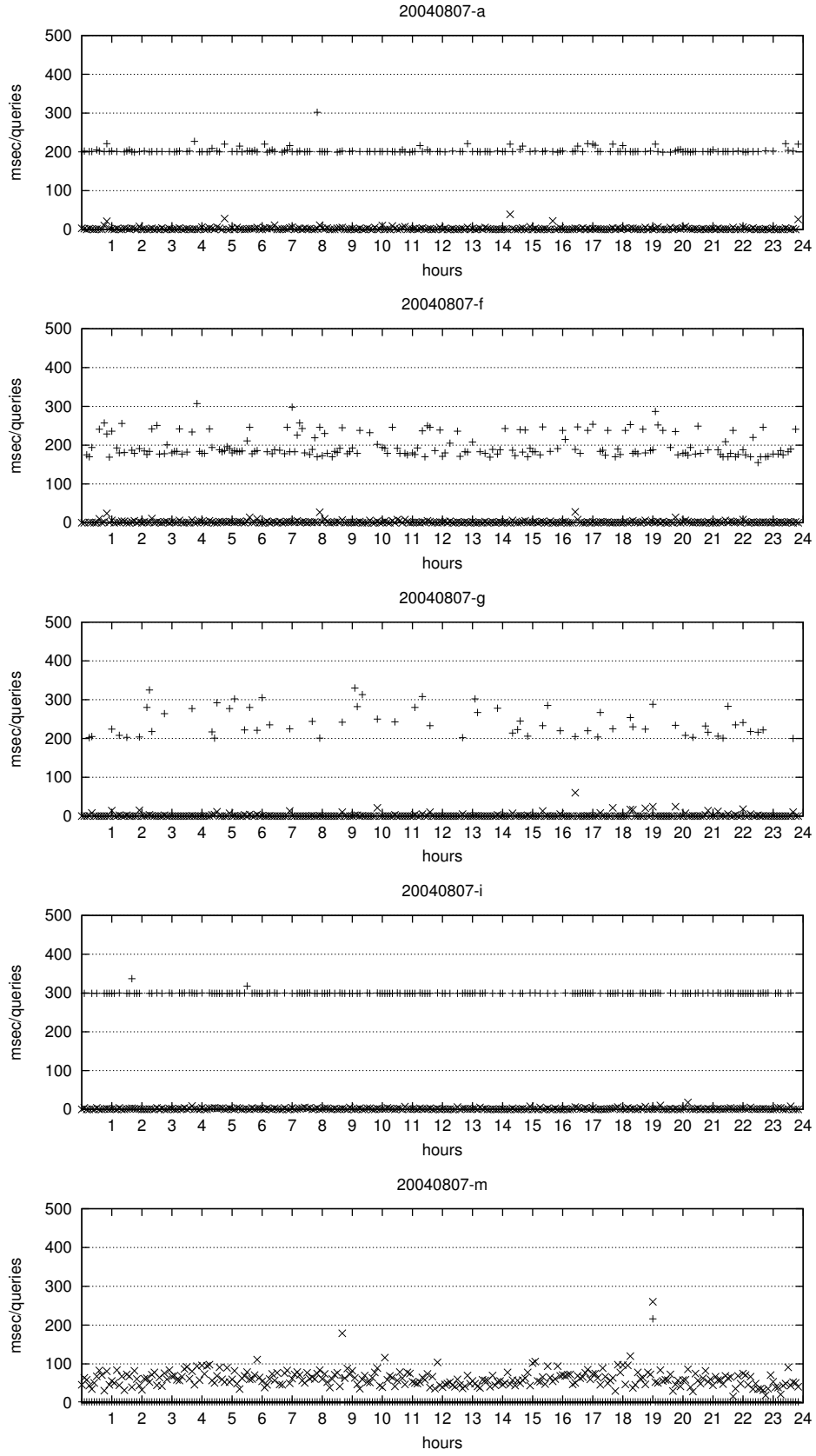


図 4.7: 2004 年 8 月 7 日 : 慶應義塾大学

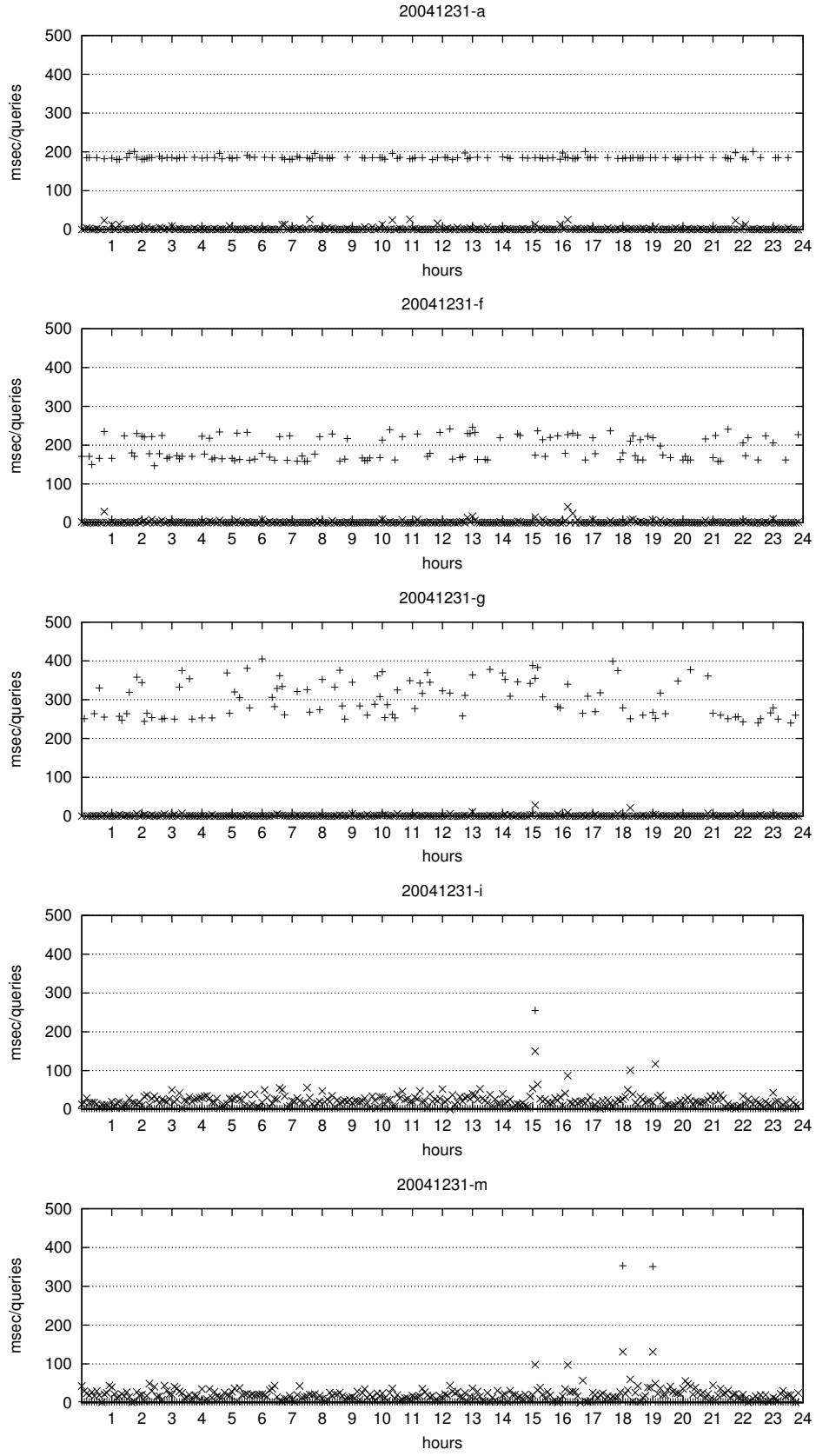


図 4.8: 2004 年 12 月 31 日 : 慶應義塾大学

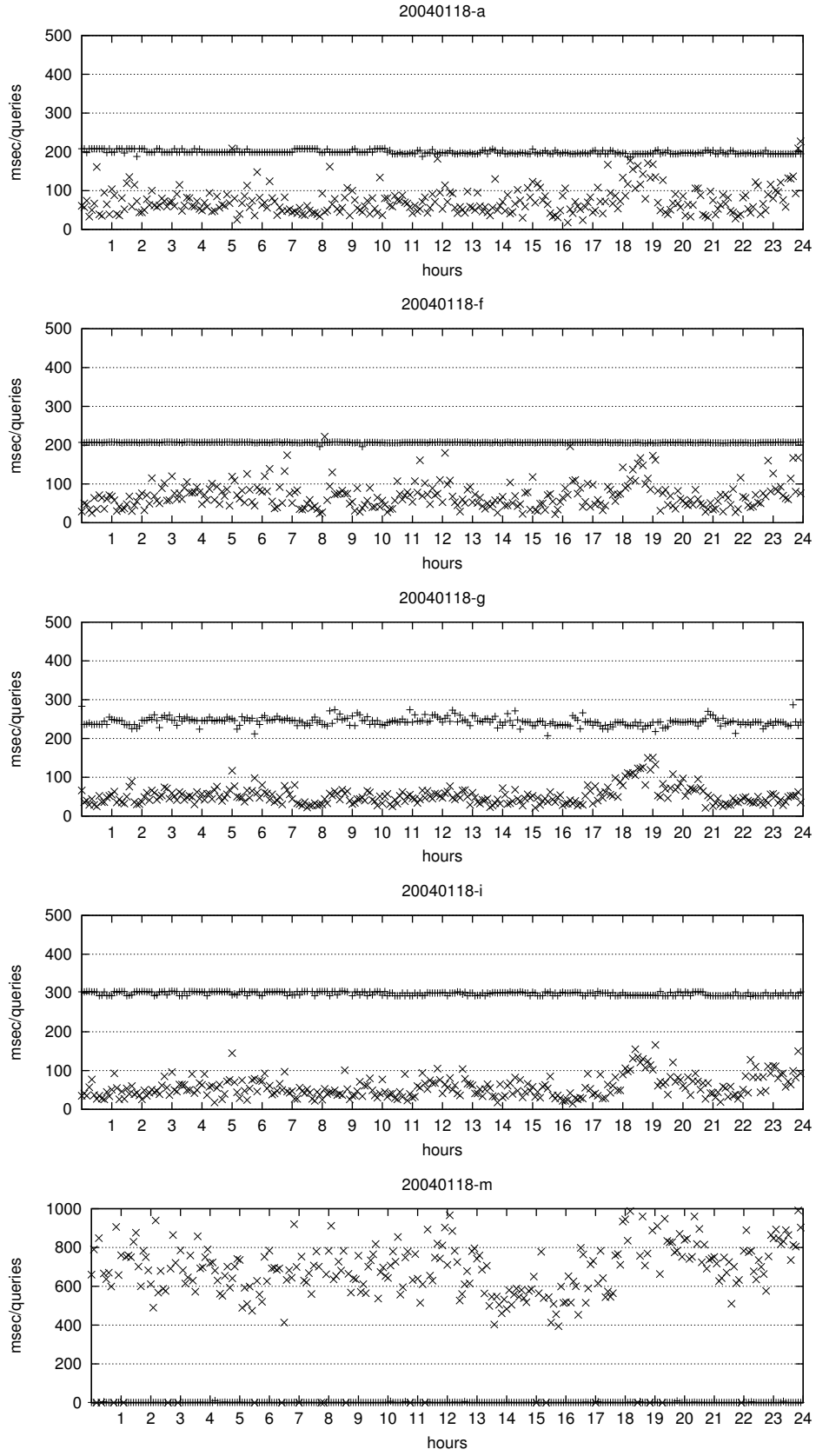


図 4.9: 2004 年 1 月 18 日：東京大学

4.3.2 節　DNS monitoring

NeTraMet 以外にも，ルート DNS サーバや gTLD サーバを定点から計測しているプロジェクトが存在する．インターネットレジストリである RIPE[33] によって行われている，dnsmon[55] というツールを利用した計測である． 世界各地に設置された dnsmon サーバから，多くの DNS サーバに対して DNS 名前解決パケットを投げ，応答が返ってくるまでの時間 (RTT) を測定するという方式で行われている．つまり，定点にて動的に計測を行う方式である．

dnsmon の計測地点を図 4.10 に示す．この図によれば，dnsmon 計測地点は欧州方面に集中している．これは RIPE が欧州地域の RIR であることに起因している．アジア方面は，日本に 1 台あるだけである．

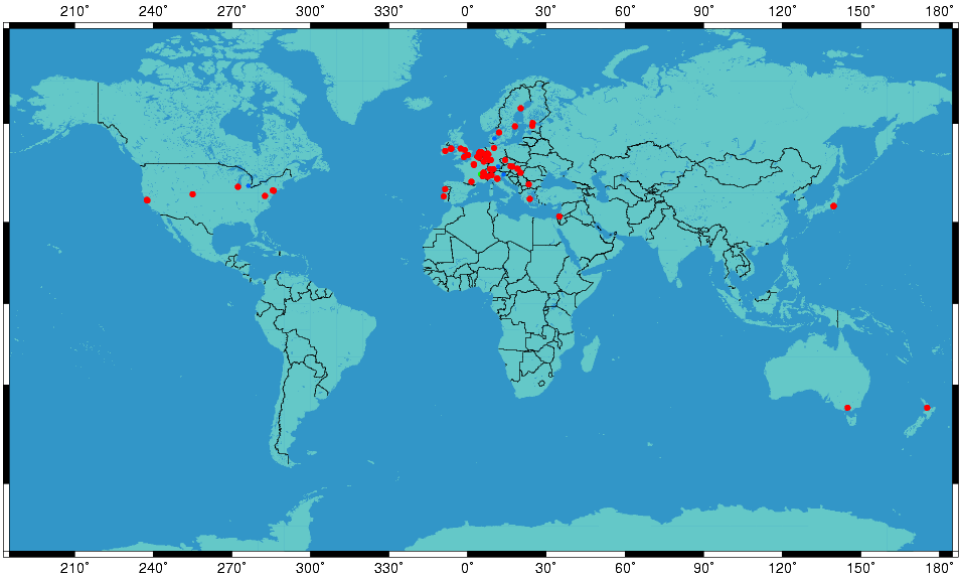


図 4.10: dnsmon 計測拠点

4.3.3 節　Skitter

図 4.11 に示すように，NeTraMet や dnsmon は，ユーザから計測対象の DNS サーバに往復するパケットで RTT 等を計測する方式である． 一方，計測対象となる DNS サーバにおいてトラフィックを複製して計測し，DNS サーバからユーザに往復するパケットにて RTT 等を計測し

た研究がある．CAIDA によって行われた，skitter[56] と呼ばれるプロジェクトである．

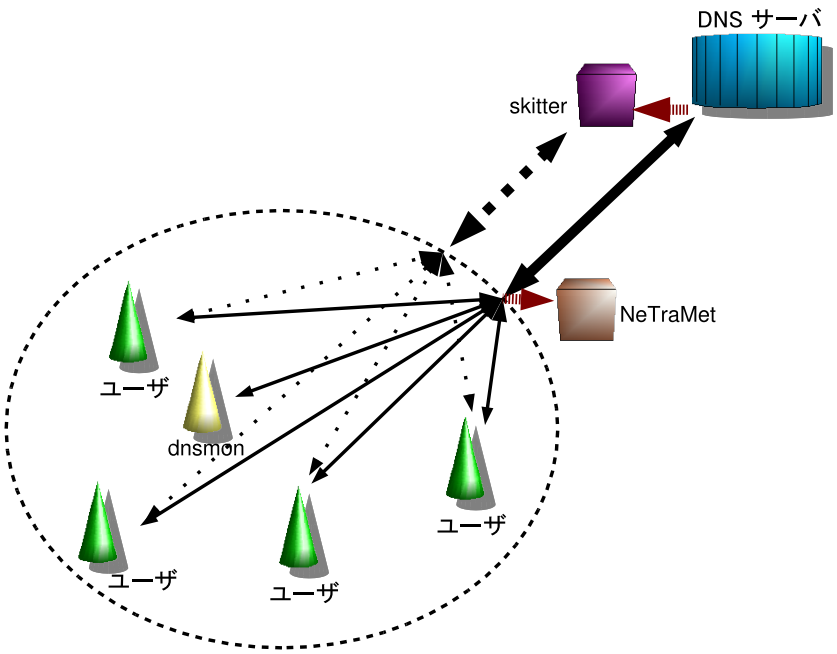


図 4.11: NeTraMet v.s. dnsmon v.s. skitter

DNS サーバに対して送られてきた名前解決クエリを監視し，名前解決クエリを送信してきたユーザに対して ICMP を送ることで，RTT を測定する．また，ユーザに対して traceroute を行い，ユーザまでの経路情報を記録する．

この方式も，NeTraMet と同じく DNS サーバへのトラフィックの複製を必要とする．また，計測対象となる DNS サーバの近隣に skitter ホストを設置する必要があるため，エニーキャストによる分散拠点が増加すると，拠点ごとに skitter のための設備が必要となる．

この skitter を利用して，各ルート DNS サーバに名前解決要求を送ってくるユーザのネットワーク的な所在地を分析するという研究がなされた．これは論文 [57] にて述べられている．

しかし，この skitter による計測は，すべてのルート DNS サーバを網羅していない．さらに，エニーキャスト分散拠点の増加によって，skitter を設置しているルート DNS サーバの割合が減少したため，現在は実質上 skitter による計測は中断している．

4.3.4 節　DNS サーバへのトラフィックの分析

DNS サーバに送られてくるトラフィックを，DNS サーバの手前で複製し，その中身を検証することによって，DNS へのトラフィック傾向を分析した研究である． この成果は，論文「ISP の DNS サーバの DNS トラヒックの解析 [58]」にて述べられている．

この研究によると，DNS サーバに送られてくる名前解決要求のほとんどが，無効な名前解決クエリであることがわかった．つまり，実際の名前解決には関係のない，設定の不備やユーザの間違い，もしくは乗っ取られたホストからの DDoS と思われる名前解決クエリがほとんどであることがわかった．

同様な研究は CAIDA によっても行われており，論文「DNS Measurements at a Root Server[59]」に述べられている． 同様に DNS サーバに送られる名前解決クエリのほとんどが不正なクエリであると結論づけている．

すなわち，DNS サーバは常に不正なクエリにさらされ続けており，その中で通常のサービスを支障無く提供する必要があることがわかる．

4.4 節　ルート DNS サーバの配置に関する研究

本節では，サーバの配置によるサービスの信頼性と到達性に関する関連研究 [60] について述べる．

インターネット上にて多くの人から利用されるサービスでは，耐障害性や負荷の分散といった観点から，複数台のサーバを分散して設置する例が見られる．この研究においては，サーバが分散して存在している場合の，サーバの配置状況とサーバの選択方法による，ユーザへの到達性の影響に関して述べている．

また，ルート DNS サーバをサービスの例として取り上げ，DNS サーバの配置状況によるサービスへの影響について考察を行った．

この研究では，サーバの選択のアルゴリズムを以下の 3 種類に分類し，それぞれの分析ならびに評価を行っている．

(1) Best Server アルゴリズム

RTT や サーバまでのホップ数で各サーバを評価し，もっとも良いサーバのみを利用する方式．

(2) Uniform アルゴリズム

全サーバをランダム，もしくはラウンドロビン方式などで均一に利用する方式.

(3) Reciprocal アルゴリズム

Best Server アルゴリズムと同じく，各サーバを評価するが，一定確率で最良サーバ以外のサーバも利用する.

効率的なサーバの配置方法は，これらアルゴリズムによって異なる． そこで，ルート DNS サーバの配置状況を例に，配置方法に関してシミュレーションによる考察を行った．その結果，(1) のアルゴリズムでは，サーバへの到達性が悪い場所に新たなサーバを設置することにより，一時的に到達性は向上する．しかし，新たなサーバに引き寄せられるユーザの数によっては，新たなサーバの負荷が一気に増大し，またユーザが別のサーバに振り分けられるといった，到達性の揺らぎが発生してしまう場合があることがわかった．また，(2) のアルゴリズムでは，新たなサーバを配置することで，1 台のサーバにかかる負荷は少し減るものの，ユーザにとっての到達性はあまり変化しないことがわかった．(3) のアルゴリズムでは，最良サーバ以外のサーバを利用する際の確率を変えるパラメータをうまく導入することによって，サーバを新たに設置した際に，負荷分散とユーザからの到達性両面において，最も効果がある結果となった.

5章　DNS 運用基盤分析システム

本章では，3 章にて述べた DNS の現在の運用状況と，3.4 節にて述べた分析モデルをふまえ，DNS 運用基盤分析システムの設計と実装，並びに分析手法に関して述べる．

本研究にて構築したシステムは，図 3.9 に示す通り，次の 3 つのサブシステムによって構成される．

- DNS 到達性調査システム
- DNS 委譲情報調査システム
- DNS 運用情報識別システム

それぞれのシステムについて詳細を述べる．

5.1 節　DNS 到達性調査システム

本節では，DNS 到達性調査システムの設計と実装，ならびに計測手法について述べる．

5.1.1 節　システムの設計

DNS サーバへの到達性計測のために作成した，dnsprobe というツールの設計と計測手法について述べる．

dnsprobe は，あらかじめ定義された DNS のサーバセットに対して DNS の名前解決要求を投げ，応答が帰ってくるまでの RTT を測定するソフトウェアである．

dnsprobe は，以下の点を目標として設計された．

- バックグラウンドにて軽く動作すること
- 多くの OS にて動作すること
- 一般ユーザ権限で動作すること
- 専用サーバや，特別な事前の準備を必要としないこと

dnsprobe の動作概要を図 5.1 に示す．この図に示すとおり，あらかじめ定義された DNS サーバセットに対して，dnsprobe ホストから，DNS サーバセットに含まれる全ての DNS サーバに対して名前解決要求を送信し，RTT を計測する．この図では，ホスト X, Y, Z が計測を行うホストである．計測結果は電子メールにて計測結果収集ホストに送信，もしくは dnsprobe を実行しているホストのローカルファイルに記録される． この際，dnsprobe を動作させたホストの IP アドレスも同時に記録される．

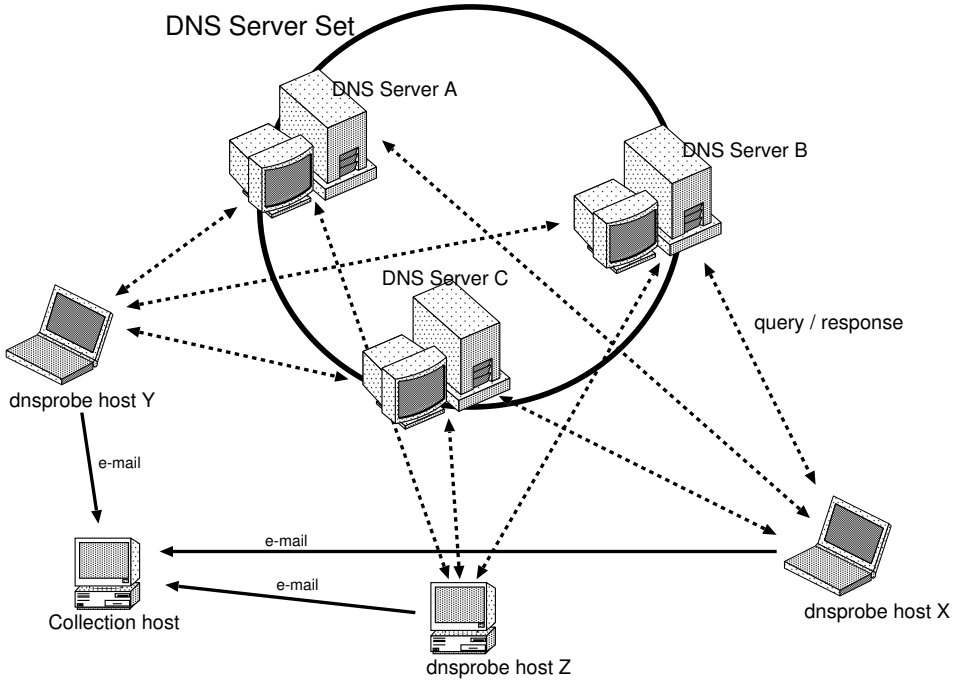


図 5.1: dnsprobe の動作概要

5.1.2 節　　システムの実装

本節では，dnsprobe の実装に関して述べる．dnsprobe は C 言語にて作成されており，その結果，各種 UNIX OS ならびに MacOS X，Microsoft Windows 上にて動作させることが可能である．dnsprobe を起動すると，デーモンとしてバックグラウンドで動作する．また，dnsprobe は一般ユーザ権限で実行できるため，様々なホストにて気軽に動作させることが可能である．

図 5.2 に，dnsprobe の動作フローを示す．

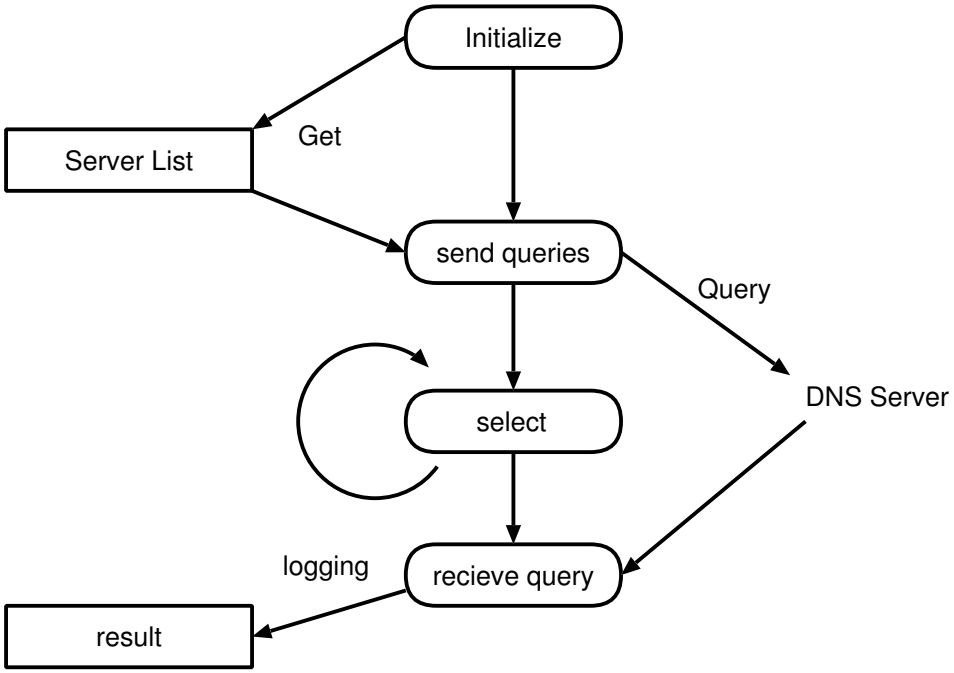


図 5.2: dnsprobe の動作フロー

dnsprobe は，あらかじめ定義された DNS サーバ群に対して，名前解決要求を送信し，その応答が帰ってくるまでの時間を記録する．この際，複数のサーバに対して並列に名前解決要求を送信することによって，調査時間を短縮している．

図 5.3 に，dnsprobe の動作結果例を示す．

この実行結果は，dnsprobe をルート DNS サーバに対して実行した結果である．図中の各行は，左から，応答が得られた時間，計測を行ったホストの IP アドレス，計測をおこなったホストのインタフェース名，計測対象となった DNS サーバ名，問い合わせを送信してから応答が得られるまでにかかった時間を示している．

この結果では，A.root-servers.net から M.root-servers.net までのルート DNS サーバへの名前解決応答に要した時間を示している．G.root-servers.net からの応答は，60 秒待っても得られなかったことを示している．これは，何らかの原因で名前解決要求もしくは応答が失われたことを意味している．

また，M.root-servers.net からは SERVFAIL というエラーメッセージが返答されている．DNS サーバからエラーメッセージが返答された場合には，このようにエラーメッセージの種類を表示する．

dnsprobe が判別するエラーメッセージは，以下の通りである．

```
1026688779 133.93.XX.1 eth1 A: rtt 210 ms
1026688785 133.93.XX.1 eth1 B: rtt 159 ms
1026688790 133.93.XX.1 eth1 C: rtt 250 ms
1026688793 133.93.XX.1 eth1 D: rtt 180 ms
1026688798 133.93.XX.1 eth1 E: rtt 110 ms
1026688803 133.93.XX.1 eth1 F: rtt 130 ms
1026688808 133.93.XX.1 eth1 G: timed out (60sec)
1026688812 133.93.XX.1 eth1 H: rtt 190 ms
1026688816 133.93.XX.1 eth1 I: rtt 268 ms
1026688820 133.93.XX.1 eth1 J: rtt 210 ms
1026688827 133.93.XX.1 eth1 K: rtt 264 ms
1026688830 133.93.XX.1 eth1 L: rtt 130 ms
1026688837 133.93.XX.1 eth1 M: SERVFAIL
```

図 5.3: dnsprobe 実行結果例

- FORMERR

  DNS サーバに送信した名前解決要求が不正なものであった場合に DNS サーバから返答されるエラーメッセージである.

- SERVFAIL

  何らかの理由により，解決要求した名前が解決できなかった場合に返答されるエラーメッセージである.

- NXDOMAIN

  名前解決要求した名前が，DNS サーバが保持している名前空間の中に存在しない場合に返答されるエラーメッセージである.

- NOTIMP

  要求された機能が，DNS サーバにまだ実装されていなかった場合に返答されるエラーメッセージである.

- REFUSED

  何らかの理由により，名前解決要求が拒否されたことを示すエラーメッセージである.

標準の設定では，dnsprobe は 5 分に 1 回計測を行い，結果を報告する.

次に，これまで述べてきた dnsprobe を用いて，DNS サーバへの到達生を調査する場合の調査手法について述べる．

5.1.3 節　ダイヤルアップによる計測手法

dnsprobe は，ホスト上にて動作させるソフトウェアであるため，基本的には，計測を行いたい拠点にホストが存在し，そのホストを利用できる権限あることが前提となる．

しかし，ある DNS サーバセットに対する世界中からの到達性と到達時間を計測するためには，世界中の多くの地点から dnsprobe を行う必要がある．すなわち，計測を行いたい地点に利用可能なホストが存在しない限り，計測は不可能となる．

この問題の解決策としては，dnsprobe をフリーソフトウェアとして公開することによって協力者を募り，数多くの地点からの結果を集めるという方法が考えられる．確かにこの方法にてある程度の協力者を得ることは可能である．しかし，必ずしも計測したい地点に協力者が現れるとは限らず，計測地点が偏ってしまう場合もある．

そこで本システムでは，測定するにあたって次の 2 種類の測定手法を採用した．

(1) native probe

(2) dialup probe

native probe とは，測定を行う地点にホストが存在し，そのホストにて dnsprobe を実行する手法である．一方，dialup probe とは，測定を行う地点に存在する Internet Service Provider(ISP) のアクセスポイントにダイヤルアップ PPP 接続を行い，ダイヤルアップ接続したホストにて dnsprobe を実行する手法である．それぞれの実行形態を図 5.4 にまとめる．

dialup probe を利用することによって，利用できるホストが存在しない計測地点からも，手軽に計測を行うことが可能となる．しかし，native probe と dialup probe は明らかに計測手法が異なるため，dialup probe にて計測した結果と native probe にて計測した結果をそのまま比較することはできない．

多地点からの計測結果を基に，DNS サーバ群への到達生を分析するためには，native probe と dialup probe の計測結果を同様に扱えるように補正することが必要となる．次に，その補正方法について述べる．

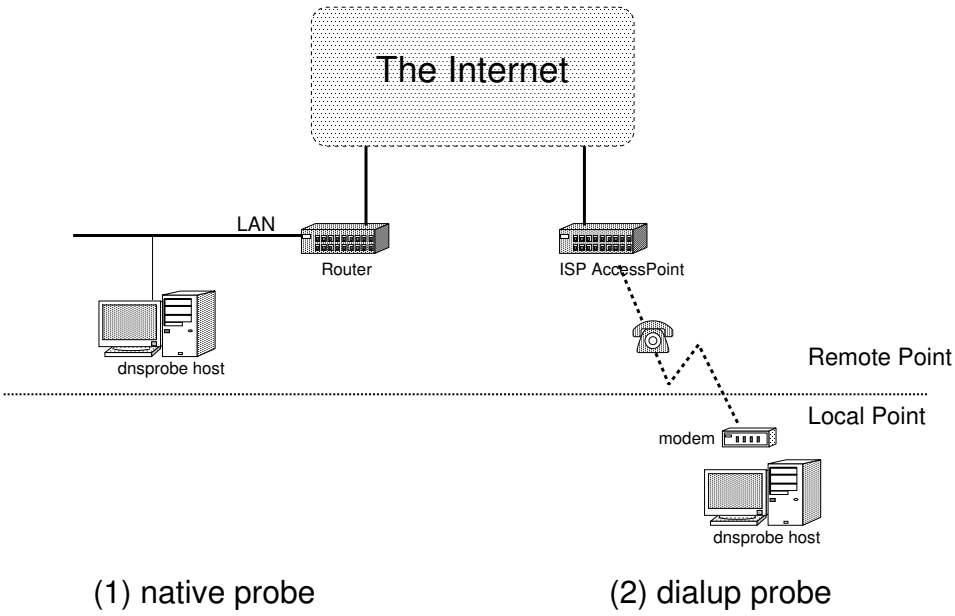


図 5.4: native probe と dialup probe

5.1.4 節　測定値の補正

dialup probe にて測定したデータと native probe にて測定したデータを同列に扱うためには，dialup probe にて収集したデータを補正する必要がある．

このため，同地点から両方の方法にて計測したデータをもとに，測定値を補正する際の補正方法について述べる．

まず，native probe と dialup probe 結果を比較するために，米国 California 州 Los Angeles 市のデータセンタ内に，LAN によるインターネットへの接続性を有しており，かつモデムによるダイヤルアップ接続も受け付けているホストを用意した．ルート DNS サーバ群に対して，このホストから native probe による測定と dialup probe による測定を行った．その結果を図 5.5 に示す．またこの結果を累積分布関数 (CFD) を用いて示した結果を，図 5.6 に示す．縦軸は累積分布を示し，横軸は応答が得られるまでの RTT を示す．なお，dialup probe は，神奈川県から V.34 方式をサポートしたアナログ回線モデムを用いて行った．

ルート DNS サーバ群は，a.root-servers.net から m.root-servers.net までの 13 個のサーバによって運用されている．このグラフでは，観測地点である Los Angeles 市のデータセンターから，それぞれ 13 種類の ルート DNS サーバまでの応答時間を示している．なお，これ以降本論文にて示されるグラフにおいて，A から M までのアルファベットはそれぞれ a.root-servers.net から m.root-servers.net までのルート DNS サーバに対応するものとする．

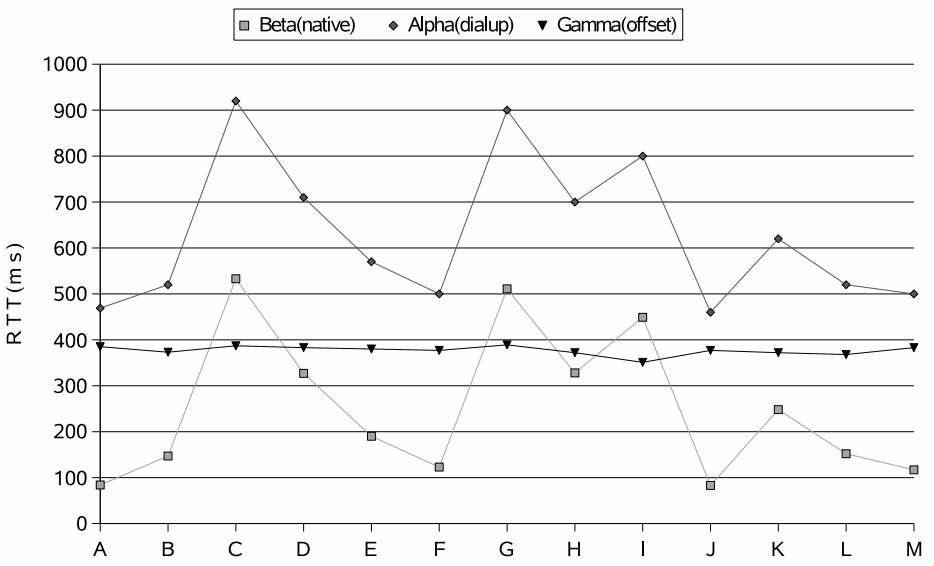


図 5.5: native probe と dialup probe の比較 (Los Angeles 市データセンタ内)

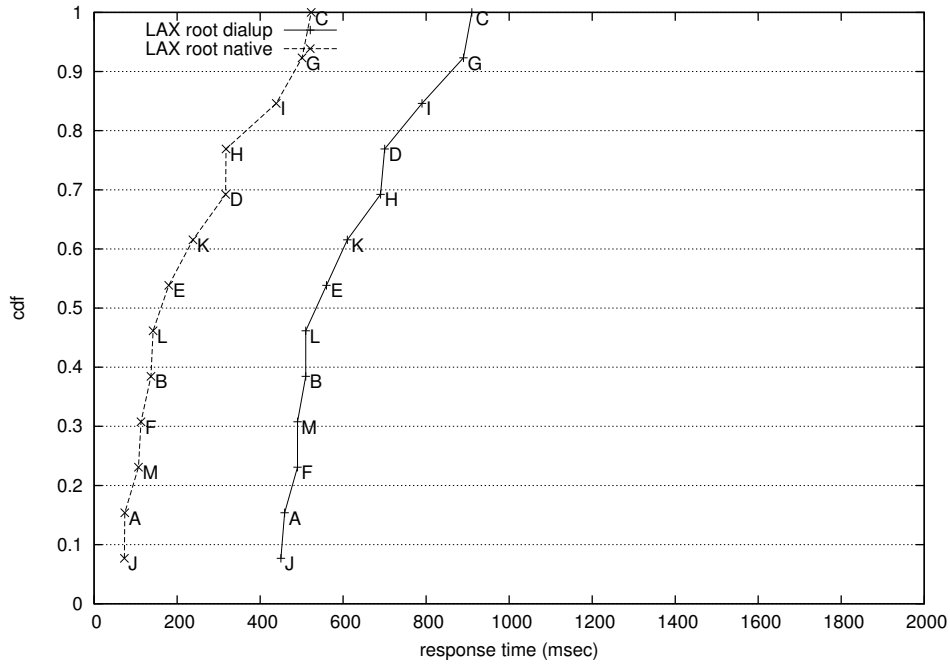


図 5.6: native probe と dialup probe の比較 (Los Angeles 市データセンタ内) - CFD

また，図 5.7 と図 5.8 に，同様にして慶應義塾大学湘南藤沢キャンパス内に，LAN の接続性を持ち，かつモデムによるダイヤルアップ接続を受け付けるホストを設置して測定を行った結果を示す.

これら 2 つの測定結果にて，ルート DNS サーバ n に対する dialup probe の RTT を α_n, native probe の RTT を β_n とすると，native probe と dialup probe の差 γ_n は，$\gamma_n = \alpha_n - \beta_n$ となる．この結果にて導かれる γ_n は図 5.5, 5.7 に示すように，ほぼ一定の値を示す．この他にも，日本国内で ADSL にて接続性を有し，かつモデムによるダイヤルアップ接続を受け付けるホストを設置して同様な測定を行った．その結果も同様に，γ はほぼ一定の値を示した．したがって，dialup probe の結果に対して補正を行うことにより，native probe の結果と同等に扱うことができると言える.

5.1.5 節　　補正値

dialup probe にて行った計測結果を補正するにあたって，補正値 γ をどのように決定するかという問題がある．本研究では，補正方法として，以下の 2 つの方法を用いた.

(1) dnsprobe 実行ホストから，ダイヤルアップサーバまでの RTT を補正値 γ とする

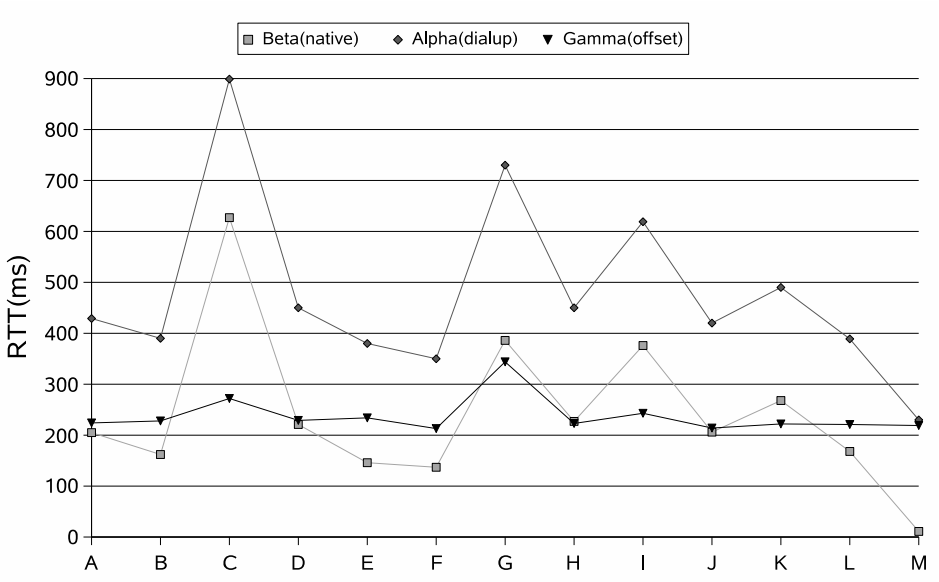


図 5.7: native probe と dialup probe の比較 (慶應義塾大学内)

(2) 基準 DNS サーバ群を定義し，dnsprobe を実行したホストから最も近い基準 DNS サーバへの RTT を補正値 γ とする

(1) の方法は，ISP のダイヤルアップサーバが設置された地点から，測定対象となる DNS サーバセットまでの RTT 値を算出することができる．すなわち，dialup probe において，ダイヤルアップ区間によって生じる RTT の差のみを除くための補正方法である．dialup probe に対してこの補正方法を用いると，ISP のダイヤルアップサーバがある地点において native probe を行った場合に近似した値が得られる．

しかし，ISP のダイヤルアップサーバは，ISP の中核となるバックボーンから数ホップ離れた位置に存在している場合がある．つまり，ISP 内部のネットワークトポロジは ISP 毎に異なるため，ISP のバックボーンからダイヤルアップサーバまでのネットワーク的な距離は同一条件であるとは言えない．

そこで，計測地点によるネットワーク構成の差異を吸収するため，基準 DNS サーバを用いて補正を行う方式を考案した．これが (2) の方法である．なお，基準 DNS サーバの定義については，5.1.6 節にて述べる． (2) の方法では，計測を行った地点から最もネットワーク的に近くにある基準 DNS サーバまでの RTT を補正値として用いる．

すなわち，図 5.9 に示すように，dnsprobe を行うホストからダイヤルアップサーバまでの RTT を RTT(a)，dnsprobe を行うホストから最も近い基準 DNS サーバまでの RTT を RTT(b) とす

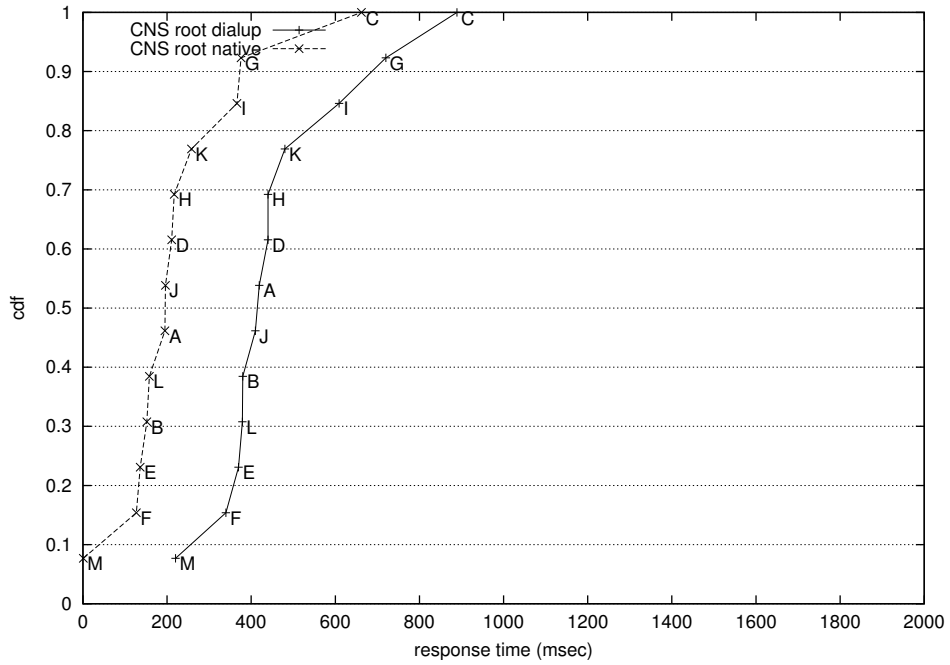


図 5.8: native probe と dialup probe の比較 (慶應義塾大学内) - CFD

ると，(1) の方法では補正値は $\gamma = median(RTT(a))$ で，(2) の方法では，$\gamma = median(RTT(b))$ である．

なお，この場合の RTT(a) の母集団は，dnsprobe にて測定を行う度にダイヤルアップサーバまで ping を行い，その RTT を記録したものである．RTT(b) の母集団は，最も近い基準 DNS サーバまでの dnsprobe の応答結果である． また，補正値として中間値を用いているのは，突発的なネットワークの不安定による極端な値の振れの影響を除くために，平均値ではなく中間値を用いることとした．

つまり，DNS のパフォーマンスを測定するにあたって，同一測定地点からの複数の DNS サーバ群に対する測定結果を比較する場合や，測定地点からある DNS サーバ群までの RTT，すなわち応答が得られるまでの時間を調査したい場合には，補正法 (1) を利用すればよい．

一方，補正法 (2) は，複数の測定地点から，単一の DNS サーバ群に対しての測定結果を比較したい場合に用いる補正である．すなわち，ダイヤルアップ先の ISP 内部の構成の差異を補正するための方法である．また，ダイヤルアップに限らず，異なる測定地点からの測定結果を比較する際に，測定地点のネットワークの状況による測定結果の揺らぎを減らしたい場合に有効な手法である．

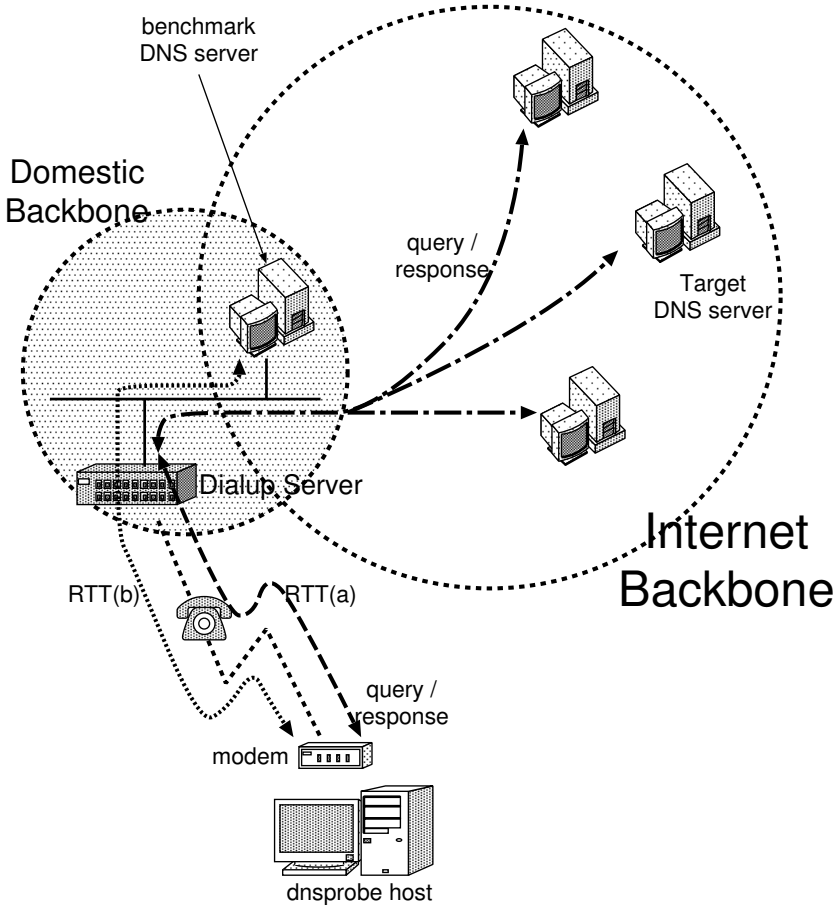


図 5.9: Dialup probe の補正方法

5.1.6 節　基準 DNS サーバの定義

本節では，補正に必要な基準 DNS サーバの定義にて述べる．前述の通り，基準 DNS サーバを用いた補正は，dialup probe を行った場合における，計測地点周辺のネットワークの差異を吸収するために利用される． 従って，基準 DNS サーバとして利用される DNS サーバは，どのような計測地点からも等しく到達できるような DNS サーバが好ましい．しかし，現実としてそのような DNS サーバは存在しない． そのため，本システムにおける計測では，次の条件を満たす DNS サーバを，基準 DNS サーバとして用いた．

(1) どのような計測地点の周辺にも存在している DNS サーバ

(2) どのような計測地点においても頻繁に利用される DNS サーバ

すなわち，(1) は,「全世界に分布している DNS サーバ」を意味し，(2) は「公共性の高い DNS サーバ」を意味する．そこで，このような基準 DNS サーバを選出するために，公共性の高い DNS サーバの調査と，全世界に分布している DNS サーバの調査を行った．

5.1.7 節　公共性の高い DNS サーバ

本節では，基準 DNS サーバを決定するために必要な調査として行った，公共性の高い DNS サーバの調査について述べる．一般的に公共性の高い DNS サーバとして挙げられるのが，ルート DNS サーバや gTLD DNS サーバ，ccTLD DNS サーバなどである．

これらの DNS サーバは，名前解決を行う際に頻繁に参照される．また，ある特定の組織の名前空間を有するための DNS サーバではなく，国別ドメインや属性ドメインといった，広く利用される名前空間を保持するための DNS サーバである．そこで，これらの公共性の高いと思われる DNS サーバが，実際にどのように利用されているのか調査した．

この調査では，dnstop [61] というツールを用いて，UDP でポート 53 宛に送られたパケットを抽出し，宛先となっている DNS サーバのアドレスを記録する，という方法にて解析した．

表 5.1 に，日本国内のある大学から上流のインターネットに送出された，DNS 名前解決に関するパケットの統計を示す．なお，この解析は実際のトラフィックから DNS 名前解決に関する 1,000,000 パケットを抽出して行われた．また，プライバシーの観点から，IP アドレスの 3 オクテット目と本研究の測定に直接関係のない DNS サーバの名前を一部伏せて示されている．

パケットが送信された DNS サーバの，上位 40 台までを示した．1 位から 3 位までの DNS サーバには飛び抜けてパケットが集中している傾向にあり，測定時において，ウィルスの活動に関連した問い合わせや Denial of Service(DoS) による攻撃である可能性がある．上位 40 サーバのうち，ルート DNS サーバ，gTLD サーバ，ccTLD サーバの割合は，表 5.2 の通りである．

表 5.2: 上位 40 サーバ中における ルート/gTLD/ccTLD DNS サーバの数

種別	数
ルート DNS サーバ	1
gTLD サーバ	13
JP ccTLD サーバ	5

ルート DNS サーバは，m.root-servers.net のみ上位 40 サーバの中に登場している．これは，論文 [60] にて述べられている，DNS サーバ選択の結果によって，13 個のルート DNS サーバのうち m.root-servers.net が多く選ばれているためと考えられる．広く利用されている DNS サーバ実装である BIND[1] では，あるゾーン (名前空間) を保持している DNS サーバが複数ある場合，それら DNS サーバに対する応答の RTT によって，重み付けを行い，利用するサーバを選択する．

13 個のルート DNS サーバのうち，データ収集時点で日本国内に存在していたのは m.root-servers.net のみであった．他のルート DNS サーバは米国もしくはヨーロッパ方面に存在する．したがって，測定地点からもっともネットワークトポロジ的に近かった m.root-servers.net に対して問い合わせが多く発生したと考えられる．

gTLD サーバは，全部で 13 サーバ存在している．今回の測定においては，全 13 サーバが上位 40 サーバ中に登場している．また，ccTLD サーバも 7 サーバ登場しており，そのうち 5 サーバが .jp の ccTLD サーバである．今回の測定時においては，.jp の ccTLD サーバは全部で 6 サーバ存在しており，そのうち 5 サーバが上位 40 サーバ中に登場している．

以上の結果から，DNS 名前解決パケットは，ルート DNS サーバや gTLD DNS サーバ，ccTLD DNS サーバといった DNS サーバ群に対して多く送出されていることがわかった． これは，ルート DNS サーバや，gTLD DNS サーバ，ccTLD DNS サーバといった DNS サーバが公共性の高いサーバであることを示している．

5.1.8 節　基準 DNS サーバの選定

次に，公共性が高い DNS サーバ群の分布状況を調査した．

この調査は，調査対象となる DNS サーバが，物理的にどの国に存在しているかを調べることによって，それぞれのサーバ群の物理的な分散状況を調査することを目的として行った．具体的には，インターネットにて宛先への経路を調査するためのコマンドである traceroute コマンドと，IP アドレスから国名を調査することのできる GeoIP [62] を用いて，2002 年 6 月と 2003 年 11 月に行った．

その結果，ルート DNS サーバは，13 台中 10 台が米国内に存在しており，gTLD サーバも，13 台中 8 台が米国内に存在していることがわかった．どちらのサーバ群も，分布が米国に偏っていると言える．

[1] http://www.isc.org/

一方，ccTLD サーバは，243 の地域もしくは国のトップレベルドメインを持つ ccTLD が，667 台に分散して登録されている． 2003 年 11 月の分布の様子を図 5.10 に示す．この図では，自国他国関係なく，なんらかの国別ドメインを保持する ccTLD サーバが，7 台以上 (ccTLD サーバ全体の 1%以上)，国内に存在している国を図示した．この結果からも，米国内だけではなく，ヨーロッパ方面，アジア方面，ロシア，オセアニア方面と多方面にわたって存在していることがわかる．

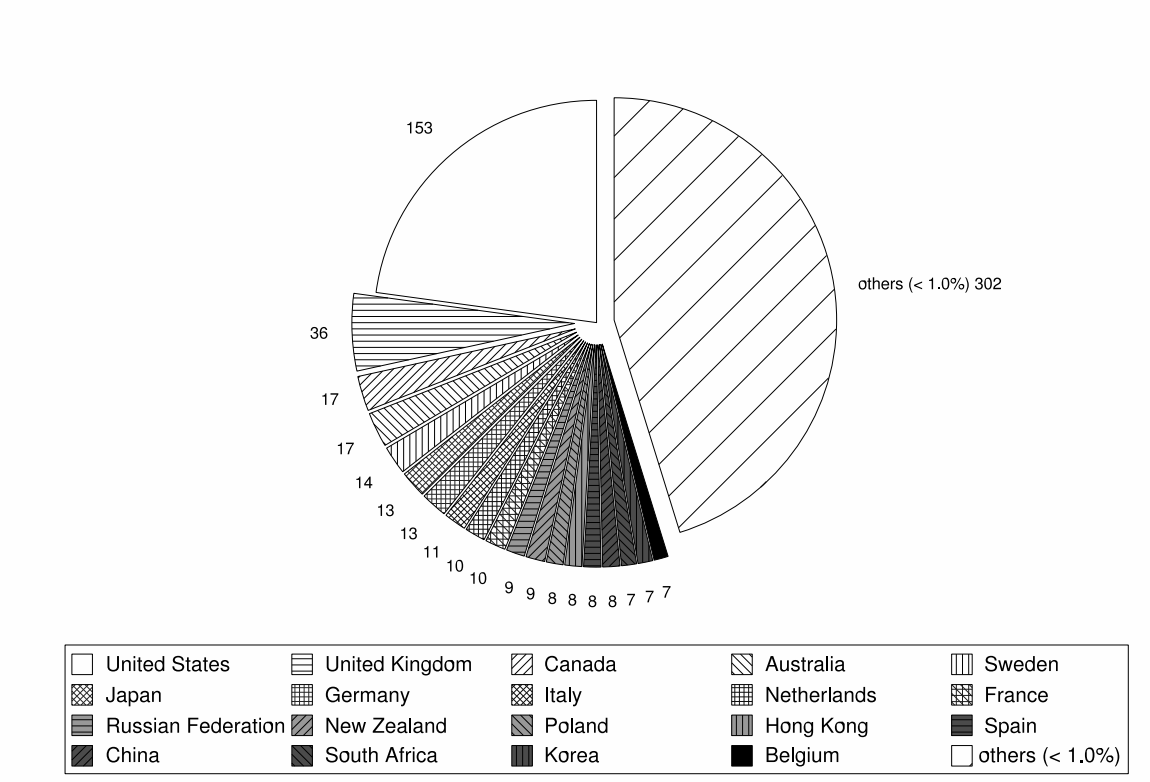


図 5.10: ccTLD サーバの分布状況

また，図に示されていない 1% 未満，すなわち 6 台以下，国内に ccTLD サーバが存在している国は，138 ヶ国存在した．この結果から，ccTLD を持つ半分以上の国や地域が，なんらかの ccTLD サーバを国内に有していることがわかる．この場合，自国の ccTLD サーバを自国内におかず，他国の ccTLD サーバを自国内にホスティングしている場合も考えられる．しかし，多くの場合には，自国の ccTLD サーバを自国内に 1 台も置かずに他国の ccTLD サーバのみをホスティングしているとは考えにくい．表 5.1 の結果にも表れている通り，自国の ccTLD ゾーンは，自国内から検索されることが多いため，少なくとも 1 台は自国内に設置する場合が多いと考えられる．

以上の調査結果により，頻繁に問い合わせが行われる公共的な DNS サーバの中では，ccTLD サーバが一番世界中に分布して存在していることがわかった．

しかし，ccTLD サーバはルート DNS サーバや gTLD サーバと違い，同一の名前空間を保持しているわけではない．各国の国別のトップレベルドメインを保持しているサーバを総称して ccTLD サーバと呼んでいる．本来なら，同一の名前空間を保持するサーバ群で，全世界に万遍なく存在している DNS サーバ群が望ましいが，そのような DNS サーバ群は存在しない．そこで，本研究では，ccTLD サーバを基準 DNS サーバとして用いることとした．

その理由は，ccTLD サーバ群は，頻繁に参照される公共的な DNS サーバであり，世界中にできる限り分散して存在している DNS サーバ群として現存する DNS サーバ群の中で最良であると考えたからである．そこで，本研究では，ccTLD サーバ群を基準 DNS サーバとして用い，測定結果の補正・評価を行った．

もし自国内に ccTLD サーバが全く存在しない場合には，ネットワークトポロジ的に最も近い ccTLD サーバが，その測定地点から見て最も近い公共的な DNS サーバということになる．

具体的には，測定地点から最もネットワークトポロジ的に近い 基準 DNS サーバを測定値補正に用い，基準 DNS サーバ群への到達性測定結果を，到達性の傾向評価に用いた．

もちろん，さらに基準 DNS サーバとして適している DNS サーバ群があれば，その DNS サーバ群を基準 DNS サーバとして用いることが可能である．しかし，その場合も本研究の手法は変わることなく適用可能である．

5.2 節　DNS 委譲情報調査システム

本節では，DNS 委譲情報調査システムの設計と実装について述べる．

5.2.1 節　システムの設計

本システムでは，DNS の信頼性・耐障害性を分析するため，以下の点を目標としてシステム設計を行った．

- DNS の委譲ツリー全体の状態を監視する
- DNS のバージョン情報を監視する

- DNS 管理者に対して，注意を促す

上記の目標を実現するためには，以下の情報が必要である．

- DNS の委譲木構造
- DNS サーバ毎の IP アドレス
- DNS サーバ毎のバージョン情報
- DNS サーバ毎の，所有組織と管理者に関する情報

DNS の委譲木をルート DNS サーバからたどっていくことにより，委譲木の正確性を確認することが可能となる．また，委譲木をたどっていく過程において発見された個々の DNS サーバに関して，IP アドレスや管理者情報，サーバのバージョン等の情報を収集する．

そこで，上記の情報を効果的に収集するためのシステムとして，図 5.11 に示す DNS 情報収集システムを設計した．このシステムは，次の 4 つのモジュールにて構成される．

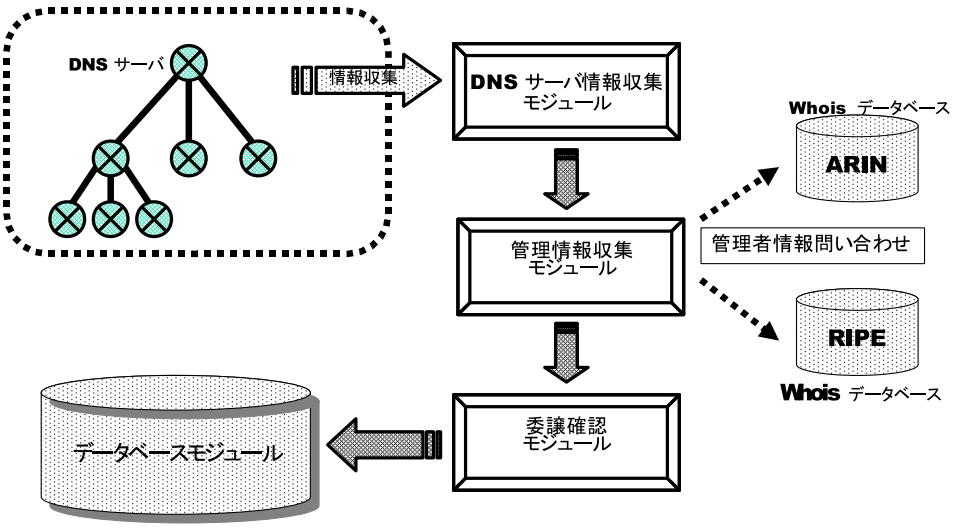


図 5.11: DNS 委譲情報収集システム

1. DNS サーバ情報収集モジュール
2. 管理情報収集モジュール
3. 委譲確認モジュール

4. データベースモジュール

次に，各モジュールの役割について説明する．

1. DNS サーバ情報収集モジュール

   DNS の委譲木をルート DNS サーバからたどることによって，あるゾーンを管理する DNS サーバの FQDN と IP アドレス，委譲の有無，さらに DNS サーバのバージョンを調査するモジュールである．具体的には，ゾーン中の SOA(Start of Authority) レコード，NS レコード，ゾーンを保持する DNS サーバの名前と IP アドレス，DNS サーバのバージョン番号を記録する．そして，ゾーン中にて委譲が行われている場合には，委譲先も記録する．委譲があった場合には，委譲先のサーバにて同様なデータ収集を行う．これを再帰的に繰り返し，末端の DNS サーバまでの委譲情報を収集する．

2. 管理情報収集モジュール

   DNS サーバの所有組織と管理者に関する情報を収集するモジュールである．DNS サーバ情報収集モジュールにて収集した，ゾーンを保持する DNS サーバの FQDN と IP アドレス情報をもとに，RIR[2] の whois データベースに問い合わせを行い，情報を収集する．

3. 委譲確認モジュール

   DNS 委譲の正確性を判定するモジュールである．上位の DNS サーバから委譲されているゾーンに関して，その NS RR にある DNS サーバが，本当に委譲先のゾーンの権威をもつサーバとなっているかを確認する．また，委譲元の NS RR に列挙されている DNS サーバと，委譲先のゾーン先頭に列挙される DNS サーバが一致するかを確認する．

4. データベースモジュール

   上記の 3 つのモジュール処理を経て生成された，各 DNS サーバとゾーン，さらにゾーンの委譲に関するデータを格納するデータベースである．

5.2.2 節　　システムの実装

前述の設計に基づき，情報収集システムを実装した．情報収集システムの実装は，次に述べる環境にて行った．

[2]Regional Internet Registries の略．主に APAN, ARIN, RPIE を示す

- Sparc Station 20
- RedHat Linux 6.2 (kernel-2.2.15)
- SCSI DISK 20G * 2

各モジュールの実装について述べる．

- DNS サーバ情報収集モジュール

  DNS サーバ情報収集モジュールは，perl にて作成した．DNSwalk[63] というソフトウェアに改造を加え，以下の情報を抜き出すよう作成した．

  - ゾーンの SOA，MNAME，RNAME，NS，委譲の有無
  - DNS サーバの FQDN，IP アドレス，バージョン番号

- 管理情報収集モジュール

  本モジュールも perl にて作成した．DNS サーバ情報収集モジュールが集めた DNS サーバの IP アドレスをもとに，適切な RIR の whois データベースに問い合わせを行い，IP アドレスに対応する管理組織と管理者のデータを取得する．そして，DNS サーバ情報収集で得た情報と，whois データベースから取得した管理組織と管理者の情報をもとに，データベースモジュールに送るデータを作成する．

- 委譲確認モジュール

  本モジュールも perl にて作成した．以下の 2 つの条件を満たす場合に，正確に委譲がなされていると判断するよう作成した．

  1. 委譲先の DNS サーバに対して問い合わせを出し，権威付き回答 (Authoritative Answer) が返答される
  2. 委譲元の NS レコードに列挙されている DNS サーバと，委譲先のゾーン NS レコードに列挙されている DNS サーバが一致する

- データベースモジュール

  データベースモジュールは，rwhois[64] サーバに改造を加えて作成した．データベースサーバとして rwhois サーバを選択した理由は，次の 2 点である．1 点は，普遍性である．通常の rwhois もしくは whois クライアントを用いて情報検索が行えるという利点がある． も

う 1 点は，拡張性である．本システムを分散して設置する場合，rwhois プロトコルにて規定されている「Referral(参照)」という仕様により，容易にデータの分散を行うことが可能である．

以上の 4 つのモジュールを組み合わせ，DNS 情報収集システムを構築した．

5.2.3 節　委譲情報調査

本節では，委譲情報調査システムを用いて調査を行う方法について述べる．

本研究では，逆引きゾーンに対して委譲情報の調査を行った．これは，DNS の逆引きをサーバやホストの認証として利用している場合が多く，また 1 つの IP アドレスには 1 つの逆引きの名前を設定するのが通例となっているため，DNS の完全性を調査するためには，まず逆引きを調査する方が良いと考えたためである．

まず，DNS サーバ情報収集モジュールにて，逆引き委譲木の調査が行われた．これは前述の環境を用いて，約 1 週間を要した．さらに，この調査データをもとに，管理情報収集モジュールにて，DNS サーバ単位，ならびにゾーン単位における管理情報の作成を行った．これにも約 1 週間を要し，データは約 4GB の容量となった．図 5.12 に，10.in-addr.arpa 逆引きゾーンに対して作成された管理情報の例を示す．通常の whois 情報に加え，SOA や M-Name，Serial，Parent-Zone，Name-Server といった情報が付加されている．

次に，委譲確認モジュールにおいて，各ゾーンから委譲されているゾーンに対して，上位ゾーンの NS レコードと，下位ゾーンの NS レコードが一致しているかの調査を行った．

最後に，管理情報と，委譲の判定結果を合わせてデータベースに入れ，Web から検索できるようインタフェースを構築した．

このようにして，DNS の委譲情報調査は行われた．

```
ID:             10.in-addr.arpa.
Netblock:       10.0.0.0 - 10.255.255.255
SOA:            blackhole.isi.edu.
M-Name:         bmanning@isi.edu.
Serial:         19971502
Org-Name:       IANA
Street-Address: Internet Assigned Numbers Authority
Street-Address: Information Sciences Institute
Street-Address: University of Southern California
Street-Address: 4676 Admiralty Way, Suite 330
City:           Marina del Rey
State:          CA
Postal-Code:    90292-6695
Country-Code:   US
Contact-Name:   Internet Assigned Numbers Authority
Contact-Email:  iana@IANA.ORG
Phone:          (310) 823-9358
Parent-Zone:    10.in-addr.arpa.
Name-Server:    ns1.10.in-addr.arpa.
```

図 5.12: 10.in-addr.arpa ゾーン管理情報

表 5.1: 日本国内のある大学から出された DNS パケットの統計

宛先アドレス	パケット数	%	DNS サーバ名	種別
195.149.XXX.59	67072	6.7	XXX.camcontacts.net	-
66.70.XXX.20	65971	6.5	-	-
66.70.XXX.175	63668	6.3	-	-
200.206.XXX.201	17483	1.7	XXXXXXterra.com.br	-
202.12.XXX.33	12350	1.2	m.root-servers.net	root
218.16.XXX.164	10348	1.0	-	-
150.100.XXX.3	9086	0.9	ns-jp.sinet.ad.jp	ccTLD(jp)
219.153.XXX.10	8359	0.8	-	-
172.29.XXX.16	8078	0.8	-	-
61.120.XXX.100	7124	0.7	ns-jp.nic.ad.jp	ccTLD(jp)
192.26.XXX.30	6742	0.7	c.gtld-servers.net	gTLD
192.48.XXX.30	6674	0.7	j.gtld-servers.net	gTLD
192.55.XXX.30	6347	0.6	m.gtld-servers.net	gTLD
192.41.XXX.30	6336	0.6	l.gtld-servers.net	gTLD
211.14.XXX.10	6177	0.6	XXXXXX.yahoo.co.jp	-
12.1.XXX.251	5341	0.5	-	-
12.1.XXX.253	5295	0.5	-	-
170.66.XXX.3	4944	0.5	XXXX.bb.com.br	-
170.66.XXX.2	4856	0.5	XXXX.bb.com.br	-
192.31.XXX.30	4773	0.5	d.gtld-servers.net	gTLD
192.12.XXX.30	4643	0.5	e.gtld-servers.net	gTLD
202.96.XXX.90	4576	0.5	-	-
213.193.XXX.137	4548	0.5	XXX.multimania.net	-
192.35.XXX.30	4423	0.4	f.gtld-servers.net	gTLD
202.232.XXX.34	4384	0.4	ns0.iij.ad.jp	ccTLD(jp)
192.33.XXX.30	4381	0.4	b.gtld-servers.net	gTLD
192.54.XXX.30	4334	0.4	h.gtld-servers.net	gTLD
192.5.XXX.30	4315	0.4	a.gtld-servers.net	gTLD
192.52.XXX.30	4224	0.4	k.gtld-servers.net	gTLD
202.12.XXX.131	4193	0.4	XXX.apnic.net	-
192.43.XXX.30	4117	0.4	i.gtld-servers.net	gTLD
202.12.XXX.131	3865	0.4	ns0.nic.ad.jp	ccTLD(jp)
202.229.XXX.120	3700	0.4	XXXXXXX.yahoo.co.jp	-
213.193.XXX.130	3594	0.4	XX.multimania.net	-
165.76.XXX.98	3372	0.3	dns0.spin.ad.jp	ccTLD(jp)
65.57.XXX.120	3294	0.3	XXX.hsphere.cc	-
65.57.XXX.100	3272	0.3	XXX.hsphere.cc	-
192.26.XXX.32	3255	0.3	c3.NSTLD.COM	ccTLD
202.12.XXX.140	3254	0.3	karashi.apnic.net	ccTLD
92.5.XXX.32	2796	0.3	a3.NSTLD.COM	gTLD

5.3 節　DNS 運用情報識別システム

本節では，DNS の運用情報ならびに個体識別情報を調査するために作成した，運用情報識別システムの設計について述べる．

5.3.1 節　システムの設計

3.2 節にて述べたとおり，現在の DNS には，エニーキャストという技術が導入され始めている．これによって，全く同じ DNS サーバが複数地点に存在することとなり，ユーザがサービスを受ける場合，実際にはどの DNS サーバからサービスを受けているのかを意識せずに利用する．これによって多くのユーザから利用される DNS サーバの負荷を分散することが可能となり，ユーザに対してより安定した到達性を提供することができる．

しかしその一方，どのエニーキャスト DNS サーバからサービスを受けているのか把握できなくなる．そのため，ユーザからの報告によってエニーキャスト DNS サーバの障害が判明した場合，どのエニーキャスト DNS サーバに障害が発生したのかを特定することが難しい． そのため，平常時からどのエニーキャスト DNS サーバがどのサービス範囲を担当しているのかを把握しておく必要がある．

そこで，DNS 運用情報識別システムが必要となる．DNS 運用情報識別システムは，次の点を目標として設計された．

- 平常時のエニーキャスト DNS のサービス範囲を調査する
- DNS サーバ単位の運用情報を蓄積し，検索できるようにする

委譲の目標を実現するため，DNS 運用情報識別システムは，DNS 到達性調査システム並びに DNS 委譲情報調査システムと連携してデータを構築する．図 5.13 に示す通り，DNS 運用情報識別システムは 2 つのモジュールから構成される． 以下に，そのモジュールに関して述べる．

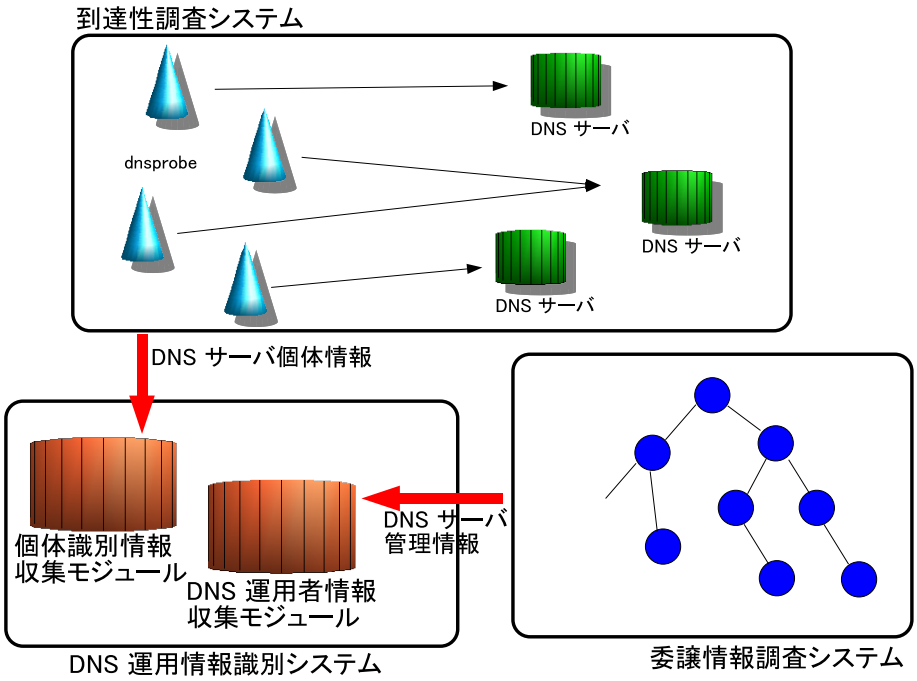


図 5.13: DNS 運用情報識別システム

- DNS 運用者情報データベースモジュール

  DNS 委譲情報調査システムにて収集した DNS 管理組織ならびに管理者データをもとに，DNS サーバ単位にデータを整理し，次に述べる個体識別情報収集モジュールによる個体識別情報とともにデータベースを作成するためのモジュールである．

- 個体識別情報収集モジュール

  DNS サーバの個体情報を収集するためのモジュールである．3.2 節にて述べたエニーキャスト DNS サーバを識別するためのモジュールである．

5.3.2 節　　システムの実装

本節では，5.3.1 節にて述べた，DNS 運用情報識別システムを構成する 2 つのモジュールの実装について述べる．

- DNS 運用者情報データベースモジュール

DNS 委譲情報調査システムにて収集した DNS 管理組織ならびに管理者データは，rwhois のデータ形式にて蓄積されている．しかし，本研究で用いた rwhois 実装のデータ形式がテキストであったため，データ更新や削除，追加といった加工にあまり適した形式ではない．そこで，rwhois のデータを Structured Query Language(SQL) 形式に変換し，MySQL[65] データベースサーバを用いてデータベースを構築した．

この際，個体識別情報収集モジュールから得た個体識別情報とあわせてデータベースを作成した．また，データを SQL データベースに蓄積したことで，Web やその他のクライアントから，標準的な SQL のインタフェースにて情報を検索することが可能となった．

図 5.14 に，本システムにて作成した Web インタフェースを示す．

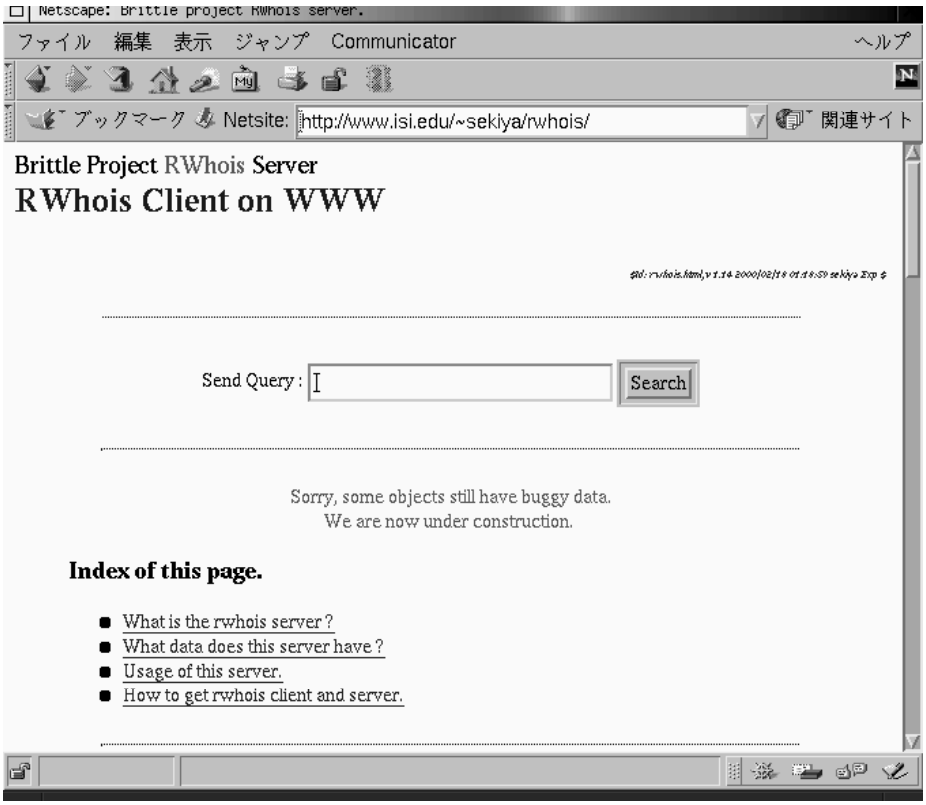


図 5.14: DNS 運用情報検索 Web インタフェース

- 個体識別情報収集モジュール

  本モジュールでは，エニーキャスト DNS の個体情報を識別するため，現在標準化が進められている DNS Server ID[66] の機能を利用した．この機能は，DNS サーバに対して特別

な名前解決要求を送ることによって，DNS サーバから個体識別できる情報を得られるものである．dnsprobe ツールにこの機能を付加し，到達性の調査と同時に個体識別情報の調査も行った．

図 5.15 に，実行結果例を示す．

```
1092719088 130.69.XXX.116 eth1 A: rtt 189 ms: hostname ns11-aroot
1092719120 130.69.XXX.116 eth1 B: rtt 111 ms: hostname b2.isi.edu
1092719059 130.69.XXX.116 eth1 C: rtt 205 ms: hostname iad1a.c.root-servers.org
1092719103 130.69.XXX.116 eth1 D: rtt 186 ms: hostname d-root.net.umd.edu
1092719074 130.69.XXX.116 eth1 E: rtt 158 ms: hostname e4.arc.nasa.gov
1092719093 130.69.XXX.116 eth1 F: rtt 156 ms: hostname lax1a.f.root-servers.org
1092719115 130.69.XXX.116 eth1 G: rtt 198 ms: hostname g.root-servers.net
1092719068 130.69.XXX.116 eth1 H: rtt 176 ms: hostname H2
1092719065 130.69.XXX.116 eth1 I: rtt 295 ms: hostname s2.sth
1092719083 130.69.XXX.116 eth1 J: rtt 256 ms: hostname jns3-kgtld.j.root-servers.net
1092719099 130.69.XXX.116 eth1 K: rtt 264 ms: hostname k2.linx
1092719110 130.69.XXX.116 eth1 L: rtt 263 ms: hostname l2.l.root-servers.org
1092719079 130.69.XXX.116 eth1 M: rtt 0 ms: hostname MB
```

図 5.15: 個体識別情報収集モジュール 実行結果例

実行結果は，dnsprobe によって到達性調査を実行した場合とほぼ同じである．異なるのは，最後の hostname の項目に，名前解決要求に対して応答してきた DNS サーバの個体識別情報が入っているという点である．

6章　本手法による結果と評価

本章では，本研究にて提案した分析モデルに基づいて構築した，DNS の運用基盤分析システムによる分析結果と，その考察について述べる．

6.1 節　DNS 委譲情報調査結果

本節では，委譲情報調査システムにて調査したデータをもとに，DNS の信頼性・耐障害性を分析した．

6.1.1 節　DNS 委譲情報収集システムの運用

本論文で述べた DNS 情報収集システムを用いて，DNS 委譲情報の調査を行った．今回は，逆引きゾーンである，in-addr.arpa ゾーンならびに ip6.int ゾーンに対して，本システムを実行した．その結果，逆引き委譲木を走査しデータを生成するのに，約 6 週間を要した．データ容量は，約 4GB ほどとなった．なお，今回示す調査結果は，2000 年の 4 月に行われたものであり，University of Southern California にある Information Sciences Institute (ISI) に位置するホストから行った．

なお，アドレスによってゾーン転送を制限している DNS サーバからは，ゾーンのデータを収集することができない．したがって，その DNS サーバが，逆引き委譲木の上位に位置している場合には，そこから下位の委譲を把握することが不可能となってしまう．そこで，ゾーン転送を制限している DNS サーバのうち，委譲木の把握上重要と思われる DNS サーバに関しては，管理者に対して協力要請のメールを送ることによって，できる限り解決した．したがって，今回の調査では逆引きゾーン全てを完全に走査することはできていない．また，バージョン番号の取得に関しても，取得に成功したサーバについてのみ結果を示した．しかし，DNS の現状を把握するにあたって十分と思われるデータ量が取得できた．

6.1.2 節　統計情報から見る DNS の現状

まず，委譲に関する結果を，表 6.1 に示す．走査した全 183476 委譲点のうち，約 5.6% にあたる 10322 委譲点が，正常に委譲されていないという結果が出た．5.6% という数値は，意外に低いと思われるかもしれない．しかし，これは IP アドレスという限りある資源に基づいて形成される，逆引き委譲木に対して行った場合の結果であることを考えると，一概に低い数値だと言うことはできない．さらに，委譲木が完全に壊れていた場合，それより先に委譲木を走査することができないため，その先の委譲がどの程度正確に行われているかを把握することは難しい．

表 6.1: 委譲の正確性に関する調査結果

	委譲点数	割合
全調査委譲点	183476	-
正確な委譲点	173154	94.4%
不正確な委譲点	10322	5.6%

次に，DNS サーバのバージョン番号に関する統計を，図 6.1 に示す．様々なバージョンの DNS サーバが，現在稼働されていることがわかる．bind-8.2.2-P3 以下や，bind-4.9.6 以下は，有名な幾つかのセキュリティホールが発見されており [24]，これを悪用された場合，DNS サーバのサービス停止や，最悪の場合 DNS サーバのルート権限を搾取される可能性がある．また，本調査で発見された DNS サーバのうち，実装として bind を用いている 14094 台の DNS サーバに関して，安全なバージョンと危険なバージョンの割合をまとめたグラフを，図 6.2 に示す．危険なバージョンのまま運用されている bind が 69% にものぼることがわかる．アップグレードされないまま放置され，運用されている DNS サーバが多いことがうかがえる． これらの bind は，早急にアップグレードされるべきである．

セキュリティ向上のための 2 つの前提条件に関して，その現状を把握するためのデータを示した．データから考えると，現状の委譲状態では，DNSSEC を有効に機能させることが困難だと考えられる．したがって，DNSSEC を有効に導入するためには，本システムによって発見された不正確な委譲点を抜き出し，管理者に対して改善するよう注意を促す必要がある．また，DNS サーバのバージョン番号に関しても，危険な DNS サーバは早急にアップグレードする必要がある．危険な DNS サーバの管理者に対しても，注意を促すことが必要である．

以上より，現状の DNS においては，信頼性ならびに耐障害性を向上させるための新技術を適用する土台が整っておらず，仮に新技術をそのまま適用したとしても，本来の効果を期待することはできないと言える．

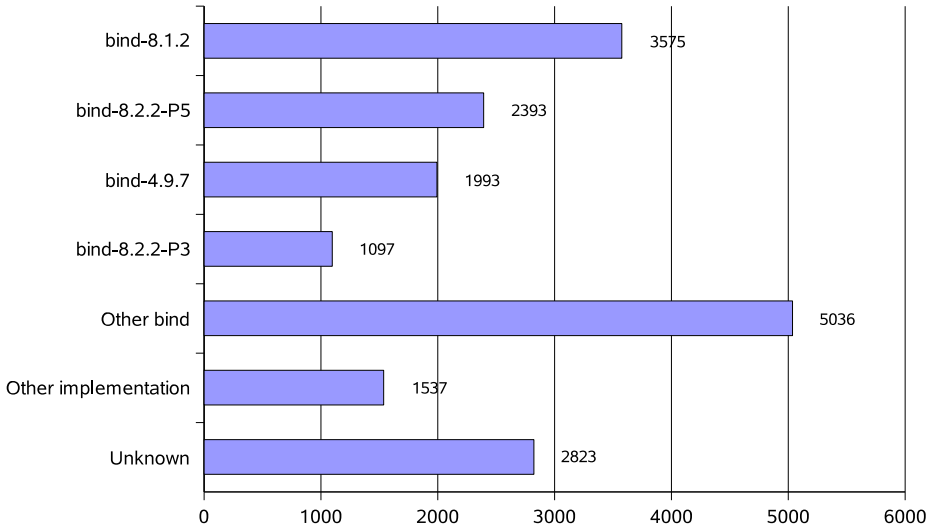


図 6.1: DNS サーバのバージョン番号調査結果

6.1.3 節　DNS 委譲情報分析結果の考察

3.3 節では，DNS の信頼性と耐障害性を向上させるために解決しなければならないセキュリティ問題について述べ，その解決策となる新技術について述べた．その上で，現状の DNS に対して新技術を適用することができるかどうか分析するために，委譲情報調査システムを構築し，分析を行った．

その結果，新技術を有効に導入するためには，委譲ツリーをもっと正確なものにする必要があることがわかった．すなわち，新技術導入の基盤を整えることが，これからの DNS に課せられている急務の課題であると言える．

これからのインターネットは，計算機がつながるだけではなく，様々な機器が接続されると予想される．つまり，インターネットはそれら機器をつなぐインフラとして利用され，今まで以上の信頼性が求められるようになると考えられる．すると，DNS にもさらなる信頼性が要求され，セキュリティ向上が必須要件となる．

現在の DNS にとって，信頼性と耐障害性を向上させるためには，DNSSEC に代表されるセキュリティ新技術の導入が求められている．DNSSEC を導入するためには，委譲ツリーの正確性を向上させなければならない．そのためには，DNS の完全性を分析し，委譲ツリーの正確性を向上させるためのシステムが必要であり，本システムはその約割を担うことができる．例えば，不正確な委譲点に関しては，本システムを利用すれば，不正確な委譲点となるゾーンと，その両

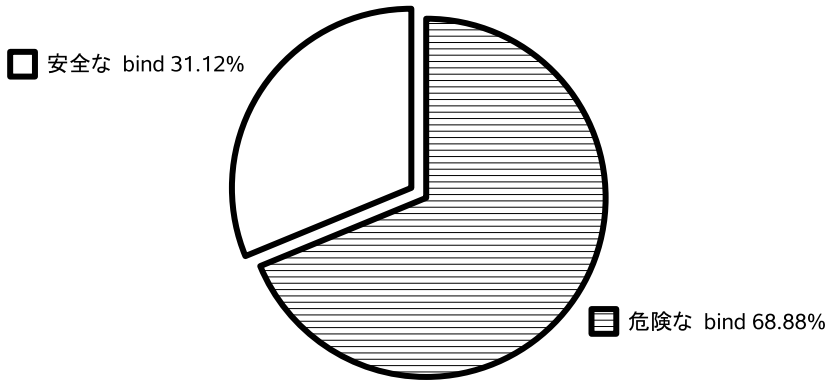


図 6.2: 危険な DNS サーバの割合

DNS の管理者に関する情報を容易に抜き出すことができる．また，危険な DNS サーバの発見に関しても，本システムを用いれば，危険な DNS サーバとその管理者に関する情報を容易に抜き出すことができる．

6.2 節　DNS 到達性調査結果

本節では，DNS 到達性調査システムにて計測した，世界各地からルート DNS サーバならびに ccTLD DNS サーバに対する到達性調査結果を示す．ルート DNS サーバは，DNS 木構造データベースの頂点となっており，名前検索の際の起点となるため，必ず検索される DNS サーバである．また，ccTLD DNS サーバは，各国の国別ドメイン名を保持している DNS サーバであり，ルート DNS サーバとあわせて，公共性の高い DNS サーバである．

6.2.1 節　dnsprobe による計測結果

計測は主にダイヤルアップを用いて行い，27ヶ所からの計測を行った．内訳は，アフリカ地域から 3 カ所，アジア地域から 4 カ所，ヨーロッパ地域から 5 カ所，ロシア地域から 1 カ所，オセアニア地域から 2 カ所，北米地域から 7 カ所，南米地域から 5 カ所である．なお，ダイヤルアップは神奈川県から，アナログモデムを用いて行った．

計測結果は，地点ごとに 3 種類の情報として集計され，分析される．例としてブラジルにある計測ホストから native probe にて計測した結果を示す．まず，図 6.3 はルート DNS サーバならびに ccTLD DNS サーバへの RTT を累積分布関数にて表したものである．次に，図 6.4 は，名前解決パケットの損失割合を示す．最後に，図 6.5 は，計測地点から各ルート DNS サーバへの RTT を，時系列に従って示したものである．

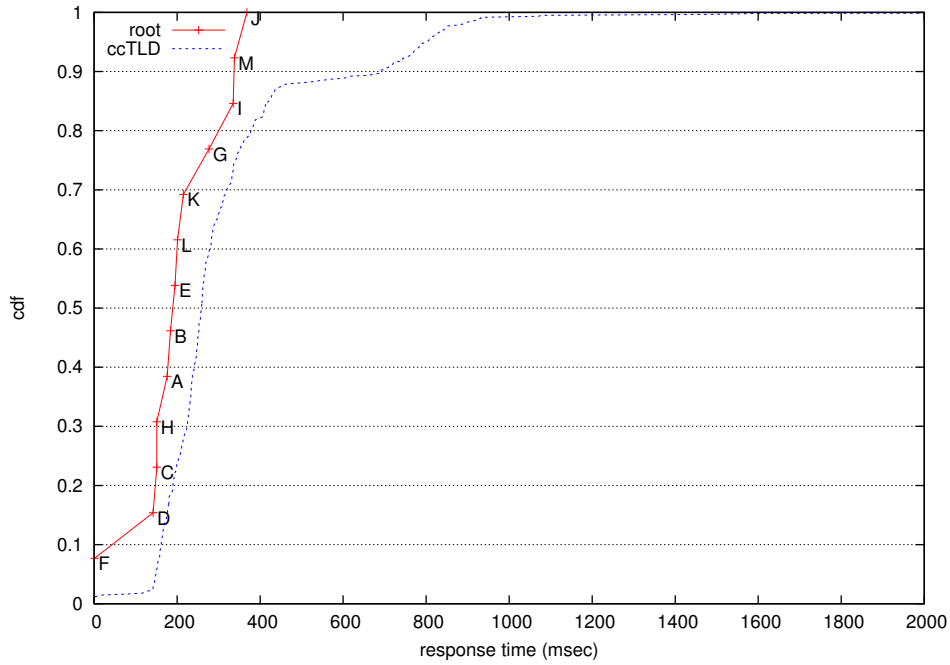


図 6.3: ルート DNS サーバならびに ccTLD DNS サーバへの到達性

なお，各計測地点毎の詳細な結果に関しては，本論文の付録として添付する．

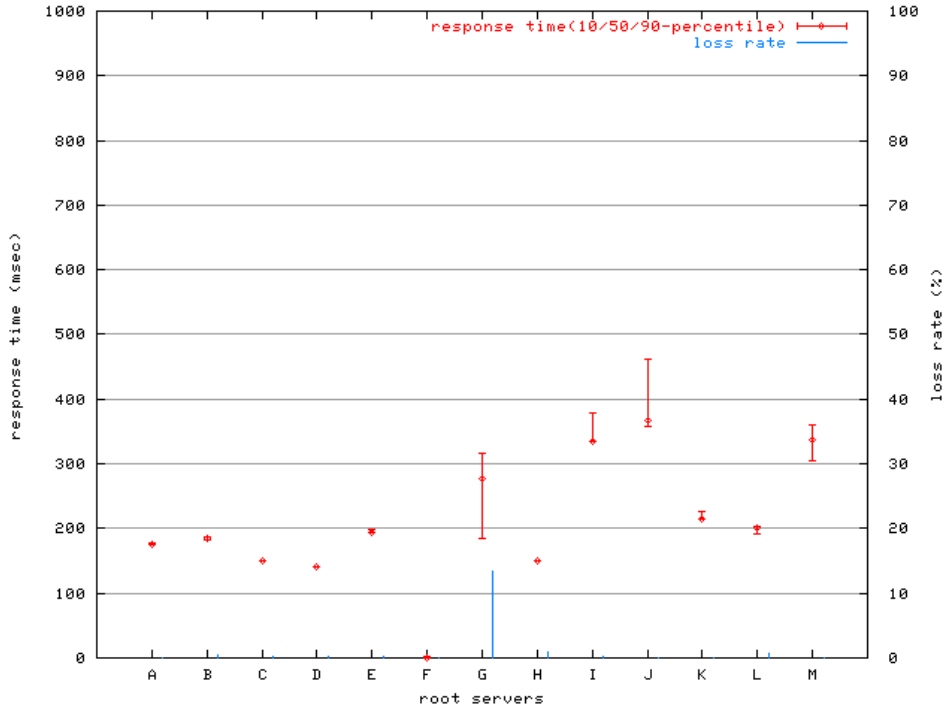


図 6.4: ルート DNS サーバへの名前解決応答損失率

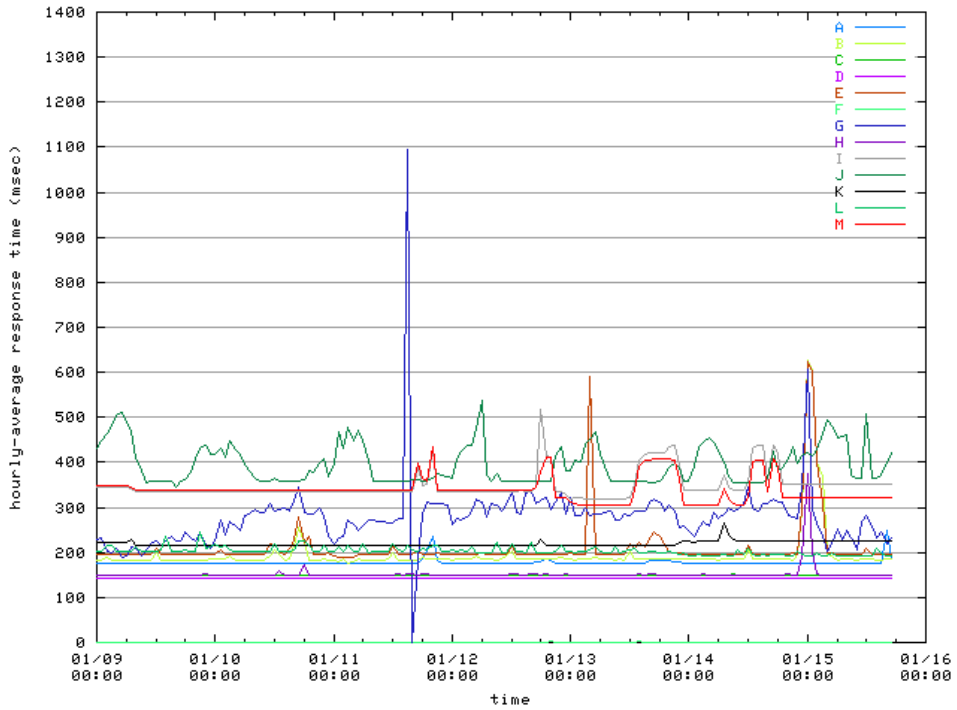


図 6.5: ルート DNS サーバへの到達性の時間による変化

6.2.2 節　到達性の分析結果

これら全ての計測地点からの分析結果をもとに，世界各地からの計測結果を図 6.6 にまとめる．なお，この計測においては，native probe と dialup probe が混在しているため，5.1.5 節にて述べた補正法 (2) を用いて補正を行った．各地点，最低 20 回の計測を行い，結果には中間値 (median) を用いた．

上記の結果から，北米方面からのルート DNS への到達性は良好で，多くのルート DNS サーバから 200ms 以下で応答が得られていることがわかる．一方，アフリカ，オセアニア方面と比べても，アジアの一部の国からの到達性が悪いことがわかった．

また，基準 DNS サーバ群のへの測定結果とルート DNS サーバへの測定結果の比較による，到達性判定の例をいくつか示す．米国 California 州 Palo Alto 市からの結果を図 6.7 に，中国からの結果を図 6.8 に示す．

Palo Alto からの測定結果は，ルート DNS サーバ群への到達性が，基準 DNS サーバである ccTLD サーバ群への到達性をすべて上回るという結果を示した．これは北米方面からのすべてのルート DNS サーバへの到達性が良好であることを示す．

中国からの測定結果は，ルート DNS サーバの示す曲線と，基準 DNS サーバの示す曲線が，L ルート DNS サーバから H ルート DNS サーバまでの 9 個のルート DNS サーバにて曲線がほぼ重なり合う結果となった．これは，基準 DNS サーバ群とほぼ同一の到達性を得られており，残る 4 個のルート DNS サーバは，基準 DNS サーバよりも良好な到達性を得られていると言える．

このように，基準 DNS サーバと測定対象 DNS サーバとの到達性を比較することによって，測定地点から見た，測定対象 DNS サーバへの到達性の傾向が読み取れる．これに関しては，さらにデータを集め，分類することによって，定地点から，目的とする DNS サーバ群への到達性の傾向を，単独の測定でつかむことが可能となると考えられる．

さらに，ccTLD サーバに対して同様の調査を行った結果をもとに，各測定拠点からルート DNS サーバならびに世界中の ccTLD サーバに対しての到達性の傾向をまとめた．その結果を図 6.9 に示す．

この図において，横軸を dnsprobe の測定拠点，縦軸をルート DNS サーバならびに ccTLD サーバが置かれている位置として，名前解決応答が得られるまでの時間を色濃度にして表示した結果である．結果として，やはり北米方面の到達性の良好さと，アジア方面からのルート DNS サーバならびに ccTLD サーバへの到達性の悪さが見て取れる．

6.2.3 節　分析結果の考察

以上の分析結果により，DNS の中でも公共性の高い，ルート DNS サーバと ccTLD DNS サーバへの，世界各地からの到達性を明らかにすることができた．ネットワークの接続性にも恵まれ，ルート DNS サーバの大半が存在する北米方面は，多くの公共的な DNS サーバ群に対して，良好な到達性を得られていることがわかった．一方，アジア方面やオセアニア方面からは，公共的な DNS サーバに対する到達性が，北米や欧州方面に比べ劣っていることがわかった．特に東南アジアや中国の到達性が劣っていることがわかった．

これらの分析結果より，今後エニーキャスト等の分散技術を用いて公共的な DNS サーバを増やす場合，アジア方面からの接続性が良好な地域を選択することが望ましいことがわかった．

6.3 節　DNS 運用情報調査結果

本節では，DNS 運用情報調査システムを用いて DNS の運用情報を分析した結果を示す．

図 6.10 ならびに図 6.11 に，DNS 運用者情報データベースモジュールにて作成されたデータベースの検索結果を示す．この例では，3ffe:501::/32 という IPv6 アドレス空間を保持している DNS サーバと，その管理者情報について検索した結果を示している．

この例に示すように，アドレスの利用者に関する情報とともに，そのアドレス空間を保持している DNS サーバに関する情報も同時に検索することができる．また，その逆も可能であり，DNS サーバの名前を入力すると，その DNS サーバが保持している名前空間の一覧を検索することも可能である．これによって，名前解決できない名前や，到達性が無い DNS サーバを発見した場合，その管理情報と影響範囲をすばやく把握することが可能となる．

次に，個体識別モジュールの調査結果を示す．3.1.1 節にて述べたとおり，ルート DNS サーバには，エニーキャスト 技術が導入されている．そこで，世界の 19 箇所から，13 種類すべてのルート DNS サーバに対して，個体識別調査を行った．

ルート DNS サーバの中でも，エニーキャスト拠点が特に多い，f.root-servers.net と j.root-servers.net への調査結果を，表 6.2 に示す．

また，定常的な運用情報を分析することによって，分散サービス拠点の増加を検知できた例を示す．東京大学内にある，130.69.251.116 というホストから i.root-servers.net に対して到達性調査と同時に運用情報識別調査も行った結果を示す．SQL データベースに対して，図 6.12 に示すクエリを発行し，130.69.251.116 というホストから i.root-servers.net という DNS サーバに関す

表 6.2: f.root-servers.net に対する個体識別調査結果

調査地点	f.root-servers.net	j.root-servers.net
Adelaide-Australia	bne0a.f.root-servers.org	jns2-kgtld
Athens-Greece	sfo2b.f.root-servers.org	jns6-hgtld
Beijin-China	sfo2b.f.root-servers.org	jns1-kr
Belgrade-Yugoslavia	sfo2b.f.root-servers.org	jns4-igtld
Bethlahem-Israel	tlv1b.f.root-servers.org	jns1-kgtld
BuenosAires-Argentina	sfo2b.f.root-servers.org	jns3-dgtld
Calgary-Canada	sfo2a.f.root-servers.org	jns1-kr
Copenhagen-Denmark	sfo2b.f.root-servers.org	jns5-hgtld
Cork-Ireland	sfo2b.f.root-servers.org	jns6-hgtld
Dusseldorf-Germany	sfo2b.f.root-servers.org	jns4-hgtld
Helsinki-Finland	sfo2a.f.root-servers.org	jns2-hgtld
HongKong-HongKong	hkg1a.f.root-servers.org	jns1-kgtld
Jakarta-Indonesia	cgk1b.f.root-servers.org	jns1-mgtld
PhnomPenh-Cambodia	pao1c.f.root-servers.org	jns1-kr
Quito-Ecuador	sfo2a.f.root-servers.org	jns1-fgtld
SaoPaulo-Brazil	gru1a.f.root-servers.org	jns3-fgtld
Seoul-Korea	sel1a.f.root-servers.org	jns1-kr
Shenzhen-China	pek1a.f.root-servers.org	jns1-kr
Sofia-Bulgaria	pao1f.f.root-servers.org	jns4-hgtld

る到達性調査ならびに運用情報識別調査を行った結果を，調査時間を添えて出力するよう要求するした.

```
select Timestamp,HOSTNAME,RTT from dnsprobe
where IPaddress='130.69.251.116' && ServerNAME='I' && HOSTNAME not like 'NULL';
```

図 6.12: SQL サーバへの調査結果の問い合わせ

その問い合わせ結果を，同様な行が並んでいる部分を略し，注目する部分のみ図 6.13 に示す. この結果から，調査を開始した 2004 年 8 月 17 日から，2004 年 9 月 20 日までは，130.69.251.116 から i.root-servers.net への問い合わせには約 295ms の時間を要しているのに対し，2004 年 9 月 20 日のある時間をもって，問い合わせ時間が 1ms に減少しているのがわかる．また，運用情報識別システムによる，個体識別情報を見ると，RTT が 295ms 程度の場合には，s2.sth という個

体識別情報を持つ i.root-servers.net DNS サーバに問い合わせが行われていたことがわかる．一方，RTT が減少してからは，s1.tok という i.root-servers.net DNS サーバに対して問い合わせが行われていることがわかる．これは i.root-servers.net が，分散サービス拠点を東京にも設置したことが原因である．s2.sth はアメリカ合衆国のシアトルに位置する分散サービス拠点であり，s1.tok は東京に位置するサービス拠点を示す．このように，定常的に運用情報の収集を行うことによって，プロアクティブに運用基盤の安定化をはかるための分析結果を得ることができる．

```
Timestamp       HOSTNAME        RTT
20040817050425  s2.sth          295
20040817050956  s2.sth          295
20040817051450  s2.sth          295
20040817052010  s2.sth          295
20040817052439  s2.sth          295
             (略)
20040920125436  s2.sth          296
20040920125953  s1.tok 1
             (略)
20050104104317  s1.tok          1
20050104104512  s1.tok          1
20050104104814  s1.tok          1
```

図 6.13: SQL サーバへの問い合わせ結果

以上の結果から，DNS 運用情報識別システムを用いることにより，いままで困難とされてきたエニーキャスト DNS サーバの運用基盤分析を行うことが可能となった．これによって，平常時におけるサービス拠点の識別を行うことが可能となり，インフラにとってのプロアクティブな信頼性を確保するための運用基盤分析が可能となった．

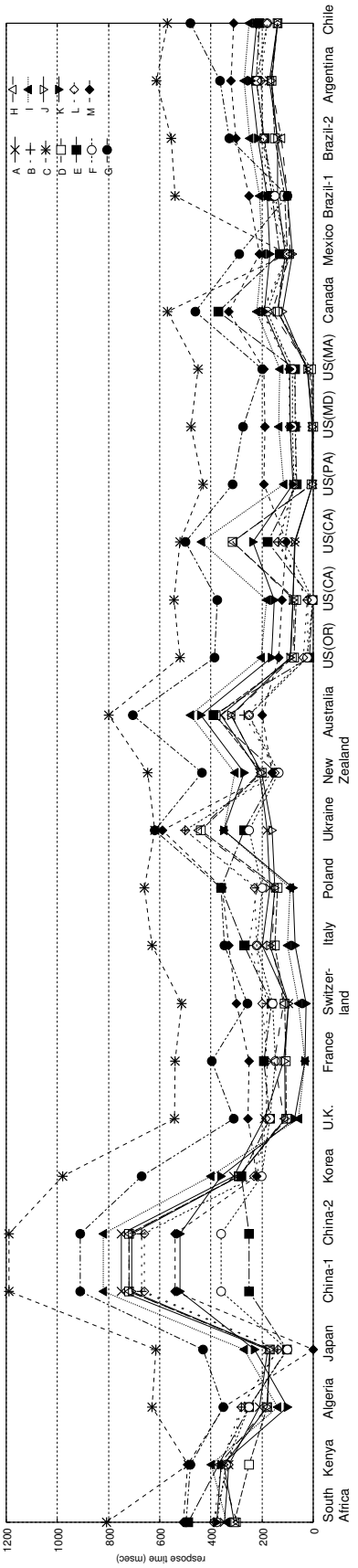


図 6.6: ルート DNS サーバに対する世界各地からの到達性

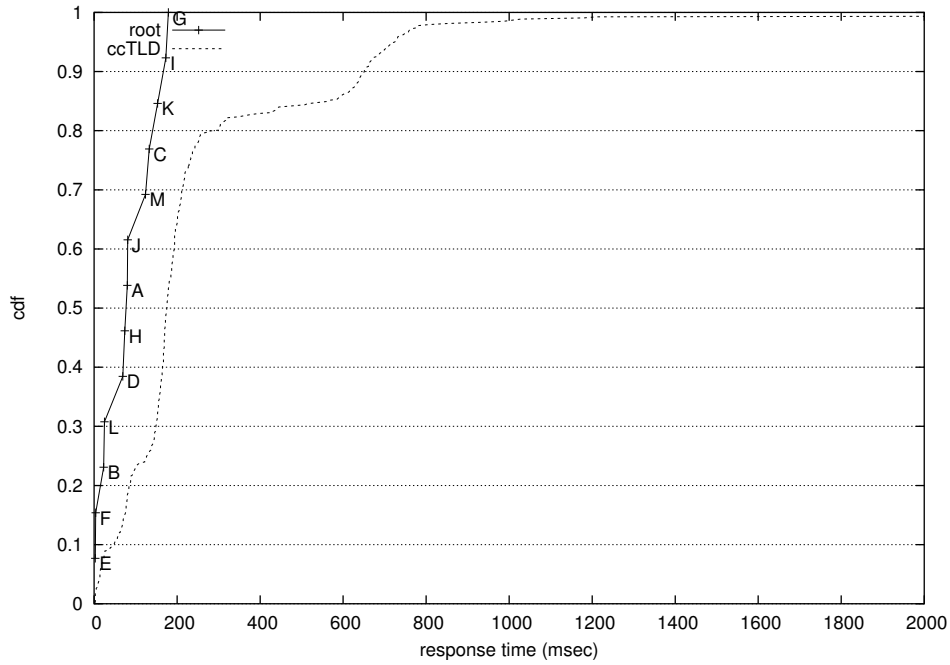


図 6.7: アメリカ (Palo Alto) からのルート DNS サーバへの到達性

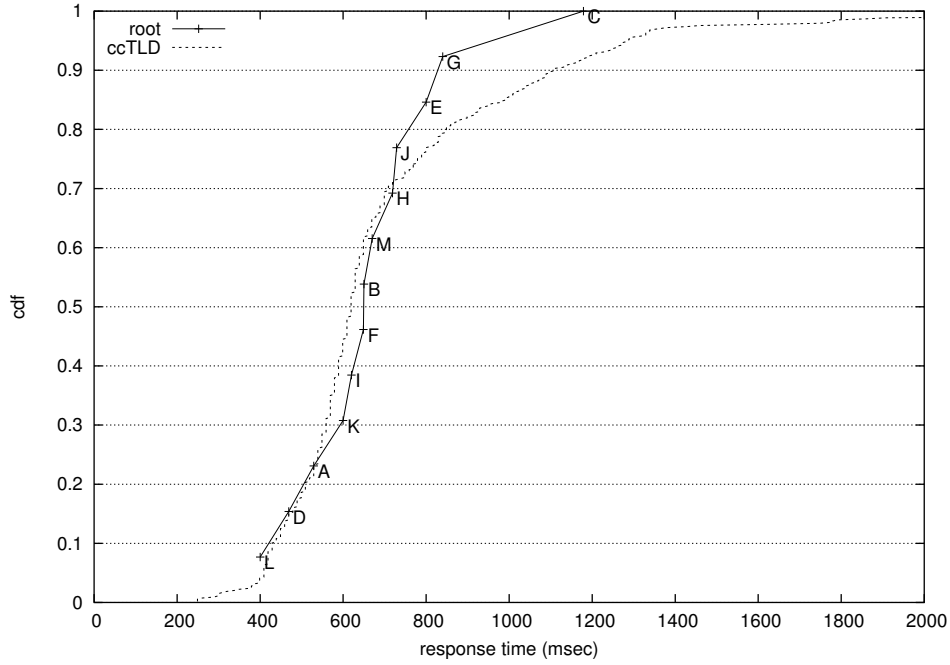


図 6.8: 中国からのルート DNS サーバへの到達性

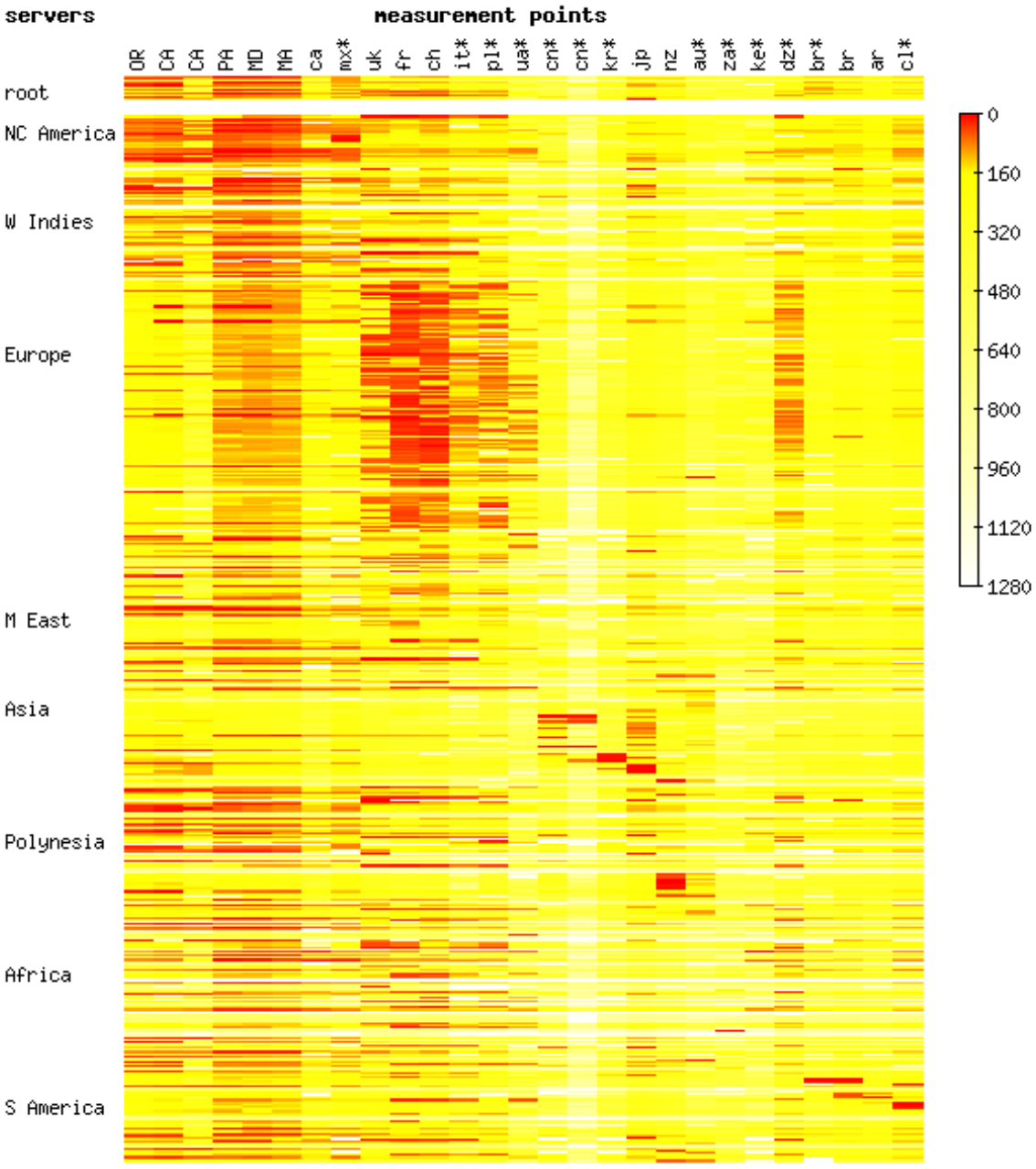


図 6.9: 測定拠点からのルート DNS サーバならびに ccTLD サーバへの到達性

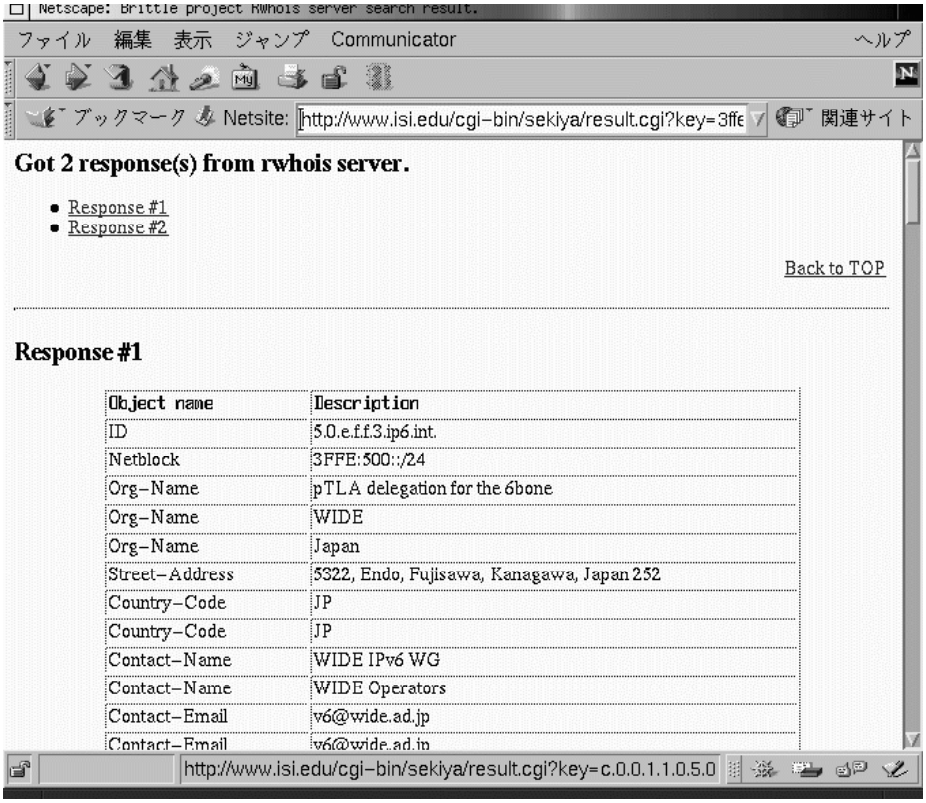


図 6.10: IPv6 アドレス空間からの検索結果

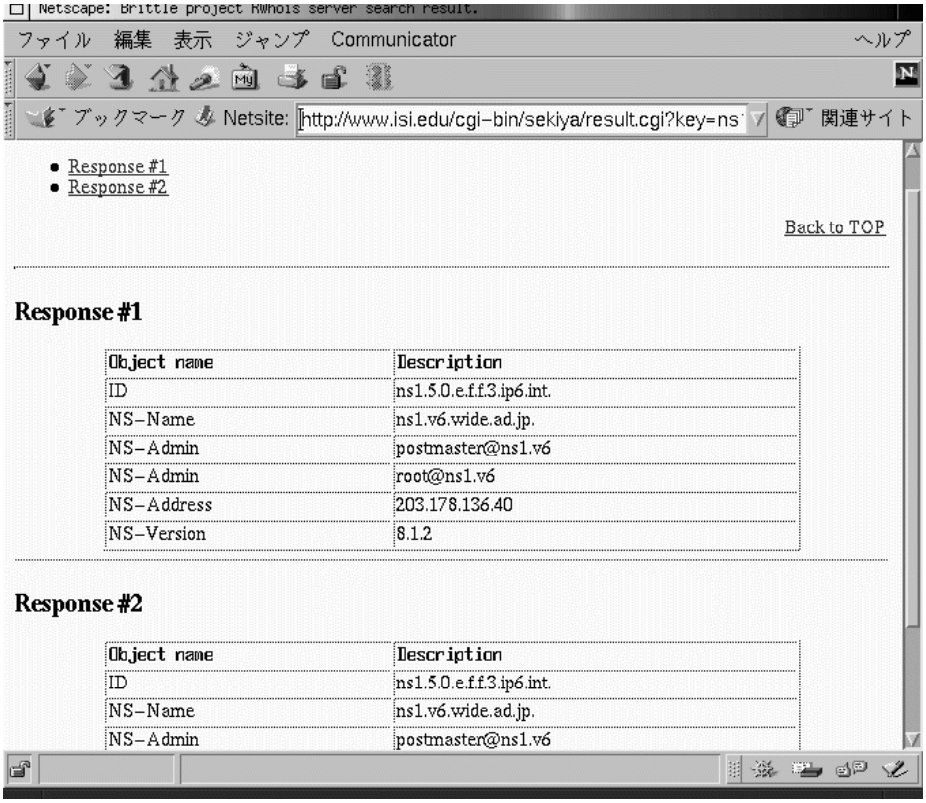


図 6.11: アドレス空間の名前情報を持つ DNS サーバの検索結果

7章　おわりに

本研究では，大規模かつグローバルなインフラの運用基盤分析を行う手法を提案し，実践することによって検証した．

本研究では，まずインフラストラクチャの分類を行った．インフラのサービス範囲やサービス拠点の分散方法によって，従来のインフラと，大規模かつグローバルなインフラに分類した．本研究の対象とする大規模かつグローバルなインフラとは，サービス範囲が世界規模であり，サービス拠点が分散して存在し，それぞれが自律分散的に協調してサービス空間を構築しているような，情報通信インフラである．

次に，インフラがインフラとして機能するために必要な条件について論じ，本論文では次の点をインフラの要件として定義した．それは、(1) サービスの公共性m (2) サービスの信頼性・耐障害性、(3) サービス拠点の識別性，の 3 点である．これをふまえ，分析対象となるインフラストラクチャが，これらの要件を満たしているかどうか調査する手法を確立した．その結果，本研究では，(1) ユーザの視点からのサービス分析，(2) サービスの完全性分析、(3) サービス識別情報の分析，の 3 点を軸とした分析手法を基に，運用分析モデルを構築した．

構築した運用分析モデルを実証するために，本研究では， DNS を大規模かつグローバルなインフラの課題事例としてとりあげ，具体的な運用分析モデルの設計を行った．DNS の運用分析モデルを構築するために，DNS がどのように大規模かつグローバルなインフラとして機能しているのかの現状分析を行い，この分析をふまえ，本論文にて提案した運用分析モデルを適用して DNS 運用基盤分析モデルを構築した．このモデルは運用分析モデルにおける 3 つの手法に対応しており，それぞれ (1)DNS サーバ到達性の分析，(2)DNS 委譲ツリーの分析，(3)DNS サーバ識別情報の分析，の 3 手法により構成された．

本研究のモデルによる DNS 運用基盤分析では，個々の DNS サーバを分析するのみならず，DNS 全体をひとつの系としてモデル化し，分析することができた．これは従来の研究にない特色であり，本研究にて提案したモデルの優位性を検証することができた．

また，これらの分析結果は，いままでの運用基盤分析手法では明らかにできなかった DNS のシステム全体の信頼性を明らかにした．また，ユーザからの視点による分析データを蓄積することによって，障害時の迅速な復旧に役立つための情報を提供することができた．この成果により，本論文にて確立した分析手法が，本研究の定義する大規模かつグローバルなインフラの運用基盤分析において有効に機能することを確認できた．

以上の結果により，本研究によって，今まで手法が確立されていなかった大規模でグローバルなインフラの運用基盤分析を行うことが可能となった．本研究の成果は，DNS のみならず，さ

らに増加していくであろう大規模かつグローバルかつなインフラを，正常に運用するための運用基盤分析に利用することが可能である．

謝辞

本研究におきましては，非常に多くの方からご助言・ご助力をいただきました．特に，慶應義塾大学の村井純教授，徳田英幸教授，楠本博之助教授，中村修助教授，東京大学の江崎浩助教授，加藤朗助教授，中山雅哉助教授，IIJ 技術研究所の長健二朗氏，東芝研究開発センターの神明達哉氏には，日頃より本当にたくさんのご指導を賜りました． この方々のおかげで本研究を進め，本論文を完成させることができました．心から感謝の意を表します．

また，日頃から様々な激励をしていただいた，慶應義塾大学の南政樹氏，重近範行氏，植原啓介氏，湧川隆次氏，南政樹氏，石原知洋氏，川喜田祐介氏，渡里雅史氏，岡田耕司氏，小柴晋氏，IIJ 技術研究所の山本和彦氏，萩野純一郎氏，東京大学江崎研究室の山本成一氏，吉田薫氏に感謝致します．

さらに，修士課程から慶應義塾大学に入学し，始めて所属した村井研究室の研究グループである，IPng-WG の諸氏，ならびにその後の慶應義塾大学における活動の拠点となった，SING WG の諸氏，さらに大学院のプロジェクトであった，MAUI メンバーの諸氏には，数えきれないくらいの多くのご助力を頂きました．これらの活動の場が無かったならば，本論文の完成はあり得ません．

一方，私の活動の範囲を飛躍的に増大させてくれ，多くの恩師や師匠と仰ぐ方々に出会う機会を与えてくれ，研究のきっかけや動機を与えてくれた，WIDE Project の存在に多大なる感謝の意を表します．WIDE Project 無しにはこの研究は為し得なかったものであります．dialup probe の数十万円に及ぶ電話代や，海外渡航の費用等を含め，様々な形でサポートをしていただきました．

特に，WIDE Project の中においては，私の論文完成をひたすら祈ってくださった Area Director 諸氏，また私の論文執筆をなるべく邪魔をしないようにとサポートしてくださった TWO wg の諸氏，私のもう一つの大きな活動場所であった，USAGI WG の諸氏にも，多大なる感謝の意を表します．また，本研究に対して様々なご助力を頂いた，WIDE プロジェクト内の IPv6 WG の諸氏，DNS WG の諸氏，MAWI WG の諸氏に，感謝致します．

最後に，私的な面にて様々なサポートを行ってくれた，愛妻の関谷弥生と愛猫のクロタブミオに感謝します．

この論文完成の奇跡を祝って．
2005 年 2 月 16 日
関谷 勇司

付 録A　研究履歴

A.1　著者による主論文に関連する査読論文

1. 関谷 勇司, 長 健二郎, 加藤 朗, 村井 純, “基準 DNS サーバを利用した DNS のパフォーマンス測定並びに評価手法に関する研究”, 電子情報通信学会, Vol.J87-B No.10, pp.1542–1551, 2004 年 10 月

2. Yuji Sekiya, Hiromi Wakai, Shu Nakamae, Kenji Hirose and Jun Murai, “The Mechanism for Scalable Registry System with Aggregatable Address Allocation on WIDE 6bone”, IEICE Transactions on Communications, Special Issue on Internet Technology and Its Applications, Vol.E82-D No.4, pp.888–895, Jan. 1998

A.2　著者による主論文に関連する国際学会発表論文

1. Yuji Sekiya, Kenjiro Cho, Akira Kato, Ryuji Somegawa, Tatsuya Jinmei and Jun Murai, “Root and ccTLD DNS server observation from worldwide locations”, Proceedings of Passive and Active Measurement 2003, pp.117–129, Apr. 2003

A.3　著者による主論文に関連するその他の論文

1. 関谷 勇司, 石原 知洋, “DNS のセキュリティ対策と運用状況の調査ツール”, 情報処理学会会誌, Vol.41 No.12, pp.1373–1379, 2000 年 12 月

A.4　著者が含まれる主論文に関するその他の査読論文

1. Ryuji Somegawa, Kenjiro Cho, Yuji Sekiya and Suguru Yamaguchi, “The Effects of Server Placement and Server Selection for Internet Services”, IEICE Transaction on Communication, Vol.E86-B No.2, pp.542–551, Feb. 2003

2. 加藤 朗, 関谷 勇司, “ISP の DNS サーバの DNS トラヒックの解析”, 電子情報通信学会, Vol.J87-B No.3, pp.327–335, Mar. 2004

A.5 著者が含まれる主論文に関するその他の論文

1. 石原 知洋, 関谷 勇司, 南 政樹, “DNS キャッシュ通信機構の設計および実装”, 日本ソフトウェア科学会, 第 4 回インターネットテクノロジーワークショップ 論文集, pp.58–65, 2001 年 9 月

A.6 著者によるその他の論文

1. Yuji Sekiya, Hideaki Yoshifuji, Mitsuru Kanda and Kazunori Miyazawa, “Evaluation and Improvement of IPv6 Protocol Stack by USAGI Project”, Linux Conference, Proceedings of Linux Conference 2002, Jun. 2002

A.7 著者が含まれるその他の論文

1. Shu Nakamae, Yuji Sekiya and Jun Murai, “A Study into a Visualization of an IPv6 Network”, Internet Society, Proceedings of INET99, Jun. 1999

2. Osamu Nakamura, Yuji Sekiya, Suguru Yamaguchi, Noriyuki Shigechika and Naoto Morishima, “ShowNet on INTEROP Tokyo 2002”, SAINT2003 IPv6 and Applications Workshop, Jan. 2003

3. 吉藤 英明, 神田 充, 高宮 紀明, 関谷 勇司, 江崎 浩, 村井 純, “Linux における IPv6 基本ソフトウェアの研究開発”,

4. Hideaki Yoshifuji, Kazunori Miyazawa, Masahide Nakamura, Yuji Sekiya, Hiroshi Esaki and Jun Murai, “Linux IPv6 Stack Implementation Based on Serialized Data State Processing”, IEICE Transactions on Communications, Vol.E87-B No.3, Mar. 2003

5. Hideaki Yoshifuji, Kazunori Miyazawa, Yuji Sekiya, Hiroshi Esaki and Jun Murai, “Linux IPv6 Networking - Past, Present and Future”, Linux Conference, Proceedings of Linux Conference 2003, Jul. 2003

6. 吉藤 英明, 神田 充, 高宮 紀明, 関谷 勇司, 江崎 浩, 村井 純, “USAGI プロジェクトによる IPv6 基本ソフトウェアの開発”, 電子情報通信学会, Vol.J85-B No.8, pp.1339-1346, 2002 年 8 月

A.8 著者による刊行物

1. 江崎 浩, 吉藤 英明, 関谷 勇司, 石原 知洋, “詳説図解 IPv6 エキスパートガイド”, 秀和システム, ISBN 4798003131, 2002 年 5 月

A.9 著者によるその他の活動

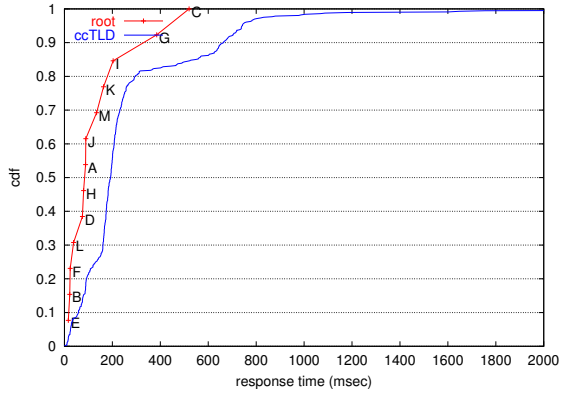
1. W. Biemolt, A. Durand, D. Finkerson, A. Hazeltine, M. Kaat, T. Larder, R. van der Pol, Y. Sekiya, H. Steenman and G. Tsirtsis, “An overview of the introduction of IPv6 in the Internet”, draft-ietf-ngtrans-introduction-to-ipv6-transition-08.txt, Internet Draft, Internet Engineering Task Force, Feb. 2002

2. Visiting Researcher at USC/ISI, 1999 – 2000 (6 months)

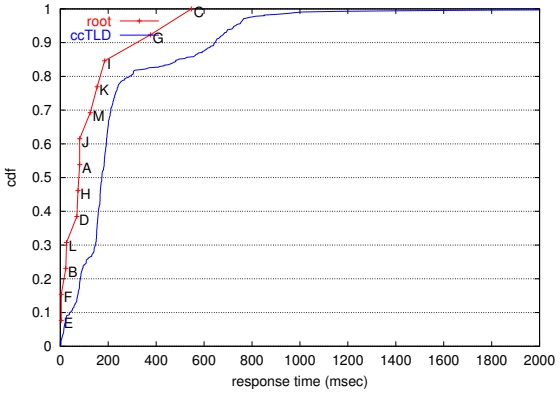
3. Core Member of USAGI Project - Linux IPv6 Development, Oct. 2000

付 録B　dnsprobe による計測結果

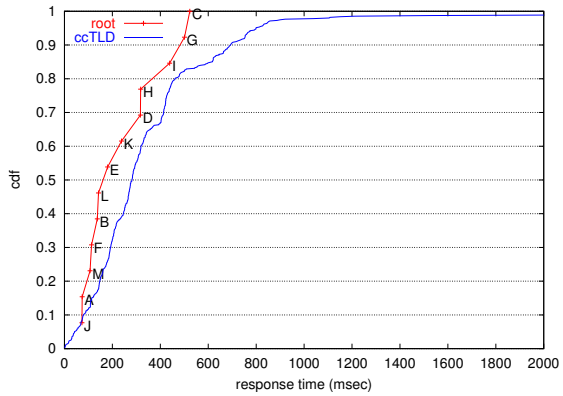
dnsprobe にて計測を行った地点のうち，ルート DNS サーバへの到達性と ccTLD DNS サーバへの到達性の両方を分析できた計測地点の，累積分布関数を示す.



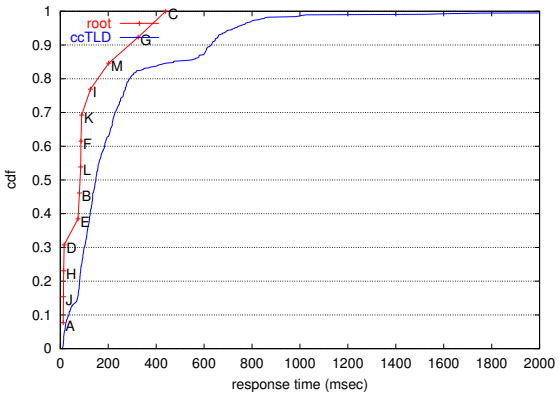
university, Oregon, USA



university, Palo Alto, USA

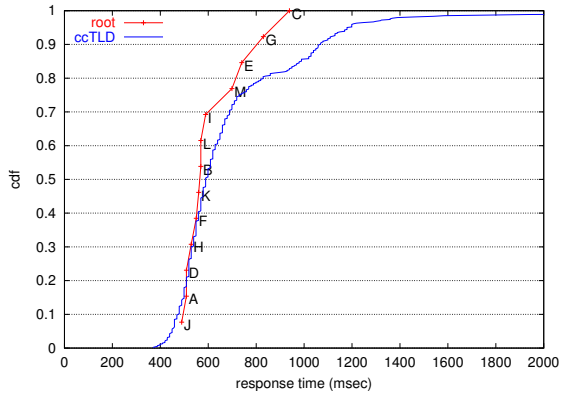


data center, Los Angeles, USA

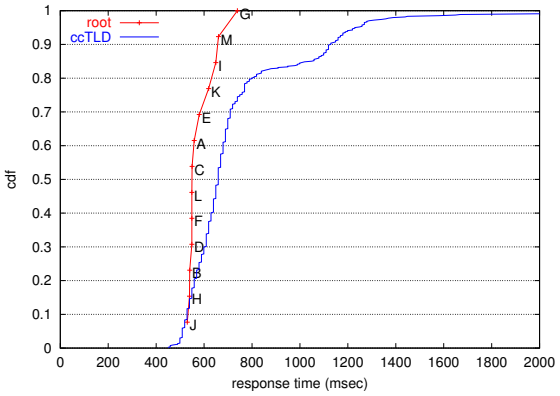


university, Pittsburgh, USA

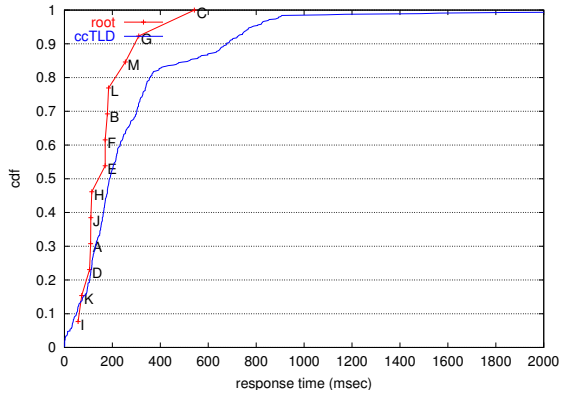
図 B.1: median response time of root and ccTLD DNS servers (1/6)



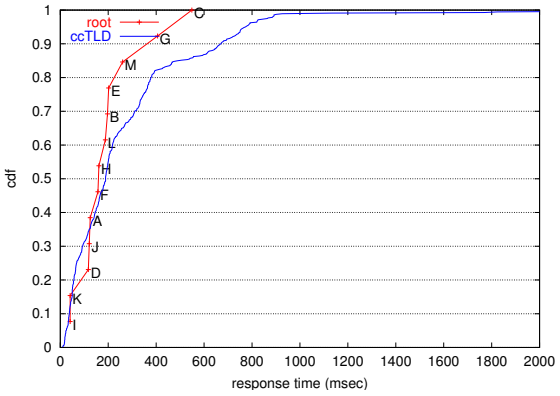
dialup, Ottawa, Canada



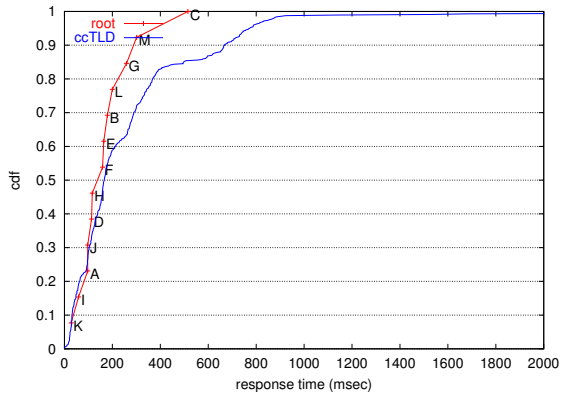
dialup, Cordoba, Mexico



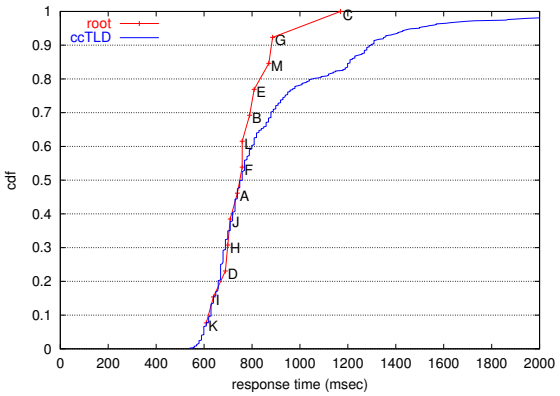
isp, London, UK



isp, Paris, France

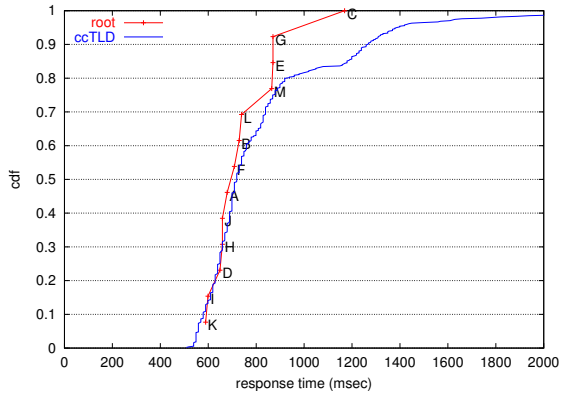


university, Zurich, Switzerland

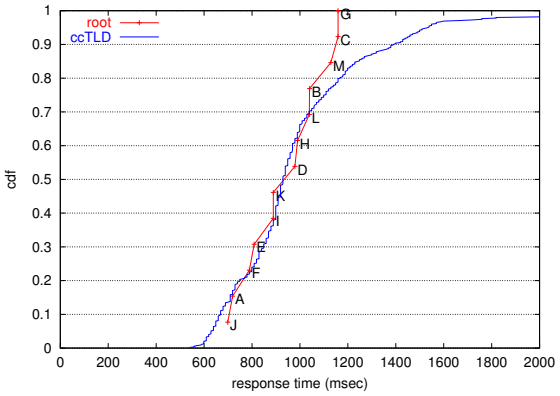


dialup, Parma, Italy

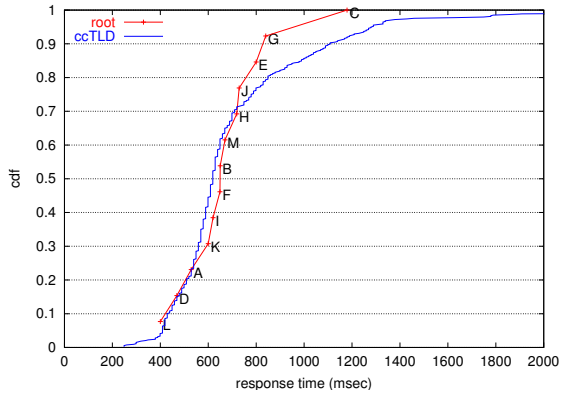
図 B.2: median response time of root and ccTLD DNS servers (2/6)



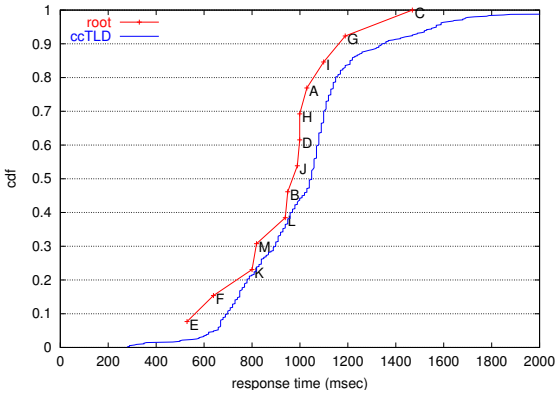
dialup, Torun, Poland



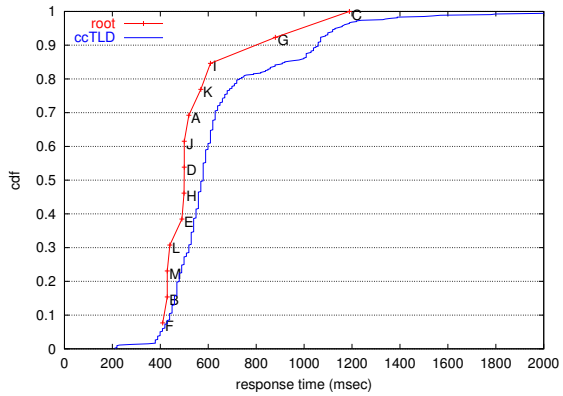
dialup, Ukraine



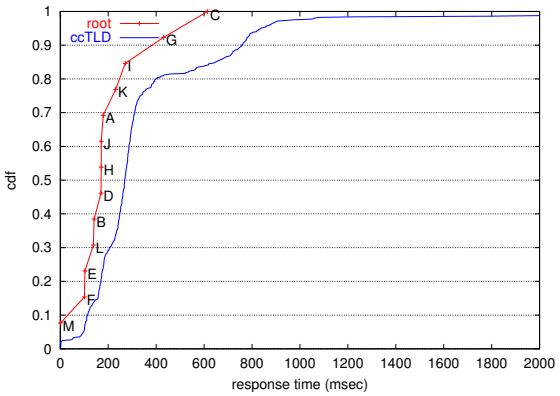
dialup, Shanghai, China



dialup, Beijing, China

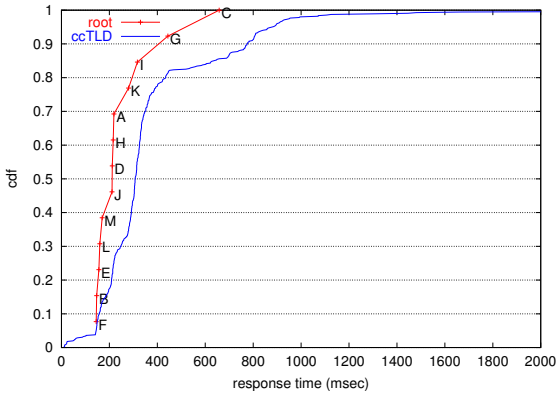


dialup, Seoul, Korea

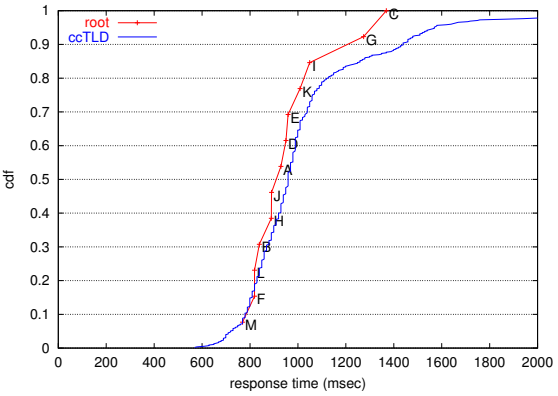


home, Tokyo, Japan

図 B.3: median response time of root and ccTLD DNS servers (3/6)

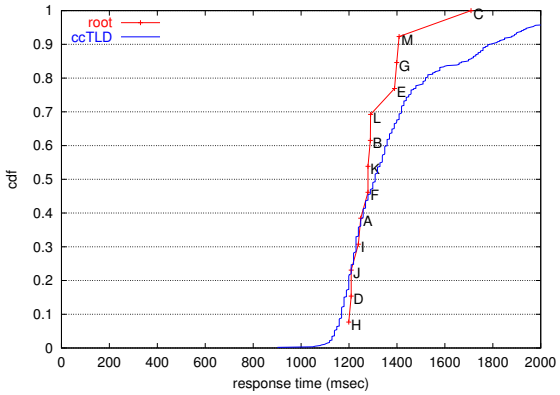


university, Hamilton, New Zealand

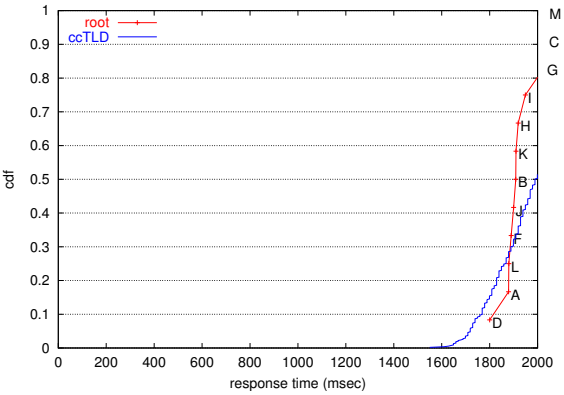


dialup, Canberra, Australia

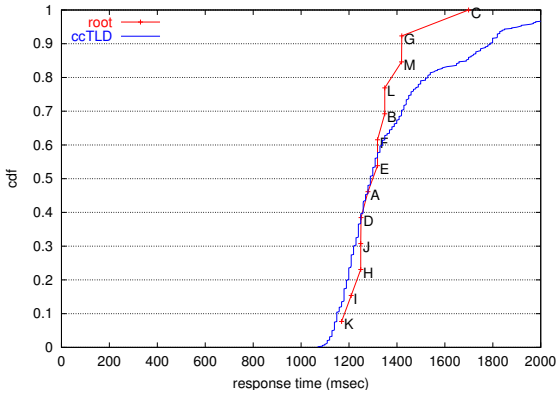
p



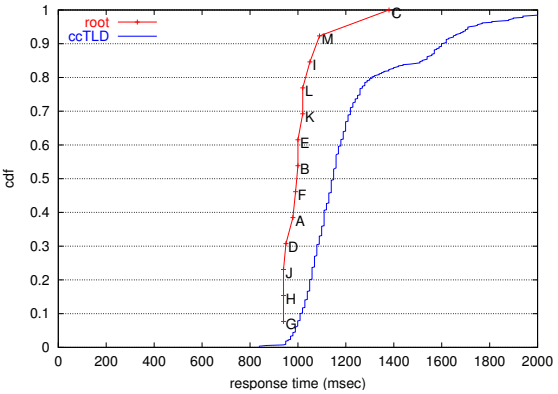
dialup, Cape Town, South Africa



dialup, Eldoret, Kenya

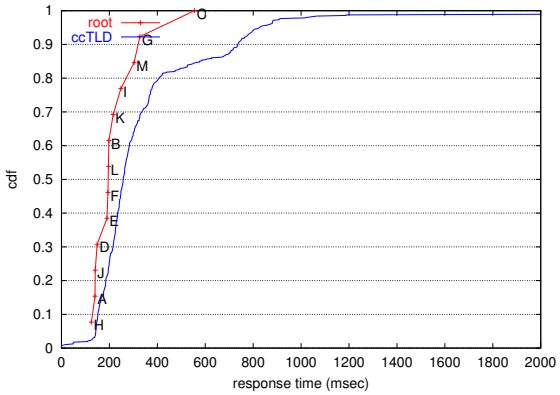


dialup, Algiers, Algeria

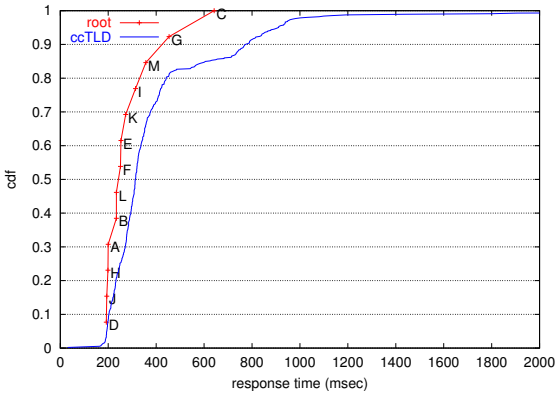


dialup, Salvador, Brazil

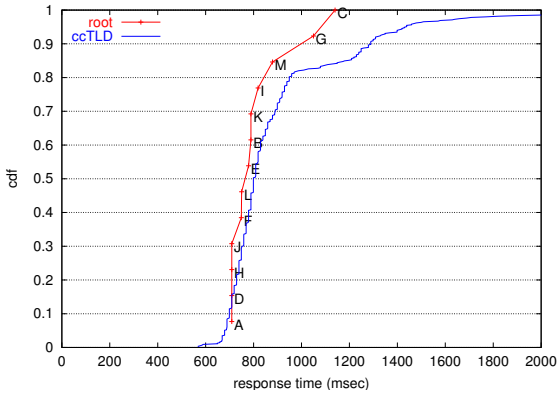
図 B.4: median response time of root and ccTLD DNS servers (4/6)



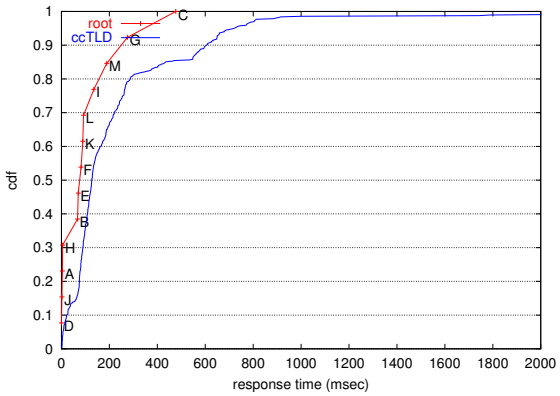
data center, Sao Paulo, Brazil



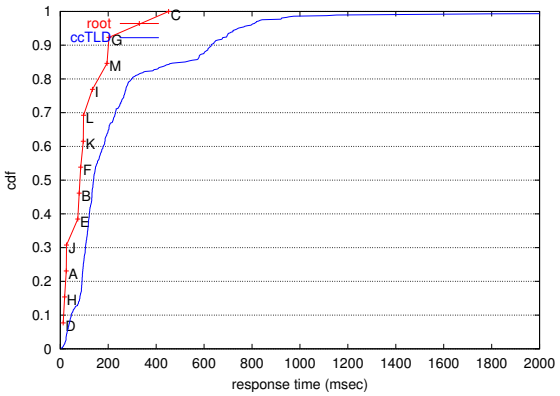
home, Buenos Aires, Argentina



dialup, Talca, Chile

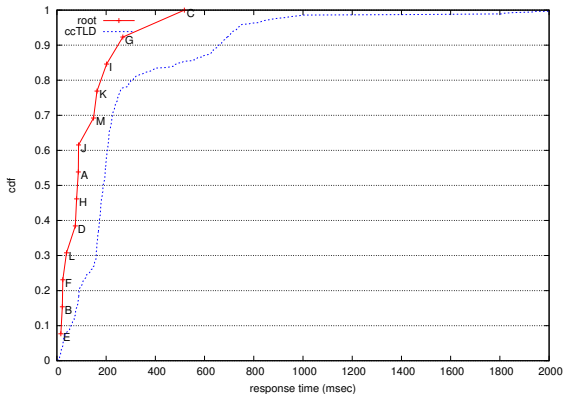


university, Maryland, USA

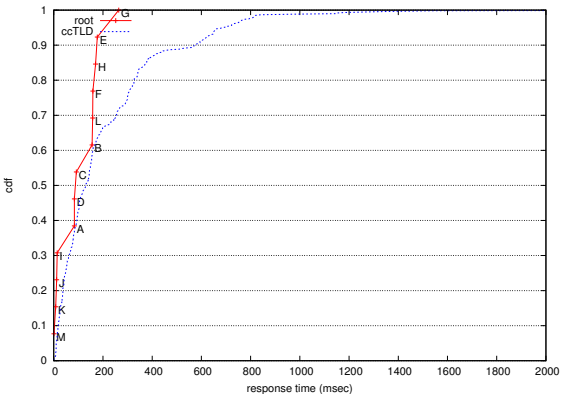


university, Cambridge, USA

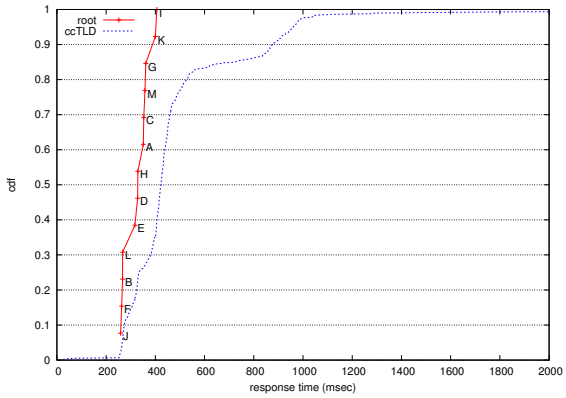
図 B.5: median response time of root and ccTLD DNS servers (5/6)



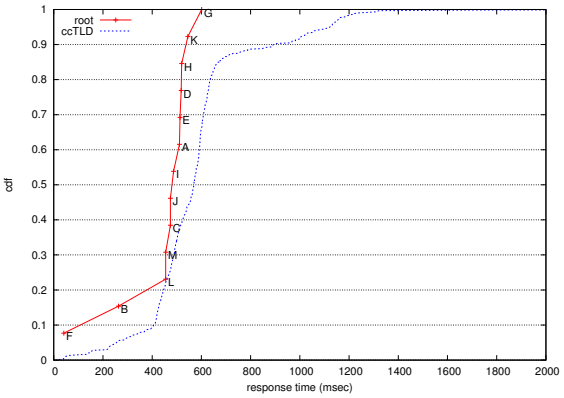
data center, Eugene, OR, USA



data center, Paris, France



ISP, Hat Yai, Thailand



University, Bandung, Indonesia

図 B.6: median response time of root and ccTLD DNS servers (6/6)

なお，計測を行った地点におけるソース IP アドレスの一覧は，以下のとおりである．

218.224.32.178,	171.64.92.77,	200.32.115.54,	210.130.102.74,	213.117.72.113,
12.162.27.126,	18.26.0.154,	200.39.233.210,	210.130.103.210,	213.117.73.172,
128.223.220.55,	192.134.1.9,	200.41.122.65,	210.23.149.202,	213.117.88.127,
128.8.244.130,	192.150.250.54,	200.45.245.218,	210.78.24.102,	213.130.35.153,
129.132.66.28,	192.150.250.67,	200.58.41.52,	211.222.61.137,	213.139.9.199,
130.217.250.13,	192.168.1.36,	200.63.219.21,	211.222.61.173,	213.169.32.110,
130.69.251.116,	192.168.1.8,	200.73.43.177,	211.222.61.176,	213.169.32.47,
130.69.251.121,	192.168.72.216,	202.134.66.197,	211.226.225.154,	213.169.32.65,
131.113.71.3,	192.168.81.188,	202.134.66.233,	211.39.29.165,	213.2.220.253,
133.11.123.222,	192.42.61.254,	202.134.66.236,	211.39.29.171,	216.6.44.156,
133.138.1.148,	192.50.6.2,	202.169.35.73,	211.39.29.178,	216.6.44.164,
133.27.24.11,	192.88.114.82,	202.169.35.77,	211.39.29.189,	216.6.51.241,
133.27.24.44,	193.192.166.46,	202.210.220.18,	211.39.49.86,	217.97.165.253,
133.27.25.33,	193.192.166.53,	202.232.15.103,	211.8.49.211,	218.222.42.40,
133.27.5.4,	195.123.249.39,	202.38.119.1,	212.106.79.122,	218.222.42.51,
134.159.108.232,	195.202.85.218,	202.51.156.5,	212.106.79.185,	220.126.232.251,
145.100.54.13,	195.232.52.4,	202.51.232.45,	212.106.79.80,	220.214.138.106,
148.240.87.195,	195.232.62.35,	202.56.150.6,	212.106.92.92,	221.143.6.72,
149.225.74.151,	195.242.160.105,	203.159.31.12,	212.211.93.11,	57.67.132.60,
149.225.80.48,	195.252.80.208,	203.159.31.94,	212.211.94.3,	57.70.6.55,
150.101.64.49,	195.72.162.205,	203.167.7.94,	212.37.2.75,	57.72.132.5,
150.101.64.91,	196.22.220.196,	203.178.138.146,	212.43.196.25,	57.74.4.76,
152.104.123.19,	196.22.220.243,	203.178.138.149,	213.116.22.249,	64.167.156.215,
152.104.167.229,	198.32.39.13,	203.178.140.236,	213.116.22.54,	80.78.138.140,
152.104.167.232,	198.32.6.68,	203.178.142.219,	213.116.22.57,	80.78.138.186,
152.104.226.197,	198.49.1.218,	203.178.143.79,	213.116.40.69,	
152.104.226.252,	200.160.0.6,	203.189.129.91,	213.117.112.120,	
157.25.168.85,	200.160.7.130,	203.93.165.152,	213.117.112.216,	
161.142.8.114,	200.211.206.59,	204.248.20.49,	213.117.112.35,	
167.205.8.2,	200.212.98.169,	205.206.148.57,	213.117.128.233	

参考文献

[1] GSM Assosiation. Gsm world. http://www.gsmworld.com/index.shtml.

[2] National Aeronautics and Space Administration (NASA). Gps applications exchange. http://gpshome.ssc.nasa.gov/.

[3] U.S. Coast Guard. LORAN-C. http://www.navcen.uscg.gov/loran/default.htm.

[4] Jr. Spilker, James J., and Bradford W. Parkinson. *Global Positioning System: Theory and Applications*, Vol. 1, chapter Overview of GPS Operation and Design, pp. 29–56. American Institute of Aeronautics and Astronautics, Inc., 1996.

[5] 社団法人 交通工学研究会. ITS - インテリジェント交通システム. 1999 年.

[6] 財団法人 自動車走行電子技術協会. ITS の標準化 2004. 2004 年 3 月.

[7] 佐藤雅明. インターネットにおける自動車情報の抽象化およびデータ辞書モデルの設計. Master's thesis, 慶応義塾大学大学院 政策・メディア研究科, 2001 年.

[8] 財団法人インターネット協会. インターネット白書 2004. 株式会社インプレス, 7 月 2004 年. ISBN : 4-8443-1948-5.

[9] J. Postel. RFC 791: Internet Protocol, September 1981.

[10] S. Deering and R. Hinden. RFC 2460: Internet Protocol, Version 6 (IPv6) specification, December 1998.

[11] APNIC, ARIN, and RIPE NCC. IPv6 address allocation and assignment policy, January 2003. http://www.ripe.net/ripe/docs/ipv6policy.html.

[12] R. Hinden, S. Deering, and E. Nordmark. RFC 3587: IPv6 global unicast address format, August 2003. Status: INFORMATIONAL.

[13] L. Daigle. RFC 3912: Whois protocol specification, September 2004. Status: DRAFT STANDARD.

[14] Yuji Sekiya, Hiromi Wakai, Shu Nakamae, Kenji Hirose, and Jun Murai. The Mechanism for Scalable Registry System with Aggregatable Address Allocation on WIDE 6bone. *IEICE Transactions on Communications*, Vol. E82-D, No. 4, pp. 888–895, January 1998.

[15] P. V. Mockapetris. RFC 1035: Domain names — implementation and specification, November 1987. Status: STANDARD.

[16] J. Postel. RFC 1591: Domain Name System Structure and Delegation, March 1994. Status: INFORMATIONAL.

[17] T. Hardie. RFC 3258: Distributing Authoritative Name Servers via Shared Unicast Addresses, April 2002. Status: INFORMATIONAL.

[18] Carnegie Mellon University Software Engineering Institute. Cert coordination center. http://www.cert.org/.

[19] CERT/CC. Incident Notes. http://www.cert.org/incident_notes/.

[20] Merit Network. The North American Network Operator's Group. http://www.nanog.org/.

[21] R. Bush, D. Karrenberg, M. Kosters, and R. Plzak. RFC 2870: Root Name Server Operational Requirements, June 2000. Status: BEST CURRENT PRACTICE.

[22] Jelena Mirkovic and Peter Reiher. A Taxnomy of DDoS Attack and DDoS Defense Mechanisms. *ACM SIGCOMM Computer Communications Review*, Vol. 34, No. 2, pp. 39–54, April 2004.

[23] Ryan Naralne. Massive DDoS Attack Hit DNS Root Servers, October 2002. http://www.esecurityplanet.com/trends/article.php/1486981.

[24] CERT Coordination Center. CERT Advisory CA-1999-14 Multiple Vulnerabilities in BIND, April 2000. http://www.cert.org/advisories/CA-1999-14.html.

[25] Jaeyeon Jung, Emil Sit, Hari Balakrishnan, and Robert Morris. DNS Performance and the Effectiveness of Caching. *IEEE/ACM Transaction on Networking*, Vol. 10, No. 5, pp. 589–603, October 2002.

[26] P. V. Mockapetris. RFC 1034: Domain names — concepts and facilities, November 1987. Obsoletes RFC0973, RFC0882, RFC0883. See also STD0013 . Updated by RFC1101, RFC1183, RFC1348, RFC1876, RFC1982, RFC2065, RFC2181, RFC2308. Status: STANDARD.

[27] CERT Coordination Center. CERT Advisory CA-1997-22 BIND - the Berkeley Internet Name Daemon, August 1997. http://www.cert.org/advisories/CA-1997-22.html.

[28] CERT Coordination Center. CERT Incident Note IN-2001-11, August 2001. http://www.cert.org/incident_notes/IN-2001-11.html.

[29] D. Eastlake 3rd. RFC 2535: Domain Name System Security Extensions, March 1999. Status: PROPOSED STANDARD.

[30] P. Vixie, O. Gudmundsson, D. Eastlake 3rd., and B. Wellington. RFC 2845: Secret Key Transaction Authentication for DNS (TSIG), May 2000. Status: PROPOSED STANDARD.

[31] D. Eastlake 3rd. RFC 2931: DNS Request and Transaction Signatures (SIG(0)s), September 2000. Status: PROPOSED STANDARD.

[32] D. Eastlake 3rd. RFC 2930: Secret Key Establishment for DNS (TKEY RR), September 2000. Status: PROPOSED STANDARD.

[33] RIPE Network Coordination Centre. RIPE/NCC. http://www.ripe.net/.

[34] Asia Pacific Network Information Centre. Apnic. http://www.apnic.net/.

[35] American Registry for Internet Numbers. Arin. http://www.arin.net/.

[36] African Network Information Center. AfriNIC. http://www.afrinic.net/.

[37] Latin American Network Information Center. Lanic. http://lanic.utexas.edu/.

[38] R. Hinden and S. Deering. RFC 3513: Internet Protocol Version 6 (IPv6) Addressing Architecture, April 2003. Status: PROPOSED STANDARD.

[39] IAB and IESG. RFC 3177: IAB/IESG Recommendations on IPv6 Address Allocations to Sites, September 2001. Status: INFORMATIONAL.

[40] D. Atkins and R. Austein. RFC 3833: Threat Analysis of the Domain Name System (DNS), August 2004. Status: INFORMATIONAL.

[41] Internet Engineering Task Force. IETF. http://www.ietf.org/.

[42] Yuji Sekiya. DNS spoofing software - uso800d. http://dnstap.nc.u-tokyo.ac.jp/USO800d/.

[43] Kazunori Fujiwara. dnsattack software. Not published globally.

[44] Widely Integrated Distributed Environment Project. http://www.wide.ad.jp/.

[45] Institute of Electrical and Inc. (IEEE) Electronics Engineers. IEEE 802.11 wireless world. http://www.802wirelessworld.com/index.jsp.

[46] Nevil Brownlee. NeTraMet. http://www.caida.org/tools/measurement/netramet/.

[47] Nevil Brownlee, Kc Claffy, and Evi Nemeth. DNS Root/gTLD Performance Measurement. *USENIX, LISA2001*, December 2001.

[48] The Cooperative Association for Internet Data Analysis. CAIDA. http://www.caida.org/.

[49] N. Brownlee. RFC 2720: Traffic Flow Measurement: Meter MIB, October 1999. Status: PROPOSED STANDARD.

[50] N. Brownlee. RFC 2721: TFM: Applicability Statement, October 1999. Status: INFORMATIONAL.

[51] N. Brownlee, C. Mills, and G. Ruth. RFC 2722: Traffic Flow Measurement: Architecture, October 1999. Status: INFORMATIONAL.

[52] N. Brownlee. RFC 2723: SRL: A Language for Describing Traffic Flows and Specifying Actions for Flow Groups, October 1999. Status: INFORMATIONAL.

[53] S. Handelman, S. Stibler, N. Brownlee, and G. Ruth. RFC 2724: RTFM: New Attributes for Traffic Flow Measurement, October 1999. Status: EXPERIMENTAL.

[54] 学術情報ネットワーク. SINET. http://www.sinet.ad.jp/.

[55] RIPE Network Coordination Centre. Ripe ncc dns monitoring. http://dnsmon.ripe.net/.

[56] B.Huffaker, A.Broido, kc claffy, M.Fomenkov adn K.Keys, E.Lagache, and D.Moore. Skitter AS Internet Graph. http://www.caida.org/tools/measurement/skitter/.

[57] M.Fomenkov, kc claffy, B.Huffaker, and D.Moore. Macroscopic Internet topology and performance measurements from the DNS root name servers. *USENIX, LISA2001*, December 2001.

[58] 加藤朗, 関谷勇司. ISP の DNS サーバの DNS トラヒックの解析. *IEICE Transaction on Communications*, Vol. J87-B, No. 3, pp. 327–335, 3 月 2004 年.

[59] Duane Wessels. Is Your Caching Resolver Polluting the Internet ? *SIGCOMM 2004 NetTS Wokrshop*, September 2004.

[60] Ryuji Somegawa, Kenjiro Cho, Yuji Sekiya, and Suguru Yamaguchi. The Effects of Server Placement and Server Selection for Internet Services. *IEICE Transaction on Communications*, Vol. E86-B, No. 2, pp. 542–551, February 2003.

[61] The Measurement Factory. dnstop. http://dnstop.measurement-factory.com/.

[62] MaxMind LLC. GeoIP free country. http://maxmind.com/.

[63] David Barr. dnswalk - DNS database debugger. http://www.visi.com/ barr/dnswalk/.

[64] S. Williamson, M. Kosters, D. Blacka, J. Singh, and K. Zeilstra. RFC 2167: Referral Whois (RWhois) Protocol V1.5, June 1997.

[65] MySQL AB. MySQL Database Server. http://www.mysql.com/.

[66] S. Woolf and D. Conrad. Identifying an Authoritative Name Server. Work in Progress, July 2004.