現在の名前解決システムの課題と汎用名前解決エンジンの提案

山田 竜也† 嶋田 雄二郎† 島津 伸行† 倉富 修† 岡 光秋† 岡本 利夫† 栄 光宏†

†株式会社東芝 SI 技術開発センター 〒183-8512 東京都府中市片町 3-22

E-mail: †{ymd, shimada, shimazu, kuratomi, oka, okamoto, sakae}@sitc.toshiba.co.jp

あらまし 現在の名前解決システムでは、DNS が利用できない環境での近隣ノードの名前解決、ノードが移動したときの DNS サーバ位置の自動発見、DNS サーバの適応的選択、ファイアウォールで分断されたネットワーク環境、一般に名前を公開しない機器への対応、などについて課題がある。これらの問題を解決するためにクライアントアプリケーションの実装に影響を与えずに、ローカルホストにおいてデーモンとして動作する汎用名前解決エンジンを実装し、それが実現する予定の機能の一部分を実装した。また、特定のネットワーク環境下でその動作を検証し、その有効性を確認したことを報告する。

キーワード 名前解決、IPv6、モバイル

Problems about Current Name Resolution System and A Proposal of The General-Purpose Name Resolution Engine

Tatsuya YAMADA† Yujiro SHIMADA† Nobuyuki SHIMAZU† Osamu KURATOMI†

Mitsuaki OKA† Toshio OKAMOTO† and Mitsuhiro SAKAE†

† Systems Integration Technology Center, Toshiba Corporation

3-22 Katamachi, Fuchu-shi, Tokyo, 183-8512 Japan

E-mail: †{ymd, shimada, shimazu, kuratomi, oka, okamoto, sakae}@sitc.toshiba.co.jp

Abstract On a current name resolution system, there are problems such that are how to resolve name of neighbor nodes without DNS, find automatically DNS server location at moving a node, select a DNS server adoptively, adopt network environment separated by firewall and deal with appliances of non-publishing name. In order to solve these problems, we propose the general-purpose name resolution engine and some modules working as a daemon process in localhost without any effect to client application implementations. Furthermore, we report its effectiveness by tests of it's features under specific network environment.

Keyword DNS, IPv6, Mobile

1. はじめに

IPv6[1]が普及してくると、あらゆるデバイスに対して IPv6 アドレスを付与してネットワークへ接続することが可能になる。また、接続形態も様々になることが予想されることから、ネットワーク構成を柔軟かつ動的に変更・運用することが必要となってくる。さらにこれらの環境に対して、一般のユーザが IP アドレスを覚えて機器を利用することは非常に困難なので、名前によってアクセスできる必要がある。

現在の名前解決システムでは DNS[2][3]が一般に用いられているが、様々な面で問題がある。DNS が利用できない環境での近隣ノードの名前解決、ノードが移動したときの DNS サーバ位置の自動発見、DNS サーバの適応的選択、ファイアウォールで分断されたネットワーク環境、一般に名前を公開しない機器への対応、などである。

これらの問題を解決するためにクライアントアプリケーションの実装に影響を与えずに、ローカルホストにおいてデーモンとして動作する汎用名前解決エンジンを設計し、予定している機能の一部を実装した。また、特定のネットワーク環境下でその動作を検証し、その有効性を確認した。

本論文の構成は以下のとおりである。2章で現在

の名前解決システムの問題点とその解決のために我々がとった手段､3章で汎用名前解決エンジンの概要、4章で実施した動作検証の概要、5章、6章で具体的な検証結果、についてそれぞれ述べる。

2. 名前解決システムの現状の課題

2.1. 問題点

現在の名前解決システムは DNS に強く依存しており、この DNS を利用するためには、ネットワークのどこかに静的にネームサーバを設置しておく必要がある。これによって、その管理のためのコストが必要になるばかりでなく、ある程度の技術的知識を備えた人間が管理にあたる必要がある。

また、ネットワーク障害やクライアントの移動によりネームサーバにアクセスできなくなったときに通信相手のホストとは接続性があるにもかかわらず名前ではアクセスできないという状況が起きることもある。

さらに、ファイアウォールが介在するネットワーク環境をまたいでクライアントを移動させると、双方で名前空間の構成木が異なっていることがあり、片方からもう一方の名前が解決できないという状況が起こるため、ユーザがネットワークの移動を気にして利用しなくてはならない。

そして､家庭内で IP ネットワークが普及するに従って､前述したように DNS に依存せずに､技術的知識の無い人間でも簡便に利用できるような名前解決方式が必要になってくる。さらに、プライバシを考慮した場合、例えば、家庭にある家電機器の名前解決は、家庭内のノードや外部からでも認証されたノードならばできて欲しいが、第三者からは名前が見えて欲しくない場合もある。

2.2. 解決へのアプローチ

以上のような問題は現在の名前解決の枠組みでは解決できない。これに対して、ローカルホストのリゾルバによる名前解決を肩代わりして実行する汎用名前解決エンジンを提案する。

2.1節で述べた問題について､今回取り上げたのは以下の二点である。

1. 利用しているネームサーバの生存管理
2. ファイアウォールで分断されたネットワークへの適応

1に関して､従来のリゾルバでは問合せ先のネームサーバの生存管理を行っていなかった。本提案の汎用名前解決エンジンを用いて、過去に問い合わせた履歴を管理しておき、一定期間応答を返さなくなっているネームサーバに対しては問合せを行わない。これによってネームサーバのフォールバックに要する時間を短縮し効率的な名前解決を実現する。

2に関して､ファイアウォールによって名前空間が分断されたネットワーク環境でそのようなドメインは存在しないという回答をするネームサーバと適切なレコードを回答するネームサーバがあったとき、これまでは最初に返ってきた答えを信用していた。これによって、接続性はあるのに名前ではアクセスできない状況が起こっていた。これを本提案の汎用名前解決エンジンによって、より適切なレコードを採用し、クライアントアプリケーションに返す機能を実現する。

3. 汎用名前解決エンジン

3.1. 概要

2.2節で述べたような機能は現在の名前解決の枠組みでは実現できないため、汎用名前解決エンジン[1]を実装・利用する。これは、ローカルホスト上にて一つのデーモンプロセスとして動作し、リゾルバによる名前解決を肩代わりする機能を持つ。さらに、モジュールとして最新の名前解決機能を実装することが可能であり、任意に機能を追加できる。また、クライアントアプリケーションからはローカルホストで動作するネームサーバに見えるため、アプリケーション自身の実装を変更することなしに利用可能である。構成を図 1に示す。

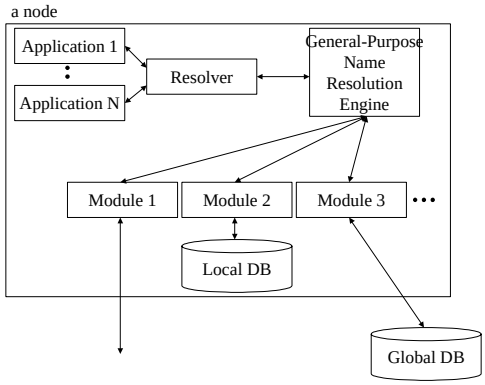

図1 汎用名前解決エンジン構成図

3.2. 実装した機能

今回、開発したものは、汎用名前解決エンジンの基幹部分と、2.2節における、1（ネームサーバの生存記憶管理)、2（返答レコードの適応的選択）

[1] 汎用名前解決エンジンは一構成要素であるが、我々が開発したシステムの総称を示す場合もある。

の機能を実現するモジュールである。

- ネームサーバの生存記憶管理

名前解決エンジンが知っているネームサーバにほぼ同時に問合せをし、結果の返ってきたものを採用する。数回（デフォルトは 5 回)、結果が返ってこないものをダウンしたと判断し、以降の問合せを行わない。図 2に動作フロー図を示す。

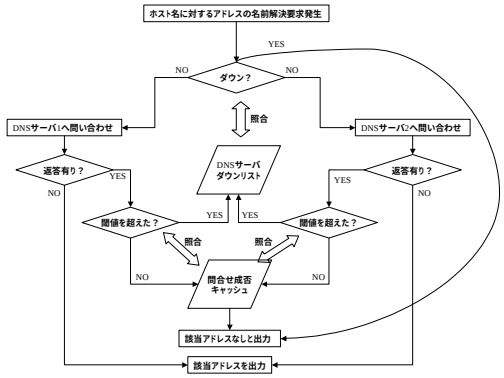

図2 ネームサーバの生存記憶管理

- 返答レコードの適応的選択

名前解決エンジンが知っているネームサーバにほぼ同時に問合せをする。得られた複数の回答が食い違っていた（NXDOMAIN と正しそうなレコード（A、AAAA など）が得られた）場合、正しそうなレコードを選択し、呼び出し元のアプリケーションに返却する。図 3に動作フローを示す。

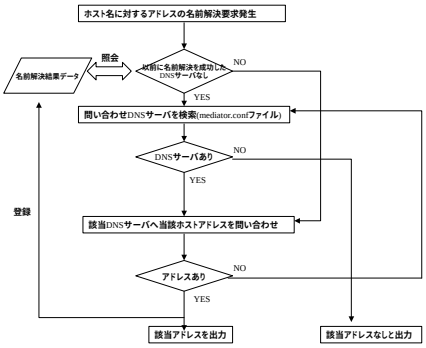

図3 返答レコードの適応的選択

4. 動作検証

3.2節で示した二つの機能について動作検証を実施した。「ネームサーバの生存記憶管理」を5章、「返答レコードの適応的選択」を6章で解説する。検証の基本的な手順として、通常のリゾルバを用いたときにおきる現象と、汎用名前解決エンジンを用いたときに起きる現象とを比較する。

使用したソフトウェアなどは以下のとおり。

- ホスト OS: FreeBSD-4.6R
- リゾルバ: ホスト OS に付属のもの
- ネームサーバ: BIND-9.2.1
- 汎用名前解決エンジン: 試作版
- ルータ: Cisco 1605R

また、以下では IPv4 についての検証結果を記述しているが、IPv6 についても同様の動作検証を行い、同様の結果を得た。

5. 検証 1（ネームサーバの生存記憶管理）

5.1. ネットワーク構成

本構成では、同一リンク上に 2 台のネームサーバ（それぞれプライマリとセカンダリ）があり、リゾルバが両方に対して問合せを行う設定になっているとき、優先して問い合わせることになっているネームサーバの応答が無かったときの動作を検証する。図 4にネットワーク構成を示す。

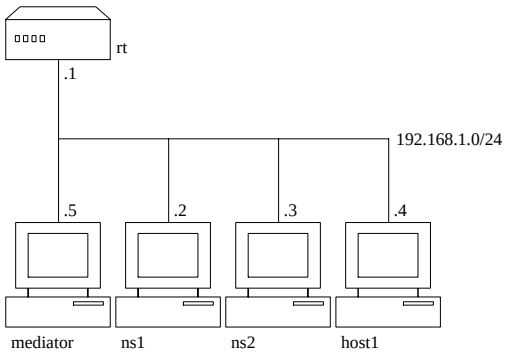

図4 検証用ネットワーク 1

凡例は以下のとおり。

- rt: ルータ。
- mediator: 汎用名前解決エンジンのクライアント。このノードから名前解決を行う。
- ns1: プライマリネームサーバ。
- ns2: ns1 のセカンダリネームサーバ。
- host1: 一般のホスト。

5.2. 手順

1) ns1, ns2 の両方が稼動している状態で rt 等に ping を行い、要した時間を測定する。

2) プライマリネームサーバ(ns1)が動作を停止している状態で 1)と同様に ping を行い、要した時間を測定する。また、その状況を tcpdump で観察する。

なお、汎用名前解決エンジンの動作時に、他の

ホストへの名前解決も正常に行えるかを試すため、ns1 以外のホストに対しても ping を行った。

5.3. 汎用名前解決エンジン未使用時

クライアント(mediator)の/etc/resolv.conf を以下のように設定する。まずプライマリサーバに問合せ、その後セカンダリサーバに問い合わせる。

```
nameserver 192.168.1.2
nameserver 192.168.1.3
```

- 両方のネームサーバにアクセス可能なとき

`rt, host1, mediator, ns1, ns2` に順に ping を実行し、tcpdump で出力を観察した。名前解決に要した時間は 1 秒未満である（下線部を比較）。ネットワークが正常に機能しているので当然の結果とみなせる。全て同じ結果なので最初のもののみ示す。

```
09:44:46.969164
        mediator.tao-mng.org.1089 >
          ns1.tao-mng.org.domain:
            10588+ A? rt.tao-mng.org. (35)
09:44:46.970058
         ns1.tao-mng.org.domain >
         mediator.tao-mng.org.1089:
       10588* 1/2/2 A rt.tao-mng.org (129)
09:44:46.970758
  mediator.tao-mng.org > rt.tao-mng.org:
                        icmp: echo request
```

- プライマリにアクセスできないとき

セカンダリにフォールバックするまで 5 秒かかっている（下線部を比較）。

```
09:47:05.777977
        mediator.tao-mng.org.1099 >
          ns1.tao-mng.org.domain:
            13060+ A? rt.tao-mng.org. (35)
09:47:05.948765
        mediator.tao-mng.org.1100 >
          ns1.tao-mng.org.domain:
39376+ PTR? 2.1.168.192.in-addr.arpa. (42)
09:47:10.787473
        mediator.tao-mng.org.1101 >
          ns2.tao-mng.org.domain:
            13060+ A? rt.tao-mng.org. (35)
09:47:10.788356
         ns2.tao-mng.org.domain >
         mediator.tao-mng.org.1101:
       13060* 1/2/2 A rt.tao-mng.org (129)
09:47:10.789029
  mediator.tao-mng.org > rt.tao-mng.org:
                        icmp: echo request
```

5.4. 汎用名前解決エンジン利用時

クライアントの/etc/resolv.conf を自分自身宛に解決するように設定する。

```
nameserver 0.0.0.0
```

汎用名前解決エンジンの設定ファイル（mediator.conf）にネームサーバのリストを書く。

```
nameserver: 192.168.1.2
nameserver: 192.168.1.3
```

予め汎用名前解決エンジンを起動しておく。

- 両方のネームサーバにアクセス可能なとき

汎用名前解決エンジン未使用時と同様なため略。

- プライマリにアクセスできないとき

クライアント mediator (192.168.1.5)から ns1 (192.168.1.2)と ns2 (192.168.1.3)へほぼ同時に問合せをし、先に回答を得た方に ping を行っている。これによって、名前解決を 1 秒未満で完結している。

```
11:39:17.615359
    192.168.1.5.1249 > 192.168.1.3.53:
                0+ A? rt.tao-mng.org. (35)
11:39:17.616081
    192.168.1.5.1249 > 192.168.1.2.53:
                1+ A? rt.tao-mng.org. (35)
11:39:17.616201
    192.168.1.3.53 > 192.168.1.5.1249:
             0* 1/2/2 A 192.168.1.1 (129)
11:39:17.630756
        192.168.1.5 > 192.168.1.1:
                        icmp: echo request
```

汎用名前解決エンジンの動作検証のため、アクセスできないネームサーバに対して名前解決を行ったときのステータスを一部出力した。これによると、ネームサーバのダウンを検出していることがわかる（網掛部）。

- ns1(192.168.1.2)

#	sent	recv	sendfail	down?
rt	1	0	0	UP
host1	2	0	0	UP
mediator	3	0	0	UP
ns1	4	0	0	DOWN
ns2	4	0	1	DOWN

6. 検証 2（返答レコードの適応的選択）

6.1. ネットワーク構成

本構成では、名前空間の異なるネットワークが相互に接続可能な状況を想定する。例えば、インターネットと企業内ネットワークを接続する場合、企業ではファイアウォールによって内部のネット

ワーク構成を外部に公開しない場合が多い。すなわち、ファイアウォールの外部にあるネームサーバと内部にあるネームサーバに問い合わせた結果が異なるという状況が起こる。このため、企業内ネットワークに接続しているにも関わらず、外部のネームサーバに問い合わせた場合、「そのような名前をもつノードは存在しない(NXDOMAIN)」という応答を返され、通信できない状態に陥る。

このような状況にあって、名前空間の異なるネットワークが接続しているときに、移動ノードの名前解決に関わる設定を変更せずにネットワークを移動した場合、どのような動作をするかを検証する。図 5にネットワーク構成を示す。

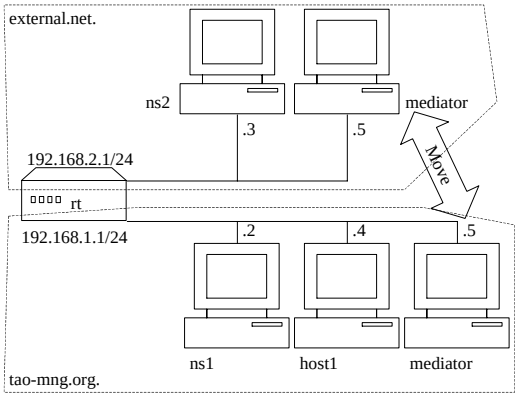

図5 検証用ネットワーク 2

凡例は以下のとおり。

- rt: 192.168.1.0/24 と 192. 168.2.0/24 のネットワークをルーティングする。
- ns2: external.net について権威をもつネームサーバであり、同時に external.net に対するルートの名前空間も持つ。
- ns1: tao-mng.org について権威をもつネームサーバであり、同時に tao-mng.org に対するルートの名前空間も持つ。
- mediator: 汎用名前解決エンジンのクライアント。192.168.1.0/24 と 192.168.2.0/24 のネットワークを移動するノード。このノードから名前解決を行う。
- host1: 一般のホスト。

ns2 と ns1 はお互いのことは知らないため、ns2 から tao-mng.org の名前空間にはアクセスできず、その逆も同様である。すなわちネットワークの到達性はあるが、名前解決のできない環境である。

6.2. 手順

1) mediator の/etc/resolv.conf の内容をそれぞれの場合で固定。
2) mediator を external.net ドメインと tao-mng.org ドメインを移動させ、両方のネットワークにおいて ns2.external.net.と ns1.tao-mng.org.への到達性を試す。

6.3. 汎用名前解決エンジン未使用時

クライアント(mediator)の/etc/resolv.conf を以下のように設定し、ネットワークを移動するときも変更しない。

```
nameserver 192.168.1.2
nameserver 192.168.2.3
```

- mediator が external.net 側に接続しているとき

最初に聞いたネームサーバ(192.168.1.2)が NXDOMAIN を返してきたら、それを信じてしまい、ns2.external.net を知っている 192.168.2.3 のネームサーバには問い合わせない。したがって、同じリンク上にいる ns2.external.net には接続できないという現象が起きた。

```
13:52:00.807005
    192.168.2.5.1027 > 192.168.1.2.53:
           3251+ A? ns2.external.net. (39)
13:52:00.812351
    192.168.1.2.53 > 192.168.2.5.1027:
                  3251 NXDomain* 0/1/0 (96)
13:52:00.815860
    192.168.1.2.53 > 192.168.2.5.1028:
                3252 NXDomain* 0/1/0 (109)
ping: cannot resolve ns2.external.net:
                                Unknown host
```

一方、ns1 は正しく A レコードが返却されたため、到達できた。

```
13:52:00.823729
    192.168.2.5.1029 > 192.168.1.2.53:
           50452+ A? ns1.tao-mng.org. (38)
13:52:00.826917
    192.168.1.2.53 > 192.168.2.5.1029:
           50452* 1/1/0 A 192.168.1.2 (68)
13:52:00.827615
        192.168.2.5 > 192.168.1.2:
                        icmp: echo request
```

- mediator が tao-mng.org 側に接続しているとき

/etc/resolv.conf の設定が同じため、external.net 側に接続しているときと同じ結果になった。出力は同様なため省略。

6.4. 汎用名前解決エンジン利用時

設定項目は5.4節と同様である。クライアントの

/etc/resolv.conf を以下のようにし、

```
nameserver 0.0.0.0
```

汎用名前解決エンジンが知っているネームサーバの名前を以下のように設定する。

```
nameserver: 192.168.2.3
nameserver: 192.168.1.2
```

- mediator が external.net 側に接続しているとき

汎用名前解決エンジンが知っている全てのネームサーバに問合せに行き、返ってきた結果で有効そうなものを選択してクライアント・アプリケーション（ここでは ping）に返却している。

ns2.external.net.への ping を試行。

```
13:27:55.401929
    192.168.2.5.1064 > 192.168.2.3.53:
            1+ A? ns2.external.net. (39)
13:27:55.402803
    192.168.2.3.53 > 192.168.2.5.1064:
                    1* 1/1/0 A[|domain]
13:27:55.402881
    192.168.2.5.1064 > 192.168.1.2.53:
            0+ A? ns2.external.net. (39)
13:27:55.407276
    192.168.1.2.53 > 192.168.2.5.1064:
                  0 NXDomain* 0/1/0 (96)
13:27:55.415688
       192.168.2.5 > 192.168.2.3:
                        icmp: echo request
```

ns1.tao-mng.org.への ping を試行。

```
13:27:57.604097
    192.168.2.5.1064 > 192.168.1.2.53:
             2+ A? ns1.tao-mng.org. (38)
13:27:57.604733
    192.168.2.5.1064 > 192.168.2.3.53:
             3+ A? ns1.tao-mng.org. (38)
13:27:57.605426
    192.168.2.3.53 > 192.168.2.5.1064:
                  3 NXDomain* 0/1/0 (96)
13:27:57.607590
    192.168.1.2.53 > 192.168.2.5.1064:
             2* 1/1/0 A 192.168.1.2 (68)
13:27:57.618538
       192.168.2.5 > 192.168.1.2:
                        icmp: echo request
```

- mediator が tao-mng.org 側に接続しているとき

こちらも同様に、external.net 側にいるときと同様にネームサーバからの回答の中で有効そうなものを選択してクライアント・アプリケーションに返却している。検証結果は同様なため省略。

7. おわりに

この論文では、名前解決における諸問題を解決するために汎用名前解決エンジンを提案した。実装では、リゾルバが複数のネームサーバを知っているときは全てのネームサーバに問合せ、もっともらしい結果を使用する機能、複数のネームサーバに問い合わせた結果、あるサーバがダウンしているときはそのサーバの回答を待たない機能、を実装した。加えて、その機能検証のために試験ネットワークを構築し、仕様どおりの機能が実現できていることがわかった。

今後は、DNS が存在しないネットワーク環境においても名前によってアクセスできるように、モジュールの実装を進める。また、機能モジュールが実現する機能に対して試験ネットワークにおける評価を実施していく。

8. 謝辞

本研究は、通信・放送機構(TAO)の委託研究テーマ「モバイル環境やセキュリティを考慮した名前解決方式とその検証環境の研究開発」の一環として行われているものである。ここに記して謝意を表す。

また、本研究にあたり、東芝研究開発センターの石山政浩氏、神明達哉氏、井上淳氏に有益な議論を頂いた。

文　献

[1] S. Deering, R. Hinden, “Internet Protocol, Version 6 (IPv6) Specification”, RFC2460, 1998.

[2] P. Mockapetris, “DOMAIN NAMES – CONCEPTS AND FACILITIES”, RFC1034, 1987.

[3] P. Mockapetris, “DOMAIN NAMES – IMPLEMENTATION AND SPECIFICATION”, RFC1035, 1987.