

複数無線基地局同時利用のためのLIN6の拡張

藤川 賢治[†] 森岡 仁志[†] 真野 浩[†]

[†]ルート(株)

E-mail: †{fujikawa,hmorioka,hmano}@root-hq.com

あらまし 本研究の目的は、複数の無線基地局から電波を受け取れるような環境で複数基地局を同時に活用しながらIPv6によるインターネット接続することである。このため移動透過プロトコルとしてはLIN6を用い、LIN6を改良することで、1) 複数基地局から受け取る prefix を同時に利用可能とし、2) また位置情報取得にかかる遅延を短縮することで通信時のSYN及びSYN+ACK送信時の遅延時間が短縮され、高速ハンドオーバーが可能となった。また本方式に関する評価を行う。

キーワード 無線インターネット,IPv6,LIN6,複数基地局,MIAKO

LIN6 Extensions for Simultaneous Utilization of Multiple Wireless Base Stations

FUJIKAWA KENJI[†], MORIOKA HITOSHI[†], and MANO HIROSHI[†]

E-mail: †{fujikawa,hmorioka,hmano}@root-hq.com

Abstract The purpose of this paper is to implement a mechanism, that a wireless mobile terminal can utilize multiple wireless base stations simultaneously and have access to the IPv6 Internet via them, under the environment where it receives different waves from them. For this purpose, we adopted LIN6 as a mobility-transparent protocol, and implements the following functions by adding extensions to LIN6; 1) a LIN6 node can utilize the multiple prefixes received from multiple wireless base stations, and 2) the delays of sending TCP SYN and SYN+ACK are reduced by means of improving the procedure of getting location information. We also evaluate our proposed methods.

Key words Wireless Internet,IPv6,LIN6,multiple base stations,MIAKO

1. はじめに

近年、無線LAN(IEEE802.11a/b/g)の普及により、移動しながらインターネットを利用できる環境が整ってきている。またIPv6に対応した機器も市場で手に入るようになってきている。

本研究の目的は、複数の無線LANアクセスポイントなどの無線基地局から電波を受け取れるような環境で複数基地局を同時に活用しながらIPv6によるインターネット接続することである。

このために我々はLocation Independent Network for IPv6(LIN6)[1]を利用する。LIN6はIPv6においてモビリティとマルチホーミングを提供するプロトコルである。LIN6では各ノードは64ビットのノードIDを持ち、128ビットのIPv6アドレス全体はLocatorと呼ばれる。どのようなLocatorを持っていたとしても、各ノードはそのIDによって識別される。

無線移動端末は複数基地局から複数のIPv6 Prefixを受け取

り、異なる複数のPrefixからなるIPv6アドレス(Locator)を持つことになる。また無線移動端末での利用であるため、IPv6アドレスは頻繁に追加・削除される。よって複数のIPv6アドレスを活用できるに加えて、このような頻繁なアドレスの追加・削除に即座に対応して利用できるアドレスを即座に使えるようにする仕組みが必要である。

しかし、これまでのLIN6では次のような問題点があった。

- アプリケーションから複数のLocator情報を活用できない
 - connect時のSYNパケットがMNに送信されるまで時間がかかる
 - accept時のSYN+ACKがMNに送信されるまで時間がかかる
 - DNS resolverとLIN6 resolverが別々に動作し、情報が共有できておらず、DNS queryとLIN6 queryが遅延される
- そこでLIN6に次のような機能改良及び拡張を施した。
- アプリケーションがCNのLocatorを指定できる仕組み

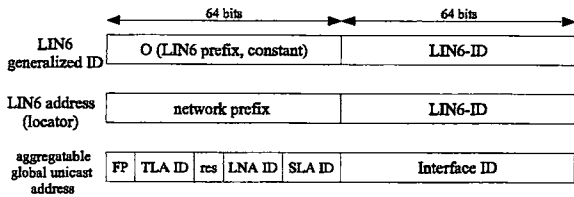


図2 The LIN6 generalized ID and LIN6 address

の導入

- MA のアドレスが MN の AAAA だけからも引けるように変更

- LIN6 resolver と DNS resolver との連携
- 受信パケットの始点アドレスも Locator として採用
これにより LIN6 において次のような改良を行うことができた。

- LIN6 ノード上のアプリケーションから複数基地局から受け取る複数 Locator 情報が活用可能

- TCP SYN や SYN+ACK 送信時の遅延を短縮

本研究によって、複数基地局を同時に活用しながら IPv6 インターネットに接続する方法が確立された。

以下、2 章では従来の LIN6 とその問題点について述べ、3 章では従来 LIN6 を拡張した *MIAKO LIN6* を提案する。4 章では *MIAKO LIN6* の実装と評価についてを述べ、5 章では LIN6 と他の位置透過・マルチホーミングプロトコルと比較する。

2. 従来 LIN6 の概要とその問題点

提案手法の説明に先立ち、従来 LIN6 の概略を述べ、LIN6 ノードから MN に対して通信を行う場合の動作手順について説明し、その問題点を明かにする。

2.1 従来 LIN6 の概要と MN への通信時における動作

LIN6 のアドレス形式には O+ID と Prefix+ID の二つがある。前者は DNS の正引きの AAAA エントリに書かれる形式である。後者は各インタフェースに付けられるアドレスであり、またパケットに実際に埋め込まれるアドレスである。(図 2)

ある ID の Prefix を管理する Mapping Agent (MA) は、DNS を用いて ID からその MA のアドレスを検索することができる。MN は MA に自身が現在持つ Prefix を常に登録・更新しており、MN と通信しようとする Correspond Node (CN) は MA に問い合わせることで、MN が現在持つアドレス (Locator と呼ぶ) を取得することができる。

以下に MN へ TCP コネクションを確立する場合の手順を示す。(図 1)

- アプリケーションが DNS の AAAA レコードを引き、O+ID という IPv6 アドレスを得る。
- アプリケーションが O+ID に対して connect システムコールを発行する。
- socket インタフェースによりカーネルが取得した MN の Locator は O+ID なので、パケットを送信を遅延させ、lin6resolvd と呼ばれる daemon に ID から Locator の解決を依頼するための

upcall を行う。

- lin6resolvd が DNS query により MN の MA のアドレスを調べ、MA に対して LIN6 query を行い Locator を解決する。そして ID の現在のアドレスが Locator であることをカーネルに登録する。

- カーネル内で遅延していたパケットが Locator で示されるアドレスに送信される。

2.2 従来 LIN6 における問題点

従来 LIN6 では次のような問題点がある。

- アプリケーションから複数の Locator 情報を活用できない。

- DNS resolver と LIN6 resolver が別々に動作し、情報が共有できておらず、DNS query と LIN6 query が遅延される。このため次のような問題が生じる。

- connect 時の SYN パケットが MN に送信されるまで時間がかかる。

- accept 時の SYN+ACK が MN に送信されるまで時間がかかる。

まず (1) の問題点について説明する。

MN が複数アドレスを持っている場合に、アプリケーションが対応する MA に関問い合わせれば、その複数アドレスを取得することはできる。しかし LIN6 を使うためにはアプリケーションは O+ID というアドレスで socket を使用しなければならないという仕様になっていたため、任意の Locator を選んでパケットを送信することはできない。

次に (2) の問題点について説明する。従来 LIN6 においては、LIN6 ID に対する Locator の解決は、その ID を用いた通信が行われることをカーネルが認識して、LIN6 resolver に解決を upcall し、その結果を得るまで遅延されてしまう。

最後に (3) の問題点について説明する。LIN6 では、パケット受信時にもその応答処理が遅延されてしまう場合がある。それは初めて MN からパケットを受信する場合であり、例えば MN から TCP SYN を受けた場合などである。この場合、カーネルが TCP SYN+ACK パケット送信をしようとして、Locator を解決するため、前節の 4) の手順を行うことになる。

上述の問題は次のような環境下で特に問題となる。無線移動端末は複数基地局から複数の IPv6 Prefix を受け取り、異なる複数の Prefix からなる IPv6 アドレス (Locator) を持つことになる。また無線移動端末での利用であるため、IPv6 アドレスは頻繁に追加・削除される。

このような環境下では、複数の IPv6 アドレスを活用できることに加えて、頻繁なアドレスの追加・削除に即座に対応して利用できるアドレスを即座に使えるようにする仕組みが必要である。次章で上述の問題を解決する LIN6 の拡張である *MIAKO LIN6* を提案する。

3. *MIAKO LIN6* 方式の提案

前章で述べた従来 LIN6 の問題点を解決するための *MIAKO LIN6* を本章で提案する。*MIAKO* (Mobile Internet Access in Kyoto) とは、京都を中心とした無線インターネットアクセスを実

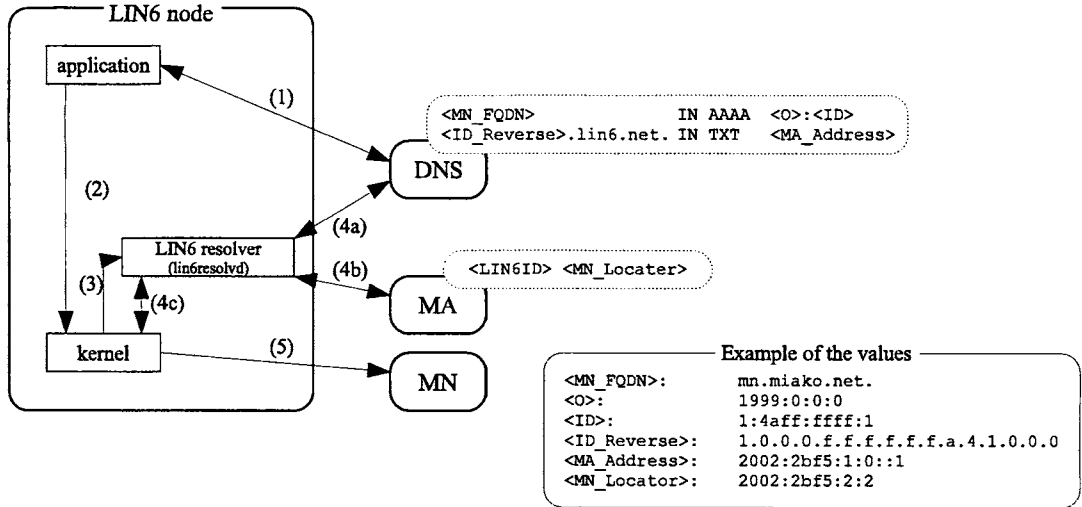


図1 The procedure of TCP connection to an MN in the current LIN6

現するためのプロジェクトであり[2],[3]、そのプロジェクト一貫として MIAKO LIN6 を開発している。

MIAKO LIN6 では次のような拡張を行なう。

(1) アプリケーションが Locator を指定できるよう、カーネルの動作を変更

(2) MA の位置がすぐに分かるよう、DNS エントリ登録方法を変更

(3) DNS の正引き情報を活用して MA の query を素早く行うための新しいデーモン lin6dnsproxy の導入

(4) 受信したパケットが LIN6 ID 形式であった場合、そのパケットの始点アドレスも、locator として採用以下、それぞれの拡張項目について説明する。

3.1 アプリケーションによる Locator 指定のためのカーネル動作の変更

従来 LIN6 では、

- MN のアドレスを O+ID として指定する必要があり、
- Prefix+ID 形式のアドレスのパケットへの埋め込みはカーネル内で行った。

MIAKO LIN6 ではカーネルの動作を以下のように変更した。

- MN のアドレスを O+ID として指定したときの動作は同じ
- MN のアドレスを Prefix + ID として指定した場合、Prefix 部はカーネルで書き換えずに、LIN6 として動作

これによって、CN が Locator を複数持っているときに、アプリケーションは Prefix+ID で CN を指定することで、特定の Locator に対する通信を行うことができるようになる。

3.2 DNS 登録方法の変更

MA の位置がすぐに分かるよう、DNS エントリ登録方法を変更する。

従来 LIN6 では図 3 のように MA のアドレスが登録されていた。

```

<MN_FQDN>          IN AAAA <O>:<ID>
<ID Reverse>.lin6.net. IN TXT <MA_Address>

```

Example:

```

mn.miako.net.          IN AAAA 1999::1:4a00:0:1
1.0.0.0.0.0.0.0.0.a.4.1.0.0.0.lin6.net.
                        IN TXT 2002:2bf5:1:0::1

```

図3 DNS Records of the current LIN6

```

<MN_FQDN>          IN AAAA <MA_Prefix>:<ID>
<ID Reverse>.lin6.net IN PTR <MN_FQDN>

```

Example:

```

mn.miako.net.          IN AAAA 2002:2bf5:1:0:8302:2bf5:1:1
1.0.0.0.1.0.0.0.5.f.b.2.2.0.3.8.lin6.net.
                        IN PTR mn.miako.net.

```

図4 DNS Records of MIAKO LIN6

MIAKO LIN6 では図 4 のように MA のアドレスを登録できるようにする。

MIAKO LIN6 では MA のアドレスは <MA_Prefix>:0100:: と解釈される。AAAA を検索するだけでよいため、迅速な MA のアドレス解決が可能である。また ID 空間の委譲 (delegation) を必ずしも必要としないという利点もある。(ただし ID の乗取りを防ぐには逆引き設定が必要であるため、そのためには ID 空間の委譲は必要である)

また ID 部は従来 LIN6 では

```
0001:4aXX:XXXX:XXXX
```

の形式であったが、MIAKO LIN6 では

```
83XX:XXXX:XXXX:XXXX
```

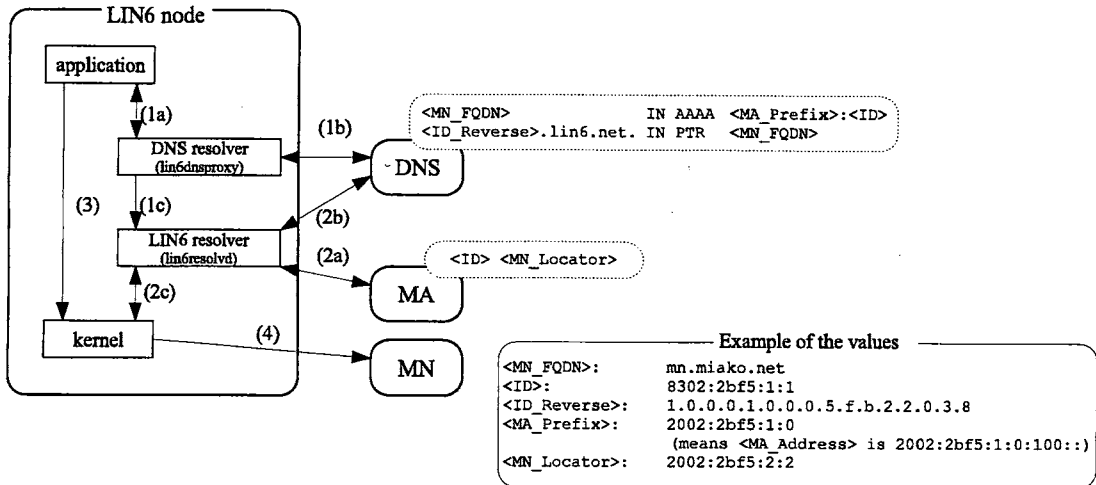


図5 The procedure of TCP connection to an MN in MIAKO LIN6

も利用できるような変更し、LIN6 で使える空間を拡張した。

これにより、IPv6 Prefix を委譲されると自動的に LIN6 ID 空間も委譲される仕組みが導入できるようになった。我々は、Prefix として 2002:2bf5:1::/48 というアドレスを所有すれば、ID として 8302:2bf5:1:0000~8302:2bf5:1:ffff を使ってよいという規則を採用している。

また上の例は 6to4 のアドレスであり、例に示されているように IPv4 のアドレス (例では 43.245.0.1 = 0x2bf50001) を持っているだけでも LIN6 ID の空間が委譲されるようになる。

3.3 LIN6 DNS proxy デーモンの提案

新しく lin6dnsproxy と呼ぶ daemon を提案する。主な機能は次の三つである。

- (1) 一般的な DNS resolver 機能 (正確には forwarder 機能)
- (2) AAAA の query に対して下位 64bit が LIN6 ID だと判断すると、アプリケーションには O+ID として通知。
- (3) AAAA の query に対して下位 64bit が LIN6 ID だと判断すると、ID と AAAA (すなわち MA のアドレス Prefix) を lin6resolvd に通知。

(1) と (2) の機能により、図 4 のような DNS エントリの場合でも、アプリケーションは O+ID の形式のアドレスを取得できる。

アプリケーションが DNS の正引きを lin6dnsproxy に依頼し、lin6dnsproxy が MN の AAAA を得られたら即座に lin6dnsproxy は lin6resolvd に MA の query を促す。従来 LIN6 では lin6resolvd はカーネルからの upcall を待って MA query を行っていたが、MIAKO LIN6 ではそれを必要としない。

このように MIAKO LIN6 では MA への query が遅延されるという問題を解決することができる。

MN に接続するさいの動作

図.5 において TCP で MN に接続するときの手順を示し、どのように問題が解決されているかを示す。

(1) アプリケーションが DNS の AAAA レコードを lin6dnsproxy 経由で引くと O+ID が返ってくる。(a, b) その際 lin6dnsproxy は Locator を取得し、lin6resolvd にも ID と MA のアドレス情報を通知する。(c)

(2) LIN6 resolver は MA に関ひ合わせ CN の Locator を取得する。(a) それと同時に ID からの DNS の逆引き及び正引きを行う。(b) (b) の結果、異なる MA が見つかった場合は MA への query をやり直す、DNS が正常に設定されている限り、このやり直しは起きないカーネルに、ID に対応する取得した locator を登録する。(c)

(3) アプリケーションが connect システムコールを発行する。(2) とは並列実行である。

(4) (2) の処理が終了するまでパケットが遅延され Locator で示されるアドレスに送信される。

処理 2. の終了を待たずに処理 3. が行えるため、MIAKO LIN6 では従来 LIN6 よりも、TCP SYN 時のパケットを遅延させずに送信することができる。このように従来 LIN6 の問題のうちの一つが解決される。

3.4 受信パケットの始点アドレスを Locator として登録

既に述べたように、TCP SYN を受ける時のように初めて MN からパケットを受信する場合、従来 LIN6 では Locator 解決のために応答処理が遅延される。

我々はその問題を解決するため、LIN6 ID パケット受信時にその始点アドレスの ID のカーネル内マッピングテーブルが無ければ、受信パケットの始点アドレスを Locator としてエントリに追加するという拡張を施す。

これによって MIAKO LIN6 ではエントリの存在しない始点アドレスのパケットを受信した場合でも、即座に TCP SYN+ACK などの返答をすることができる。また同時に、従来 LIN6 と同様にカーネルは lin6resolvd に ID に対する Locator の解決の依頼も行われ、別途 Locator が得られた場合はカーネル内のマッ

ピングテーブル内のその ID のエントリは Locator で上書きされる。

4. 実装及び評価

MIAKO LIN6 方式を NetBSD 3.1 上に実装し、図 6 のような実験環境を構築した。i386 ベースの PC を無線端末 (MN) 及び通信相手 (CN)、無線基地局ルータ (BS1, BS2) とした。無線基地局ルータは IEEE802.11a/b/g に対応している。なお BS1 は MN に対する MA の機能も兼ね備えている。

実験環境化では MN は BS1 及び BS2 の無線基地局ルータに同時接続しており、複数の IPv6 アドレスを所有している。また CN もマルチホームしており、複数アドレスを所有している。

CN から MN に対して UDP で映像ストリームを転送した。その際 MN が持つ二つのアドレスに同時に同じパケットを転送することで、MN は BS1 経由と BS2 経由の二つの経路からパケットを受け取る。BS1 の無線のみを停止した場合でも、BS2 の無線のみを停止した場合でも、MN で受信した映像の再生が途切れないことが確認できた。

CN 上のストリームを伝送するためのアプリケーションは、従来 LIN6 では MN が現在持っているアドレスを直接指定することはできなかったが、MIAKO LIN6 では直接指定することができ、複数経路を同時に利用することができるようになった。

次に、

(1) DNS の正引きを行い、CN から MN へ telnet コマンドによる TCP コネクションを確立する場合及び、

(2) CN が MN からの TCP SYN パケットを受け取り TCP SYN+ACK パケットを返信する場合の動作検証及び実測による評価を行う。

それぞれ図 7 及び図 8 に示す動作シーケンスで動作することが確認できた。また図には MN 上でのパケット送受信タイミングの実測値 (平均値) も記載した。ただし (a) DNS query/reply と (c) LIN6 query/reply は DNS サーバ/LIN6 サーバと実験環境とのネットワーク的な距離によって大きく変動するため、5ms-500ms と幅のある表記とした。

図 7 から、MIAKO LIN6 は従来 LIN6 より SYN パケットが出るまでの時間が短縮できていることが示される。ただしこれは、(c) LIN6 query/reply にかかる時間に大きく左右される。(c) が 5ms 程度と極端に短い場合や、逆に 500ms 程度と極端に大きな場合、どちらの場合でも MIAKO LIN6 の従来 LIN6 に対する長所は薄れる。

図 8 から MIAKO LIN6 の SYN+ACK を返すまでの時間は、通常の IPv6 と変わらず、従来 LIN6 に比べて非常に優れていることが分かる。

5. 関連研究との比較

本章では MIAKO LIN6 を、他の位置透過プロトコルやマルチホームに対応したプロトコルである、MIPv6 [5] や MAT [6]、HIP [7], [8] と比較する。

5.1 MIPv6 及び MAT との比較

Mobile IPv6 (MIPv6) は IPv6 における標準の移動透過プロト

コルである。その特徴として、トンネルによる CN と MN との Home Agent (HA) を介した通信と、Home Address Option (HoA) による CN と MN 直接転送も提供などが挙げられる。

Mobile IP with Address Translation (MAT) では Mobile Address (MIPv6 での Care-of address に相当) と Home Address の変換テーブルを CN 内に持つことに特徴がある。これによって MN が HAO を付けて CN にパケットを出すのは最初にパケットを送るときと、移動した直後のみですみ、それ以外の時は HOA を付ける必要がない。

元々 LIN6 は MIPv6 と比べて、パケットヘッダサイズが通常の IPv6 パケットと変わらないという利点を持っていた。(パケットがトンネリングのためなどにサイズが大きくなると、様々な問題が引き起こされていることが一般に知られている) またわずかではあるが、HoA を用いるものと比べると、オプションヘッダを使わない分、帯域の利用効率が高い。しかし MIPv6 や MAT に比べると、LIN6 は MN に対するパケットの送信や返信が遅延されるという問題を持っていたが、今回 MIAKO LIN6 ではこれらの問題を解決した。

また MIAKO LIN6 では LIN6 ノードが複数 IP アドレスを持っているときに、他のノードがそのアドレスを指定し、そのアドレスを利用した通信を簡単にできるようになっている。

5.2 HIP との比較

Host Identity Protocol (HIP) は新しいホスト ID 空間を導入し、マルチホームやモビリティに対応することが期待されている。

しかし HIP は HIP ノード間の通信に先立ち、IPSec ベースのネゴシエーションを必要とする。つまり TCP のような通信は非常に遅延させられることになる。代わりに HIP は常に強いセキュリティを通信相手と確立することができる。

HIP に比べて MIAKO LIN6 は、LIN6 ノード間のコネクションの確立を圧倒的に高速に行えるという利点を持っている。一方セキュリティに関しては HIP ほど強くない。しかし強固なセキュリティを必要とするときには、通常の IPv6 と同じく LIN6 ノード間でセキュリティアソシエーションを確立すればよいため、それが問題になることはない。

6. 結論

本研究では LIN6 に次のような機能改良及び拡張を施した。

- アプリケーションが CN の Locator を指定できる仕組みの導入
- MA のアドレスが MN の AAAA だけからも引けるように変更
- LIN6 resolver と DNS resolver が連携し MN のアドレス情報を共有
- 受信パケットの始点アドレスも Locator として採用する。これにより LIN6 において次のような改良を行うことができた。
- LIN6 ノードが複数基地局から受け取ったアドレスを活用
- TCP SYN や SYN+ACK の送信時間を短縮

MIAKO LIN6 によって複数の無線基地局から電波を受け取れる

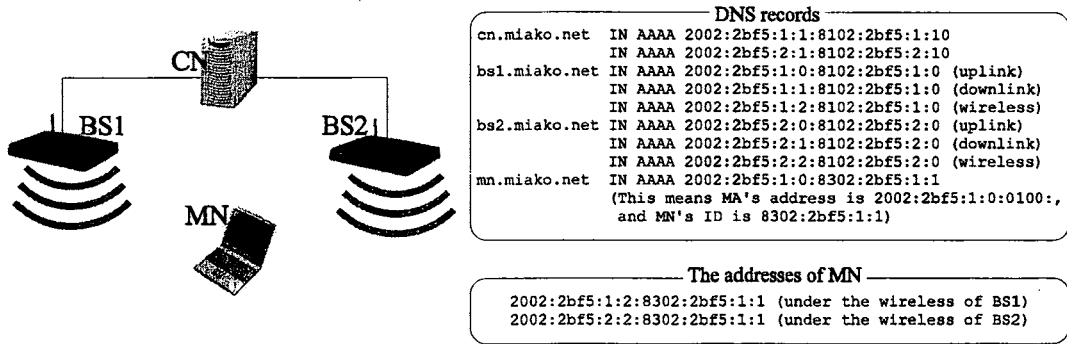


図 6 A MIAKO LIN6 network environment

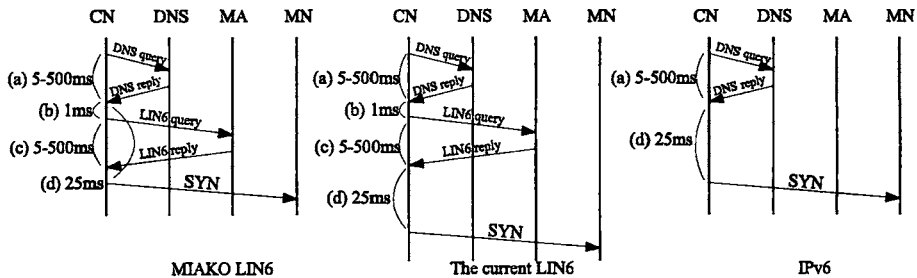


図 7 The protocol sequence of sending TCP SYN

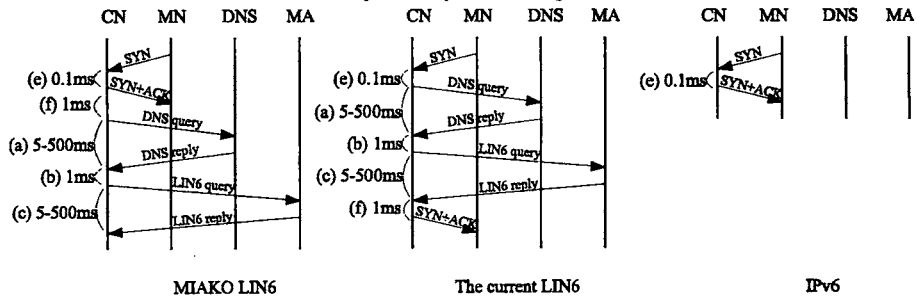


図 8 The protocol sequence of receiving TCP SYN and sending TCP SYN+ACK

ような環境で複数基地局を同時に活用しながら IPv6 によるインターネット接続を行えるようになった。

今後は IPsec との連携による安全な通信路を確保した儘でのハンドオーバーや、IPv4 ホストや通常の IPv6 ホストの通信時にもモビリティ機能を活用する方法について検討する予定である。

文 献

[1] M. Ishiyama, M. Kunishi, and F. Teraoka, "An Analysis of Mobility Handling in LIN6," International Symposium on Wireless Personal Multimedia Communication, 2001.

[2] T. Komura, K. Fujikawa, and Y. Okabe, "The MIAKO.NET Public Wireless Internet Service in Kyoto," Proc. of WMASH 2003, September 2003.

[3] <http://www.miako.net>.

[4] B. Carpenter and K. Moore, "Connection of IPv6 Domains via IPv4 Clouds," RFC3056, February 2001.

[5] D. Johnson, C. Perkins, and I. Arkkio, "Mobility Support in IPv6," RFC3775, June 2004.

[6] R. Inayat, R. Aibara, K. Nishimura, T. Fujita, and K. Maeda, "An End-to-End Network Architecture for Supporting Mobility in Wide Area Wireless Networks," IEICE Transaction on Communication, Vol.E87-B, No.6, pp.1584-1593, June 2004.

[7] R. Moskowitz, P. Nikander, P. Jokela, and T. Henderson, "Host Identity Protocol," Internet Draft, draft-ietf-hip-base-03.txt (work in progress), June 2005.

[8] R. Moskowitz and P. Nikander, "Host Identity Protocol Architecture," Internet Draft draft-ietf-hip-base-03.txt (work in progress), August 2005.