Dual Stack Implementation of Mobile IPv6 Software Architecture

Dr. K.L.Bansal
Associate Professor
Department of Computer Science
H.P.University, Shimla (H.P.) India-171005

Chaman Singh
Research Scholar
Department of Computer Science
H.P.University, Shimla (H.P.) India-171005

ABSTRACT

IPv6 is introduced mainly to resolve the address space issues and also provides several advanced features. IPv6 is estimated to replace IPv4 in a very near future. Dual Stack Mobile IPv6 (DSMIPv6) is an extension of Mobile IPv6 to support mobility of devices irrespective of IPv4 and IPv6 network. This paper provides an architectural overview of the existing DSMIPv6 implementation and software architecture to understand the significant modifications which have been made on DSMIPv6 basic implementation to achieve the requirements. The scope of the paper is to implement the Dual-stack Mobile IPv6 (DSMIPv6) protocol as per the IETF (Internet Engineering Task force) draft. The entities which have been implemented are 'DSMIPv6 Home Agent' and 'DSMIPv6 Mobile Node'. The paper covers overview of NEPL (Network Mobility platform for Linux) and DSMIPv6 implementation and briefly describes the features supported by DSMIPv6 architecture. It also focuses on our Solution Approach and explains the high level view of modules used in DSMIPv6 using a block diagram schematic.

General Terms

Implementation MIPv6

Keywords

Dual Stack, IPv4, IPv6, MIPv6

1. INTRODUCTION

The application interface is required to exchange mobility information with Mobility subsystem [1]. Mobile IPv6 (MIPv6) is a protocol developed as a subset of Internet Protocol version 6 (IPv6) [2] to support mobile connections. MIPv6 [3] allows a mobile node to transparently maintain connections while moving from one subnet to another. The Mobile IPv6 protocol takes care of binding addresses between Home Agent (HA) and Mobile Node (MN). It also ensures that the Mobile Node is always reachable through Home Agent. Each mobile node is always identified by its home address [4], regardless of its current point of attachment to the Internet. While situated away from its home, a mobile node is also associated with a care-of address, which provides information about its current point of attachment to the Internet. The protocol provides for registering the care-of address with a home agent. The home agent sends datagram's destined for the mobile node through a tunnel to the care-of address. After arriving at the end of the tunnel, each datagram is then delivered to the mobile node. Currently, two mobility management protocols are defined for IPv4 and IPv6. Deploying both in a dual stack mobile node introduces a number of problems. This has been improved [5]. Mobile IPv6 uses IPSec (IP Security) to protect signaling between the home agent and the mobile node [6]. Generic Packet Tunneling [7] Specifies a method and generic mechanisms by which a packet is encapsulated and carried as payload within an IPv6 packet. The resulting packet is called an IPv6 tunnel packet. The forwarding path between the source and destination of the tunnel packet is called an IPv6 tunnel. The technique is called IPv6 tunneling. A typical scenario for IPv6 tunneling is the case in which an intermediate node exerts explicit routing control by specifying particular forwarding paths for selected packets. This control is achieved by pre-pending IPv6 headers to each of the selected original packets. The current Mobile IPv6 [3] and Network Mobility [8] specifications support IPv6 only. These extend those standards to allow the registration of IPv4 addresses and prefixes, respectively, and the transport of both IPv4 and IPv6 packets over the tunnel to the Home Agent. [9] Allows the Mobile Node to roam over both IPv6 and IPv4, including the case where Network Address Translation is present on the path between the mobile node and its home agent.

2. ARCHITECTURAL REPRESENTATION OF DSIMIPV6

NEPL (NEMO Platform for Linux) [10] is a freely available implementation of DSMIPv6 for Linux platform. The original NEPL release was based on MIPL (Mobile IPv6 for Linux) [11]. In Figure-1: Basic Operation of DSMIPv6, all Mobile Nodes (MN) has a fixed address, called a Home Address (HoA), assigned by Home Agent. When the Mobile Node moves to other networks, it gets Care-of Address (CoA) from foreign network. Mobile Node sends a Binding Update (BU) message to its Home Agent. Then Home Agent replies to the Mobile Node with a Binding Acknowledgement (BA) message to confirm the request. When Mobile Node is moved to any foreign network all packets sent to the Home Agent will be IPSec encrypted. A bi-directional tunnel is established between the Home Agent and the Care of address of the Mobile Node after the binding information has been successfully exchanged.

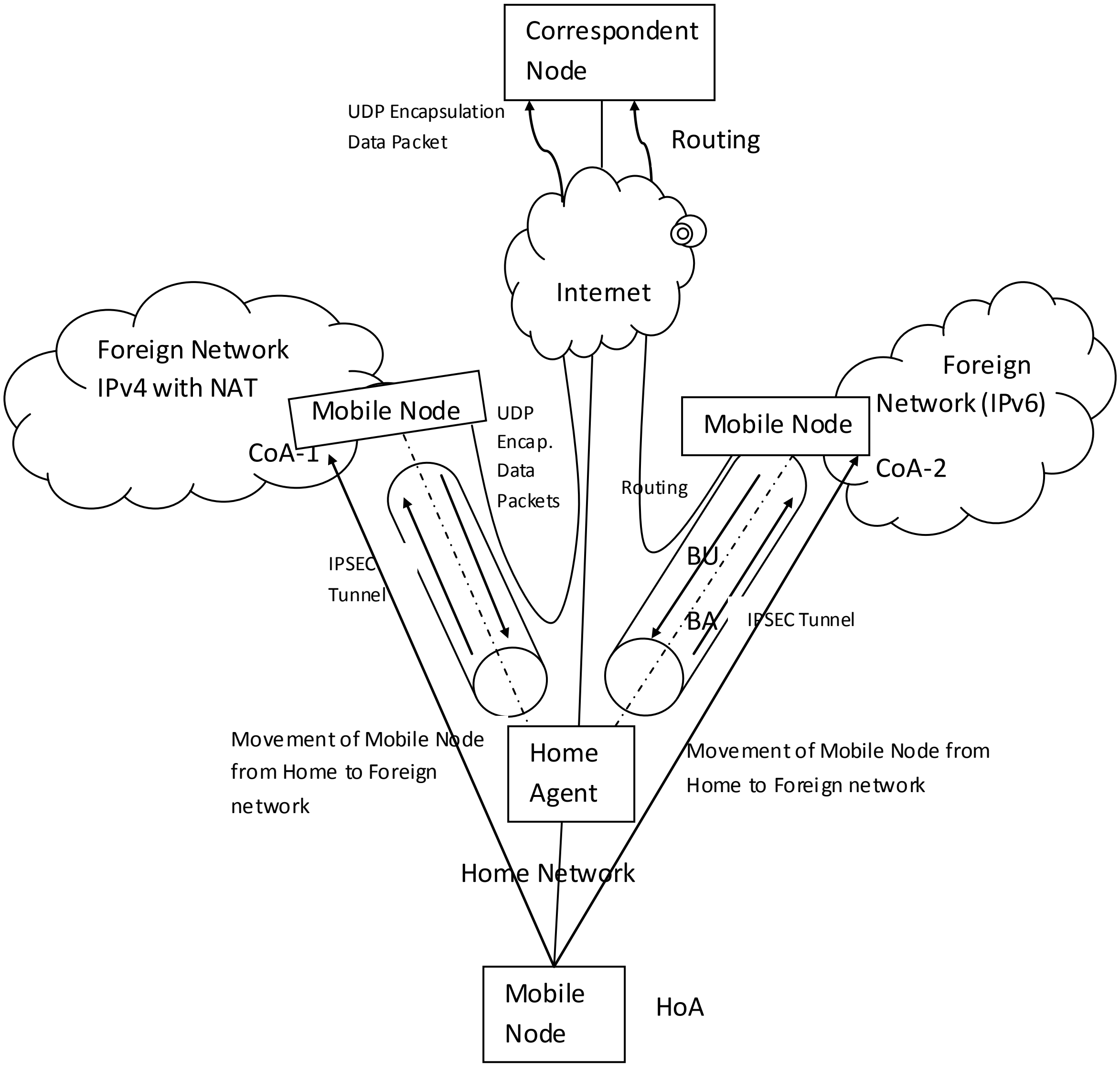

Fig 1: Basic Operation of DSMIPv6

DSMIPv6 extends the MIPv6 and NEMO [12] Basic Support standards to allow mobile nodes to roam in both IPv6 and IPv4-only networks. The following features are supported by the DSMIPv6 Architecture.

1. The mobile node can register an IPv6/IPv4 Care of address to its Home Agent and thus roam in IPv6-only networks and IPv4-only networks by the use of IPv6 tunnels and IPv6-in-IPv4 tunnels between the Mobile Node and its Home Agent.
2. A Network Address Translation Detection and Traversal Mechanism allow the Mobile Node to communicate with its Home Agent even though it uses an IPv4 private address as a Care of address. The signaling messages are always UDP encapsulated in IPv4 network. However, when the Mobile Node is located behind a NAT, data traffic is also encapsulated in UDP.
3. Securing the signaling packets between Home Agent and Mobile Node when Mobile Node is moved to foreign network.
4. Session management on movement from one foreign link to another.

2.1 Solution Description

The solution is an extension to the existing NEPL solution provided by Nautilus [10]. We validated the DSMIPv6 functionality as per the requirements provided against the draft, along with other IETF standards. We took the baseline architecture implementation from the Nautilus6 which uses Linux platform. The below mentioned steps are taken by us to achieve the requirements:-

1. Have setup DSMIPv6 Test Lab using Kernel 2.6.28.2 and UMIP veMyon 0.4. In order to test the basic functionality between Home Agent and Mobile Node

according to [3] the Test Bed has been setup.

2. Code changes have done in mip6d daemon and Linux kernel and also applied the open source patches/packages on Test Lab to meet the requirements.
3. The Routing Advertisement daemon (radvd), IPSec daemon (strongswan) and Web Server (httpd) daemon has been configured on Home Agent.
4. The Mobile Node is configured with IPSec daemon (strongswan). Mobile Node gets IPv6 address whenever it is moved to any IPv6 foreign network through the radvd server running on the router.
5. When Mobile Node is moved to IPv4 network, it gets configured with IPv4 Care of address from the DHCP server running on IPv4 Router.
6. In IPv4 network, DHCP is configured on the private network behind router. The network behind IPv4 router can be public or private

2.2 Block Diagram of Module Representation in DSMIPv6

MIPL (Mobile IPv6 for Linux) is an open-source implementation of the Mobile IPv6 standard for the GNU/Linux operating system. MIPv6 is a user space for Mobile Node and Home Agent which aims at providing the necessary changes to MIPL in order to run on the latest kernels.Figure-2: Block Diagram of MIPv6 shows the internal data flow between two major components i.e. Home Agent and Mobile Node. Both of these two components consist of several helper modules which are also shown in this figure.

2.3 Module Description

2.3.1 DNA/DHCP Module

This section describes IPv4 address assignment mechanism used by DSMIPv6.DHCP DNA module is used to obtain IPv4 address from the DHCP server running on IPv4 network.

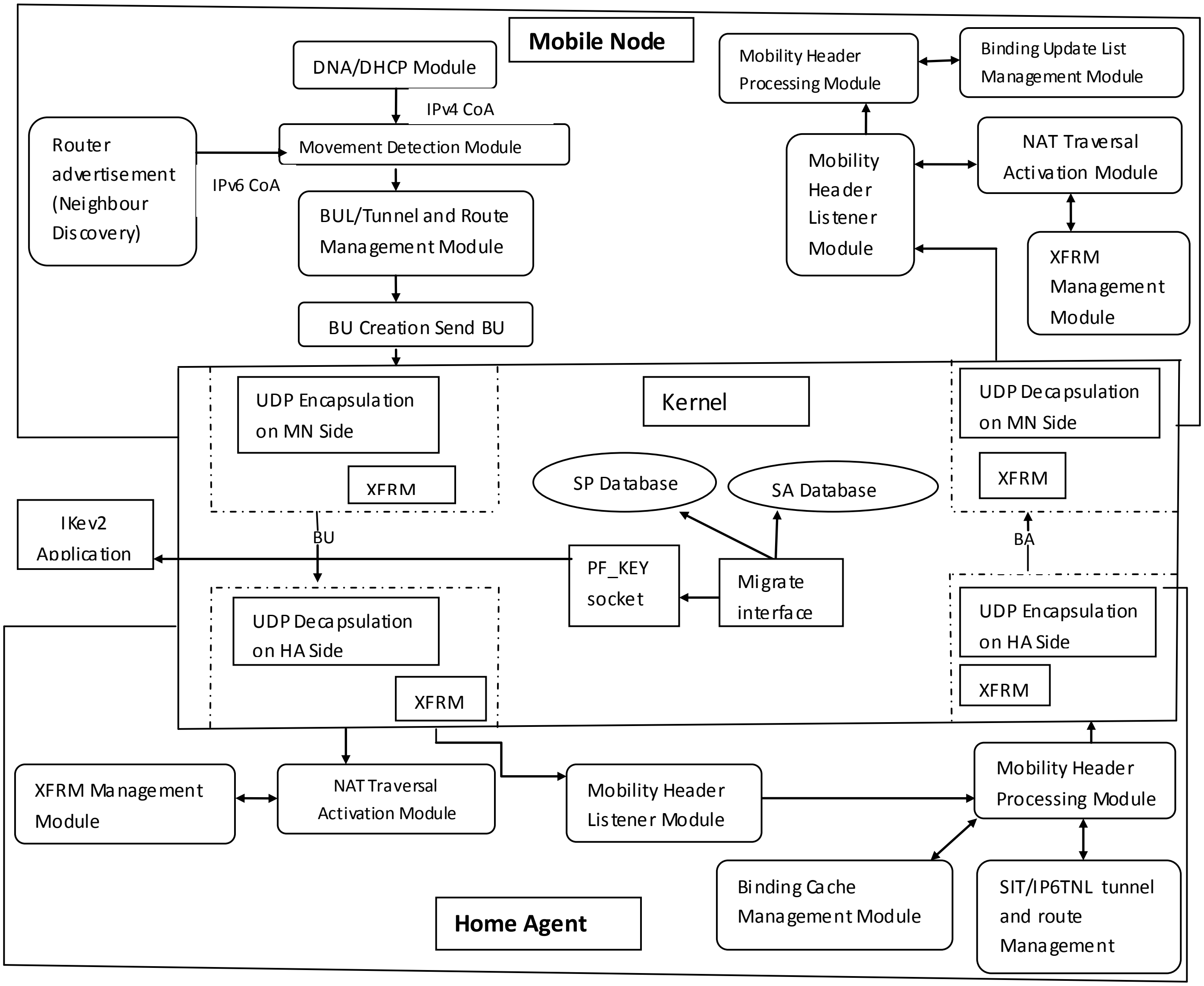
Mobile Node
DNA/DHCP Module
IPv4 CoA
Router advertisement (Neighbour Discovery)
Movement Detection Module
IPv6 CoA
BUL/Tunnel and Route Management Module
BU Creation Send BU
Mobility Header Processing Module
Binding Update List Management Module
Mobility Header Listener Module
NAT Traversal Activation Module
XFRM Management Module
Kernel
UDP Encapsulation on MN Side
XFRM
UDP Decapsulation on MN Side
XFRM
SP Database
SA Database
IKev2 Application
BU
BA
PF_KEY socket
Migrate interface
UDP Decapsulation on HA Side
XFRM
UDP Encapsulation on HA Side
XFRM
XFRM Management Module
NAT Traversal Activation Module
Mobility Header Listener Module
Mobility Header Processing Module
Binding Cache Management Module
SIT/IP6TNL tunnel and route Management
Home Agent

Fig. 2: MIPL for Home Agent and Mobile Node.

2.3.1.1 Process Description

When Mobile Node moves to IPv4 FL (Foreign Link) and its egress interface becomes enabled, Mip6d code in Mobile Node listens for Router Advertisement message, and since it does not receives Router Advertisement message in IPv4 FL, it gets timeout and sends Router Solicitation message (that will request the router to generate the Router Advertisement message immediately rather than at there next scheduled time), and Mobile Node wait for some time interval for Router Advertisement message before repeating the same procedure of sending Router Solicitation message. Meanwhile after sending Router Solicitation message, mip6d daemon will check the presence of DHCP server on the Egress interface link of Mobile Node by sending the DHCP discover message and wait for DHCP offer packet. Since the DHCP server is running on the IPv4 FL, it gets the IPv4 address from DHCP server and then mip6d code maps IPv4 address to IPv6 address, which is further used as Care of address. Mip6d daemon sets the default route on Mobile Node fig 3.

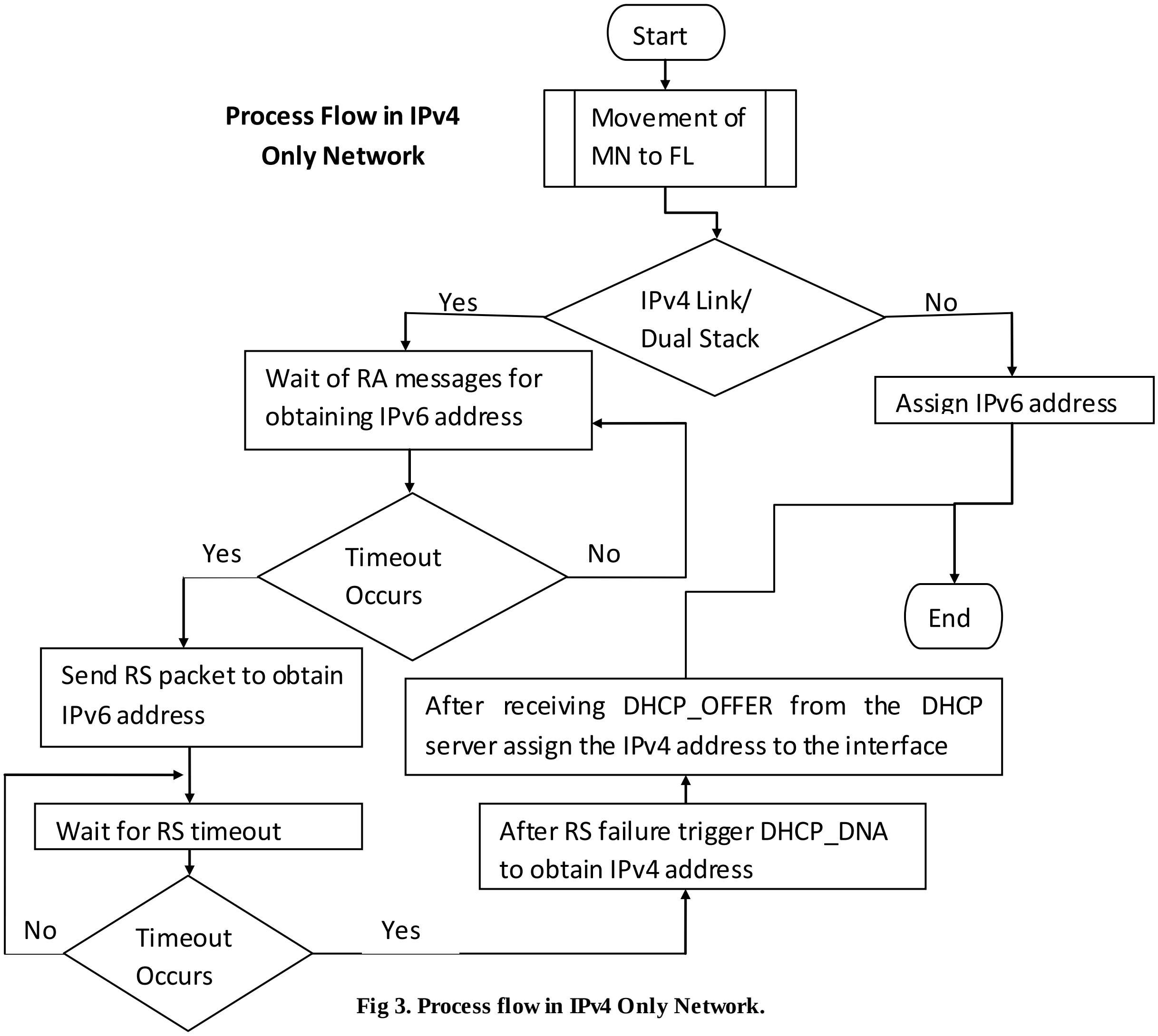

Fig 3. Process flow in IPv4 Only Network.

2.3.1.2 Data Structure

Struct dhcp_dna_control_s contains the DHCP client state machine, lease time and timeout information. File mipv6-daemon-umip-0.4/src/dhcp_dna.h is used.s and second Struct dhcp_message contains the information about DHCP messages send to and received from the server. File mipv6-daemon-umip-0.4/src/dhcp_dna.c is used. When Mobile Node moves to IPv4 only network or dual stack network then only this DHCP DNA module comes into the picture. Mobile Node first tries to acquire IPv6 Care of address and failure in IPv6 address configuration triggers the DHCP DNA code which sends dhcp discover messages on the network to acquire the IPv4 Care of address. The table-1 show the various methods used.

2.3.2 Movement Detection Module

This section describes Movement Detection module in DSMIPv6 implementation. The movement of a mobile node away from its home link is transparent to transport and higher-layer protocols and applications.

Table 1. DNA/DHCP Methods.

Function	Description	Input Parameter	Return Value	Caller	Define in file
Send_discover	Broadcast a DHCP discover packet to the network with an optionally requested IP	dhcp_ctrl : DHCP information long xid: Client IDrequested: Optionally requested IP	In case of error return value<1	DHCP_listen	DHCP_DNA.C
Send_select	Broadcast a DHCP request packet to the network	Server: SERVER ID & same as in Function 1	In case of error return value<1	DHCP_listen	DHCP_DNA.C
Send_renew	Broadcast a DHCP renew request packet on the network	Server: SERVER ID & same as in Function 1	In error return value<1	DHCP_listen	DHCP_DNA.C
Send_release	Unicasts a DHCP release message	Server: SERVER ID & same as in Function 1	In case of error return value<1	DHCP_listen	DHCP_DNA.C
Dhcp_dna_init	Starts the dhcp client state machine	none	In case of error return value<1	main	DHCP_DNA.C
Dhcp_listen	contains dhcp client state machine logic	args: argument passed to the function	In case of error return value<1	Dhcp_dna_int	DHCP_DNA.C
Get_packet	Read a packet from socket fd	dhcp_message: dhcp_message received ,fd: socket	In case of error return value<1	Dhcp_dna_int	DHCP_DNA.C

The Movement detection uses Neighbor Unreachability Detection [13] to detect when the default router is no longer bi-directionally reachable, in which case the mobile node must discover a new default router (usually on a new link). However, this detection only occurs when the mobile node has packets to send. When the mobile node detects handover, it expire the previous routers and Care-of-Address (es) and selects a new default router as a consequence of Router Discovery, and then performs Prefix Discovery with that new router to form new care-of address (es). This is handled in 'movement.c'. It then registers its new primary care-of address with its home agent. After updating its home registration, the mobile node then updates associated mobility bindings in correspondent nodes. It triggers a movement event and then detects that Mobile Node is in foreign network. Route is modified and Home Address which was previously assigned to the physical interface is now moved to the tunnels. This is handled in 'mn.c'.

2.3.3 Binding Management Module

When a Mobile Node moves between different networks, it is essential that binding update messages are sent to that node's Home Agent and Correspondent Nodes as soon as possible, in order to facilitate a fast handoff. Mobile Nodes therefore cannot rely on the soft state timeout mechanism used in binding caches to refresh stale bindings maintained by Correspondent Nodes (typical binding lifetimes are of the order of minutes). An additional data structure, the binding update list, is therefore kept by Mobile Nodes, which maintains state on any Correspondent Nodes or Home Agents. And in Home Agent mobile IPv6 binding cache maps home addresses to the current care-of addresses for each mobile node. This allows the home agent to tunnel traffic to the mobile node at its current location, and allows a correspondent node to send packets directly to a mobile node at its current location. Binding Management Module is classified as:

- Binding Update List
- Binding Cache

Binding Update: When Mobile Node moves to any Foreign Link, Mip6d code in Mobile Node initiate binding update functionality after configuring Care of address either from Router Advertisement message received in IPv6 or Dual stack FL or from DHCP server in IPv4 FL on the egress interface of Mobile Node. After Mobile Node configures Care of address on its egress interface, it creates BUL which consists of information about Mobile Node's Home Address, Care of address, CN address, life time and delay time of this binding message in seconds and set of various flag.

Binding Update List format is as follows:

```
== BUL_ENTRY ==

Home address    2001: x: x: x: x: x: x
Care-of addresses 2001:  x: x: x: x: x: x
CN address      2001:  x: x: x: x: x: x
Lifetime = 32, delay = 1500
Flags: IP6_MH_BU_HOME
IP6_MH_BU_ACK
IP6_MH_BU_TLV
```

When Mip6d code in Mobile Node creates BUL successfully, it send Binding Update message to Home Agent to register the new Care of address of Mobile Node to Home Agent. So that Home Agent gets updated with current Care of address of Mobile Node.

Binding Cache: Mip6d code in Home Agent received Binding Update message sent by Mobile Node when it moves to any Foreign Link or when it directly boots in IPv6/IPv4 FL. Home Agent process BU by performing DAD (dynamic address discover)[14] and latter parse Binding Update message. After paMyng the binding update message Home Agent creates or updates its Binding Cache entry. Binding Cache entry consists of Mobile Node's Home Address, Mobile Node's Care of address, its local address, lifetime and sequence no.

Binding Cache entry format is as follows:

```
          == BC_ENTRY ==

HoA 2001: a: b: 0:0:0:0:1 status registered
 CoA 2001: a: d: 1:20c:29ff:fea0:4026 flags AH-- Local 2001: a:
b: 0:0:0:0:1000
 Lifetime 23 / 32 seq 3435
Unreach 0 / 959299 retry -2
```

After updating its Binding Cache entry, the mip6d code creates devices to tunnel traffic to Mobile Node. Mip6d code in Home Agent creates and send Binding Acknowledgment message to Mobile Node so that Mobile Node gets acknowledged that it's new Care of address gets successfully registers with Home Agent. When Mobile Node receives Binding Acknowledgment message, mip6d code in Mobile Node parses the BA packet and update the BUL entry. It checks for various options set in BA and proceed accordingly. If NAT is detected between Mobile Node and Home Agent, it set xfrm policies/states to UDP Encapsulate IPv6/IPv4 data traffic to bypass NAT. Mip6d daemon sets the callback function to resend the BA, once lifetime of BUL entry is expired. And finally set the binding update timer to decrease the lifetime of BUL entry. When Mobile Node moves back from any FL to Home Link, Mip6d code in Mobile Node sends Binding Update message with lifetime set as zero to Home Agent to indicate that it as returned to Home Link mip6d code in Mobile Node deletes corresponding BUL entry. On Home Agent side Mip6d code receives BU message with lifetime set as zero, it indicate that Mobile Node moved to Home Link, so it deletes corresponding Binding Cache entry and send Binding Acknowledgment back to Mobile Node.

2.3.4 Tunnel and Route Management Module

Tunnel and Route Management module is mainly responsible for tunneling, when mobile node changes from IPv6 to IPv6, IPv6 to IPv4 and vice versa network. This module configures sit and ip6tnl interface via IOCTL system call which in turns performs the task at kernel level. Route Management handles the return routability with CN. Some data structures are being used between some of the important functions in Tunnel Management module. Some user land data structures used in various routines in tunnel management module. Some data structures used in various routines in tunnel management module.

2.3.5 XFRM and IPSec Module

XFRM [15] is a packet transformation framework residing in the Linux kernel. It performs operations on IP packets such as inserting, modifying headers, UDP encapsulation and de-capsulation. DSMIPv6 XFRM module will take the advantage of existing IPSEC transformation and defines a simple UDP encapsulation scheme. IPSEC module is responsible for interaction with IKE through MIGRATE messages. IPSec will be used to protect the following traffic between Home Agent and Mobile Node.

1. BU/BA messages.
2. Mobile prefix sollicitation and advertisement messages.
3. Normal traffic between Mobile Node and Home Agent.
4. All tunneled normal traffic between Mobile Node and correspondent Node.

2.3.5.1 Module name and Functionality

In Mip6d, the Mobile Node (MN) and the Home Agent (HA) uses IPsec Security Associations (SAs) in transport mode to protect BU/BA messages, since the MN may change its attachment point to the Internet, it is necessary to update its endpoint address of the IPsec SAs. This indicates that corresponding entry in IPsec databases (Security Policy (SPD) and SA (SAD) databases) should be updated when Mobile Node performs movements. IPSec is used to protect the following traffic between Home Agent and Mobile Node:

1. BU/BA messages

Process Description (IPSec Protection for BU/BA) When Mobile Node move in FL a new Care of address is assigned to the Mobile Node by FL network. After detecting the movement following steps are taken to create IPSec tunnel.

1. Mip6d issues a PF_KEY MIGRATE message to the PF_KEY socket.
2. The operating system validates the message and checks if corresponding security policy entry exists in SPD.
3. When the message is confirmed to be valid, the target SPD entry is updated according to the MIGRATE message. If there is any target SA found that are also target of the update, those should also be updated.
4. After the MIGRATE message is successfully processed inside the kernel, it will be sent to all open PF_KEY sockets. The IKE daemon receives the MIGRATE message from its PF_KEY socket and updates its SPD

and SAD images. The IKE daemon may also update its state to keep the IKE session alive.

5. After that ESP protected BU is send with K–bit set.

Mobile IPv6 specifies a flag named Key Management Mobility Capability bit (K-bit) in Binding Update (BU) and Binding Acknowledgement (BA) messages, which indicates the ability of IKE sessions to survive movement. When both the Mobile Node and Home Agent agree to use this functionality, the IKE daemons dynamically update the IKE session when the Mobile Node moves. The following methods are used.

Table 2. XFRM and IPSec Methods.

Function	**Description**	**Input Parameter**	**Return Value 0**	**Caller**	**Define in file**
mn_ipsec_trns_update	Update transport mode SA used for signaling	haaddr: Home Agent Address hoa: Mobile Node home address arg: bule containing old and new CoA.	In case of error return integer value less than 0	mn_pol_ext_clean up mn_send_home_b u	ipsec.c
mn_trns_update	Update transport mode SA used for signaling by issuing migrate message	haaddr :Home Agent Address hoa : MN home address ipsec-policy: IPSec policy entry arg :bule contains old, new CoA	In case of error return integer value less than 0	mn_ipsec_trns_up date	ipsec.c
xfrm_sendmigrate	Send migrate message to the kernel	xfrm_userpolicy_info : Home Agent Address xfrm_userpolicy_info: MN home address, Source, dst: Destination.	In case of error return integer value less than 0	mn_trns_update	ipsec.c
ha_ipsec_trns_update	Update transport mode SA used for signaling on HA	haaddr: Home Agent Address hoa: MN home address old: old CoA, : new CoA. tunnel : Tunnel Info	In case of error return integer value less than 0	home_tnl_add home_tnl_chg	ipsec.c
ha_trns_update	Update transport mode SA used for signaling by issuing migrate message on HA	haaddr : Home Agent Address hoa : MN home address ipsec-policy :IPSec Policy Entry arg: BULE containing old and new CoA.	In case of error return integer value less than 0	ha_ipsec_trns_upd ate	ipsec.c

2.3.6 NAT Detection and Traversal Module

NAT (Network Address Translation) is the translation of an Internet Protocol address (IP address) used within one network to a different IP address known within another network. One network is designated the inside network and the other is the outside. In DSMIPv6 the mip6d daemon should bypass NAT, when Mobile Node is behind NAT' ed device in IPv4 FL. NAT detection is done when the initial Binding Update message is sent from the mobile node to the home agent. When located in an IPv4-only foreign link, the mobile node sends the Binding Update message encapsulated in UDP (User Datagram Protocol) and IPv4; this is handled in xfrm.c file. The mip6d daemon adds xfrm policy/state for UDP encapsulation for BU packet. When the home agent receives the encapsulated Binding Update, it compares the IPv4 address of the source address field in the IPv4 header with the IPv4 address included in the IPv4 care-of address option. If the two addresses match, no NAT device is in the path. Otherwise, a NAT is detected in the path and the NAT detection option is included in the Binding Acknowledgement. The Binding Acknowledgement, and all future packets, is then encapsulated in UDP and IPv4. Note that the home agent also stores the port numbers and associates them with the mobile node's tunnel in order to forward future packets. This is handled in ha.c file. The mip6d daemon adds the xfrm polices/states for UDP encapsulation of BA and IPv6/IPv4 data traffic. Upon receiving the Binding Acknowledgement with the NAT detection option, the mobile node sets the tunnel to the home agent for UDP encapsulation. Hence, all future packets to the home agent are tunneled in UDP and IPv4. If no NAT device is detected in the path between the mobile node and the home agent then IPv4/IPv6 data traffic is not UDP encapsulated. A mobile node will always tunnel the Binding Updates in UDP when located in an IPv4-only network. Essentially, this process allows for perpetual NAT detection. Similarly, the home agent will encapsulate Binding Acknowledgements in a UDP header whenever the Binding Update is encapsulated in UDP. This is handled in mn.c and xfrm .c file. The mip6d daemon adds xfrm polices/states for UDP encapsulation of IPv6/IPv4 data traffic, when NAT is detected between Mobile Node and Home Agent.

2.3.7 Mobility Listener Module

The Mobility Header is an extension header used by mobile nodes, correspondent nodes, and home agents in all messaging related to the creation and management of bindings. The Mobility Header is identified by a Next Header value of 135 in the immediately preceding header. This Header is used to carry the following messages:

Home Test Init

A mobile node uses the Home Test Init (HoTI) message to initiate the return routability procedure and request a home keygen token from a correspondent node. The Home Test Init message uses the MH Type value 1.

Home Test

The Home Test (HoT) message is a response to the Home Test Init message, and is sent from the correspondent node to the mobile node. The Home Test message uses the MH Type value 3.

Care-of Test Init

A mobile node uses the Care-of Test Init (CoTI) message to initiate the return routability procedure and request a care-of keygen token from a correspondent node. The Care-of Test Init message uses the MH Type value 2.

Care-of Test

The Care-of Test (CoT) message is a response to the Care-of Test Init message, and is sent from the correspondent node to the mobile node. The Care-of Test message uses the MH Type value 4.

Binding Update

The Binding Update (BU) message is used by a mobile node to notify ther nodes of a new care-of address for itself. The Binding Update uses the MH Type value 5.

Binding Acknowledgement

The Binding Acknowledgement is used to acknowledge receipt of a Binding Update. The Binding Acknowledgement has the MH Type value 6.

Binding Refresh Request

The Binding Refresh Request (BRR) message requests a mobile node to update its mobility binding. This message is sent by correspondent nodes. The Binding Refresh Request message uses the MH Type value 0.

Binding Error

The Binding Error (BE) message is used by the correspondent node to signal an error related to mobility, such as an inappropriate attempt to use the Home Address destination option without an existing binding. The Binding Error message uses the MH Type value 7.

3. ASSUMPTION AND LIMITATIONS

1) Dynamic IPv4 Home Address assignment to Mobile Node using IKEv2.

2) Mobile Node with multiple tunnel interfaces.

3) The reqid (request ID's) defined in the IPsecPolicy lines of the mip6d.conf files currently must exactly match the reqid assigned by strongswan to the corresponding IPSec SA. Strongswan does the assignment using a linear counter starting with reqid 1. Otherwise the communication between the mip6d and strongSwan daemons via MIGRATES and ACQUIRE kernel messages is simply not going to work. Thus make sure that Mobile Node-Home Agent connections are started in the correct order, i.e. in our example first the connection from Mobile Node carol ((request ID's) 1 and 2) and only after that the connection from Mobile Node dave ((request ID's) 3 and 4).

4) Home Agent behind NAT

5) DHAAD (Dynamic Home Agent Address Detection)

4. CONCLUSION

The paper represents the software architecture of dual stack implementation of mobile IPv6. This allover implementation is done in computer laboratory. Various functions, structure, servers are used to implement this paper. Today's we are going to implement network address translation (NAT) and its detection and traversal on dual stack implementation on Mobile IPv6. NAT (Network Address Translation) is the translation of an Internet Protocol address (IP address) used within one network to a different IP address known within another network. One network is designated the inside network and the other is the outside. In DSMIPv6 the mip6d daemon should bypass NAT, when Mobile Node is behind NAT device in IPv4 Foreign Link.

5. REFERENCES

[1] T.Momoseetal, The Internet Engineering Task Force, July 2005 "The application interface to exchange mobility information with Mobility subsystem", Internet Drafts draft-momose-mip6-mipsock-00.

[2] Vida, R. and L. Costa, Eds., RFC 3810, June 2004. "Multicast Listener Discovery VeMyon 2 (MLDv2) for IPv6".

[3] Perkins, C., RFC 3344, August 2002. "IP Mobility Support for IPv4".

[4] Johnson, D., Perkins, C., and J. Arkko, RFC 3775, June 2004. "Mobility Support in IPv6".

[5] G.Tsirtsis, Qualcomm, H. Soliman, Elevate Technologies, [RFC 4977], August 2007. "Dual Stack Mobility".

[6] Arkko, J., Devarapalli, V. and F. Dupont, RFC 3776, June 2004. "Using IPsec to Protect Mobile IPv6 Signaling Between Mobile Nodes and Home Agents".

[7] Conta, A. and S. Deering, RFC 2473, December 1998. "Generic Packet Tunneling in IPv6 Specification".

[8] F Ralf Spenneberg, ipsec-howto, 2003-08-18.

[9] H. Soliman, Ed., Elevate Technologies, November 3, 2008. Mobile IPv6 Support for Dual Stack Hosts and Routers draft-ietf-mext-nemo-v4traversal-06.txt.

[10] NEPL (NEMO Platform for Linux) how to, June 24th, 2009.

[11] MIPL (Mobile Ipv6 for Linux), how to, 2004-4-20.

[12] Vijay Devarapalli, Ryuj Wakikawa, Alexandru Petrescu, and Pascal Thubert. Network Mobility (NEMO) basic support Protocol. RFC 3963, IETF Jan 2005.

[13] W. Simpson, Daydreamer, H. Soliman, December 2007. "Neighbor Discovery for IP version 6 (IPv6)", RFC 4861.

[14] Bauer, C., Ehammer, October 2008, "Securing Dynamic Home Agent Address Discovery with Cryptographically Generated Addresses".

[15] Yoshifuji Hideaki and al., In special section on internet technology IV, IEICE Trans Comumun, Vol.E87-B, No3 March 2004. Linux IPv6 Stack Implementation based on Serialized Data State Processing.