

Linux-2.6における Mobile IPv6 (MIPL2) の設計と実装

杉本 信太

shintasugimoto@ericsson.com

日本エリクソン株式会社
エリクソン・リサーチ・ジャパン

2006/12/08

Internet Week 2006

1

本チュートリアルの内容

- 導入
 - 背景
 - プロトコル概要
- MIP2の設計と実装
 - 設計方針
 - システム概要
 - MIPL2におけるパケット処理の仕組み
 - XFRM機能への拡張
 - ルーティング機能への拡張
 - MIPL2におけるホームレジストレーションの仕組み
 - MNの移動シーケンス
 - HAのBU受信シーケンス

2006/12/08

Internet Week 2006

2

背景

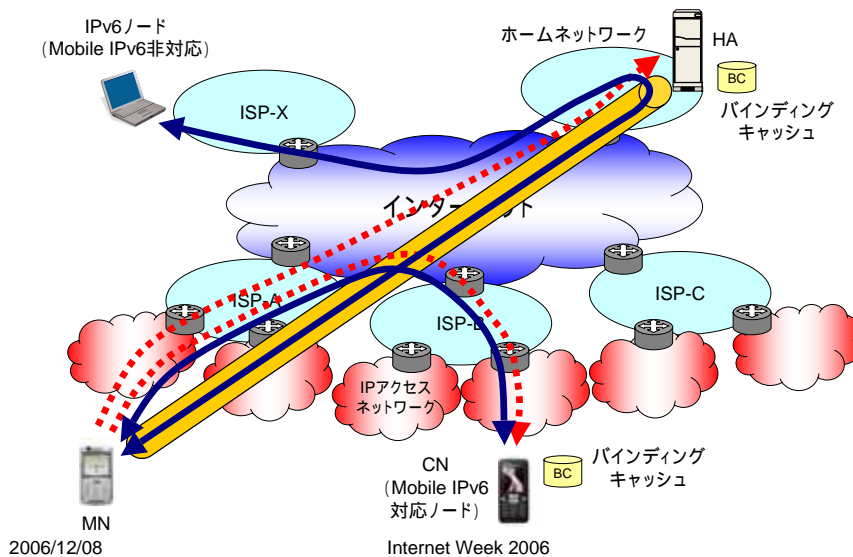
- Mobile IPv6の標準化(2004年6月)
- Linux-2.4におけるMobile IPv6の実装
 - ヘルシンキ工科大学(HUT)GO-Core Project
 - 主にカーネル内部での実装
- Linux-2.6におけるMobile IPv6の実装
 - Linux-2.4から大幅にネットワーク・スタックの設計が変更される
 - HUT GO-Core ProjectとUSAGI Projectによる共同開発
 - Linuxカーネルメンテナからカーネル修正の最小化するよう要望

Mobile IPv6用語・略語

- Mobile Node (MN)
- Correspondent Node (CN)
- Home Agent (HA)
- Home Address (HoA)
- Care-of Address (CoA)
- Return Routability
- Binding Update (BU)
- Binding Acknowledgement (BA)
- Binding Cache (BC)
- Binding Update List (BUL)

Mobile IPv6プロトコル概要(1)

システム・アーキテクチャ



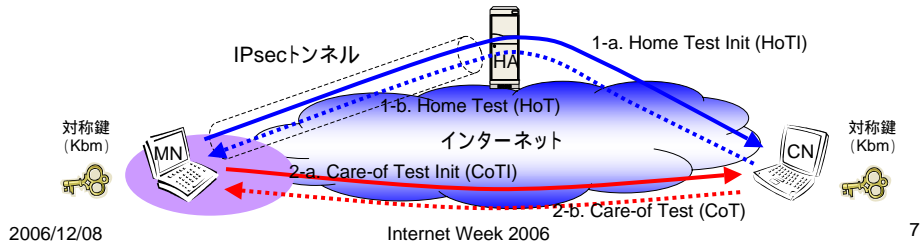
Mobile IPv6プロトコル概要(2)

プロトコルの目的・ゴール

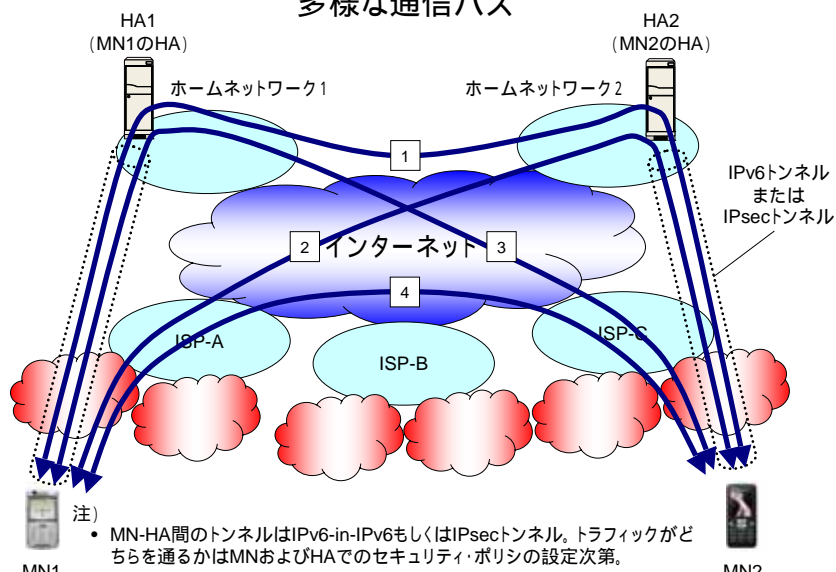
- セッション継続性
 - 上位層プロトコル(例 . TCP) がMNの移動に伴ってセッションを維持することを可能にする
 - 通常、上位層はIPアドレスとポート番号の対を通信のエンドポイントとして利用するため、この一部が動的に変化した場合セッションが切断されてしまう
- 着信可能性
 - 接続先IPサブネットが変化し、IPアドレスが変化した場合にも、通信相手から常に同一のIPアドレスにて着信可能であることを保証すること

Mobile IPv6プロトコル概要(3) セキュリティ機構

- MN-HA間のセキュリティ機構
 - IPsecによるモビリティ・シグナリングの保護(正真性・秘匿性)
 - HoAとCoAのセキュアなバインディング
 - HoAとHA addressをエンドポイントとしたトランスポート・モードSAペア
- MN-CN間のセキュリティ機構
 - Return Routability
 - MN-HA間のIPsecトンネルを利用
 - 任意のMNがそのHoAおよびCoAによって本当に到達可能かどうか確認
 - MNはCNから得たkeygen tokenを元にBU送信用の鍵(Kbm)を生成



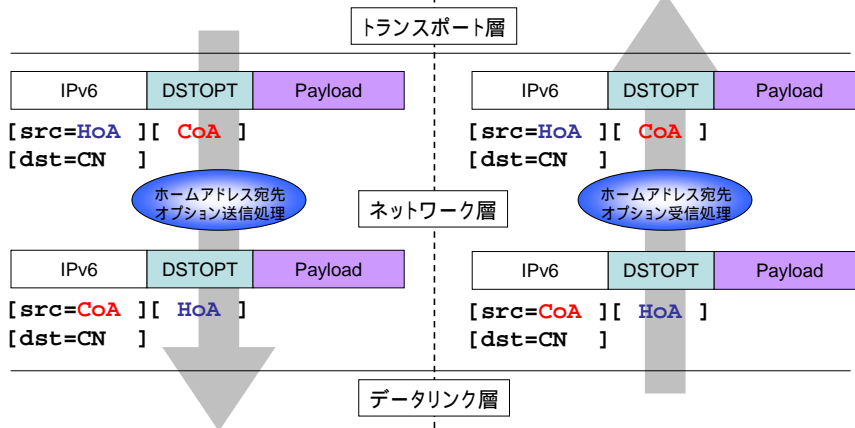
Mobile IPv6プロトコル概要(4) 多様な通信パス



注)
• MN-HA間のトンネルはIPv6-in-IPv6もしくはIPsecトンネル。トラフィックがどちらを通るかはMNおよびHAでのセキュリティ・ポリシーの設定次第。

Mobile IPv6特有のIPv6拡張ヘッダ処理 ホームアドレス宛先オプション

送信元ホスト(MN) : 受信ホスト(CN)



注) 上位層のpseudo-headerの計算には、常にHoAが用いられる。

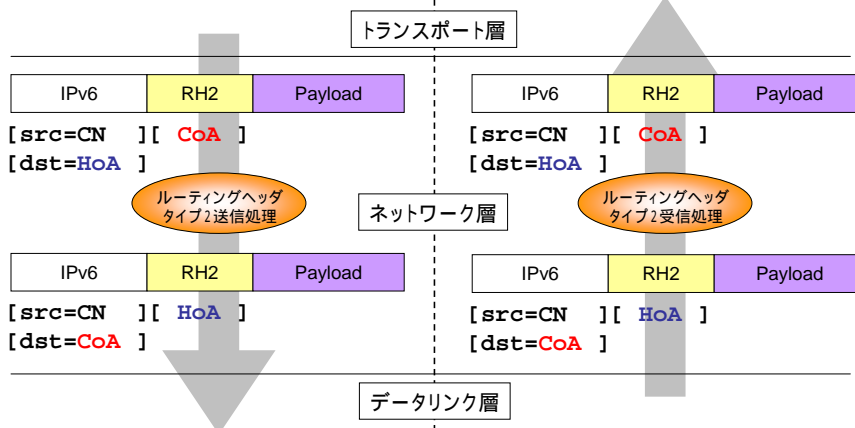
2006/12/08

Internet Week 2006

9

Mobile IPv6特有のIPv6拡張ヘッダ処理 ルーティングヘッダタイプ2

送信元ホスト(CN) : 受信ホスト(MN)



注) 上位層のpseudo-headerの計算には、常にHoAが用いられる。

2006/12/08

Internet Week 2006

10

MIPL2

- MIPL2とは
 - HUT GO-Core ProjectとUSAGI Projectによる共同開発
 - MN/HA/CNの全ての機能を共通のプラットフォーム上で実現
 - カーネルパッチ(対Linux-2.6) + デモンプログラム(mip6d)
 - GPL
- 基本設計方針
 - カーネル修正を必要最低限に抑える
 - IPv6の基本機能に悪影響を与えない
- 開発の目的
 - 基本仕様(RFC3775, RFC3776)に基づいた参照コードの提供
 - IPv6およびMobile IPv6の普及促進

2006/12/08

Internet Week 2006

11

本チュートリアル資料における留意点

- ソースコード
 - 2005年12月末時点のUMIP-0.1(MIPL2派生ブランチ)を参照
 - ユーザランドおよびカーネル・ソースアーカイブを展開したディレクトリをそれぞれ\$MIPL2USER、\$MIPL2KERNELと表記する
 - 例.\$MIPL2USER/src/ha.c
- ipコマンドの出力結果
 - MIPL2のパッチが当たったipコマンドの出力

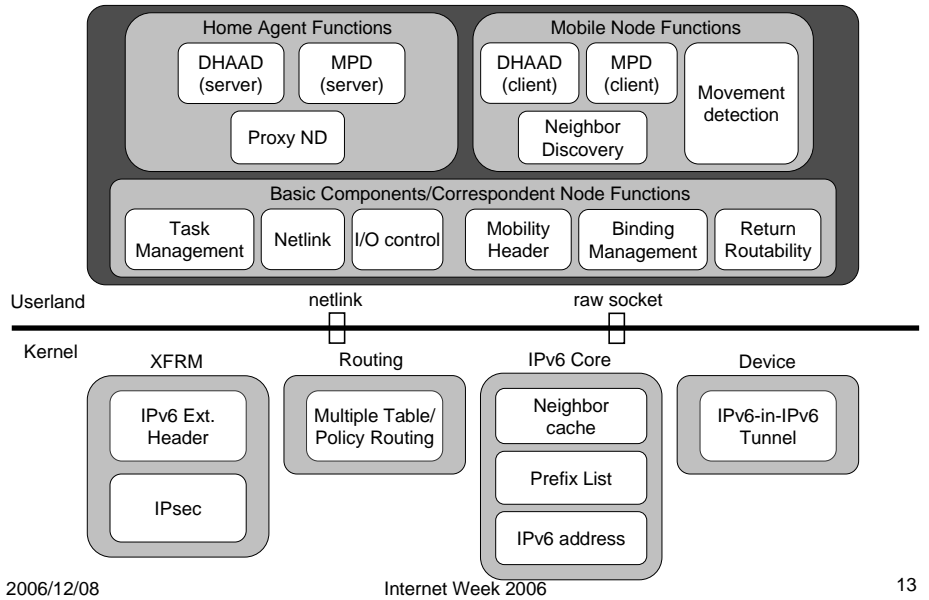
2006/12/08

Internet Week 2006

12

MIPL2システム概要

デーモンプログラム (mip6d) とカーネル内関連機構



MIPL2におけるパケット処理手法

1. IPv6トンネルデバイス

- HA-MNの双方向トンネル(IP-in-IPトンネル)
- 仮想デバイスとして扱うことが可能
- 処理の対象はデータ・プレーン
- ルータとしてのIPパケット処理

2. XFRM拡張

- 経路最適化・IPsec処理
- 汎用的なトランスフォーム機能の有効利用
- 処理の対象はコントロール・プレーンおよびデータ・プレーン
- エンドホストとしてのIPパケット処理

XFRMに対するMIPv6拡張(1)

MIPv6専用ポリシー・ステート

- Linux-2.6カーネル内のXFRM機能を拡張し、MIPv6特有のステート情報およびパケット処理を実現
 - XFRMポリシー…どのフロー(パケット)に対して、
 - XFRMテンプレート…どのような処理を適用するか
 - XFRMステート…具体的な処理内容
- MIPv6専用ポリシー
 - Template部分にMIPv6特有の処理が記述
 - 場合によっては「バンドル」となる
- MIPv6専用ステート
 - ホームアドレス宛先オプションまたはルーティングヘッダタイプ2の処理
 - 実質的にはカーネル内のBCエントリおよびBULエントリに相当

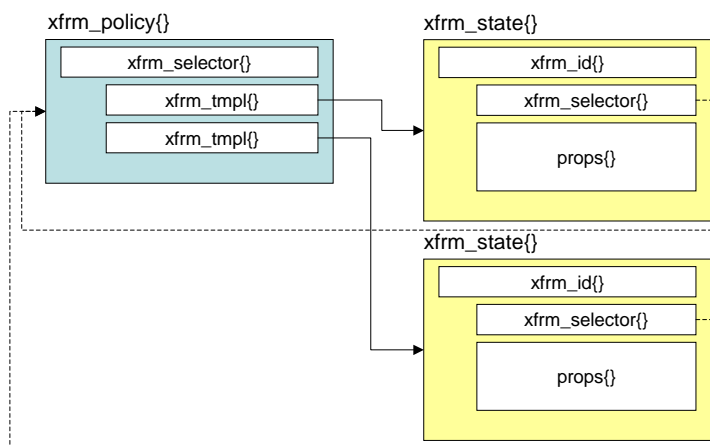
2006/12/08

Internet Week 2006

15

XFRMに対するMIPv6拡張(2)

MIPv6専用ポリシー・ステート



2006/12/08

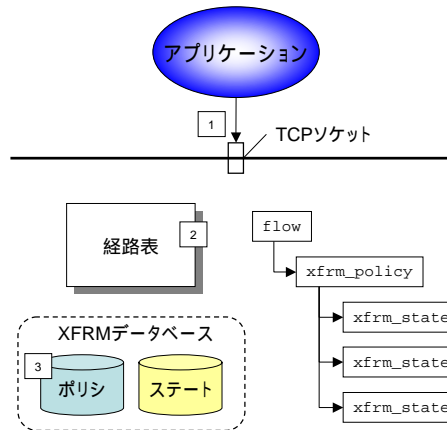
Internet Week 2006

16

パケット処理の流れ

例: TCPソケットを利用したアプリケーション

1. ソケットに対してデータが書かれる
2. フローの情報を元に、経路検索 (ip6_dst_lookup) が行われる
3. XFRMデータベースが参照され、ポリシー検索 (xfrm_lookup) を行う
4. 該当ポリシーとそれと関連したステートが探し出され、該当の処理ルーチンがスタック化される
5. スタック化された各dst_entryの出カルーチンが順に呼び出される (dst_output)



2006/12/08

Internet Week 2006

17

Mobile IPv6用XFRMポリシー (1)

モビリティ・シグナル処理用ポリシー (MN)

BU送信用ポリシー

```
src <HoA>/128 dst <HA address>/128 proto 135
dir out priority 2
IPsecテンプレート:  tmpl src :: dst ::
                      proto esp reqid 1 mode transport
MIPv6テンプレート:  tmpl src <HA address> dst <HoA>
                      proto hao reqid 0 mode ro
                      level use
```

BA受信用ポリシー

```
src <HA address>/128 dst <HoA>/128 proto 135
dir in priority 2
IPsecテンプレート:  tmpl src :: dst ::
                      proto esp reqid 2 mode transport
MIPv6テンプレート:  tmpl src :: dst ::
                      proto route2 reqid 0 mode ro
                      level use
```

2006/12/08

Internet Week 2006

18

Mobile IPv6用XFRMポリシー (2)

IPsecトンネルの設定用ポリシー (MN)

IPsec tunnel (outbound) ポリシ

```
src <HoA>/128 dst ::/0 proto 135 dev ip6tnl1
dir out priority 9
IPsecテンプレート  tmpl src <CoA> dst <HA address>
                    proto esp reqid 3 mode tunnel
```

IPsec tunnel (inbound) ポリシ

```
src ::/0 dst <HoA>/128 proto 135 dev ip6tnl1
dir in priority 9
IPsecテンプレート  tmpl src <HA address> dst <CoA>
                    proto esp reqid 4 mode tunnel
```

注)

- 上記ポリシーはHAへのバインディングの登録が成功した際に設定される。以後、MN-HA間のIPsecトンネルが有効となる。

2006/12/08

Internet Week 2006

19

Mobile IPv6用XFRMポリシー (3)

経路最適化ポリシー (MN)

データパケット経路最適化ポリシー (outbound)

```
src <HoA>/128 dst <CN address>/128 dev ip6tnl1
dir out priority 14
MIPv6テンプレート  tmpl src <HoA> dst <CN address>
                    proto hao reqid 0 mode ro
                    level use
```

注)

- 上記ポリシーはCNへのバインディングの登録が成功した際に設定される。以後、該当CN宛のパケットは経路最適化される。

2006/12/08

Internet Week 2006

20

Mobile IPv6用XFRMポリシー (4) 経路最適化ポリシー (CN)

Wild receiverポリシー (inbound)

```
src ::/0 dst ::/0 proto 135 type 5
dir in priority 8
MIPv6テンプレート  tmpl src :: dst ::
                    proto hao reqid 0 mode ro
                    level use
```

注)

- 上記ポリシーはCNの起動時に設定される。不特定のノードからのBUの受信を許可する。
(ただし、Return Routabilityが成功しないとmip6dはBUは処理されない)

2006/12/08

Internet Week 2006

21

Mobile IPv6用XFRMポリシー (5) モビリティ・シグナル処理用ポリシー (HA)

BU受信用ポリシー

```
src <MN1_HoA>/128 dst <HA address>/128 proto 135
dir in priority 2
IPsecテンプレート  tmpl src :: dst ::
                    proto esp reqid 1 mode transport
MIPv6テンプレート  tmpl src :: dst ::
                    proto hao reqid 0 mode ro
                    level use
```

BA送信用ポリシー

```
src <HA address>/128 dst <MN1_HoA>/128 proto 135
dir out priority 2
IPsecテンプレート  tmpl src :: dst ::
                    proto esp reqid 2 mode transport
```

注)

- 上記ポリシーはHAの起動時にコンフィギュレーションファイルを元に設定される。

2006/12/08

Internet Week 2006

22

Mobile IPv6用XFRMポリシ (6) IPsecトンネルの設定用ポリシ (HA)

IPsecトンネル(inbound)

```
src <HoA>/128 dst <HA address>/128 proto 135
dir in priority 9
IPsecテンプレート  tmpl src <CoA> dst <HA address>
                    proto esp reqid 3 mode tunnel
```

IPsecトンネル(outbound)

```
src ::/0 dst <MN1_HoA>/128 proto 135 dev ip6tnl1
dir out priority 9
IPsecテンプレート  tmpl src <HA address> dst <MN1_CoA>
                    proto esp reqid 4 mode tunnel
```

注)

- 上記ポリシはMNのバインディングが成功した際に設定される。以後、MN-HA間のIPsecトンネルが有効となる。

2006/12/08

Internet Week 2006

23

ポリシ・ルーティング(1)

- ポリシ・ルーティング
 - 古典的な経路検索を拡張し、宛先アドレスのみでなく送信元アドレスやトラフィック・クラス等を考慮に入れた経路探索
 - ポリシ・データベース
- マルチプル・テーブル
 - 経路検索の“ルール”を設定可能に
 - ルールに従って、検索対象となる経路表が選択される
- Mobile IPv6とポリシ・ルーティングの関連性
 - 特定の送信元アドレス (HoA) を考慮した経路探索 (Source Address Based Routing)
 - MNとして動作するかどうかは、実質的にアプリケーションが選択する送信元アドレスに基づいて行われる (任意のフローに対してモビリティサポートを提供するか否か)
 - HAにおいては、エラー処理のため自発の packets だけは例外的 (トンネルデバイスを使わず) に処理する必要がある

2006/12/08

Internet Week 2006

24

ポリシー・ルーティング(2)

初期時のMNにおけるルール設定

```
#ip -6 rule
0:          from all lookup local
32766:     from all lookup main
```

バインディング成功時のMNにおけるルール設定

```
#ip -6 rule
0:          from all lookup local
1001:     from <HoA> lookup 252
1002:     from fe80::/64 lookup main
1002:     from <on-link subnet prefix> lookup main
32766:     from all lookup main
```

MN-HA間双方向トンネルの実現(1)

- IPv6トンネルデバイス(ip6tnl)
 - バインディング有効時には、送信元アドレスがHoAのトラフィックにはデフォルトでこの経路が選択される
 - トンネルデバイス(CONFIG_IPV6_TUNNEL)の機構はmainlineカーネルにて利用可能
- MNにおけるトンネルデバイスの取扱い
 - 起動時にトンネルデバイスを生成
 - バインディング有効時(在Foreign Network時)には、HoAをトンネルデバイスに設定する
- HAにおけるトンネルデバイスの取扱い
 - MN(バインディング)毎に動的にトンネルデバイスを管理

MN-HA間双方向トンネルの実現(2)

バインディング有効時のMNにおけるトンネルデバイス

```
#ifconfig ip6tnl1
ip6tnl1 Link encap:UNSPEC HWaddr XX-XX-XX-XX-XX-XX-XX-XX-XX-XX-
XX-XX-XX-XX-XX
inet6 addr: <HoA>/128 Scope:Global
inet6 addr: <link-local address>/64 Scope:Link
UP POINTOPOINT RUNNING NOARP MTU:1460 Metric:1
RX packets:0 errors:0 dropped:0 overruns:0 frame:0
TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:0
RX bytes:0 (0.0 b) TX bytes:0 (0.0 b)
```

バインディング有効時のHAにおけるトンネルデバイス

```
#ifconfig ip6tnl1
ip6tnl1 Link encap:UNSPEC HWaddr XX-XX-XX-XX-XX-XX-XX-XX-XX-XX-
XX-XX-XX-XX-XX
inet6 addr: <link-local address>/64 Scope:Link
UP POINTOPOINT RUNNING NOARP MTU:1460 Metric:1
RX packets:0 errors:0 dropped:0 overruns:0 frame:0
TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:0
RX bytes:0 (0.0 b) TX bytes:0 (0.0 b)
```

2006/12/08

Internet Week 2006

27

MN-HA双方向トンネルの実現(3)

バインディング有効時のMNにおける経路表

```
#netstat -nr -A inet6
Kernel IPv6 routing table
Destination
  Flags Metric Ref    Use Iface          Next Hop
:::/0
  U      128    0     0 ip6tnl1         ::
<HA address>/128
  UC     0      1     1 ip6tnl1         <HA address>
<HoA>/128
  U      192    0     0 ip6tnl1         ::
...
```

バインディング有効時のHAにおける経路表

```
#netstat -nr -A inet6
Kernel IPv6 routing table
Destination
  Flags Metric Ref    Use Iface          Next Hop
...
<MN1_HoA>/128
  U      192    0     0 ip6tnl1         ::
...
```

2006/12/08

Internet Week 2006

28

MIPL2のnetlink関連ルーチン

関数名	nmsg_type	エンティティ	説明
rule_add	RTM_NEWRULE	MN/HA	ルールの追加
rule_del	RTM_DELRULE	MN/HA	ルールの削除
route_add	RTM_NEWROUTE	MN/HA	経路の追加
route_del	RTM_DELROUTE	MN/HA	経路の削除
neigh_add	RTM_NEWNEIGH	MN	NDエントリの追加
neigh_del	RTM_DELNEIGH	MN	NDエントリの削除
pneigh_add	RTM_NEWNEIGH	HA	プロキシNDエントリの追加
pneigh_del	RTM_DELNEIGH	HA	プロキシNDエントリの削除
addr_add	RTM_NEWADDR	MN	アドレスの追加
addr_del	RTM_DELADDR	MN	アドレスの削除
prefix_add	RTM_NEWPREFIX	MN	プレフィックスの追加
xfrm_policy_add	XFRM_MSG_NEWPOLICY, XFRM_MSG_UPDPOLICY	MN/HA/CN	XFRMポリシーの追加
xfrm_policy_del	XFRM_MSG_DELPOLICY	MN/HA/CN	XFRMポリシーの削除
xfrm_state_add	XFRM_MSG_ADDSA	MN/HA/CN	XFRMステートの追加
xfrm_state_del	XFRM_MSG_DELSA	MN/HA/CN	XFRMステートの削除

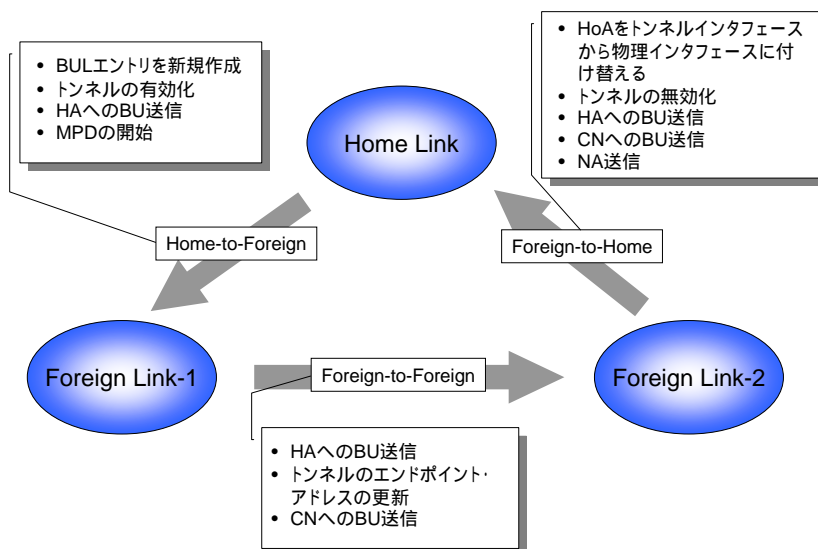
2006/12/08

Internet Week 2006

29

MNの移動(1)

異なる移動のタイプ



2006/12/08

Internet Week 2006

30

MNの移動(2)

HL FL / FL FL移動シーケンス

```
md_trigger_movement_event(...)
```

```
↳ mn_movement_event(struct movement_event *e)
```

```
↳ mn_move(struct home_addr_info *hai)
```

HL FLの場合、addr_do()を呼び出し、トンネルインタフェースにHoAを設定する

```
↳ do_handoff(struct home_addr_info *hai, struct mn_addr *coa)
```

```
↳ mn_send_home_bu(struct home_addr_info *hai)
```

```
↳ mn_send_bu_msg(struct buletnry *bule)
```

2006/12/08

Internet Week 2006

31

MNの移動(2)

FL HL移動シーケンス

```
md_trigger_movement_event(...)
```

```
↳ mn_movement_event(struct movement_event *e)
```

```
↳ mn_do_dad(struct home_addr_info *hai)
```

```
↳ mn_move(struct home_addr_info *hai)
```

```
↳ do_handoff(struct home_addr_info *hai,  
             struct mn_addr *coa)
```

```
↳ mn_send_home_bu(struct home_addr_info *hai)
```

```
↳ mn_send_bu_msg(struct buletnry *bule)
```

HAからBAを受信

```
mn_rcv_ba(...)
```

```
↳ mn_send_home_na(struct home_addr_info *hai)
```

2006/12/08

Internet Week 2006

32

MNの移動(3)

実際のBU送信処理の呼び出し

\$MIPL2USER/src/mn.c: do_handoff()

```
static int do_handoff(struct home_addr_info *hai, struct mn_addr *coa)
{
    if (!IN6_IS_ADDR_UNSPECIFIED(&hai->ha_addr)) {
        if (hai->pend_ba)
            mn_send_home_bu(hai);
        else if (!pending_bas) {
            if (!hai->at_home ||
                hai->home_reg_status != HOME_REG_NONE)
                mn_send_home_bu(hai);
            bul_iterate(&hai->bul, mn_rr_start_handoff, coa);
            hai->verdict = MN_HO_NONE;
        }
    } else if (!pending_bas) {
        if (hai->at_home)
            dhaad_stop(hai);
        else
            dhaad_start(hai);
    }
    return 0;
}
```

2006/12/08

Internet Week 2006

33

MNにおけるホーム・レジストレーション処理

\$MIPL2USER/src/mn.c: mn_send_home_bu()

1. BULエントリを検索
 - 該当エントリが無ければ新たに作成し、process_first_home_bu()を呼び出す
 - この過程で移動のタイプ(HL FL、FL FL、FL HL)を判定
2. BU送信(mn_send_bu_msg)
3. 各種トンネルの処理
 - HL FL移動の場合、IPsecトンネルを有効化
 - IPsecトンネルのエンドポイント・アドレスを更新
 - MN-HA間双方向トンネルのエンドポイント・アドレスを更新(tunnel_mod)
4. 通信相手(CN)に対するバインディング・アップデート処理
 - BULを検索して必要であればCNに対してBUを送信する

2006/12/08

Internet Week 2006

34

MNにおけるホーム・レジストレーション処理

\$MIPL2USER/src/mn.c: process_first_home_bu()

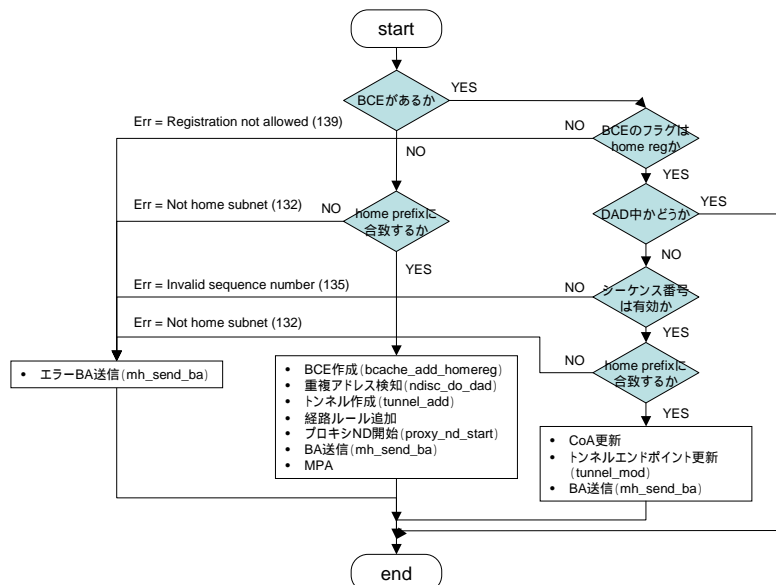
- MN-HA間双方向トンネルを有効にする
(mn_tnl_state_add)
 - 経路最適化用ポリシーの追加 (mn_ro_pol_add)
 - 送信元アドレスがHoAの場合、トンネルデバイスに出力するデフォルト経路を追加
- MPD (クライアント側) のタイマを起動

2006/12/08

Internet Week 2006

35

HAにおけるホーム・レジストレーション処理 (1)



2006/12/08

Internet Week 2006

36

HAにおけるホーム・レジストレーション処理(2)

1. BCエントリ作成 (bcache_add_homereg)
2. 重複アドレス検知 (ndisc_do_dad)
3. トンネル作成 (tunnel_add)
4. 経路ルール追加
5. プロキシND開始 (proxy_nd_start)
6. BA送信 (mh_send_ba)
7. MPD (サーバ側) の開始

2006/12/08

Internet Week 2006

37

BAエラー・ステータスコード一覧

status	MIPL2 define	エラー内容・意味
128	IP6_MH_BAS_UNSPECIFIED	原因不明(以下のエラー以外のもの)
129	IP6_MH_BAS_PROHIBIT	管理者的観点から禁止
130	IP6_MH_BAS_INSUFFICIENT	リソース不足(メモリ割り当て、経路テーブル)
131	IP6_MH_BAS_HA_NOT_SUPPORTED	HAとして動作しない旨の通知
132	IP6_MH_BAS_NOT_HOME_SUBNET	不正なHome Prefix
133	IP6_MH_BAS_NOT_HA	該当MNには元々HAとしてサービスしていない
134	IP6_MH_BAS_DAD_FAILED	HoAのDADが失敗
135	IP6_MH_BAS_SEQNO_BAD	シーケンス番号が不正
136	IP6_MH_BAS_HOME_NI_EXPIRED	Home Nonceインデックスが不正
137	IP6_MH_BAS_COA_NI_EXPIRED	Care-of Nonceインデックスが不正
138	IP6_MH_BAS_NI_EXPIRED	Nonceインデックスが無効
139	IP6_MH_BAS_REG_NOT_ALLOWED	レジストレーションタイプが変化

注)

- ピンク色の網掛け項目はHA特有のエラー
- 緑色の網掛け項目は、CN特有のエラー
- その他の項目はHA、CN共通のエラー

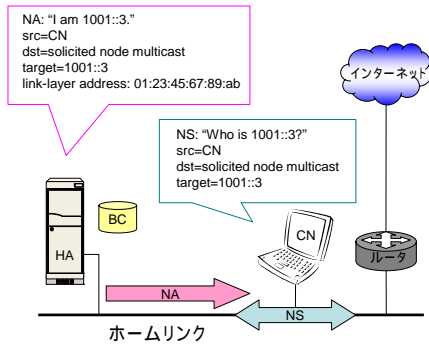
2006/12/08

Internet Week 2006

38

HAにおけるプロキシND(1)

- HAはBCにエントリのあるMN(在フォーリンネットワーク)のHoAに対してProxy NDを行う
 - ホームリンクにおいてMNのHoA宛のパケットを捕捉するため
 - ホームリンク上の他ノード(ルータあるいはCN)に対してMNの代わりにNAを送信
 - プロキシNDのNAでは、R-bitは立てない
- ホームネットワークに戻ったMNの挙動
 - overrideフラグを立て近隣ホスト・ルータのNCを更新する
 - Home de-registrationが済んだ後、HAはプロキシNDエントリを削除する



2006/12/08

Internet Week 2006

39

HAにおけるプロキシND(2)

- 任意のMNのBCエントリを最初に作成する際に、プロキシNDの処理を開始する
 - proxy_nd_start()を呼び出す
 - 続いて呼び出されるneighmod()は、netlinkメッセージをカーネルに対して発行する

```

バインディング有効時のHAにおけるNeighbor Cache          プロキシNDエントリ
#ip -6 neigh show
<HA address> dev <Home I/F> lladdr <MAC address of Home I/F> STALE
<MN1_HoA> dev <Home I/F> proxy PERMANENT

プロキシNDエントリの追加ルーティン($MPIPL2USER/src/ndisc.c)      コマンド(nlmsg_type)
int pneath_add(int ifindex, uint8_t flags, struct in6_addr *dst)
{
    return neigh_mod(NLM_F_CREATE|NLM_F_REPLACE, RTM_NEWNEIGH,
                    ifindex, NUD_PERMANENT, flags, NTF_PROXY, dst,
                    NULL, 0);
}
                                                    フラグ(nlmsg_type)
    
```

2006/12/08

Internet Week 2006

40

IPsec/IKEとの協調・連携

- 移動環境におけるIPsecトンネルの利用
 - MNの移動に応じてIPsecトンネルを有効・無効化する必要性がある
 - MNの移動に応じてエンドポイント・アドレスを更新する必要性がある
 - XFRMポリシー(IPsec SP)中のテンプレートに含まれるアドレス
 - XFRMステート(IPsec SA)に含まれるアドレス
- K-bitサポート
 - IKE SAのエンドポイント更新機能
- Mobile IPv6からIPsec/IKEへの通知
 - 新しいエンドポイント・アドレスを非同期に通知する機構
 - 正しくターゲットを選択するために古いエンドポイント・アドレスの他、プロトコル等の情報も必要
 - MIPL2ではMIGRATE機構(参考資料5)を実装

2006/12/08

Internet Week 2006

41

Socket APIに対するMobile IPv6拡張

- RFC4584
 - ユーザランドプログラムがどのようにMobile IPv6特有のIPv6拡張ヘッダ領域にアクセスするか
 - 各種MIPv6関連アプリケーションで共通に利用可能なデータ構造
- MIPL2における実装状況
 - 現在はライブラリとして実装
 - 実際このライブラリを用いてmip6dは実装されている
 - \$MIPL2USER/src/libmissing/*

2006/12/08

Internet Week 2006

42

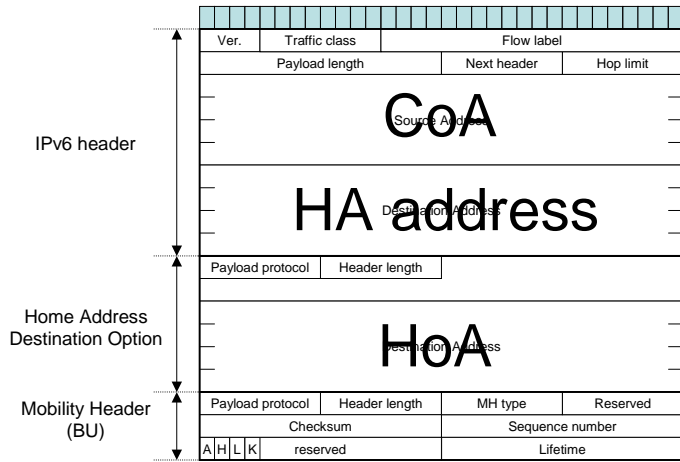
まとめ

- MIPL2 - RFC3775、RFC3776に準拠したオープンソース Mobile IPv6スタック
- カーネルへの修正を必要最低限に抑える設計
 - ユーザランドプログラム (mip6d) がシグナリングおよび状態管理を行う
 - カーネル内では必要最低限なパケット処理状態情報を保持する
- 二つの異なるパケット処理機構を採用
 - HA-MN間の双方向トンネルはIPv6トンネルデバイスによって処理
 - 経路最適化パケットの処理はXFRM機構の拡張によって処理

参考資料

1. RFC3775 Mobility Support in IPv6, D. Johnson, C. Perkins, J. Arkko, June 2004.
2. RFC3776 Using IPsec to Protect Mobile IPv6 Signaling Between Mobile Nodes and Home Agents, J. Arkko, V. Devarapalli, F. Dupont, June 2004.
3. RFC4225 Mobile IP Version 6 Route Optimization Security Design Background, P. Nikander, J. Arkko, T. Aura, G. Montenegro, E. Nordmark, December 2005.
4. RFC4584 Extension to Sockets API for Mobile IPv6, S. Chakrabarti, E. Nordmark, July 2006.
5. draft-sugimoto-mip6-pfkey-migrate-03.txt PF_KEY Extension as an Interface between Mobile IPv6 and IPsec/IKE, S. Sugimoto, F. Dupont, M. Nakamura, work-in-progress.

付録1：パケットフォーマット Home Registration BU (MN → HA)

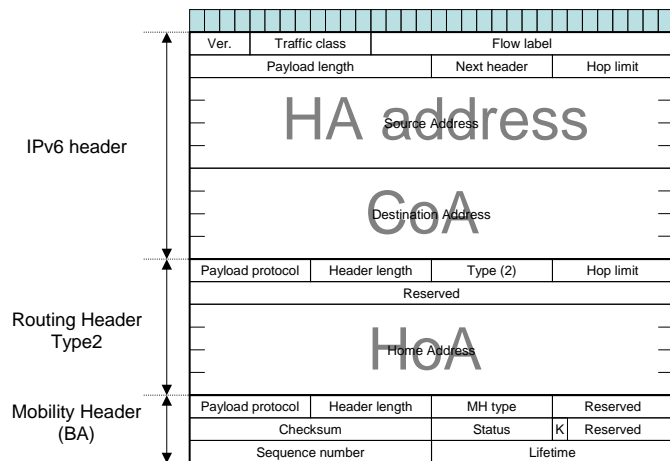


2006/12/08

Internet Week 2006

45

付録2：パケットフォーマット Home Registration BA (HA → MN)

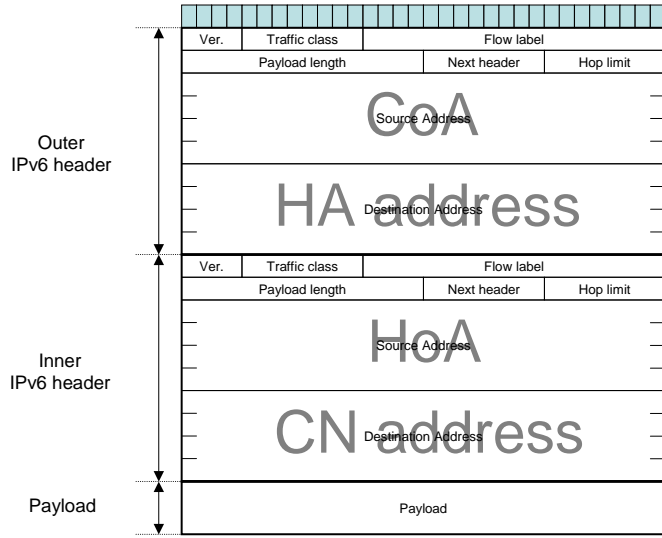


2006/12/08

Internet Week 2006

46

付録3：パケットフォーマット トンネルされたデータパケット(MN-CN)

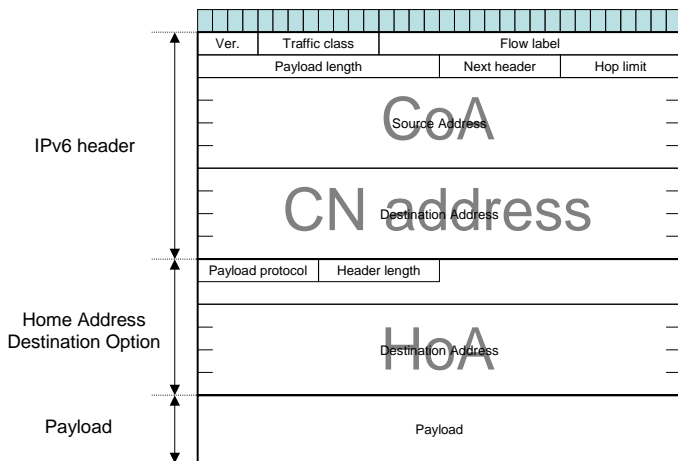


2006/12/08

Internet Week 2006

47

付録4：パケットフォーマット 経路最適化されたデータパケット(MN-CN)



2006/12/08

Internet Week 2006

48

付録5：パケットフォーマット

経路最適化されたデータパケット(CN MN)

