Title: USAGI Project 2004年度 IPv6 Mobility の設計と開発活動報告
Author(s): usagi-core@linux-ipv6.org
Date: 01/28/2005

-- 目次

-- 本文

1. 背景

USAGI Project では、ヘルシンキ工科大学（HUT）で開発された Mobile IPv6プロトコルスタックの実装を元に、 IPsec との協調処理などの USAGI Project 独自の拡張機能をLinux 2.4 系カーネル上で実装してきた。
2003 年 3 月より開発対象を 2.5 系カーネルに移し、同年 10 月より HUT と本格的な共同作業を開始した。さらに本年度は、メインラインカーネルが2.6 系へ更新されたことに追従する形で 2.6 系カーネルを開発対象としている。
一方で、 Mobile IPv6 プロトコル仕様は 2004 年 6 月に RFC 3775 及び RFC 3776 として発行された。
USAGI Project は、 RFC 3775 と RFC 3776 を実現するスタックが Linux の機能として取り入れられることを目標としている。

2. 2004年度のステータス

2.1 主な活動

2004/10 対向ノード（Correspondent Node; CN）機能は TAHI conformance test（ct-mipv6-cn-2.0b3）の全項目をクリア
2004/11 機能制限版のバージョンを MIPL-2.0-rc1 としてリリース（HUT）

2.2 実現された機能

MIPL-2.0-rc1 では、以下が実現されている。
- RFC 3775 のうち、Mobile Prefix discovery 以外の全ての機能
- RFC 3776 のうち、モバイルノード（Mobile Node; MN）とホームエージェント（Home Agent; HA）間の位置登録更新要求（Binding Update）と応答（Binding Acknowledgement）メッセージの IPsec による保護機能(手動鍵設定のみ)

3. Mobile IPv6プロトコルスタックの設計

Mobile IPv6 の機能がメインラインカーネルに取りこまれるためには、カーネルメンテナから出来るだけ軽微なカーネル修正で済むことが好ましいとアドバイスされている。
そこで USAGI Project では、カーネル内の既存の枠組みを極力流用し、カーネル内で

持つ必要のないデータ構造や処理はカーネルと分離してユーザランドで実装するデザインを採用した。

3.1 カーネルの設計

カーネルの機能ブロック図を図 3-1 に示す。

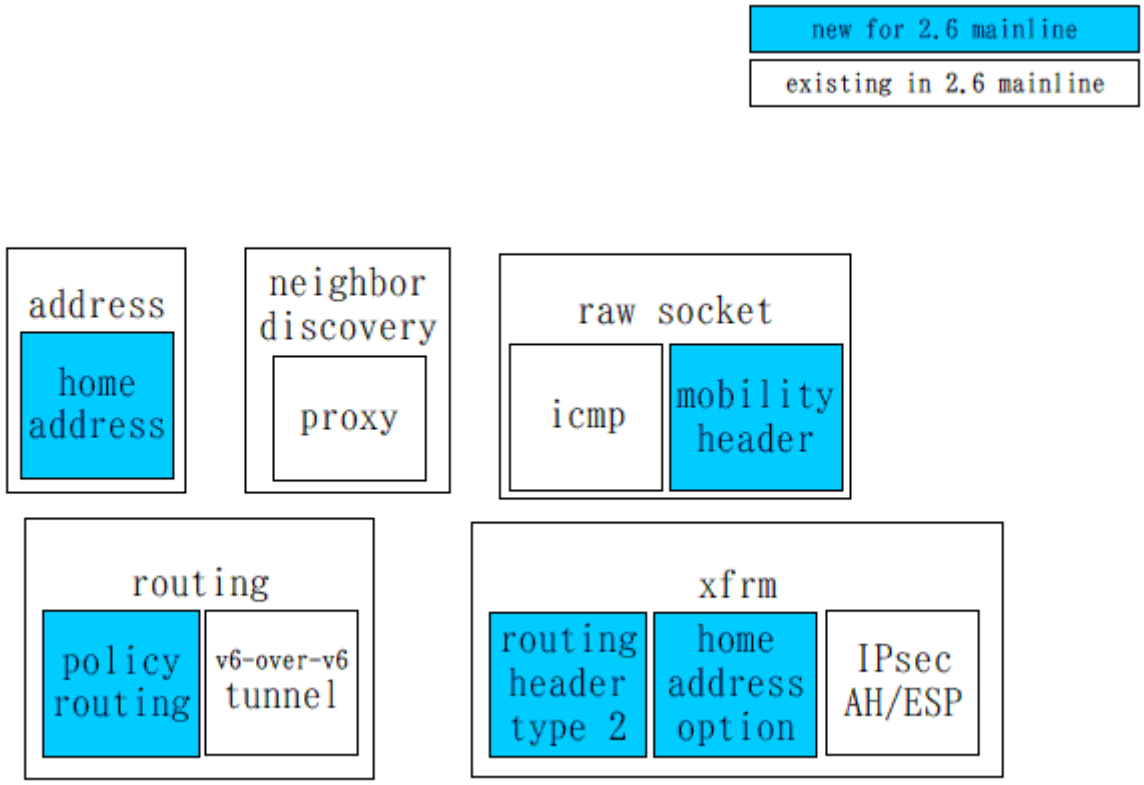

図 3-1 カーネルの機能

・アドレス管理部の拡張
MN 用ホームアドレスの識別情報と関連する処理を追加した。

・経路制御部の拡張
経路表を多段構成にし、ユーザランドから指定されるポリシーによって経路選択できる機能を追加した。この機能は Policy Routing と呼ばれ、これによって例えば送信アドレスを条件として経路選択が可能になる。

・近隣探索プロキシ機能（Proxy Neighbor Discovery）の修正
HA では、 MN のホームアドレス宛パケットを捕捉するために Proxy Neighbor Discovery 機能を使う。この機能自体は既にメインラインカーネルにあるが、多少の修正を入れている。

・RAW ソケットの拡張
Mobile IPv6 のシグナリングをサポートするために、モビリティヘッダの受信処理を RAW ソケットのパケット処理受信に追加した。ただし、ここではヘッダの最低限の解析処理しか行わず、メッセージはすぐさまユーザランドへ渡される。

・XFRM の拡張
XFRM とは、 ”transform”と読み、 2.5 および 2.6 系カーネルで採用されている、いわばパケット変換機構である。経路制御部や、ネットフィルターなどのフィルター部とは別定義されており、 IPsec の内部構造として使われている。

XFRM を、位置情報である Binding Cache 及び Binding Update List のサブセットを管理するように拡張し、終点オプションヘッダ用ホームアドレスオプションと経路ヘッダ(type 2)処理を追加して経路最適化機能をサポートした。また、位置情報が無いアドレスを含むパケットを検知して、ユーザランドへの通知を行うインターフェースを追加した。
例えば、 CN が受信したパケットの送信側のホームアドレスが Binding Cache (のサブセット)に存在しない場合、カーネルはユーザランドに位置登録エラーメッセージの送信を促すための通知を行う。また、 MN が送信しようとしたパケットの宛先アドレスが Binding Update List (のサブセット)に存在しない場合、認証機能である Return Routability テスト処理の開始を促すための通知を行う。

特に、Mobile IPv6 特有の新機能である Binding Cache と Binding Update List 関連処理が既存の XFRM を拡張して実現できるので、カーネルへの修正が軽微で済んでいる。

3.2 ユーザランドの設計

Mobile IPv6 で使用される位置情報管理や Mobile IPv6 のシグナリングメッセージの送受信は、一つのデーモンプログラムで実装されている。Mobile IPv6 で定義されている MN/HA/CN の各ノードの機能は、起動時にオプションとして指定することで切り替えて実行される。
Mobile IPv6 で必要とされる複数の内部処理は、軽量化および各データ処理、パケット処理の並列処理実現のためスレッドで実装している。

Mobile IPv6 が RFC となる以前のインターネットドラフトでは、すべてのシグナリングメッセージが 終点オプションヘッダのオプションとして定義されていたことがあったが、RFC 3775 ではモビリティヘッダとして 1 つの IPv6 拡張ヘッダとして定義されたため、カーネルよりもユーザランドでの処理が容易となった。

デーモンプログラムが、イベント管理、タイマー管理の多くを行う。Binding Cache や Binding Update List などの位置情報管理、端末の移動検知はデーモンで処理される(図 3-2)。

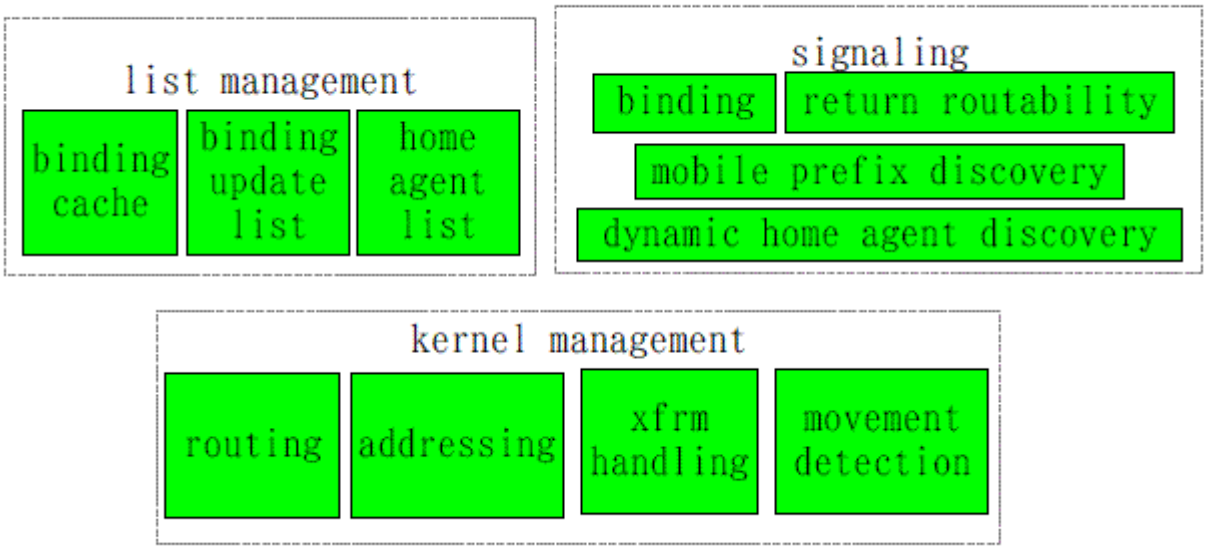

図 3-2 デーモンの機能

システム内では、デーモン内部で管理される Binding Cache と Binding Update List 、カーネル内部で保持されるパケット処理のための XFRM 情報、の 2 つのデータが存在

し、デーモンがそれらの管理および同期を行っている。

Mobile IPv6 では、RFC 3775 により IPsec の使用が必須となっているが、MIPL-2.0 の実装では、カーネルの設計で述べた通り、IPsec で使用されている XFRM を流用している。IPsec と Mobile IPv6 のパケット処理にすべて XFRM が使用されるため、Mobile IPv6 で必要とされる位置情報登録のための通信と、HA と MN 間の双方向トンネルでの通信で必要とされる IPsec の Security Policy Database (SPD) についてもデーモンの設定情報として保持される。

(MN の処理)
MN で行われる処理については、移動検出、HA への位置情報登録、 CN への位置情報登録という３つの処理が基本となる。

・移動検出
移動検出は、IP 層でのプレフィックスの変化とデフォルトルータの変化により行っている。MN の内部で現在のデフォルトルータを Router Advertisement メッセージから把握しておき、新しい Router Advertisement メッセージを受信したら現在の情報と比較する。変化があった場合は MN は移動したとみなし、移動処理を行う。

・HA への位置情報登録 (Home Registration)
MN がネットワークを移動したことを検出した後、HA に対して Binding Update メッセージを送信し、Home Registration を行う。

・CN への位置情報登録 (Return Routability/Correspondent Registration)
CN への通信が始まったことを検出した場合、MN は Return Routability テストを経て CN への Binding Update を送信し、経路最適化をおこなう。CN との通信は、最初は HA 経由の双方向トンネルを使用する。MN は、このトンネルからパケットを送信するイベントを Return Routability テストを始める契機として利用している。この契機は、XFRM から要求メッセージがデーモンに送信されることを利用している。デーモンはカーネルからのメッセージ受信待ち状態になっており、このメッセージを受信すると Return Routability テストを開始する。Return Routability テストが完了すると、Binding Update 送信用の XFRM ポリシをカーネルに設定して Binding Update を CN へ送信する。Binding Update を送信した後、経路最適化のための XFRM ポリシも登録する。

(HA の処理)
HA は主として位置情報管理、MN 宛パケットの捕捉、双方向トンネルによる MN への転送という処理を行う。

・位置情報管理
MN から Binding Update を受信すると、HA は MN の Binding Cache Entry を作成する。作成後に応答メッセージとして Binding Acknowledgement を MN へ送信する。

・MN 宛パケットの捕捉
MN からの Binding Update を正常に処理できた場合、MN のホームアドレス宛パケットが HA へ送信されるように、カーネルの Proxy Neighbor Discovery 機能を有効にする。

・双方向トンネルによる MN への転送
MN からの Binding Update を正常に処理できた場合はさらに、MN への双方向トンネルを設定し、CN からのパケットを MN へ転送する。 IPsec を使用しない場合は仮想デバイスとしてのトンネルを使用し、 IPsec によりトンネルを保護する場合は XFRM により元々のパケットをカプセル化して転送する。

(CN の処理)
CN は、MN から Binding Update を受信すると経路最適化のための XFRM ポリシをカーネルに設定する。設定された後は経路最適化による通信が可能となる。

3.3 Mobile IPv6 と IPsec の連携

USAGI Project では、 MIPL-2.0 上で IPsec を用いた各種モビリティシグナリングおよびユーザトラフィックの保護を実現すべく実装活動を進めている。これらの具体的な機能は RFC 3776 に記述されている。仕様では手動鍵設定のサポートを必須（MUST）、そして自動鍵交換のサポートを推奨（SHOULD）としている。 Mobile IPv6 を用いた実験等では、手動鍵設定が有効であるが、Mobile IPv6 の本格的な運用や高度なセキュリティ(特にリプレイ攻撃に対する防御)を実現するためには、自動鍵交換のサポートが必要である。 2.6 系カーネルでは、 IPsec のコア機能がカーネル内部で実装されており、付随するユーティリティ（ipsec-tools）が BSD から移植されている。従って、我々はこれらを利用することで上記の目的を実現することができるが、設計上いくつかの修正および拡張が必要であることが判明した。

特に、Mobile IPv6 における IPsec トンネルの利用は Mobile IPv6 と IPsec の間の密な連携が不可欠であり、これをどのように実現するかが興味深い点といえる。

（IPsec 連携機能の要件）

Mobile IPv6 では、 MN と HA の間で双方向トンネルを張り、MN のホームアドレスを利用した通信はすべてこのトンネルを経由する設計となっている。仕様では、この双方向トンネルを IPsec トンネルで代用することを提案しているが、MN が移動する度に IPsec トンネルのエンドポイント(トンネルの外側のヘッダの宛先もしくは送信元アドレス)が変化するため、これを IPsec に知らせてやる必要がある。

具体的には、特定のトンネルモード Security Association（SA）エントリに含まれるトンネルの入口・出口のアドレスの情報を、必要に応じて与えられた新たなアドレスで更新してやらなければいけない。これを実現するためには、（1）更新すべき SA エントリの特定(どのエントリを更新したら良いのか)、および（2）更新内容(どのように更新すべきか)が適切に IPsec に知らされる必要がある。

さらには、 K-bit と呼ばれる機能を実現するためには Internet Key Exchange（IKE）がこれらの情報を知る必要がある。 K-bit は、 Binding Update/Binding Acknowledgement に含まれるフラグの一部で、 MN および HA 上で動作する IKE が互いに張る論理的コネクション(IKEv1 におけるフェーズ 1 コネクション)を MN の移動に伴って維持することが可能かどうかを示すものである。

K-bit がサポートされている場合、 MN と HA 上で動作する IKE(v1) は、MN が移動して IKE のエンドポイントアドレス(すなわち気付アドレス)が変化した場合でも更新されたフェーズ 1 のコネクションを継続して利用することが可能となる。これにより、 IKE のシグナリングコストを抑え、帯域の有効利用が可能となる。なお、2.6 系カーネルのケースでは Security Association Database（SADB）に付随して SPD の更新も必要なことが分かっている。2.6 系カーネルにおける SPD エントリは、セレクタにマッチしたパケットに適応すべき IPsec 処理(すなわち該当する SA エントリ)を特定する情報をテンプレートとして保持している。このテンプレートには、SA の情報(トンネルモード SA の場合トンネルの入口・出口のアドレスを含む)を含まれているため、これらの情報も同時に更新してやる必要がある。

（IPsec 連携機能の設計と実装）

我々は KAME Project の Mobile IPv6 開発者および IPsec WG の racoon 開発者、そして Mobile IPv6 および IPsec に造詣の深い Francis Dupont 氏(ENST Bretagne) と Mobile IPv6 と IPsec の理想的な連携について議論を行った(2004 年 11 月上旬)。

この中で Dupont 氏は既に PF_KEY Version 2 の拡張を用いて上記の機能を実現していることが判明した。議論の参加者の共通の設計目標として、(1)既存のソフトウェア(Mobile IPv6 および IPsec) に対する変更が少ないこと、(2) 実装が容易であること、(3)システムになるべく依存しないことを挙げ、これを実現し得る最良の方法を検討した。

その結果、 Dupont 氏のアイディアに基づき PF_KEY の拡張で Mobile IPv6 から IPsec へのメッセージ通知を行うことで合意した。

このメッセージは PF_KEY MIGRATE と呼ばれる新たなメッセージで、更新すべき SP (Security Policy) および SA エントリを特定する情報、そして新たなトンネルモード SA の情報が含まれている。 Mobile IPv6 はこのメッセージを必要に応じて非同期に発行し、システム(IPsec および IKE) に移動の事実を知らせる。 MN は移動に伴い気付アドレスが変化するが、これをシステムに PF_KEY MIGRATE メッセージを用いて通知する。

一方、 HA は MN からの Binding Update を受信することによって、 MN の気付アドレスが変わったことを知る。 HA はこのタイミングでシステムに MN の移動を通知する。システム(カーネル内の IPsec)は、このメッセージを受信し、処理が適切に済んだ場合にこれを開かれている PF_KEY ソケットにブロードキャストする。 racoon のように PF_KEY ソケットを見張っているアプリケーションはブロードキャストされた PF_KEY MIGRATE メッセージを聞き、 Mobile IPv6 で発生した移動の事実を把握することが可能となる。図 3-3 は、Mobile IPv6 と IPsec の連携を示したブロック図である。

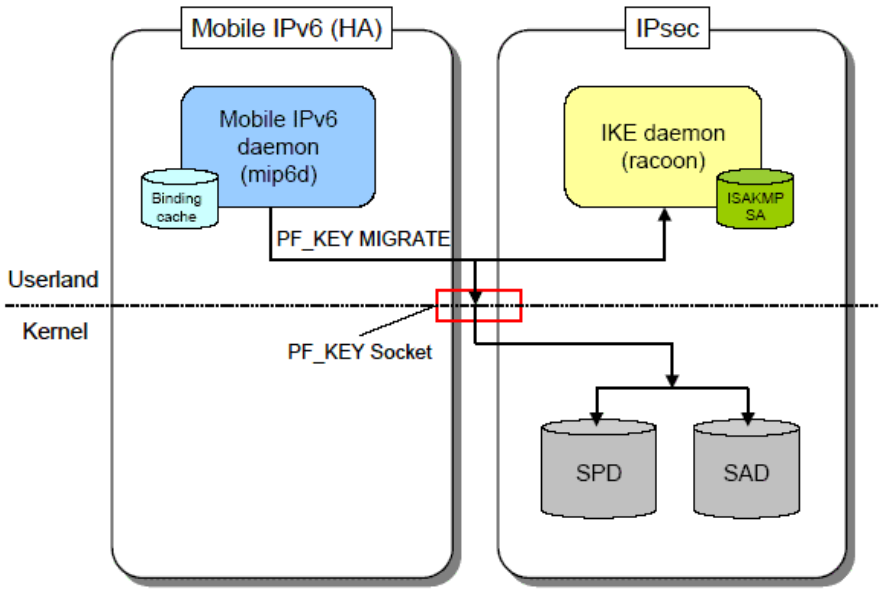

図 3-3 Mobile IPv6 と IPsec の連携

(IPsec 連携機能のステータス)

現在、 MIPL-2.0 上で PF_KEY MIGRATE およびこれに対応した racoon の動作テストを行っている。今後は、 KAME Project および慶應義塾大学と協力し、共通のメッセージ通知機構を実装する予定である。その後 IPsec/IKE に対する Mobile IPv6 の要求事項をまとめ、 IPsec WG の開発者にこれを報告する。その結果、 MIPL-2.0 および SHISA 上で動作する Mobile 環境に適応可能な IKE デーモンが実現可能となる。

4. 今後の展開

2004 年 11 月、機能制限版のバージョンを MIPL-2.0-rc1 として HUT がテスト公開した。今後は正式リリースの公開を目指してさらに HUT との共同作業を進めていくと共に、 Mobile IPv6 と IPsec との連携機能を強化し、自動鍵交換サポートに注力していく。