2004－DPS－118　(2)
2004／6／3

匿名性を考慮したIPモビリティ通信機構

安間健介[†]　石山政浩[†††]
國司光宣[†]　寺岡文男[††]

本研究では、移動透過性プロトコルである LIN6 をベースに匿名性を考慮した IP モビリティ通信機構を提案する。近年、携帯端末の性能向上と移動通信機器の普及に伴い、移動透過性プロトコルへの要求が高まってきている。既存の移動透過性プロトコルでは通信相手を特定できる識別子をパケットから取得することが可能であるため、匿名性のある通信を行うことは困難であった。そこで、initiator と responder 間でランダムに選んだ ID を一時的な識別子として利用し、匿名性のある通信を可能とする機構を提案する。

Communication Mechanism for an Anonymous-Friendly IP-Mobility

Kensuke Yasuma,[†] Masahiro Ishiyama,[†††] Kunishi Mitsunobu[†]
and Fumio Teraoka[††]

This paper presents communication mechanism for an anonymous-friendly IP-Mobility based on LIN6. Exiting mobility protocols are identified to the communication node by packets and they cannot provite anonimity to the node. So initiator and responder use temporal random ID on this protocol. It provide anonimity to initiator and responder.

1. はじめに

携帯端末の性能向上と、移動通信機器の普及に伴い、移動先から携帯端末を利用したインターネットへのアクセス、いわゆるモバイルコンピューティングが活発に行われるようになってきた。同時に、Mobile IPv6[1] や LIN6[5] といった移動透過性プロトコルへの要求も高まってきている。これらのプロトコルはネットワークレイヤで移動透過性を提供するプロトコルであり、アプリケーションへの影響がないことから、多様な応用が期待されている。しかし一方においてプライバシに対する懸念もある。多くの移動透過性プロトコルにおいてネットワークレイヤで定義される識別子、たとえば Mobile IPv6 では Home Address、LIN6 では LIN6ID を必要とし、移動透過性の恩恵を受けるためにはこれを通信相手に通知しなくてはならない。すなわち、通信パケットから通信相手を特定することが可能であり、匿名性を持った通信を行うことは困難である。しかし、現在の電話のように発呼した相手に対して自分の電話番号、すなわち識別子を通知したくないというニーズは大きいと考えられる。また、中間ノードに悪意のある第三者が潜んでいた場合、その悪意のある第三者がパケット内の識別子から移動ノードをトレースできるという問題点がある。

Mixes[6]、Crowds[7]、P5[8] といった既存の匿名通信を移動透過性プロトコルに適用することはルーティング、遅延、帯域幅、オーバヘッドを考慮すると望ましくない。我々は LIN6 における匿名通信の実現方法に関する一考察を提案している[3]。しかし、既存の提案方式では、完全な匿名性を得ることができないと言える。本提案手法は通信者間でランダムに選んだ ID を一時的な識別子として利用し、悪意のある第三者のなりすましに対処するための認証機構を備える。本研究では、移動透過性プロトコルの一つである LIN6 をベースに匿名性を提供するための手法を提案する。

2. 移動透過性プロトコルにおいて要求される匿名性

移動するノードは帯域幅の狭い無線を使用する可能性が高いと考えられるため、Mixes、Crowds、P5 といった既存の匿名通信を移動透過性プロトコルに適用することはルーティング、遅延、帯域幅、オーバヘッドを考慮すると望ましくない。このように移動透過性プロトコルにおける匿名性は一般的な通信における匿

† 慶應義塾大学大学院 理工学研究科
Graduate School of Science and Technology, Keio University.
†† 慶應義塾大学 理工学部
Faculty of Science and Technology, Keio University.
††† 東芝研究開発センタ
Corporate Research and Development Center.

名性の要求とは異なる。本章では移動透過性を失うことなく得られる匿名性について考察する。

一つ目の要求として、受信者に送信者のノード識別子を知られない匿名性が考えられる。しかし、既存の移動透過性プロトコルの多くはネットワークレイヤで定義されるノード識別子を通信相手に通知しなくてはならない。

二つ目の要求として、悪意のある第三者に追跡されない匿名性が考えられる。しかし、既存の移動透過性プロトコルの多くはパケットにノード識別子を含むため、悪意のある第三者による追跡が可能である。

三つ目の要求として、悪意のある第三者に盗聴された時の両端のホストの匿名性が考えられる。しかし、既存の移動透過性プロトコルの多くはパケットに含まれるノード識別子から悪意のある第三者に両端のノードを特定される。

移動透過性プロトコルにおいて両端のノードは移動するので、ノードのネットワークプレフィクスよりも両端のノード識別子つまり誰であるかを隠蔽したいと考えられる。

このような要求を満たす移動透過性プロトコルを実現する。

3. 移動透過性プロトコル LIN6 の概要

LIN6 は、移動透過性を保証する新しいネットワークアーキテクチャである LINA を IPv6 上へ適用した移動透過保証プロトコルである。LIN6 では位置指示子とノード識別子という 2 つの情報を概念的に分離する。ネットワーク層より上位層では、ノード識別子を用いた位置に依存しないコネクションを確立し、ネットワーク層では、位置指示子を用いた経路制御を行うことにより移動透過性を実現する。

3.1 縮退アドレスモデル

現在、IPv6 の通信で主に使用されている Aggregatable Global Unicast Address(AGUA)[2] は、上位 64bit がネットワークプレフィクス、下位 64bit がインターフェース ID を示す。LIN6 のアドレスモデルは、アドレス構造の 128bit 全体を位置指示子とし、位置指示子の中に 64bit のノード識別子を縮退させる。ネットワーク層より上位層では、LIN6 汎用識別子を用いる。LIN6 汎用識別子の上位 64bit は位置に依存しない LIN6 prefix であり、下位 64bit はノード識別子である LIN6ID である。ネットワーク層では、LIN6 アドレスを用いる。LIN6 アドレスの上位 64bit は現在のネットワークプレフィクスである。

LIN6 アドレスは従来の AGUA 形式と互換性を保ちながら、LINA における縮退アドレスと同様に位置指示子とノード識別子という分離された 2 つの情報を保持する。LIN6 におけるノード識別子は LIN6 ID と呼ばれる。

3.2 LIN6ID と LIN6 アドレスの対応づけ

LIN6 では、通信を開始する際に MA を用いて、LIN6ID と現在のネットワークプレフィクスとの対応づけを取得する。LIN6 では、この対応づけを mapping と呼び、mapping を管理する機構として Mapping Agent(MA) を用いる。MA は、要求に応じてネットワークプレフィクスを通知する。LIN6 において LIN6 ノードと MA 間には信頼関係があり、各 LIN6 ノードは LIN6ID に対応する MA に現在の位置を保持される。

3.3 LIN6 の動作概要

LIN6 の動作概要を図 1 に示す。移動ノードは MA へ現在のネットワークプレフィクスを登録する (図 1(1))。あるノードが移動ノードと通信を開始する場合、まず DNS サーバへ移動ノードの mapping を管理している MA を問い合わせる (図 1(2))。DNS には、あらかじめ移動ノードの LIN6 ID とそれを管理する MA のアドレスという静的な対応情報を登録しているため、DNS から移動ノードの mapping を管理する MA のアドレスを取得することが可能である (図 1(3))。次に通信ノードは、MA へ移動ノードの現在のネットワークプレフィクスを要求する (図 1(4))。MA は、事前に登録されている移動ノードのネットワークプレフィクスを通信ノードへ通知する (図 1(5))。LIN6 では、このように移動ノードの現在のネットワークプレフィクスを取得し、この情報を基に通信先の LIN6 アドレスを構築し通信を開始する。

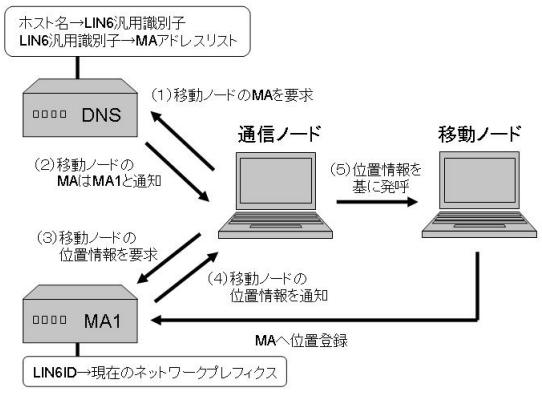

図 1 LIN6 における通信手順

4. 提 案 方 式

4.1 提 案 指 針

本研究では、移動透過性プロトコルである LIN6 をベースに initiator と responder 間でランダムに選んだ ID を一時的な識別子として利用し、匿名を考慮した移動透過性を実現する機構を提案する。

通信を開始する際に先にパケットを送信するノード

を initiator と呼ぶ。また、通信を開始する際に先にパケットを受信するノードを responder と呼ぶ。ここで initiator と responder の LIN6ID の役割の違いに着目する。initiator にとっての responder の LIN6ID は通信を開始したい相手を示し、responder にとっての initiator の LIN6ID はパケットを誰に返せば良いかを示す。本研究では、この違いを利用し、initiator と responder がそれぞれ動的に変化可能である一時的な LIN6ID を使用する方式を提案する。本研究では、この LIN6ID を anonymized LIN6ID と呼ぶ。既存の LIN6 では、LIN6ID は Authority の管理によって各ノードに静的に割り当てるものであった。本研究では、この他に動的に anonymized LIN6ID を割り当てる手法を導入する。また、この動的な anonymized LIN6ID は initiator と responder の 2 者間でのみ解決できれば良いため、initiator、responder ともに anonymized LIN6ID を管理する MA を必要としない。すなわち、この動的な anonymized LIN6ID はグローバルユニークでないので、グローバルな視点では同時に複数のノードを指し示す可能性もある。また、一定時間毎に anonymized LIN6ID を動的に変化させることで悪意のある第三者による追跡の回避や盗聴に対する両端のホストの匿名性を実現できる。また、なりすましに対処するために、responder が initiator の anonymized LIN6ID を一時的に認証する機構も実現する。このように動的な anonymized LIN6ID を利用することで initiator、responder 両ホストに匿名性を提供することができる。

電話と同様に匿名と非匿名を明示的に使い分けることを可能するために、ユーザやアプリケーションは送信先として responder の anonymized LIN6 汎用識別子を指定することで両ホストに匿名性を提供できるようにする。

以下、initiator を Ni、responder を Nr とする。Ni は Nr の LIN6ID (ID_Nr) を既知である。また、Nr と Nr の LIN6ID のマッピングを管理している MA_Nr の間には信頼関係があるものとする。

4.2 拡張縮退アドレスモデル

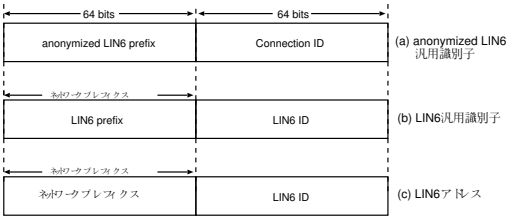

図 2 拡張縮退アドレスモデル

3.1 節で示した縮退アドレスモデルを拡張し、LIN6 汎用識別子の他に anonymized LIN6 汎用識別子を新たに追加した拡張縮退アドレスモデルを図 2 に示す。anonymized LIN6 汎用識別子の上位 64bit は anonymized LIN6 prefix と呼ばれる固定の値であり、下位 64bit は Connection ID (CID) を表す。Connection ID は通信相手を表す識別子である。アプリケーションまたはユーザが匿名通信を指定するときは anonymized LIN6 汎用識別子を利用することで匿名通信を行うことができる。

4.3 提案方式の通信モデル

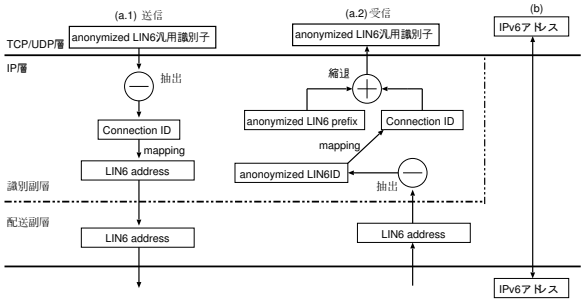

図 3 提案方式の通信モデル

提案方式の送信時の処理を図 3(a.i) に示す。Ni の CID を CID_Ni、Nr の CID を CID_Nr と表す。対象となる anonymized LIN6 汎用識別子の下位 64bit から CID_Nr を得る。次に、CID_Nr を鍵として Mapping テーブルを検索する。その結果、その通信で使われている Mapping エントリを発見でき、Ni と Nr の LIN6 アドレスが求まる。この LIN6 アドレスがパケットの配送に使われる。

受信時の処理を図 3(a.ii) に示す。対象となる LIN6 アドレスから LIN6ID 部を抽出し、この LIN6ID 部を鍵に mapping テーブルを検索する。ここで mapping テーブルを検索すると鍵として使われた LIN6ID 部が anonymized LIN6ID または既存の LIN6ID ということがわかるため、それぞれの処理を行う。anonymized LIN6ID であった場合、対応した anonymized LIN6 汎用識別子を得る。また、既存の LIN6ID であった場合、既存の LIN6 の処理を行う。

通常の IPv6 アドレスの送受信処理である即値処理を図 3(b) に示す。anonymized LIN6 汎用識別子、LIN6 汎用識別子、LIN6 アドレスでない場合、通常の IPv6 アドレスとして即値処理される。即値処理は既存の IPv6 通信と全く同じ処理である。

4.4 提案方式の動作手順

提案方式の動作手順を図 4 に示す。

Ni が Nr に対して anonymized LIN6 を利用した通信を行う場合、Ni は DNS サーバへ Nr の mapping を管理している MA の IPv6 アドレスを問い合わせる (4(1))。DNS サーバは ID_Nr とそれを管理する MA の IPv6 アドレスという静的な対応情報を管理しているので、DNS から Nr の mapping を管理する MA(MA_Nr) を取得することが可能である (図 4(2))。

次に、Ni は MA_Nr へ Nr が匿名通信を許可して

いるかを確認する (図 4(3))。MA_Nr は Nr が匿名通信を許可している場合、Ni に現在のネットワークプレフィクス (loc_Nr) と公開鍵 PK_Nr を通知する (図 4(4))。このように Ni は loc_Nr と PK_Nr を得ることができる。

Ni は loc_Nr と ID_Nr を利用し、anonymized LIN6ID を決定するフェーズに入る。loc_Ni は Ni の現在のネットワークプレフィクスとし、Ni は anonymized LIN6ID(AID_Ni) をランダムに生成する。Ni と Nr 間で使う共通鍵 K がランダムに生成され、共通鍵 K のライフタイムを L とする。これらのパラメータは Ni 上の mapping テーブルで保持される。{loc_Ni, AID_Ni, K, L}PK_Nr を含むメッセージを Nr に送信する (図 4(5))。これらのパラメータは Ni 上の mapping テーブルで保持される。Nr は Ni からのメッセージを受け取ると、秘密鍵 SK_Nr でメッセージを解読し、AID_Ni と K が Nr 上で一意であるかを確認した後、anonymized LIN6ID(AID_Nr) を生成する。また、CID_Ni は共通鍵 K から生成され、CID_Nr は Nr の LIN6ID とする。これらのパラメータは Nr 上の mapping テーブルに保持される。次に、Nr は Ni へ {AID_Nr, K}K を含むメッセージを送信する (図 4(6))。メッセージ受信した Ni は {AID_Nr, K}K を共通鍵 K で解読し、共通鍵 K を利用し mapping テーブルを検索する。responder 側と同様に、CID_Ni は共通鍵 K から生成され、CID_Nr は Nr の LIN6ID とする。検索されたエントリに AID_Nr、CID_Ni、CID_Nr を追加する。本研究では、この一連のフェーズのことをネゴシエーションと呼ぶ。

Ni が移動し、ネットワークプレフィクスが変更された場合、Ni は Nr に対して mapping 登録メッセージを送信する (図 4(7))。Ni の新しいネットワークプレフィクスを $loc2_Ni$ とすると、この mapping 登録メッセージは、{$loc2_Ni$, AID_Ni, K}K となる。Nr は mapping 登録メッセージの宛先アドレスから AID_Nr に対する AID_Ni からの mapping 登録メッセージだとわかり、AID_Nr を利用して mapping テーブルを検索する。mapping テーブルから共通鍵 K を取得し、それを利用して mapping 登録メッセージを解読したのち、mapping 登録メッセージ内の共通鍵 K との整合性を確かめる。共通鍵 K が一致したならば、Nr は mapping 登録メッセージを送信した相手が正しい Ni であることを認証できる。この認証後、Nr は自身の mapping テーブルを更新する。

このように既存の LIN6 とは異なり、Ni、Nr ともに anonymized LIN6ID を管理する MA を必要としない。

4.5 ノードの移動に対する処理

以下の 4 つの場合、mapping 登録メッセージを送信する。

- Ni のネットワークプレフィクスが変更された場合

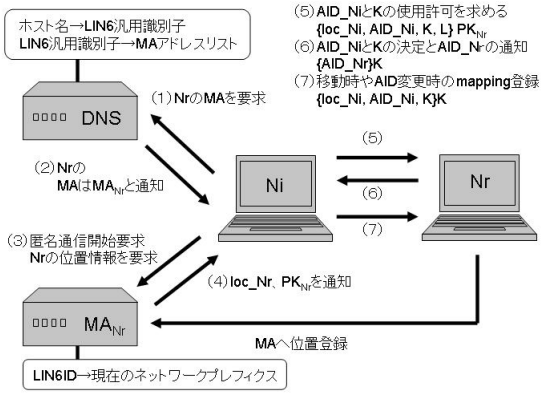

図 4 提案方式の動作手順

- Nr のネットワークプレフィクスが変更された場合
- Ni の anonymized LIN6ID が変更された場合
- Nr の anonymized LIN6ID が変更された場合

それぞれの場合、mapping 登録メッセージの宛先アドレスから anonymized LIN6ID を取得できる。この anonymized LIN6ID を利用して mapping テーブルを検索することができ、mapping 登録メッセージに含まれる共通鍵 K により mapping 登録メッセージの送信者を認証することが可能である。同様に、ネットワークプレフィクスと anonymized LIN6ID が同時に変更された時も、共通鍵 K により認証することが可能である。

Ni と Nr が同時に移動した場合、Ni が再び MA_Nr へ Nr の現在のネットワークプレフィクスを問い合わせ、再ネゴシエーションパケットを送信することで解決する。再ネゴシエーションは 4.7 節で詳細を述べる。

4.6 anonymized LIN6ID の衝突

Ni が移動した場合、リンクローカルで同じ anonymized LIN6ID が存在する可能性がある。Duplicated Address Ditection(DAD) により anonymized LIN6ID の重複を発見できる。重複の発見後、Ni は AID_Ni を AID2_Ni 変更し、Nr に対して mapping 登録を送信する。

4.7 再ネゴシエーション

再ネゴシエーションは、Ni と Nr が同時に移動した場合に利用される機構である。Ni の新しいネットワークプレフィクスを $loc2_Ni$、anonymized LIN6ID を AID2_Ni とし、Ni の新しいネットワークプレフィクスを $loc2_Nr$、anonymized LIN6ID を AID2_Nr とする。

まず、Ni が MA_Nr へ Nr の現在のネットワークプレフィクスを問い合わせると、MA_Nr は Ni に $loc2_Nr$ を通知する。次に、Ni は $loc2_Nr$ と ID_Nr を利用し、Nr に {$loc2_Ni$, AID2_Ni, K, L}PK_Nr を送信する。Nr は LIN6 のパケットヘッダのタイプフィールドにより再ネゴシエーションだと判断でき、秘密鍵 SK_Nr

を利用してメッセージを解読する。次に、Nr は mapping テーブルを共通鍵 K を鍵として検索し、エントリに $loc2_Ni$, AID2_Ni, L を更新する。Nr は Ni へ {AID2_Nr, K}K を含むメッセージを送信する。Ni はそのメッセージを共通鍵 K で解読し、共通鍵 K を利用し mapping テーブルを検索し、そのエントリに AID2_Nr を更新する。

5. 考　　察

5.1 匿名性について

本節では、考えられる 3 つの匿名性を挙げ、それぞれの匿名性が実現できることを示す。

responder(Nr) に initiator(Ni) のノード識別子を知られない匿名性

Ni は Nr に対し、anonymized LIN6ID を知らせることで LIN6ID を公開することなく通信開始し、移動透過性を保証することができる。このように、Ni は Nr に対して匿名で通信を開始することができる。

悪意のある第三者に追跡されない匿名性

短期間で見れば anonymized LIN6ID を追跡される可能性はあるが、Ni と Nr は動的に anonymized LIN6ID を変更することを許可していること、mapping 登録メッセージやネゴシエーションを暗号化していることから anonymized LIN6ID の変遷を知ることは非常に難しいと考えられる。

例えば、あるノード N が initiator である時、N の通信をすべて盗聴できる E が存在した場合を考える。N は anonymized LIN6ID ID_a1 で通信をしていた後、N は anonymized LIN6ID のライフタイムが切れるもしくはユーザまたは OS の判断で anonymized LIN6ID ID_a2 に切替えたとする。E はあるネットワークにいた ID_a1 が通信をやめ、新たに ID_a2 が現れたと言うことがわかる。統計的な情報が得られる場合には、これらの事実から E は ID_a1 と ID_a2 についての相関を得られる可能性がある。しかし、N は移動ノードであり常に N の通信を広い範囲で傍受することは現実的ではないため、一般的な環境においては大きな脅威とはなりにくいと考えられる。

悪意のある第三者に盗聴された時の両端のホストの匿名性

本研究はエンドツーエンドでの通信を重視した設計となっているため、パケットの配送に扱われるネットワークプレフィクスには変更を加えていない。そこで、悪意のある第三者が盗聴した場合、あるネットワークプレフィクスとあるネットワークプレフィクスが通信していることは分かる。しかし、動的に anonymized LIN6ID を変更することにより、ノードと anonymized LIN6ID のバインドが困難になる。前項で述べたような広域のネットワークでの傍受は、Ni と Nr が非同期に anonymized LIN6ID を変更していくため相関を得ることも難しく、一般的な環境において大きな脅威とはなりにくいと考えられる。

以上のように、本提案手法はエンドツーエンドの通信や移動透過性を保持しながら、十分な匿名性を得ることができるといえる。

5.2 悪意のある第三者のなりすましへの耐性

LIN6 ノードと MA 間である Nr と MA_Nr 間はセキュアな通信路がある前提であるので、悪意のある第三者が MA_Nr に偽のメッセージを送信し、Nr になりすますことはできない。また、Ni が Nr に対してネゴシエーションを行う時のパケットは Nr の公開鍵である PK_Nr で暗号化されるので、ネゴシエーション情報は秘密鍵である PK_Nr を持つ Nr しかネゴシエーション情報を知ることはできない。

悪意のある第三者が Ni になりすましネゴシエーションを行うには、Ni の anonymized LIN6ID が必要である。Ni の anonymized LIN6ID は暗号化し送信、または、ネゴシエーション完了後にインターフェースに付与されるため、なりすましが不可能であるといえる。

また、mapping 登録メッセージは含まれる情報全てが共通鍵 K により暗号化されるので、Ni と Nr 以外に知られることはない。情報を偽造された場合においても、mapping 登録メッセージに含まれる共通鍵 K による認証があるので悪意のある第三者によるなりすましは不可能である。同様に、再ネゴシエーションについても共通鍵 K による認証があるので、悪意のある第三者によるなりすましは不可能である。

以上のように悪意のある第三者によるなりすましへの耐性は非常に高いといえる。

5.3 ネゴシエーションによるオーバヘッド

本節では、本研究において匿名性を得るためのネゴシエーションでのオーバヘッドについて議論する。

Ni が Nr に対して初めて通信を開始する時、ネゴシエーションを行う必要がある。ここで、Ni の anonymized LIN6ID と K は Nr 上で一意である必要がある。anonymized LIN6ID は 40bit であり、Nr がすでに保持している anonymized LIN6ID の数を n_1 とすると、Ni の anonymized LIN6ID の衝突確率は $p_1 = \frac{n_1}{2^{40}}$ と表せるため、極めて小さいといえる。

K は共通鍵であるので、現在一般的に使用されている共通鍵暗号の一つ AES の例をあげると、鍵長は 128bit、192bit、256bit の 3 種類である。Nr がすでに保持している anonymized LIN6ID の数を n_2 とすると、鍵長が最も短い場合の衝突確率は $p_2 = \frac{n_2}{2^{128}}$ と表せるため、極めて小さいといえる。

このように衝突確率が極めて低く、ネゴシエーションは通信開始時 1 度だけやればよいので、ネゴシエーションによるオーバヘッドは小さいといえる。

5.4 mapping 登録のオーバヘッド

mapping 登録のメッセージの本体部分は共通鍵 K で暗号化される。この mapping 登録は定期的に送信ま

たは移動したときに送信する。このように、mapping 登録のオーバヘッドは、anonymized LIN6 での送受信時のオーバヘッドになるとは考えられない。

5.5 送受信時のオーバヘッド

本方式では、既存の LIN6 と異なり受信側も mapping table を引き、その情報を使い縮退を行う必要がある。このような受信側の縮退におけるオーバヘッドは RTT に比べ非常に小さいといえる。このように、送受信時のオーバヘッドは RTT に比べ非常に小さく無視できる範囲であると考えられる。

5.6 リンクローカルでの anonymized LIN6ID の衝突によるオーバヘッド

4.6 節で述べたように、ノード N の移動時にリンクローカルで anonymized LIN6ID が衝突する可能性がある。anonymized LIN6ID は 40bit であり、N がすでに保持している anonymized LIN6ID の数を n_3 とすると、移動時にリンクローカル上での anonymized LIN6ID の衝突確率は $p_3 = \frac{n_3}{2^{40}}$ と表せるため、極めて小さいといえる。

また、衝突後が検知された場合に送信される mapping 登録のオーバヘッドは、5.4 節で述べたように送受信そのもののオーバヘッドにはならない。

5.7 MIPv6 上での実現に関する問題点

本研究を MIPv6 (Mobile IPv6) 上で実現するためには、問題点が存在する。最も大きな問題点は、LIN6ID と Home Address の役割の違いである。LIN6ID はグローバルユニークなノード識別子であるのに対して、Home Address は Home Network 内のノード識別子である。

LIN6 の場合、LIN6ID はノード識別子であるため、initiator と responder の 2 者間で動的に変更しても MA に影響がない。しかし、MIPv6 の場合、Home Address が Home Network 内のノード識別子であるため、Home Network を動的に変化させる必要がある。このように、Home Address と Home Network のバインドを動的に変化させる必要があるため、原理的に実現は非常に困難である。

5.8 今後の課題

既存の LIN6ID と anonymized LIN6ID を混用しないような仕組みが必要である。例としては、本研究で割り当てられた OUI 以外の新たな OUI を取得し、それを anonymized LIN6ID 用の OUI とする方法や、LIN6ID の 1bit を利用して bit が 0 ならば既存の方式、bit が 1 ならば本方式とする方法が考えられる。

本研究を実装、評価し、現実のインターネット上で動作することを証明する。

6. 結　　論

本研究では移動透過性プロトコルである LIN6 において、移動透過性を失うことなく 3 つの匿名性を提供するための手法を提案した。

- responder に initiator のノード識別子を知られない匿名性
- 悪意のある第三者に追跡されない匿名性
- 悪意のある第三者に盗聴された時の両端のホストの匿名性

本研究では、ネゴシエーション終了後、initiator と responder 間で匿名通信を行うことができる。initiator や responder は anonymized LIN6ID を管理する MA を持つ必要がなく、匿名通信を行うためのオーバヘッドは小さいものであると考えられる。また、responder による initiator の認証機構により、悪意のある第三者のなりすましへの耐性を持つ。

今後の課題として、本提案を実装、評価し、現実のインターネット上で動作することを証明する。また、既存の LIN6ID と anonymized LIN6ID を混用しない仕組みが必要である。

参 考 文 献

1) D. Johnson C. Perkins and J. Arkko. Mobility Support in IPv6. Internet Draft, work in progress, June 2003.
2) R. Hinden, M. O'Dell, and S. Deering. An IPv6 Aggregatable Global Unicast Address Format. Internet RFC2374, Jul. 1998.
3) 石山政浩, 國司光宣, 河野通宗, 寺岡文男. LIN6 における匿名通信の実現方法に関する一考察. 情報処理学会 マルチメディア, 分散, 協調とモバイル (DICOMO2003) シンポジウム 論文集, p557-560, June. 2003.
4) S. Kent and R. Atkinson Security Architecture for the Internet Protocol Internet RFC2401, Nov. 1998.
5) 國司光宣, 石山政浩, 植原啓介, 寺岡文男. 移動体通信プロトコル LIN6 の性能評価. 情報処理学会論文誌, Vol. 43, No. 2, pp. 398–407, Feb. 2002. マルチメディアコミュニケーションシステム特集.
6) David Chaum. Untraceable Electronic Mail, Return Addresses, and Digital Pseudonyms. Communications of the ACM, 24(2): 84-88, Feb, 1981
7) Michael K. Reiter and Aviel D. Rubin. Crows: Anonymity for Web Transaction. ACM Transactions on Information and System Security, 1(1):66-92, 1998. Nov. 1998.
8) Rob Sherwood, Bobby Bhattacharjee, Aravind Srinivasan. P5: A Protocol for Scalable Anonymous Communication. 2002 IEEE Symposium on Security and Privacy, p58, May. 2002.