

Title:	Deliverable D6.1 Report on case studies and initial prototypes	Document Version: 2.4
---------------	---	-------------------------------------

Project Number: 027002	Project Acronym: ENABLE	Project Title: Enabling efficient and operational mobility in large heterogeneous IP networks
----------------------------------	-----------------------------------	---

Contractual Delivery Date: 31/12/2006	Actual Delivery Date: 22/12/2006	Deliverable Type* - Security**: R – PU
---	--	--

* Type: P – Prototype, R – Report, D – Demonstrator, O – Other
 ** Security Class: PU- Public, PP – Restricted to other programme participants (including the Commission), RE – Restricted to a group defined by the consortium (including the Commission), CO – Confidential, only for members of the consortium (including the Commission)

Responsible and Editor/Author: Miguel Ponce de Leon	Organization: WIT-TSSG	Contributing WP: WP6
---	----------------------------------	--------------------------------

Authors (organizations): Ronan, John (WIT-TSSG), Clancy, Niall (WIT-TSSG), Cleary, Frances (WIT-TSSG), Mayer, Karl (IABG), Fritsche, Wolfgang (IABG), Brück, Timm (IABG), Menardi Marco (TI), La Monaca, Michele (TI), Mussabbir, Qazi (Brunel), Yao Wenbing (Brunel), Yang, Ting (Huawei)
--

Abstract: This document describes a search & rescue scene management demonstration scenario which is being used to verify the technical requirements of a Mobile IPv6 service environment. The document also focuses on the planning of software developments and test-bed integration effort for functionalities being prototyped from WP 1, 2, 3, 4 and 5, which are to be included in the final ENABLE demonstration.
--

Keywords: Case study, scenarios, software development, prototypes, test-bed, IPv6, Mobile IPv6.

Revision History

The following table describes all the main changes completed on the document since its creation.

Revision	Date	Description	Author (Organization)
V0.1	14/07/2006	Document creation and initial ToC	M. Ponce de Leon (WIT-TSSG)
V0.2	02/10/2006	Additions to Section 3 Application Scenarios	M. Ponce de Leon (WIT-TSSG)
V0.3	13/10/2006	Additions to Section 3 Application Scenarios	M. Ponce de Leon (WIT-TSSG)
V0.4	20/10/2006	Modified ToC as per Meeting #3	M. Ponce de Leon (WIT-TSSG)
V0.5	07/11/2006	Additions to Section 4.1 Integrated Software architecture	M. Menardi (TI)
V0.6	16/11/2006	Updates to Section 2	M. Ponce de Leon, N. Clancy, F. Cleary (WIT-TSSG)
V0.7	22/11/2006	Additions to section 3.2.3, 4.1.2, and 4.1.3.	Karl Mayer, Wolfgang Fritsche, Timm Brück (IABG)
V0.8	22/11/2006	Section 3.2.1, 3.2.4, 5 added. Section 4.1 updated.	M. Menardi, Michele La Monaca (TI)
V0.9	23/11/2006	Added Section 3.2.6 and 4.2.2	Q Mussabbir , W Yao (Brunel)
V0.10	23/11/2006	Updates to 4.1 and 3.2.7.2	Marco
V0.11	23/11/2006	Section 3.2.5 added. Additions to Section 4.1	Pedro Garcia (UMU), Alejandro Perez (UMU)
V0.12	23/11/2006	Merged contributions and added more to section 2	M.Ponce De Leon, N. Clancy, F. Cleary (WIT-TSSG)
V0.13	24/11/2006	Obsolete	
V0.14	24/11/2006	Added a process diagram (reactive mode) in section 4.2.2.4.3	Q Mussabbir , W Yao (Brunel)
V0.15	28/11/2006	Added comments on reviewed Section 3.2.2 & 3.2.3 also added on behalf of UGOE Section 3.2.5	M.Ponce De Leon, N. Clancy, F. Cleary (WIT-TSSG)
V0.16	29/11/2006	Added/revised all UMU sections	Pedro Garcia (UMU), Alejandro Perez (UMU)
V0.17	30/11/2006	Updates to section 4.1	M. Menardi (TI)
V0.18	04/12/2006	Updates to section 3.2.5, 4.2.1	N. Steinleitner (UGOE)
V0.19	05/12/2006	Additions made to Section 2 Some modifications in section 3.2.3, 4.1.2, and 4.1.3.	M.Ponce De Leon, N. Clancy, F. Cleary (WIT-TSSG) Karl Mayer (IABG)

V0.20	07/12/2006	Some minor modifications in section 3.2.4, 4.1.2.1, 4.1.2.2, and 4.1.3.7.	M. Menardi (TI)
V0.21	07/12/2006	Additions made to Section 2 Minor modifications in section 3.2.3, 4.1.2, and 4.1.3.	M.Ponce De Leon, N. Clancy, F. Cleary (WIT-TSSG) Karl Mayer (IABG)
V0.22	08/12/2006	Section 2 & Section 4.4 additions	M.Ponce De Leon, N. Clancy, J. Ronan (WIT-TSSG)
V1.0	08/12/2006	Added Introduction, Conclusion & Executive Summary	M.Ponce De Leon, N. Clancy, J. Ronan (WIT-TSSG)
V1.1	11/12/2006	Added section 4.1.3.13, minor modifications in section 4.1.2.2, comments fixed	M. Menardi, M. La Monaca (TI)
V1.2	12/12/2006	Removed Section 4.2.2.3.1 & 4.2.2.4 and moved them to section 5.3, 5.3.1, 5.3.1.1, 5.3.2, 5.3.2.1 and 5.3.2.2 accordingly. Also added an explanation and diagram of the software architecture section 4.2.2. A diagram of the message flow chart was also added in section 4.2.2.2	Qazi Bouland Mussabbir, W. Yao (Brunel)
V1.3	14/12/2006	Review of the document until (including) section 4.1. Section 3.2.3 and the Ac interface description in section 4.1.2.4 updated	Karl Mayer (IABG)
V1.4	14/12/2006	Addressing review comments made thus far.	M.Ponce De Leon (WIT-TSSG)
V1.5	15/12/2006	Modify the style of the figures in section 4.2.2 Figure 4-28,4-30 ,4-31,4-32, 4-33	Ting Yang (Huawei)
V1.6	18/12/2006	Revision to document and added section 4.1.3.4 Revision to some sentences and comments to Figure 2-2 and section 2.2.2.5. Slight modifications to Figure 5-1 & 5-2 and additions made to section 4.1.3.13 and section 3.2.1. Revised text due to comments Addressing further review comments	M. A. Diaz (CONSULINTEL) X. Yang (UGOE) M. Menardi (TI) A. Perez (UMU) M.Ponce De Leon (WIT-TSSG)
V2.0	19/12/2006	Added Figures 3-18 & 4-29 and modified slightly Section 4.1.2.4.3	M.Ponce De Leon (WIT-TSSG)

027002	ENABLE	D6.1:Report on case studies and initial prototypes
--------	--------	--

V2.1	21/12/2006	Addressing PSC review comments	M.Ponce De Leon, N. Clancy, J. Ronan (WIT-TSSG)
V2.2	21/12/2006	Revision to 4-28,4-30 ,4-31,4-32, 4-33	Ting Yang (Huawei)
V2.3	21/12/2006	Update to Section 3.2.2 and Section 4.2.2.2 Revision of Section 3.2.5 and Sections 4.2.1.1-4.2.1.3.	Alejandro Perez (UMU), M. La Monaca (TI) N. Steinleitner (UGOE), J. Ronan (WIT-TSSG)
V2.4	22/12/2006	Cleaning up PSC review comments.	M.Ponce De Leon (WIT-TSSG)

Executive Summary

This document is a comprehensive summary of an application scenario (search & rescue scene management) and the design of software developments & test-bed integration effort for functionalities being prototyped from WP 1, 2, 3, 4 and 5 of ENABLE.

This document provides the basis for how ENABLE will achieve three of its key seven objectives. That is to say, it will show how ENABLE will develop the required technologies to enable the deployment of Mobile IPv6 in real-life environments (Objective 2), investigate solutions to improve the reliability of Mobile IPv6 and enable an optimal usage of network resources for the deployment of Mobile IPv6 in a provider network (Objective 3) and it goes some way towards showing how ENABLE will validate the results of the developed mechanisms and technologies through prototyping and laboratory testing (Objective 6).

The document starts with a description of an application scenario based on search & rescue scene management. The approach for identifying the scenario and detailed description of the search & rescue are given.

The combination of the sections on technological developments, initial prototyping and initial test-bed design gives the detailed design of the software architecture, interface descriptions and software modules for six technological components that are to be developed in the project.

It is expected that once the software components are tested and integrated, they can be run as the application scenario (search & rescue) and this will show how ENABLE research has facilitated efficient and operational mobility in large heterogeneous IP networks.

Table of Contents

1.	<i>Introduction</i>	10
2.	<i>Application Scenario</i>	11
2.1	<i>Approach</i>	11
2.1.1	IST Ambient Networks	12
2.1.2	IST Daidalos	15
2.1.3	IST ePerSpace	16
2.1.4	IST Simplicity	19
2.1.5	IST Advisory Board	20
2.1.6	IST ENABLE	21
2.2	<i>Search & Rescue Scene Management Scenario Description</i>	23
2.2.1	Environment & Assumptions	27
2.2.2	Scene 1: Search and Rescue is initiated	29
2.2.2.1	Scene Challenge	29
2.2.2.2	Supported services	30
2.2.2.3	Mobility Issues	30
2.2.2.4	User experience	31
2.2.3	Scene 2: Assets (People & vehicles) are deployed	31
2.2.3.1	Scene Challenge	31
2.2.3.2	Supported services	32
2.2.3.3	Mobility Issues	32
2.2.3.4	User experience	32
2.2.4	Scene 3: Not enough assets on site, volunteers called in	33
2.2.4.1	Case Study Detail	33
2.2.4.2	Environmental and General Assumptions	34
2.2.4.3	Scene Challenges	34
2.2.4.4	Supported Services	35
2.2.4.5	Mobility Issues	35
2.2.4.6	User Experience	36
2.2.4.7	Mapping of Scene 3 to the Test-Bed	36
2.2.5	Scene 4: Areas of location not covered by Network Operator	37
2.2.5.1	Scene Challenge	37
2.2.5.2	Supported services	38
2.2.5.3	Mobility Issues	38
2.2.5.4	User experience	38
2.2.6	Scene 5: Rescue victim found, Ambulance deployed	38
2.2.6.1	Scene Challenge	39
2.2.6.2	Supported services	39
2.2.6.3	Mobility Issues	39
2.2.6.4	User experience	40
2.2.7	Scene 6: Ambulance transports the victim from rescue scene to hospital	40
2.2.7.1	Case Study Detail	40
2.2.7.2	Environmental and General Assumptions	41
2.2.7.3	Scene Challenges	42
2.2.7.4	Supported Services	43
2.2.7.5	Mobility Issues	43

2.2.7.6	User experience	43
2.2.7.7	Mapping of Scene 6 to the Test-Bed	44
2.3	Business Entity Perspective	45
2.3.1	Access Service Authoriser	45
2.3.2	Access Service Provider	45
2.3.3	Mobility Service Provider	46
2.3.4	Mobility Service Authoriser	46
3.	Technological developments	47
3.1	Introduction	47
3.2	Functional Components of ENABLE	47
3.2.1	EAP-based MIPv6 bootstrapping	49
3.2.1.1	Configuration TLV	50
3.2.1.2	LCP packets	51
3.2.1.3	Configure-Request	52
3.2.1.4	Configure-Ack	52
3.2.1.5	Configure-Nak	52
3.2.1.6	MIPv6 Configuration Options	53
3.2.2	AAA for MIPv6	54
3.2.3	Home Agent load sharing	58
3.2.3.1	Architecture and overview	58
3.2.3.2	Selection parameters	59
3.2.3.3	HA selection	61
3.2.3.4	Message flow for HA load sharing	63
3.2.4	Interworking with IPv4 networks	64
3.2.4.1	Dual Stack Mobile IPv6 (DSMIP)	65
3.2.4.2	Movement detection	66
3.2.5	MIPv6 firewall traversal	69
3.2.6	Mobility optimizations	70
3.2.6.1	Overview of FMIPv6 Protocol	71
3.2.6.2	Description of the Reference Architecture	73
4.	Initial prototyping	75
4.1	Integrated Software architecture	75
4.1.1	Reference Architecture	75
4.1.2	Interface descriptions	76
4.1.2.1	Mobile Node	76
4.1.2.1.1	Ma (MIPL – OpenIkev2)	76
4.1.2.1.2	Mb (MIPL – Xsupplicant)	78
4.1.2.2	Home Agent	80
4.1.2.2.1	Hc (MIPL – OpenIkev2)	80
4.1.2.2.2	Hd (MIPL – OpenDiameter)	82
4.1.2.2.3	He	84
4.1.2.3	Home Agent-DB manager	84
4.1.2.3.1	Da	84
4.1.2.4	ASA / MSA	84
4.1.2.4.1	Aa (FreeRadius – SQL users database)	84
4.1.2.4.2	Ab	84
4.1.2.4.3	Ac	85

4.1.2.5	Network protocols	86
4.1.2.5.1	Pa IKEv2	86
4.1.2.5.2	Pb MIPv6.....	88
4.1.2.5.3	Pc SNMP	88
4.1.2.5.4	Pd MySQL.....	88
4.1.2.5.5	Pe EAP over RADIUS and EAP over 802.1X.....	88
4.1.2.5.6	Pf MIPv6 Diameter Application (Authorisation).....	88
4.1.2.5.7	Pg EAP Diameter Application (Authentication).....	89
4.1.2.5.8	Ph DNS.....	89
4.1.3	Software modules	90
4.1.3.1	OpenIKEv2 (MN)	90
4.1.3.2	MIPL (MN)	90
4.1.3.3	Xsupplicant (MN)	91
4.1.3.4	NAS (AP)	92
4.1.3.5	OpenIKEv2 / Diameter-EAP (HA)	92
4.1.3.6	MIPL (HA).....	93
4.1.3.7	NETSNMP (HA).....	93
4.1.3.8	OpenDiameter (HA).....	94
4.1.3.9	HA manager (HA-DB manager)	94
4.1.3.10	HA-DB (HA-DB manager)	94
4.1.3.11	FreeRADIUS (ASA)	96
4.1.3.12	HA Select (MSA).....	98
4.1.3.13	OpenDiameter (MSA).....	98
4.2	Additional ongoing developments.....	101
4.2.1	Firewall traversal based on NSIS	101
4.2.1.1	MN firewall traversal process	101
4.2.1.2	HA firewall traversal process	105
4.2.1.3	CN firewall traversal process	107
4.2.1.4	Implementation Overview.....	110
4.2.1.4.1	NTLP Implementation Overview.....	110
4.2.1.4.2	NAT/FW NSLP Implementation Overview.....	111
4.2.1.4.3	NSIS and NAT/FW NSLP for Mobile IPv6 firewall traversal - Implementation Overview	113
4.2.2	Architecture of FMIPv6	114
4.2.2.1	Network Topology of the Test-bed	115
4.2.2.2	High level of the FMIPv6 Implementation process	116
4.2.2.2.1	Functionalities of the ARs.....	118
4.2.2.2.2	The Functions of the MN	120
4.2.2.2.3	MN Process Flow	121
4.3	Development Platform	122
4.3.1	Linux distribution.....	122
4.3.2	MIPL Mobile IPv6 for Linux	122
4.3.3	IEEE 802.1X	123
4.3.4	AAA	123
4.3.5	SNMP.....	123
4.3.6	Database	123
4.4	Software development tools.....	124
4.4.1	Source code management.....	124
4.4.2	Bugtracking	127

5.	<i>Initial test-bed design</i>	128
5.1	Test-bed for the initial integration	128
5.2	Test-bed for the final integration	130
5.3	Test bed for NSIS / Mobile IPv6 firewall traversal	131
5.3.1	Hardware Requirements	132
5.3.2	Software Requirements	132
5.4	Test bed for Mobility Optimisations	133
5.4.1	AR Sub-System	134
5.4.1.1	Hardware and Software Requirements	134
5.4.2	MN System	134
5.4.2.1	Hardware Requirements	134
5.4.2.2	Software Requirements	134
6.	<i>Conclusion</i>	136
7.	<i>References</i>	138
	<i>Appendix A. Summary of high level mobility concepts</i>	141
	<i>Appendix B. Installing and using Subversion</i>	145

1. INTRODUCTION

There are two goals addressed by this document: firstly, it puts in place a specific application scenario and case study that we believe adequately supports the verification of the technical and business requirements of a Mobile IPv6 service environment and secondly, it identifies and plans the technological developments and specific functionalities from WP 1, 2, 3, 4 and 5 that will be included in the final public demonstration of the ENABLE project.

Section 2 provides the approach and specification of the ENABLE application scenario. It includes an overview of scenarios defined in deliverables from four other IST projects, IST Ambient Networks, IST Daidalos, IST ePerSpace and IST Simplicity, and also reviews and examines the Ambient Intelligence scenarios defined by the IST Advisory Board in 2001.

Section 2 moves on to give a more detailed breakdown of a search & rescue scene management scenario which is made up of six scenes. Only two of the six scenes are foreseen to be actually demonstrated in the final project trial and so these two scenes (Scene 3 & Scene 6) are described in further detail in this section 2. The main actor in these scenes is a search & rescue volunteer that comes in and goes through all the different access networks (IPv6-capable, IPv4-only and dual-stack) as the search & rescue is being carried out.

Section 3 provides an overview and assessment of the specific ENABLE technological components, and then identifies six functional components, EAP-based MIPv6 bootstrapping (WP1), AAA for MIPv6 bootstrapping (WP1) Interworking with IPv4 networks (WP2), MIPv6 firewall traversal (WP2), HA load sharing (WP3) and Fast Mobile IPv6 (FMIPv6) (WP4) as functionalities that will be used in the demonstration scenario.

Section 4 provides the development plans for the initial prototyping of these six functional components. The section also gives an explanation of how four of the six components (EAP-based MIPv6 bootstrapping, AAA for MIPv6 bootstrapping, Interworking with IPv4 networks and HA load sharing) are being integrated at an early stage and provides descriptions of the software architecture, interfaces and software modules to be created by the ENABLE partners. An initial description of additional on-going developments on NSIS and FMIPv6 is also given in this section.

Section 4 also provides a description of the common software development platform and development tools being utilised during the prototyping.

Finally Section 5 looks to the initial integration test-bed design which is being used to check the compatibility and the functionality of the software modules being created by ENABLE.

2. APPLICATION SCENARIO

This section describes a realistic and demonstrable scenario for the ENABLE project.

Firstly the approach and specification of the ENABLE application scenario is given which includes an overview of scenarios defined in deliverables from four other IST projects, IST Ambient Networks, IST Daidalos, IST ePerSpace and IST Simplicity, and also reviews and examines the Ambient Intelligence scenarios defined by the IST Advisory Board in 2001.

It then moves on to give a more detailed breakdown of a search & rescue scene management scenario which is made up of six scenes.

2.1 Approach

As stated from the ENABLE description of work, the purpose of this activity is to describe a specific application scenario of Mobile IPv6, which would adequately verify the technical and business requirements for the deployment of a Mobile IPv6 service environment.

This activity was started with a review of scenarios already defined from the “B3G System Architecture and Control” projects of the Directorate D - Network and Communication Technologies, Unit D1 - Communication Technologies IST programme.

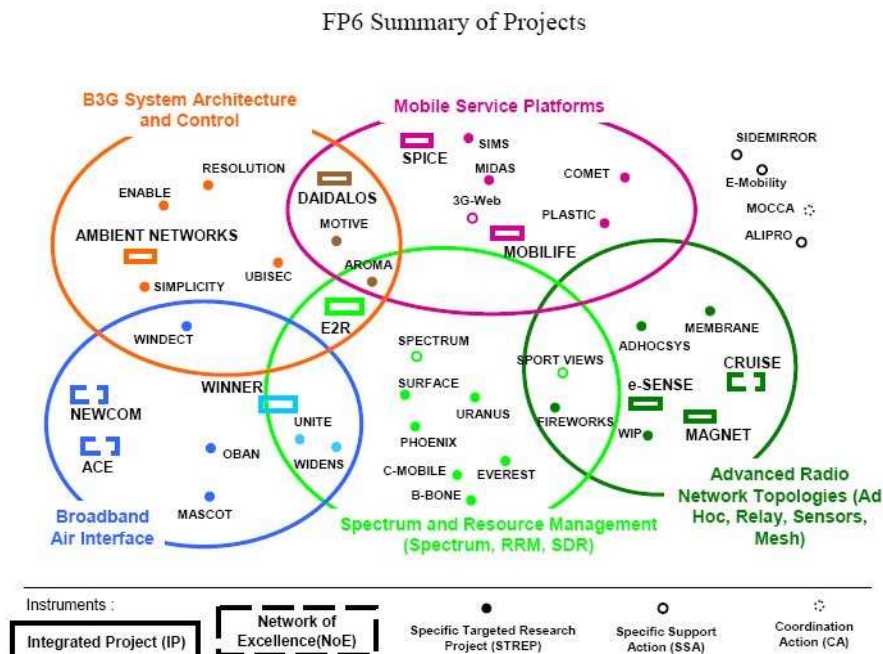


Figure 2-1: Communication Technologies IST FP6 projects

The IST projects with clear scenarios defined in deliverables include:

- IST Ambient Networks
 - D4.1, V1.0, Ambient Network Mobility Scenarios & Requirements, July 2004.
- IST Daidalos
 - D111 “Consolidated Scenario Description”, Feb. 2005.
- IST ePerSpace
 - D1.1 “Service Scenarios and Specifications”, Mar. 2004.
- IST Simplicity
 - D2101 "Use cases, requirements and business models", July 2004.

For good measure, this activity also reviewed and examined the Ambient Intelligence scenarios defined by the IST Advisory Board in 2001. The following section shall give overview of these scenarios, and an indication of how they influenced the ENABLE application scenario.

2.1.1 IST Ambient Networks

Deliverable D4-1 of Ambient Networks describes in detail the achievements of the initial project period, which was about high level mobility concepts, innovative scenarios from a mobility perspective and the definition of requirements for these different mobility perspectives. It shows how the mobility concepts have been derived from scenarios but also from other mobility related research initiatives.

There are 27 high level mobility concepts presented by Ambient Networks and the specific ones used as input to the mobility scenarios of ENABLE are summarised in the following table.

Table 2-1: High level mobility concepts

1 Traditional mobility	2 Session mobility	3 Independent flow mobility
4 Distributed flow (application) mobility	5 Managing multiple triggering sources for handovers	6 Multi-homed mobility
7 Cluster mobility	8 Inter-technology mobility (vertical handover)	9 Inter-address space mobility
10 Inter-trust domain mobility	11 Inter-provider domain mobility	12 Predictable mobility
13 Hierarchy considerations with mobility	14 Multicast mobility	15 Cluster organisation and management
16 Cluster reachability	17 Multihomed cluster mobility	18 Physical to virtual link handover
19 Optimization across different mobility events	20 Advanced location management	21 Name identifier concept for handover support
22 Mobility control space support for network connectivity	23 Managing middleboxes for control of routing streams	24 Contextual predictable mobility
25 Combined mobility management	26 Unresolved mobility	27 Handover redirection

Appendix A, contains further detailed descriptions of each of these mobility concepts.

There are 6 scenarios defined by Ambient Networks:

- Scenario #1 ‘Car scenario’
- Scenario #2 ‘An rock scenario’
- Scenario #3 ‘Business worker scenario’
- Scenario #4 ‘Developing countries’

- Scenario #5a ‘Gaming scenario’
- Scenario #5b ‘Health care scenario’
- Scenario #6 ‘Moving network scenario’

Each scenario has a template layout which covers the following headings:

- Environment and assumptions.
- Scenario story:
 - End User perspective.
 - Operator perspective.
 - Service Provider perspective.
 - Network perspective.
 - Application Developers perspective.
- Business relations.
- Roles and players.
- Value chain.
- Accounting / Compensation models.
- Trust/authorization relationships.
- Contractual responsibilities.
- User data privacy and/or integrity protection.

From conversations with partners in the AMI-Net project the most popular scenario was number #2, the RockStar Express, in which the scenario takes place somewhere in Europe during summer of 2015. It follows a rock band, Rusty Zigglers Travelling Hearts Club Band, while they tour Europe using a special rock train by which they travel between gigs.

Players in this scenario include the rock band promoter who is responsible for content production (i.e. the rock band) and content owner, the train operator as owner of infrastructure and network provider (i.e. the train itself as well as physical network access on board the trains and between

the train and its surroundings) and public network operators providing network access and extra features to its subscribers on board the train.

There will be multiple ambient network domains set up between different actors on board the trains as well as between actors on and off the train. There will also be temporary ambient network domains set up at the concerts when the rock group plays, to facilitate information sharing and content distribution between: the band and the audience, among the audience, as well as between the audience and some of their friends not being at the concert.

2.1.2 IST Daidalos

The Daidalos project as a whole adopted a methodology of scenario-based design, and in its Deliverable D111 “Consolidated Scenario Description” it describes in detail the continuous evolution of scenarios, the generation of requirements based on several stages of the scenario development process and the flowing design of the architecture.

In general the Daidalos scenarios describe the daily life in the near future from an end-user perspective and are structured into different scenes. They are user driven / user focused and demonstrate how a user will handle complex future technology and services easily and seamlessly. From this background the project defined two key scenarios which are:

- Automotive Mobility scenario.
- Mobile University scenario.

The Auto Mobility Scenario places itself in the daily life of Bart M. Watson. The scenario is structured in different scenes and shows Bart in his job and in his private environment. As an example of the type of scene

Scene 1: Bart is having morning coffee and getting dressed while watching his personalised newscast on screens around the house – his new service follows him into every room that he enters - when a call from his boss Hector is signalled.” “Bart walks to the living room, as this is where external video calls are received by default and accepts Hector’s call, who is urging him to come to the office prior to the briefing.

Scene 2: He jumps up and enters his car. The vehicle automatically activates voice call. Also, the TV program he was watching is transferred but on hold during the voice call. He can resume watching it once he has finished the call – though in sound-only driver-mode. His boss informs him, that he needs to pick up customer Rosalyn Royce at the airport.

The Mobile University Scenario describes the daily life of Dani and Maria on campus. Pervasive location based services will help new students on campus to find their classmates as well as meeting rooms on campus. For example; Reservation services for restaurants will work closely together and seamlessly integrate their own scheduling service. Less manual work will be necessary to set up a date and a reservation at the restaurant or cafeteria.

The D111 also gives an overview of how this methodology of scenario-based design helps to create questions on issues like: How could future look like? Campus, Cars, people, buildings, streets and other objects are closely integrated and will become a connected world.

As with Ambient networks in Daidalos each scenario has a template layout which covers the following headings

- General Assumptions.
- Short description of the scenes that make up the scenario.
- Business Models.
- Realisation of the Scene in each WP.
- Used Technology & Services.
- Set of Use Cases for each step in each scene.

2.1.3 IST ePerSpace

Deliverable D1.1 of ePerSpace “Service Scenarios and Specifications” mainly serves the purpose of providing an outline number of scenarios that will give input for a consistency check of architectural ideas to the functional architecture and an extensive list of service proposals consisting of high-level service definitions to trials.

From this list the project picked a subset of services to implement as trials, or to use as a basis for implementing a similar service having the same basic ePerSpace functionality. It is important to note that ePerSpace did not intend to implement the scenarios in a 1-1 fashion (i.e. to exactly recreate one or more of the scenarios in trials).

This document contains five extensive scenarios. A scenario resembles a short story told with simple words. Each scenario follows the same model: a narrative section that describes the utilisation case and a description of services at two levels. At the higher level, products that may consist of several different services, and at the lower level each of these services. The lower level

will describe services that may be included in several different products. Finally, we set out to illustrate the characteristics of each scenario with regard to the ePerSpace project.

An overview example of these scenarios include:

- A day in the van Epers Family.
 - The scenario describes a day in the life of the van Epers family. The van Epers family are ePerSpace users and can make use of services provided by the system both at home and elsewhere.
- In the car.
 - This scenario describes a day in the life of a businessman travelling to a conference.
- At Home.
 - The scenario takes into consideration a home platform compatible with the distribution of audiovisual services and the integration of several types of home devices.
- Hospital stay.
 - Lucy von Epers is a lady in her forties, and she is scheduled for an operation in the City hospital. The scenario focuses on Lucy's needs, the problems she needs to solve and tasks she needs to accomplish during her stay in the hospital and away from home.
- "Children in the weekend".
 - The scenario describes some scenes representing children through an ordinary weekend. In fact Mr. and Mrs. Servin have three children who have many hobbies and participate in several activities according to their age.

In addition to the five scenarios twenty services were identified which included:

- Doorkeeper.
- Content Management.
- Traffic Info gathering.
- In-Car.

- Traffic management.
- Presence information.
- Wireless key provision.
- Accessing private media outside the home.
- Remote control of home appliances.
- Payment via Bluetooth.
- Home personalisation.
- ePerSpace enabled content on demand.
- Video conference.
- Service and device auto discovery.
- Identity check.
- eRoom.
- Medical journal administration.
- ePerSpace VPN.
- Service continuity at home.
- Home devices usage.

The scenario template layout covered the following headings

- Summary.
- Premises.
- Participants.
- The story.
- Service definitions.
- The role of “ePerSpace” in this scenario.

2.1.4 IST Simplicity

Deliverable D2101 "Use cases, requirements and business models" of IST Simplicity describes and analyses user scenarios, use cases and business models for the Simplicity System and uses these to derive preliminary system requirements. The system requirements and functionalities outlined in this document were derived from twenty-seven (27) user scenarios which were used as the basis for four "reference scenarios"; this included:

- Mobile worker Scenario.
 - How a mobile worker could benefit from Simplicity when moving between different locations and different terminals.
- Ubiquitous Media Streaming.
 - How a private user could use Simplicity to buy and/or access multimedia material, from home or in other locations.
- Buy and Use a 'Self Learning' Simplicity Device.
 - How Simplicity can learn from a user's behaviour and adapt to the user's preferences and needs.
- Tour Guide Scenario.
 - How a tourist might use Simplicity while exploring an unknown city.

All functionalities identified in the original 27 scenarios are present in at least one of the "reference scenarios". Once fully defined, the "reference scenarios" were formalised as UML use case and sequence diagrams. A preliminary investigation defined tentative "economic requirements" for the system. Collectively these requirements provided input to the Simplicity System architecture design.

The scenario template layout covered the following headings

- Narrative Description of the Scenario.
 - With a set of Use Cases.
- Detailed Analysis of Scenario.
 - In which the specific and generic functionalities required are named and mapped across the architecture.

2.1.5 IST Advisory Board

In 2001 ISTAG compiled a report containing the “Scenarios for Ambient Intelligence in 2010”. In it the concept of Ambient Intelligence (AmI) is provided, where the emphasis is on greater user-friendliness, more efficient services support, user-empowerment, and support for human interactions. It shows a world in which people are surrounded by intelligent intuitive interfaces that are embedded in all kinds of objects and an environment that is capable of recognising and responding to the presence of different individuals in a seamless, unobtrusive and often invisible way.

The scenarios attempt to offer provocative glimpses of the future that can be realised. Each scenario has a script that is used to work out the key developments in technologies, society, economy, and markets necessary to arrive at the scenario. The central feature of the scenarios is that people (as people not just ‘users’, ‘consumers’ or ‘employees’) are at the forefront of the Information Society. This vision of people benefiting from services and applications whilst supported by new technologies in the background and intelligent user interfaces was essential to the ISTAG notion of Ambient Intelligence.

The four scenarios were constructed to provide ‘food for thought’ and include:

- Scenario 1 – Maria Road Warrior
 - Maria travels to another country to give a presentation to clients.
- Scenario 2 – Dimitiros – and the digital me
 - Dimitiros, is in the office but has a device that acts as an avatar representation of himself. Embedded in his cloths, there is a device that acts as a learning device, communications tool, with processing and decision making functionality.
- Scenario 3 – Carmen – Traffic, Sustainability & Commerce
 - Carmen is travelling to work, and requires some information on a possible carpooling opportunity. While at coffee a shopping list for the nights dinner is made. While going to work on passing some shops a suggestion for wine is made. While travelling home, pollution is noted and the car in which she is travelling has its max speed reset.
- Scenario 4 – Annette and Solomon in the ambient for social learning
 - In a study group, looking at environmental studies, the “ambient” helps new members of the group join and interact with the already resident members.

The scenario template layout covered the following headings

- Background.
- Full script.
- Socio-Political issues.
- Economics issues.
- Technology issues.
- Where is ambient intelligence in the scenario.

2.1.6 IST ENABLE

Not to forget, there were some hints for the application scenarios in the ENABLE description of work, which included:

- Location Based Services (LBS).
- Search and Rescue scene management (emergency applications).
- VoIP Application with HA failover & middlebox traversal.

These were all mentioned as possibilities, and it was from these starting points and subsequent discussions the WP6 team put together the first application/demonstration scenario on “*Search and Rescue scene management*”.

The Search and Rescue scene management scenario was chosen as it allowed for enough flexibility to comply with the basic mobility scenario requirements, such as the range of access technologies, intra-subnet/inter-subnet and intra-technology/inter-technology handover and intra-domain/inter-domain mobility, as set forth in the initial ENABLE architecture [ENABLE-D1.1].

Table 2-2 is taken from [ENABLE-D1.1] and is a summary of the different mobility scenarios discussed in section 3.4 of [ENABLE-D1.1], along with their relevance to the ENABLE project.

Table 2-2: Summary of Mobility Scenarios

		Intra-domain	Inter-domain
Intra - Technology	Wired - Wired	-----	-----
	WLAN - WLAN	Yes	Yes
	WMAN - WMAN	Yes	Yes
	Cellular - Cellular	Out-of-scope ¹	Out-of-scope ²
Inter - Technology	Wired - WLAN	Yes (High) ²	-----
	Wired - WMAN	-----	Yes
	Wired - Cellular	-----	Yes
	WLAN - WLAN	Yes	Yes
	WLAN - WMAN	Yes (High)	Yes (High)
	WLAN - Cellular	Yes (High)	Yes (High)
	WMAN - Cellular	Yes (High)	Yes (High)
	Cellular - Cellular	FFS ³	FFS ³

In most cases, on the rescue scene there is limited connectivity. For this reason an assumption made is that some local volunteers can provide connectivity using their private resources (e.g. WLAN, ADSL, etc.).

This connectivity, being opportunistic, is provided with no network planning, which means that mobility events that are normally unlikely might happen in this scenario. For example there might be overlapping (and independent) WLAN/WMAN coverage with no authentication required and multiple protocols supported (IPv4-only, IPv4-only with NATs, IPv6-only, dual-stack).

These factors made the Search and Rescue scene management scenario rich with application opportunities, and gave the possibility to incorporate the Location Based Services (LBS) and VoIP Application into the scenario.

¹ Handled at lower layer by 3GPP protocols.

² "High" indicates a mobility scenario that is expected to be very likely.

³ For Further Study. Depends on the architecture that 3GPP will choose for mobility between GPRS/UMTS and LTE.

The consortium's motivation for choosing this methodology came from the examination of existing scenarios from other projects in the IST programme and other (partner specific) national programmes. Only one instance of fully using UML to specify their scenario was found, and hence it was decided that the ENABLE project would follow a more traditional route with a text based description of the scenario.

Given all the template scenario layouts from the different projects, it was felt that the ENABLE project could best use the format as shown in IST Ambient Networks, with it's headings of environment and assumptions, scenario story, different perspectives and business relations; we took these as the basis for the ENABLE scenarios and came to the general headings of

- Scene Story.
- Scene Challenge.
- Supported services.
- Mobility Issues.
- User experience.

2.2 Search & Rescue Scene Management Scenario Description

The overall search and rescue scenario description identified for the ENABLE project is aimed at conveying the innovations of the ENABLE project and to work towards visually demonstrating efficient and operational mobility in large heterogeneous IP networks.

The identification and selection of scenarios are a very beneficial way to guide the development of conceptual models and technologies. They can also become a very important part of the integration and test-bed deployment process, by successfully supporting and determining pre-integration activities and steps and to positively aid the merging of innovative developments and technologies within the ENABLE project. The scenario chosen for the demonstration of the ENABLE technologies will be based on a Search and Rescue scene where multiple networks, actors and devices will interoperate with each other. A search and rescue scenario is ideal for the test-bed as there is an abundance of technologies interoperating with each other. In addition, due to the emergency aspect of the scenario the quality of the connection between the actors and the ASPs must be both reliable and seamless due to the risk involved of a dropped data exchange.

The scene layout for the Search and Rescue (SAR) scene is depicted in Figure 2-2.

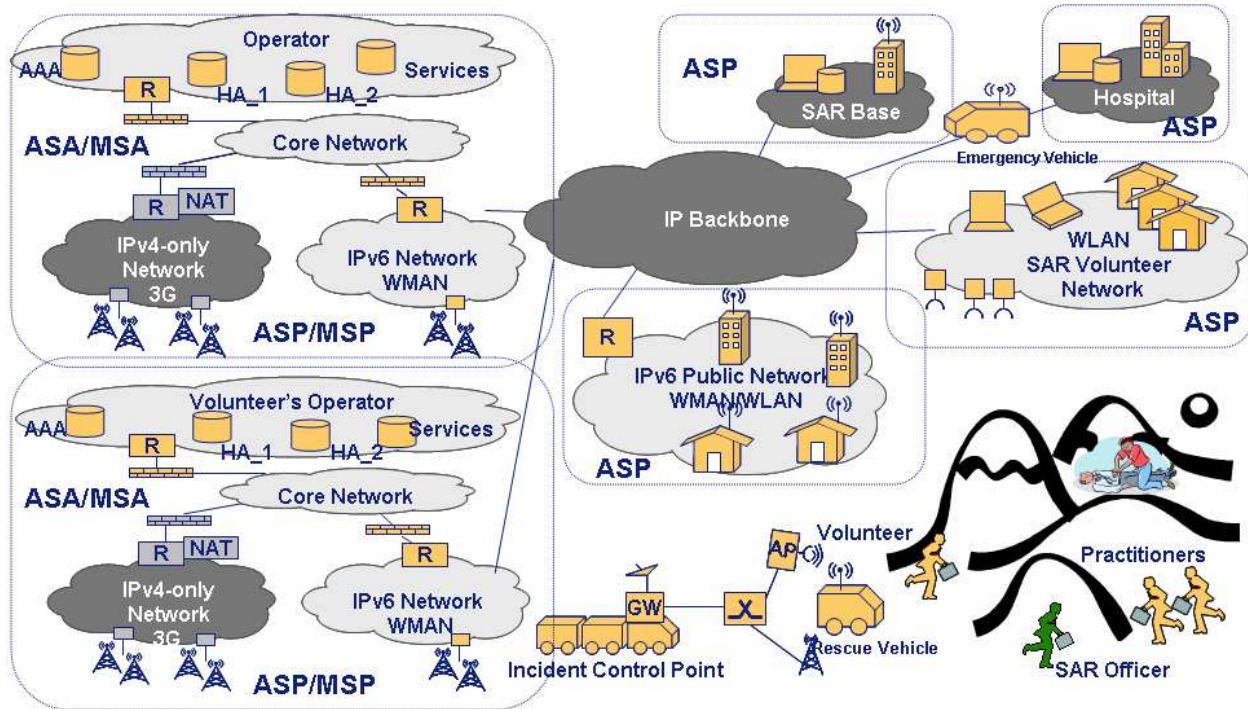


Figure 2-2: Search and Rescue Scene


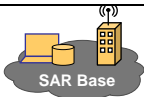


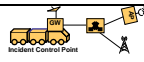
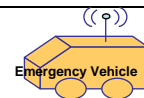


The scenario takes into account the various access technologies ranging from Local Area Networks, to Wireless Metropolitan Area Networks to cellular networks. The end user experience is a main focus of the scenario also, allowing them to benefit from services independently of the underlying access infrastructure. An example of some of these services would be to make available services subscribed by the user with the home provider anywhere and with the highest performance level. Another example would be the seamless movement across homogeneous and heterogeneous access technologies, with little or no disruption caused to ongoing applications (e.g. VoIP or video conferencing).



Other issues such as performance of mobility management procedures, network capability discovery, security and service control are other technological innovative areas being considered and taken into account as well.

This scenario consists of both fixed locations and mobile assets, and a combination of networks that serve these places and assets. A detailed description of these can be seen in Table 2-3 below.

In this scene there are fixed locations and mobile assets, as described in the following.

Table 2-3: Search and Rescue Scene locations and mobile assets

Fixed Locations		
	Rural Location	The choice of a rural location is essential as it is assumed most cities and suburban locations should expect to already have enough coverage by Mobile Operators. We even consider a mountain location in which there is limited to no Private Network Operator connectivity possible in the search area.
	Search and Rescue Base	This is the physical home base for the professional SAR officers, and their equipment.
	Search and Rescue Volunteer Location(s)	The SAR team is extended by volunteer personnel, who are called upon for large scale operations.
	Hospital	Hospital location that the Ambulance will arrive at when the search mission has concluded.
Mobile Assets		
	Incident Control Point	An incident control point (ICP) is the on-site command post for the search and rescue operation.
	SAR Ambulance	Specialised vehicle for treating on-site victims and transporting them to hospital.
	SAR Vehicle	A SAR vehicle is used for transporting assets (equipment, SAR Officers and volunteers) to and around the operational site. It is also deployed for the search functions.
	SAR Officer	Full-time professional search and rescue practitioners that are stationed permanently at the SAR Base.

027002	ENABLE	D6.1:Report on case studies and initial prototypes
	SAR Volunteer	Part-time search and rescue practitioners, who essentially work from home, and who come to a rescue location when called upon from the SAR Base.
	Practitioners	A search made up of a team leader, navigator, radio officer, medic and volunteers deployed to cover specific search areas.

Throughout the SAR scenario there are multiple operators, ASPs (Access Service Providers) and mobility access points providing networking services. These are summarised in the Table 2-4 below but have a more detailed explanation in the business entity perspective, section 2.3.

Table 2-4: Networks that serve the search & rescue locations and assets

Mobile Operator (ASP/MSP) / (MSA/ASA)	This is the private network operator that provides connectivity and mobility services to subscribed customers. In this scenario the customer is the SAR organisation.
Incident Control Point (ASP)	The ICP is the onsite command post that contains equipment to allow rescuers and practitioners access to the network. The ICP unit will act as a bridge onto the operator's network.
SAR Volunteer Network (ASP)	A closed community network, which is used by the part-time search and rescue personnel to connect with each other, and the SAR Base.
Public Network (ASP)	Open community network offered by members of the public.
SAR Base Network (ASP)	A closed enterprise network, which is used by the full-time professional search and rescue practitioners that are stationed permanently at the SAR Base.

The ASA is the provider that authorises the end users access to the ASP. The end user will initially subscribe to a 'home' ASP which is also the end users ASA. As the MN migrates around the search and rescue location area or urban environment, it may attach to multiple operators of network access provision.

Within the search and rescue scenario, the SAR base may be considered an ASP, where the vehicles bootstraps in the SAR base.

Given the starting point as shown in Figure 2-2, the scenario is further developed through 6 individual scenes:

- Scene 1 Search and Rescue is initiated.
- Scene 2 Assets (People & vehicles) are deployed.
- Scene 3 Not enough assets on site, volunteers called in.
- Scene 4 Areas of location not covered by Private Network Operator.
- Scene 5 Rescue victim found, special emergency unit vehicle deployed.
- Scene 6 Ambulance transports the victim from rescue scene to hospital.

These scenes shall be described in further detail in this section of the document. The main two scenes that will be selected from the overall Search and Rescue scenario to be demonstrated and mapped to the physical nodes in the test-bed infrastructure will include:

- Scene 3 Not enough assets on site, volunteers called in.
- Scene 6 Ambulance transports the victim from rescue scene to hospital.

2.2.1 Environment & Assumptions

Before describing each scene in detail there are some environmental issues and assumption which must be clarified:

- The location for the search and rescue is a Rural Area.

The location of the SAR area will be contained in the rural area of a town. As rapid response to emergencies is a critical requirement for saving lives, disaster vehicles will need reliable connectivity relating location and instructional information back the base and control units. This area is ideal as it will contain multiple WLAN/WMAN connectivity access points, a large search area for the units to interact with and access to multiple services providers including the SAR base, Hospital, Users Home Networks, Volunteer network and ICP/Ambulance vehicles.

- The Search and Rescue (SAR) team is made up of full-time & part-time members.

When a search is initiated the first team to be called is a full-time team of SAR practitioners. This will simulate multiple devices coming into an area and authenticating and using bootstrap procedures to log their devices onto the SAR network. However, in a real life scenario the search team might be overstretched and most organisations have a backup team of part-time members who can be called later in the scenario. This will simulate more devices bootstrapping either on IPv4/IPv6 networks and different authentication mechanisms.

- The ICP Node bootstraps at the SAR Base.

The ICP Node will initially be located at the SAR volunteer base. When a rescue is initiated the ICP node will enter the bootstrap phase at the SARs base. It will authenticate itself against the SAR network via WLAN. As the ICP moves from the SAR base it will use its three interfaces (WLAN, 3G and WMAN) to maintain connectivity back to the SAR base.

In all these cases the ICP receives its HoA (Home Address) from the Mobility Service Provider (MSP).

- The ICP Node has three network interfaces (WLAN, 3G, WMAN).

The ICP node is a network bridge with three separate connections to allow it communicate with the SARS base and other ISPs in the rescue scenario. There will be intelligent software on the ICP bridge which will detect and enable the preferred available back haul connection. This connection will be bridged to the locally deployed Ethernet and WLAN connections which will be the main network access point for the volunteers and other emergency assets that are currently on site.

- The ICP sets up a network bridge once deployed on-site.

The ICP mode shall act both as a bridge for the clients when deployed across the SAR scene and also to relay information back to the SARS base. The ICP will mainly communicate with the hospital staff and volunteers (offsite) through its backhaul connection.

- The SAR network coverage service contract allows access for IPv4 and IPv6 traffic over all interfaces.

The network coverage contract will allow connectivity from multiple ASP's. Therefore a number of the scenarios are inherent within a rescue operation including IPv4 to IPv6 transitioning, bootstrapping an IPv6 network if on an IPv4 island, multiple authentication

mechanisms, and the case where many units bootstrap on a network (e.g. HA load sharing maybe required).

- Government has a contractual agreement with a major Mobile Operator for network coverage service. The Mobile Operator is presumed to host both the ASA for authorising network access and also the MSA for authorising mobility for the SAR officers.
- Due to the terrain of the urban location and technologies involved it is likely that a SAR asset may lose direct connectivity with the ICP node, Therefore SAR assets may use available ASP's in the location such as the SAR Volunteer Network or another Public Network.

2.2.2 Scene 1: Search and Rescue is initiated

When the search and rescue is initiated, the mobile Incident Control Point (ICP) is deployed to the general location of the search and rescue. The mobile ICP command vehicle boots its MN at the SAR base before the scenario is started. It enters the bootstrap phase and begins authentication procedures with the operator it has a contract with, through the connection that the SAR base is providing. As the ICP unit leaves the SAR base it may handover from one network to another using various access technologies such as 3G, WLAN and WMAN. When the ICP unit arrives on site it will power up its WMAN/WLAN antennas to allow actors and other SAR vehicles bridge their connection to a backhaul provider that the mobile command vehicles is currently attached to.

2.2.2.1 Scene Challenge

1. Authentication of the ICP Node is an identified challenge in scene 1. Within the ENABLE project the EAP authentication framework is selected as the preferred authentication protocol. This was deemed to be most suitable as it provides multiple authentication methods over multiple data links. The ICP node will be responsible for providing the necessary credentials required by the EAP authentication method.
2. MIPv6 Bootstrapping of the ICP Node is another challenge that exists within scene 1. The keying material required in order to bootstrap the security associations for mobility service can be derived from the EAP keying framework. Bootstrapping of the ICP node will take place at the SAR base.
3. IPv6/IPv4 Interworking & NAT Traversal challenges will be encountered in this scene. Most firewalls and NATs that exist today have been specifically designed for the IPv4 networks, but these firewalls are also foreseen to be vital for protection against unwanted

traffic in IPv6 networks. As the mobile node in the search and rescue scenario roams between IPv6 and IPv4 networks, the firewall issue is a challenge that the ENABLE project will assess and implement.

4. Once on site, transition to the ICP WLAN connection (with WMAN/WLAN Bridge) is performed. The following technologies are the ones that WMAN are usually based on: IEEE 802.16REVd, IEEE 802.16e and pre-standard solutions.

2.2.2.2 Supported services

For scene 1, certain supported services can be identified as follows:

- Initially IP connections to applications such as database/fileserver. External directories interact with AAA backend servers, these external directories are used for maintaining accounting data and users profile data. The ASP may also have a database including information regarding the user session; this is useful where re-authentication of the user is required as it prevents contacting the home domain every time for this purpose (see the general bootstrapping and authorization framework being developed by WP4).
- The availability of services such as video, VoIP, Email, IRC, and HTTP.

2.2.2.3 Mobility Issues

- The support of multimedia applications should be available in multi-access networks and this would include real time applications and information sharing. This is a very important mobility issue for the search and rescue, as the need for real time applications greatly benefits the end users by having real time information available to them and hence making the whole search and rescue mission more efficient and effective. Example of real time applications would include monitoring and control applications, including high bit-rate streaming services. Real time applications would be very useful to the ICP as it has been deployed and is on route to the site.
- MIPv6 bootstrapping and IPv4 Interworking mobility issues also exist in this scene 1. If a Mobile IPv6 node enters a IPv4-only network, MIPv6 sessions will no longer be continued as the IPv4-only network prevents the MIPv6 protocol from working correctly. ENABLE must support the transporting of Mobile IPv6 messages across IPv4-only networks. This issue is taken into account in scene 1 as the ICP is deployed and moves from one network to another (e.g. IPv4 Public ASP to IPv4 Private ASP to Dual stack ASP to IPv6 ASP) until it reaches its on-site destination.

2.2.2.4 User experience

In scene 1 the user experience incorporates the user view from the people working directly from the SAR base and also it takes into account the end users connected directly into the ICP unit on at the rescue site.

- SAR base can maintain/update/monitor the ICP and all its equipment without connection loss.
- The user may have to authenticate to the ASA/MSA through the ICP unit possibly using one of the following forms of credentials; Username and Password, Smart Card, SIM/USIM and/or Public and Private key authentication.

2.2.3 Scene 2: Assets (People & vehicles) are deployed

Scene 2 of the search and rescue scenario deals with the deployment of Assets (People & vehicles) crossing the specified search location. The scene addresses many mobility issues ranging from the bootstrapping of trackers on site, authentication of trackers and assets and connectivity to the local operator through the bridges of the mobile ICP connection.

2.2.3.1 Scene Challenge

Scene 2 includes the following crucial challenges:

1. Bridge connection is available from ICP mobile node.
2. The Assets (in this scene defined as the People and Vehicles) begin deployment across the search area.
3. The assets will need to be able to maintain communication, in order to do this they may connect directly to WLAN/WMAN/3G or via ICP.
4. MIPv6 Bootstrapping, will also have to be taken into consideration as the devices will have to be initialised for connectivity and to allow access to certain application services that they need in order to successfully carry out the search & rescue. Quality of service and the use of different QoS classes that need to be authorised, is another part of this identified challenge within scene 2.

2.2.3.2 Supported services

As this scene deals with the deployment of assets across the search area, quite a few supported services would need to be considered within this scene in order for communication to be maintained successfully throughout the search and rescue timeframe. Here is a list of the main supported services identified for this scene:

- Low Bandwidth Data (positional information/instructions).
- High Bandwidth Video.
- VoIP to ICP: Voice over IP call to the Incident Control Point. This could be from a SAR rescue vehicle or tracker during the search and rescue operation.
- SMTP [RFC821]: simple mail transfer protocol is a protocol for sending email messages between servers, in this scene following the deployment of the assets, this could be used for communication purposes.
- HTTP [RFC1945]: protocol used by World Wide Web, could be used in scenario by assets to link to certain web page for up to date information.

2.2.3.3 Mobility Issues

With the deployment of the assets in this scene 2, the mobility surrounding such an operation in the search and rescue field is bound to have mobility issues that need to be taken into account. These would mainly consist of the following:

- MIPv6 Bootstrapping as previously mentioned in section 2.2.3.1.
- Many devices appear on the network and need to authenticate to the Home MSP/ASP which will provide access to the Mobility access through the MSA and also authorise against the ASA.

2.2.3.4 User experience

As trackers, SAR vehicles help in the search. Their need for direction and constant up to date reports puts a lot of priority on the level of communication required in such an operation. The mobility of the assets would require the following conditions to be taken into account

- Session continuity (WLAN/WMAN/3G handovers), as the trackers and SAR vehicles roam from one area to another. User experience shall be at its highest when the user

maintains connectivity to the high speed WMAN connection that the mobile ICP unit provides.

- The user may have to authenticate to the ASA/MSA through the ICP unit possibly using one of the following forms of credentials; Username and Password, Smart Card, SIM/USIM and/or Public and Private key authentication. This will relate to the scenario in Scene 3 where John arrives on site and connects his MN directly into the mobile ICP unit.

2.2.4 Scene 3: Not enough assets on site, volunteers called in

Once all vehicles and people have been deployed around the search location area, further assets may be required onsite if the location of the search area grows, e.g. there are not enough SAR practitioners to cover an area. In this scenario extra volunteers may be called in to the site to aid in the searching. In addition to people, extra equipment may also be brought to the site. These additional personal called into the search will be one of the main actors played in the rescue scenario as they may move between IPv6, IPv4-only, and dual stacked networks.

2.2.4.1 Case Study Detail

- 3a) John is an example of a volunteer, called by the search teams whenever there is a lack of resources at the search site. John has a MN with three network interfaces including WLAN/WMAN, 3G and LAN.
- 3b) John can access any of these networks over both IPv4 and IPv6 networks. When John is leaving the house he receives a video call.
- 3c) John decided that, as the search is quite close to his house, he will walk part of the way. It is presumed that when John leaves his house he has blanket WLAN coverage from his house to the town park over both IPv4 and IPv6 networks. John will lose the WLAN coverage when he approaches the town park which he must go through to reach the search site. However, he will then handover onto a 3G connection. John gets collected by a rescue vehicle and is then transported to the site. On route to the SAR site is continuing his video call through his MN which is fed from the SAR site.
- 3d) John reaches the site and enters the ICP. He connects his MN to the Mobile ICP with LAN cable (e.g. to download high resolution maps of the area). John on his way to the ICP area may pre authenticate himself with the mobile ICP unit before he arrives. This requires sending the correct credentials that allow John access information provided by

the SAR base. However, if John is already receiving a video stream from the SAR base he may be already authenticated.

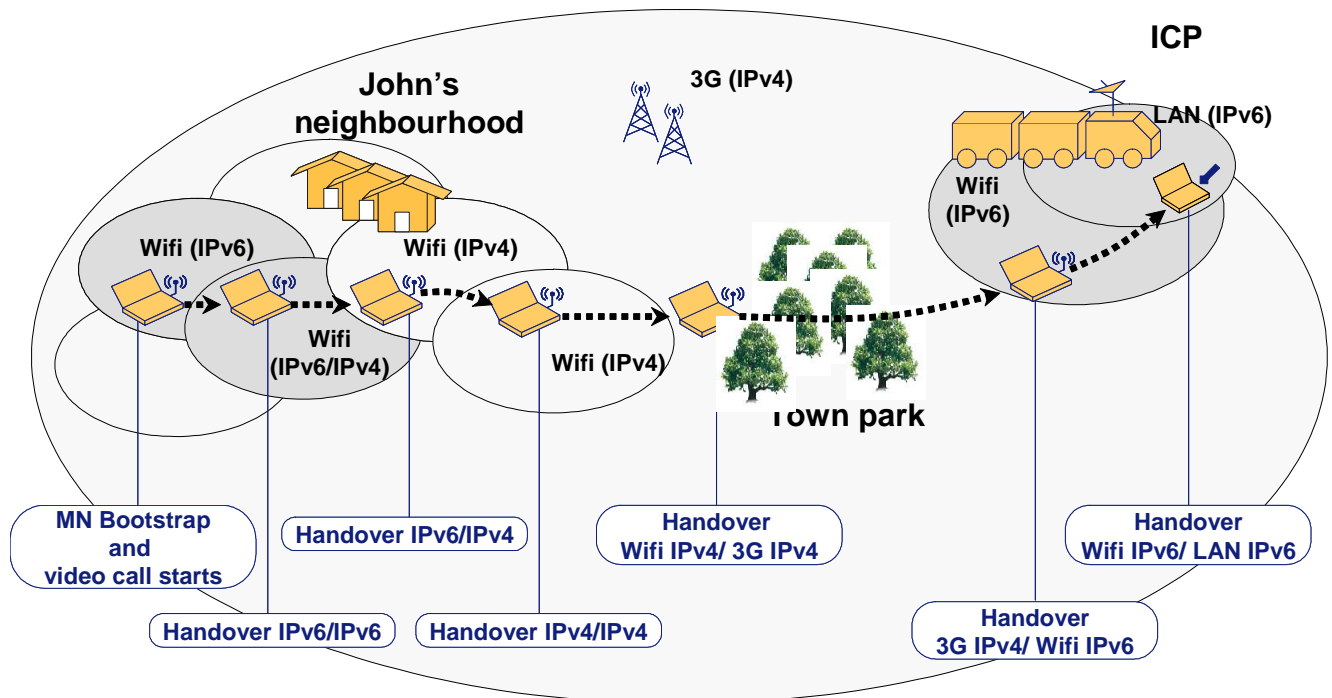


Figure 2-3: Scene 3 Case Study Detail

2.2.4.2 Environmental and General Assumptions

The main assumptions for this scene are the following:

- John has a connected device that is capable of receiving a video or audio stream via one of the three access mechanisms available to him. These include WLAN/WMAN, a LAN connection or through his local mobile operator through 3G.
- John's device will automatically connect and authenticate to these access mediums automatically without intervention. These mediums will also have both IPv4 and IPv6 connectivity.
- John's neighbourhood has blanket WLAN coverage until he reaches the town park where he must use his 3G connection.

2.2.4.3 Scene Challenges

1. As more volunteers are called in they may have different home providers (i.e. ASA/MSA). These people may also have different devices that will need to operate

within the system. As the users come closer to the Mobile ICP they may connect in via WLAN/WMAN and then subsequently via a LAN connection.

2. Direct connection to the ICP will be made via WLAN/WMAN connection. The ICP unit will be considered to be a dumb wireless bridge which will enable clients to connect. As WMAN technologies can be used here John may be able to pick up the ICP unit from greater distance and be able to avail of High Bandwidth Video that may not be possible over 3G.
3. As more users visit the site their devices will have to bootstrap and may need to authenticate against their respective ASA/MSA. Actors that are connected to WLAN/WMAN connection on the Mobile ICP will have their connections bridged. Therefore they will generate their own IPv6 address (Care-of Address, CoA) directly from the operator that the Mobile ICP unit is connected to. However, these actors must also have access credentials to be able to authenticate themselves successfully with the operator.
4. As the John is moving from one network that is IPv6 to another network that maybe IPv4-only an IPv4-IPv6 interworking solution must be provided as John may be receiving a video call from the site.

2.2.4.4 Supported Services

Supported services in this scenario can be identified as the following:

- Low Bandwidth Data (positioning information and instructions can be transmitted to the search teams).
- High Bandwidth Video can be used to relay information back to the ICP via the WLAN/WMAN connection.
- Other standard protocols can also be used such as SMTP and HTTP.

2.2.4.5 Mobility Issues

- Bootstrapping is one of the main issues that occurs with this scene as there are many new actors that may enter the search location after initial deployment.
- Authentication mechanisms must support multiple users when they arrive on the site to authenticate against their home provider.

- As John is receiving a video call when he leaves the house session connection survivability is important as the SAR may be updating John on his instructions or current situation at the site.
- Although John has blanket coverage around his living area there may be some WLAN/WMAN Operators that only operate on IPv4; therefore John will have to use IPv4 interworking methods as designed by ENABLE to overcome this problem.

2.2.4.6 User Experience

- Through this scene session continuity is important as John will be receiving a video call from the SAR scene. John will also need to seamlessly connect to the Mobile ICP unit through authentication mechanisms, possibly through the use of a username and password. Although not essential in the case of scene 3, maintaining an active connection through John's journey from house to SAR is important, along with the important issue of IPv6/IPv4 interoperation as John moves from WMAN/WLAN (IPv4 and IPv6) and 3G Networks.

2.2.4.7 Mapping of Scene 3 to the Test-Bed

Since scene 3 in the rescue scenario has now been described in detail it is now possible to map the individual actions of the actor (played by John) onto the test bed that ENABLE will deploy for demonstration purposes. The following table will detail the actions that John will take from the journey from his house to the SAR base. It will then map the test-bed components for the ENABLE test-bed to these specific actions outlining which components are used in which step.

Table 2-5 Link Scene 3 to Test-bed Components

Scene 3: John leaves his house starts a video call, migrates between networks and is collected by a vehicle	Mapping Link to Test-bed requirements
3a - before leaving his house John switches on his MN	MN, Home Agent, ASP-AAA, ASA-AAA. MSA-AAA, MSP-AAA. NAS, DNS,
3b – Video call is initiated to John from the SAR base	Home IP network contacted via HA by ICP. Video call initiated.

3c – John moves from one network to another.	<p>Access points (such as FN AP2 – FN AP3) with IPv6, IPv4 or dual stack subnets, ASP-AAA, MSA/ASA-AAA</p> <p>In this particular scene we can demonstrate all the handover types (IPv6 -> IPv6, IPv6 -> IPv4, IPv4 -> IPv4, IPv4 -> IPv6)</p>
3d – John arrives at ICP and plugs in using LAN cable	LAN Wired Connection / IPv6 Auto-configuration.

2.2.5 Scene 4: Areas of location not covered by Network Operator

2.2.5.1 Scene Challenge

- The ICP is presumed to be using WiMAX (802.16) technology with a greater coverage than standard wireless technologies. However, in some cases especially in urban areas WiMAX can be limited to a specific coverage area. Therefore, some units may not be able to connect to the Mobile ICP to relay information back. Units in this scenario will connect to a third party provider via 3G to relay information back to the ICP.
- Once a device is no longer able to receive a signal from the ICP the device will need to handover onto a new network. Challenges in this scenario will involve many sub challenges including:
 - Seamless Mobility handover when coverage to ICP is degraded. User would have to switch over onto 3G network.
 - High Bandwidth available services may be affected with switchover to lower capacity network.
- Client may handover onto IPv4-only enabled network operator.

2.2.5.2 Supported services

- As the client would be connecting to a Low Bandwidth Data provider located in the SAR scene the most important type of information that would need to be translated is positioning and instructions to the mobile units.
- Throughout the entire search process units may be in constant communication with the ICP through a VoIP call. Seamless handover is important here as the units may be receiving instructions over VoIP.
- HTTP services can be deployed over the users connection.

2.2.5.3 Mobility Issues

- As more and more devices appear on the network users may have to authenticate themselves to their ASA/MSA.
- Users that do not get covered by the ICP or move out of range may enter IPv4 only ASPs that provide 3G service. Therefore the MN may have to bootstrap in an IPv4-only environment by using IPv4/IPv6 internetworking mechanisms.

2.2.5.4 User experience

- Devices will authenticate with a username or password, EAP based authentication, Smart Card, SIM/USIM and/or Public and Private key authentication to the service.
- The most important user experience aspect in this part of the scenario is the handover from one network to another. As a user may move from the ICP to a IPv4-only mobile network they may need to reach a IPv6 network to authenticate themselves against their ASA/MSA and also to start the bootstrap process. This procedure will have to be done quite rapidly as long handover times may have an effect the user if they are in the middle of a voice or location information exchange.

2.2.6 Scene 5: Rescue victim found, Ambulance deployed

The main focus of the scene 5 is a situation where the rescue victim has been located by the search party. Immediately after they have been found, an Ambulance is dispatched to the location of the user. The position is related to the nodes involved in the rescue, the ICP, SAR base and hospital.

2.2.6.1 Scene Challenge

- Once the location of the person has been found, the location will be relayed to the ICP, Ambulance, SAR Base, and Hospital.
- In a similar situation to scene 6 the Ambulance/ICP will have a direct peer to peer communication line between them.
- Multiple handover scenarios exist in this challenge. The Ambulance can move from any number of situations including but not limited to:
 - From the WMAN/WLAN connection provided by the ICP to another WLAN connection provided by another operator.
 - WMAN/WLAN Handover onto both private and public networks.
 - WMAN/WLAN to 3G.
 - From 3G/WMAN/WLAN IPv4-only networks to networks that only support IPv6 connectivity.

2.2.6.2 Supported services

- Low and High bandwidth services can be supported in this scenario depending on the connection that the Ambulance has at any particular time. In an emergency situation the Ambulance on site could relay high bandwidth video of the current rescue scenarios back to the ICP and SAR over WLAN technologies.
- In low bandwidth situations (where the Ambulance unit has only a 3G connection available) information such as location and instructions can be transmitted over this connection.
- VoIP calls can be made from the Ambulance to the ICP/SAR base.

2.2.6.3 Mobility Issues

- As the Ambulance will be moving across multiple networks, supporting IPv4 (with both private and public addresses) and/or IPv6, and different access technologies including WLAN and 3G mobility will be an important issue for this scenario. As the Ambulance is moving across the network it may be relaying large amounts of information back to the

base which must be kept updated. ENABLE can solve these problems through IPv6/IPv4 interworking.

2.2.6.4 User experience

- The Ambulance may choose to authenticate itself against its home network as it moves across the multiple networks on the path to the search area.
- As the Ambulance is moving across multiple networks seamless mobility is a very important part of this scene. As the unit could be receiving instructions from the SAR/ICP or pre route to hospital this should be a constant exchange with any interruption.

2.2.7 Scene 6: Ambulance transports the victim from rescue scene to hospital

The main content of Scene 6 descriptive text involves an ambulance picking up the rescue victim and transporting him in the Ambulance to the hospital location. This scene includes specific mobility issues, challenges and domain issues, taking into account IPv4 interworking and IPv6 Middlebox traversal to mention but some. Environmental Issues and assumptions must also be taken into consideration when completing a more detailed work flow of this scene in order to incorporate and deal with all aspects of the scene. A detailed overview of scene 6 is described below demonstrating the user experience and supported services.

2.2.7.1 Case Study Detail

- 6a) Before leaving scene John switches on his MN, bootstrapping of John's mobile node will take place.
- 6b) John starts a video call to alert the Hospital and communicate to the medical team the patient's condition.
- 6c) The victim is put into the Ambulance, John also enters the Ambulance and will accompany the victim to the hospital.
- 6d) The Ambulance arrives at the Hospital. John gets out of the Ambulance and enters the Hospital with the victim.
- 6e) The video call finishes when care of the patient is handed over to the medical team at the Hospital.

Following the completion of the detailed scene 6, this will help the process of deriving business modelling and architecture work for this scene. The scene 6 deployment view in Figure 2-4,

provides a visual representation of the main actions within this scene completed by the actors involved.

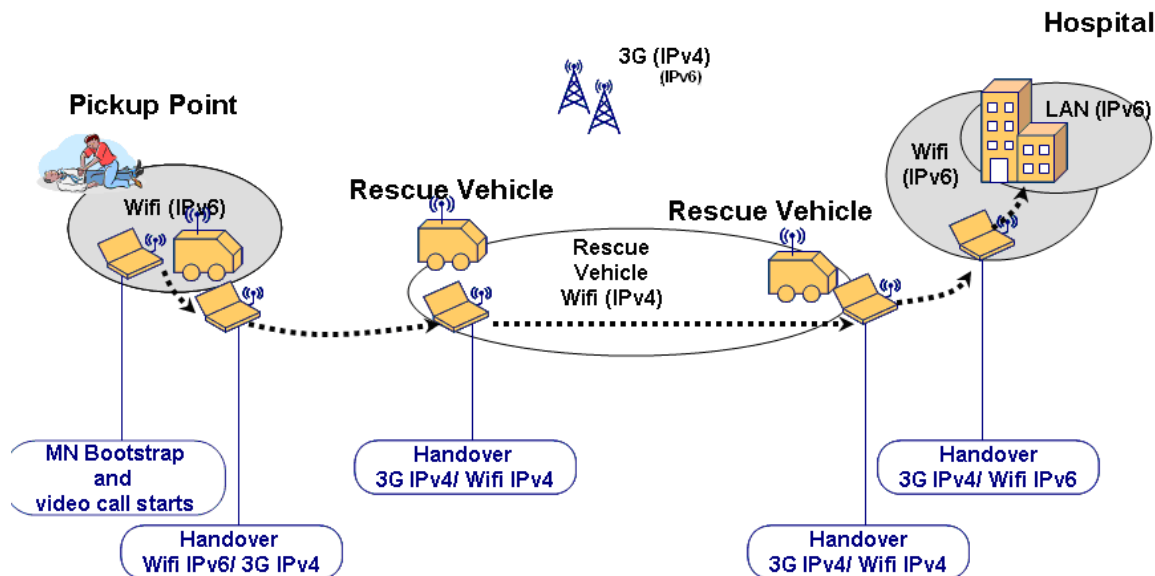


Figure 2-4: Scene 6 scenario deployment View

2.2.7.2 Environmental and General Assumptions

The main general assumptions taken into consideration for scene 6 is that, John's mobile terminal has three main network interfaces.

- WLAN/WMAN(IPv4 and IPv6).
- 3G (IPv4).
- LAN (IPv6-enabled LAN, available at Hospital).

A mobile terminal in the case of scene 6 would most likely be a PDA or cellular phone with multiple interfaces, but it could also be represented by a laptop. With the availability of these three network interfaces, it is stated in his contract that John is permitted network access over both IPv6 and IPv4 access networks, such as:

- IPv4 with public addresses.
- IPv4 with private addresses and NAT/PAT.
- IPv6 with plain end-to-end connectivity.

Another assumption taken into account for scene 6 is that any bootstrapping for the ICP will be completed at the SAR base. This ICP node will also have three network interfaces available to it which will include WLAN, 3G and WMAN. Once the ICP has been deployed on site it will be initially set up as a WMAN bridge, which is mainly IEEE802.16 with coverage measured in kilometre and connectivity without direct line of sight.

2.2.7.3 Scene Challenges

Specific challenges foreseen within this scene 6, occur during the transportation session when the Ambulance is returning to the Hospital location. The following is a list of these identified challenges.

1. Relaying of the geographical position of the Ambulance as it is transporting the victim back to the Incident Control Point (ICP) is one of the main challenges of this scene. This challenge needs to take into account the kind of mobility, whether it might come under global, localised or Intra-link mobility scene. As the mobile Ambulance node roams between different networks, all the ENABLE elements should interact efficiently, providing the end user with a seamless experience. The mobile node, in this case the Ambulance has to register its current location or point of attachment to the internet with a Home Agent which is provided by an MSP serving the Ambulance mobile node. The provision of several HAs leading to load sharing should also be considered here.
2. Direct peer to peer communication between Ambulance/ICP and the on site team will be another challenge coming from this scene 6.
3. Ambulance transitions between IPv6 public WLAN, 3G and IPv6 Network WLAN. In scene 6 the Ambulance will travel between a WLAN/WMAN IPv6 network to 3G IPv4 network to a LAN IPv6 network at its hospital destination. For these transitions to be successfully completed the Mobile Node, in this case the Ambulance, will be subjected to a handover as it moves between two access networks. This handover must be successful in order for all running applications (e.g. video call in scene 6) to be continued during the journey back to the hospital.
4. Network shutdown, is another challenge that must also be taken into account when considering this scene. Once the victim has been transferred to the hospital and then is handed over to the medical team, there is no longer any need for session continuity and communication links to remain open as the rescue has been completed. Network closedown will consider all tasks required to close down the S&R service and operation, taking into consideration the closure of any network or session activities between the ICP, Ambulance, SAR, SAR base, trackers and hospital.

2.2.7.4 Supported Services

Supported services within this scene can be identified and narrowed down to 2 specific supported services.

- Low bandwidth data (position information and instructions transmission).
- On route back to the Hospital, John is in constant contact through a video call to the hospital, this could also be communicated to the ICP to keep them up to date on ongoing activities, until the Ambulance reaches the hospital and the mission has been accomplished.

2.2.7.5 Mobility Issues

- Session continuity and IPv4 interworking also with the possibility of switching to 3G whilst the Ambulance is on route from the pick up point to the hospital are the main mobility issues encountered during scene 6.
- Session continuity refers to ongoing applications that do not have to be restarted, this occurs in scene 6 following John initialising a VoIP or a Video call to the hospital and medical team before he enters the Ambulance to begin the journey to the Hospital. As the seamless handover is completed on route this allows for session continuity when the mobile node is moving.
- IPv4 interworking in ENABLE will allow both IPv4 and IPv6 traffic to be forwarded from home network to a dual stacked (DS) MN to allow communication with IPv6-capable or IPv4-only Correspondent Nodes (CNs) without using MIPv4.

2.2.7.6 User experience

Throughout this scene 6, the end user experience is greatly improved through session continuity. The video call to the medical team at the hospital awaiting the arrival of the Ambulance to the hospital provides a very beneficial, useful and seamless service. Important medical information relayed on the video call, would be very important in a life threatening situation where the victim needs medical assistance as soon as possible. Session continuity across multi-access networks of the video call would greatly increase the victim's chances of survival, as there would be a direct communications/video link between the person attending the victim and the qualified medical team. This is one example of how this would drastically improve the end users experience in this scene 6.

IPv6 Middlebox traversal: today IPv6 firewalls do not support Mobile IPv6, but they will need to. When a user moves into another visited network (e.g. from WLAN(IPv6) to 3G (IPv4) in scene 6) there maybe firewalls along the communication lines between the MN and their home network, and this may have an impact on the Mobile IPv6 data.

IPv4 interworking: during the initial IPv6 deployment phase it is foreseen that mobile nodes are going to be in IPv4-only visited networks. IPv4 networking allows IPv4 and IPv6 traffic to be forwarded from the home network to Dual Stacked MN. Mapping of Scene 6 to Test-bed.

2.2.7.7 Mapping of Scene 6 to the Test-Bed

Following the scene selection, it is then possible to complete a scene analysis and mapping activity of scene 6 in order to work towards creating a scene 6 test-bed deployment view, producing a direct link between the scenario identification process and the test-bed deployment process. With the aid of these two processes combined it simplifies how a distribution of the ENABLE components for scene 6 can be accomplished. The mapping of scene 6 to the ENABLE test-bed will demonstrate how the scene can be mapped onto a test-bed deployment view containing the necessary machines (and therefore work towards defining the type and number of equipment required and their role), and finally advance the process towards defining a configuration of hard- and software to make the scenarios work. Table 2 6 details the link between scene 6 and test-bed, providing a mapping between the scene and the test-bed requirements.

Table 2-6 Scene 6 Link To Test-bed Components

Scene 6: Ambulance picks up victim and is returning to hospital location	Mapping Link to Test-bed requirements
6a - before leaving the scene John switches on his MN	MN bootstrapping: Mobile Node, MASA AAA server DHCP , Home Agent , DNS Server , EAP
6b - he starts a video call to alert the Hospital and communicate to the medical team the patient's condition	Home Network, Home Agent Mobile Node, Video Call Application,
6c - he enters the Ambulance	WLAN/WMAN AP Video Call Application still active

6d - he arrives at the Hospital. John gets out of the Ambulance and enters the Hospital	3G Connectivity
6e - the video call finishes when care of the patient is handed over to the medical team at the Hospital	WLAN/WMAN AP, ASP AAA server MSA/ASA-AAA

2.3 Business Entity Perspective

As defined from Section 2.3 of D1.1 [ENABLE-D1.1] there are specialised business entities which have a part to play in the Search and Rescue scenario, these consist of the following:

- Access Service Authoriser (ASA).
- Access Service Provider (ASP).
- Mobility Service Provider (MSP).
- Mobility Service Authoriser (MSA).

Further details on the Business Entity Perspective of each of these identified business entities will be further detailed in this section.

2.3.1 Access Service Authoriser

The ASA is the provider that authorises the end users access to the ASP. In the rescue scenario this is considered to be the mobile operator. SAR personnel will authenticate against their ASA based on their credentials. The mobile operator has been chosen as the ASA/MSA as it is felt that they would already have or would be willing to provide the necessary infrastructure for the service.

2.3.2 Access Service Provider

The function of the ASP is to provide required network services and connectivity to the end users requesting it. When the user roams he will enter different ASP networks. While the user is roaming in the network of an ASP, the ASA will communicate (the policies and rules to be enforced on traffic generated by the user) to the ASP. The ASP may also take the form of a

community based wireless network where a private operator might not be currently providing services in the area. This is the case in many rural areas.

Within the Search & Rescue Scene Management Scenario the ASP is represented by an Operator that owns an IPv6 WMAN, the Public Network WMAN/WLAN and the WLAN SAR volunteer network. With the SAR base representing the ASP, the volunteer will communicate with his home ASP/MSP through the connection provided by the ICP vehicle.

2.3.3 Mobility Service Provider

The delivering of an IP mobility service to an end user is completed by the Mobility Service Provider, where it provides home agents or other required global mobility anchors when needed. In the search and rescue field in the scenario, the MSP service is provided by the mobile operator.

2.3.4 Mobility Service Authoriser

The MSA is responsible for authorising the mobility service to the end user. In the rescue scenario the MSA is considered to be located in the mobile operators network.

There exists two operators. One of which (Operator) has the SAR professionals as subscribers, the other (Volunteers Operator) has the Volunteers as subscribers.

3. TECHNOLOGICAL DEVELOPMENTS

3.1 Introduction

The purpose of this section is to provide an overview and assessment of the specific ENABLE technological components, and from there provide further detail on the six functional components, EAP-based MIPv6 bootstrapping, AAA for MIPv6 bootstrapping, Interworking with IPv4 networks, MIPv6 firewall traversal, HA load sharing and Fast Mobile IPv6 (FMIPv6) which will be used in the demonstration scenario.

3.2 Functional Components of ENABLE

Each partner provided details of the different technological components they were going to develop during the ENABLE project (e.g. Middlebox, Load Sharing, Service Authorisation).

An assessment was made of the specific technological components, and out of this assessment, components were identified as ones that could be used in the demonstration scenario. The output of this activity was a list of software developments per partner, which is shown in the Table 3-1 below.

Table 3-1: Software developments per partner

		Operational Mobile IPv6 architecture	Implemented by
From WP1	Bootstrapping and control of mobility	Split Scenario. Diameter and EAP	TI/UMU
	Bootstrapping and control of mobility	IKEv2 + EAP	UMU
	Bootstrapping and control of mobility service	EAP-based bootstrapping	TI
From WP2	Middlebox traversal	NSIS solution	UGOE
	Interworking with IPv4 network	DSMIP including moving detection algorithm	TI
	Interworking with IPv4 network	MIPv6 extensions for DHCPv6 on access router	CONSULINTEL
From WP3	Load sharing	HA decision rule based on weighted	IABG
	Reliability aspects of mobility	VRRPv6 Extensions for homeagent-reliability	TSSG, Brunel
From WP4	Mobility optimizations	FMIPv6	Huawei, Brunel

An overview description of each software development design and message flows for the various sub-cases, and open issues are given below, with the exception of the reliability aspects of mobility, with the VRRPv6 extensions for homeagent-reliability, as this will be developments that are looked at during the second year of the project.

3.2.1 EAP-based MIPv6 bootstrapping

Some EAP methods (e.g. [PEAPv2], [EAP-AKA]) are able to convey generic information items along with authentication data. This flexibility allows to configure bootstrapping parameters during the MN's authentication for network access. Upon the successful completion of the authentication phase Configuration-TLVs (Section 3.2.1.1) are exchanged to deliver the bootstrapping information. Actually, these TLVs are a mere container: LCP messages and logic [PPP] are used to configure service specific information. A new network control protocol (MIPv6CP) is defined for the purpose of configuring MIPv6. This approach is somewhat similar to [MIP4-PPP] which defines a new configuration option for IPCP. New services can be configured during this phase as soon as new Configuration Protocols (CP) are defined. Services can be bootstrapped in sequence or, more efficiently, more than one Configuration TLV is inserted in one packet.

Figure 3-1 shows the message flow of occurring for the bootstrapping of MIPv6.

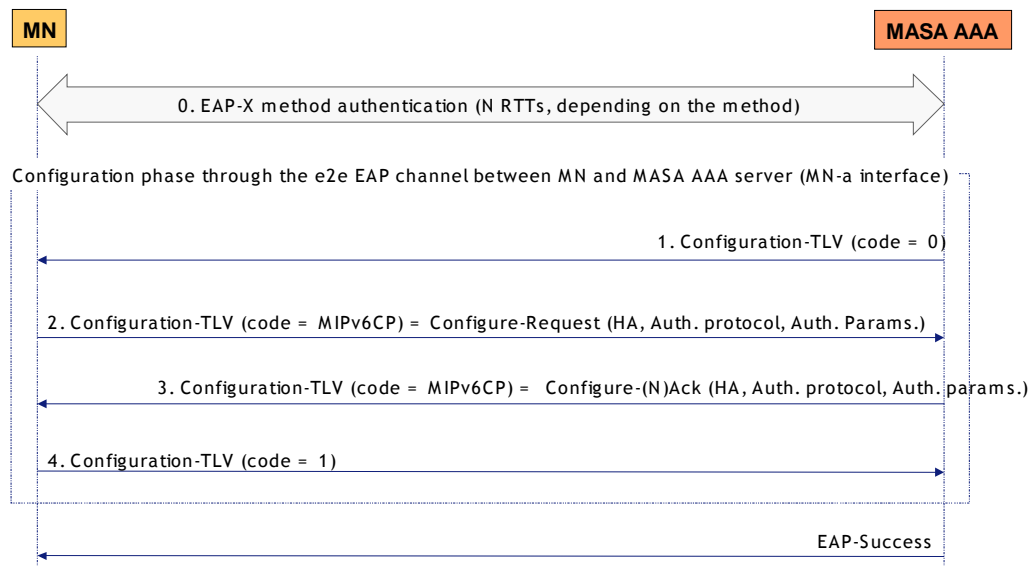


Figure 3-1: MIPv6CP message flow

- **Step 0.** EAP authentication phase completes successfully but the EAP-Success is delayed after the configuration phase (steps 1-4).
- **Step 1.** A Configuration-TLV (code = 0, no value data) is sent by the MASA to open the configuration phase.
- **Step 2.** The Configuration-TLV (code = MIPv6CP) sent by the MN contains a LCP Configuration-Request and optionally the MN may insert its preferences as the HA Agent assignment (e.g. request for a specific HA identified by its IPv6 address, request for a

local HA) and the authentication protocol to be used for establishing the MN-HA SA (i.e. IKEv2 or RFC4285).

- **Step 3.**
 - If the MASA acknowledges all MN configuration options, it sends a Configuration-TLV (code = MIP6vCP) carrying a LCP Configuration-Ack.
 - Otherwise, the MASA sends a LCP Configuration-Nak inserting the acceptable values for not acknowledged configuration options.
- **Step 4.** A Configuration-TLV (code = 1, no value data) is sent by the MN to close the configuration phase.

If the terminal does not recognize the Configuration TLV, it must send a NAK TLV and consequently the AAA endpoint must immediately close this phase sending an EAP Success message.

3.2.1.1 Configuration TLV

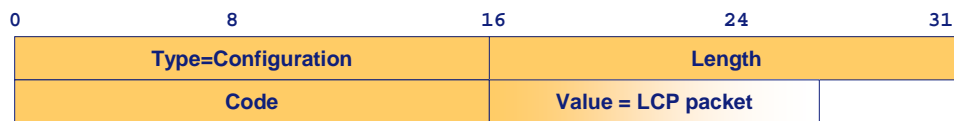


Figure 3-2: Configuration-TLV

- **Type.**
 - TBD (Configuration). Value 200 has been chosen for experimental usage.
- **Length.**
 - The length of the Value field in octets.
- **Code.**
 - 0 = Open.
 - 1 = Close.
 - TBD = MIPv6CP. Value 252 has been chosen for experimental usage.
- **Value.**

- LCP packets defined in [PPP].
- One LCP packet per TLV.

3.2.1.2 LCP packets

The format and the usage of these messages are defined in [PPP], therefore this section is meant as a quick reference for the reader and does not pretend to be exhaustive.

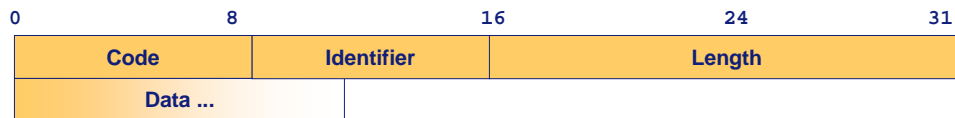


Figure 3-3: LCP packet

- **Code.** The Code field is one octet and identifies the kind of packet. When a packet is received with an unknown Code field, a Code-Reject packet is transmitted.
 - 1 Configure-Request.
 - 2 Configure-Ack.
 - 3 Configure-Nak.
 - 4 Configure-Reject.
 - 5 Terminate-Request.
 - 6 Terminate-Ack.
 - 7 Code-Reject.
- **Identifier.** The Identifier field is one octet, and aids in matching requests and replies. When a packet is received with an invalid Identifier field, the packet is silently discarded without affecting the automaton.
- **Length.** The Length field is two octets and indicates the length of the value, including the Code, Identifier, Length and Data fields. Octets outside the range of the Length field are treated as padding and are ignored on reception. When a packet is received with an invalid Length field, the packet is silently discarded without affecting the automaton.

3.2.1.3 Configure-Request

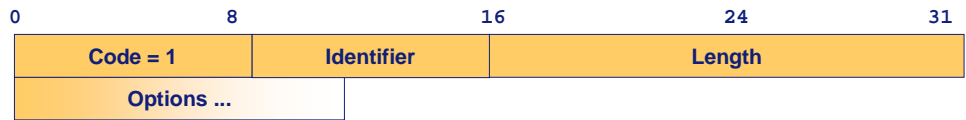


Figure 3-4: Configure-Request

Upon reception of a Configure-Request, an appropriate reply MUST be transmitted. The Identifier field must be changed whenever the contents of the Options field changes, and whenever a valid reply has been received for a previous request. For retransmissions, the Identifier may remain unchanged.

The options field is variable in length, and contains the list of zero or more Configuration Options that the sender desires to negotiate. All Configuration Options are always negotiated simultaneously.

3.2.1.4 Configure-Ack

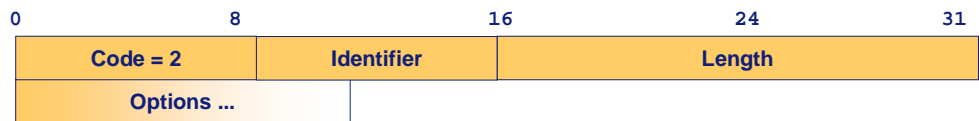


Figure 3-5: Configure-Ack

If every Configuration Option received in a Configure-Request is recognizable and all values are acceptable, then a Configure-Ack must be transmitted. The acknowledged Configuration Options must not be reordered or modified in any way. On reception of a Configure-Ack, the Identifier field must match that of the last transmitted Configure-Request.

3.2.1.5 Configure-Nak

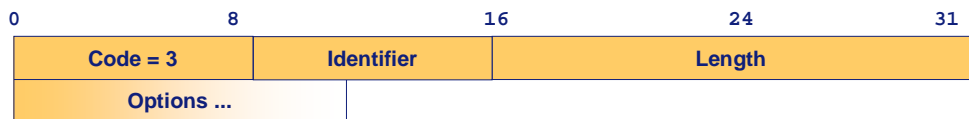


Figure 3-6: Configure-Nak

If every instance of the received Configuration Options is recognizable, but some values are not acceptable, then the implementation MUST transmit a Configure-Nak. The Options field is filled with only the unacceptable Configuration Options from the Configure-Request. All acceptable

Configuration Options are filtered out of the Configure-Nak, but otherwise the Configuration Options from the Configure-Request must not be reordered. Each Configuration Option must be modified to a value acceptable to the Configure-Nak sender

3.2.1.6 MIPv6 Configuration Options

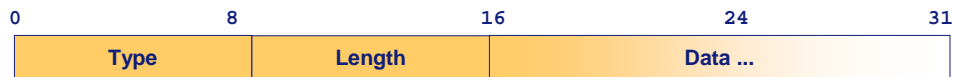


Figure 3-7: Configuration-Options

A default is specified for each option. If a Configuration Option is not included in a Configure-Request packet, the default value⁴ for that Configuration Option is assumed. The Length field is one octet, and indicates the length of this Configuration Option including the Type, Length and Data fields. If a negotiable Configuration Option is received in a Configure-Request, but with an invalid or unrecognized Length, a Configure-Nak should be transmitted which includes the desired Configuration Option with an appropriate Length and Data.

The Type field is one octet, and indicates the type of Configuration Option. These types are purposely defined for MIPv6vCP:

- 1 - HA location.
 - Length = 1.
 - Values
 - 1 – Local.
 - 2 – Remote.
 - 3 – Unspecified (default).
- 2 - HA IPv6 address.
 - Length = 0 or 16.
 - Default = NULL (i.e. length=0).
- 3 - HA FQDN.

⁴ On the following list the default values of each Configuration Option are specified.

- Length = variable.
- Default = NULL (i.e. length=0).
- If both, HA IPv6 address and HA FQDN options, are missing or set to NULL in the Configuration-Ack(Nck) reply by the MSA then MN must use DHCPv6 to retrieve the HA information.
- 4 - Authentication Protocol.
 - Length = 1.
 - Values
 - 1 - EAP-IKEv2 (default).
 - 2 - RFC4285.
 - 3 - PSK-IKEv2.
- 5 - PSK-IKEv2 length (2 octets).
 - Length = 2.
 - Default = 128.
- 6 - MIPv6 keys lifetime.
 - Length = 4.
 - The value is expressed in seconds.
 - Default = lifetime of MIPv6-USRK.

3.2.2 AAA for MIPv6

When bootstrapping MIPv6 two differentiated scenarios may be presented, each one having special requirements that must be kept in mind.

- Integrated scenario: the MSA and the ASA are the same entity.
- Split scenario: the MSA and the ASA are separated entities.

In the integrated scenario, the MSA + ASA (MASA) controls the entire bootstrapping procedure, so it can provide mobility configuration parameters piggybacked on the network authentication

process. In this scenario there are two different possibilities to provide the HA address to the MN: the MASA could deliver the HA directly within the EAP tunnel (if the MN access network allows it) or it could be delivered via DHCPv6.

In the split scenario, the ASA doesn't know anything about mobility so the MN must discover the HA address using DNS queries.

Once the HA address is known by the MN, the rest of the bootstrapping steps are the same in both scenarios. First, the MN needs to authenticate with the HA, obtain a HoA and establish the needed security associations (SAs) to protect the mobility signalling and authorise the mobility service with the MSA. All these actions are performed through:

1. IKEv2 and Diameter EAP Application.
2. MIPv6 signalling and a new defined Mobile IPv6 Authorization Application.

The following Figure 3-8 shows the message flow regarding the first part of the process:

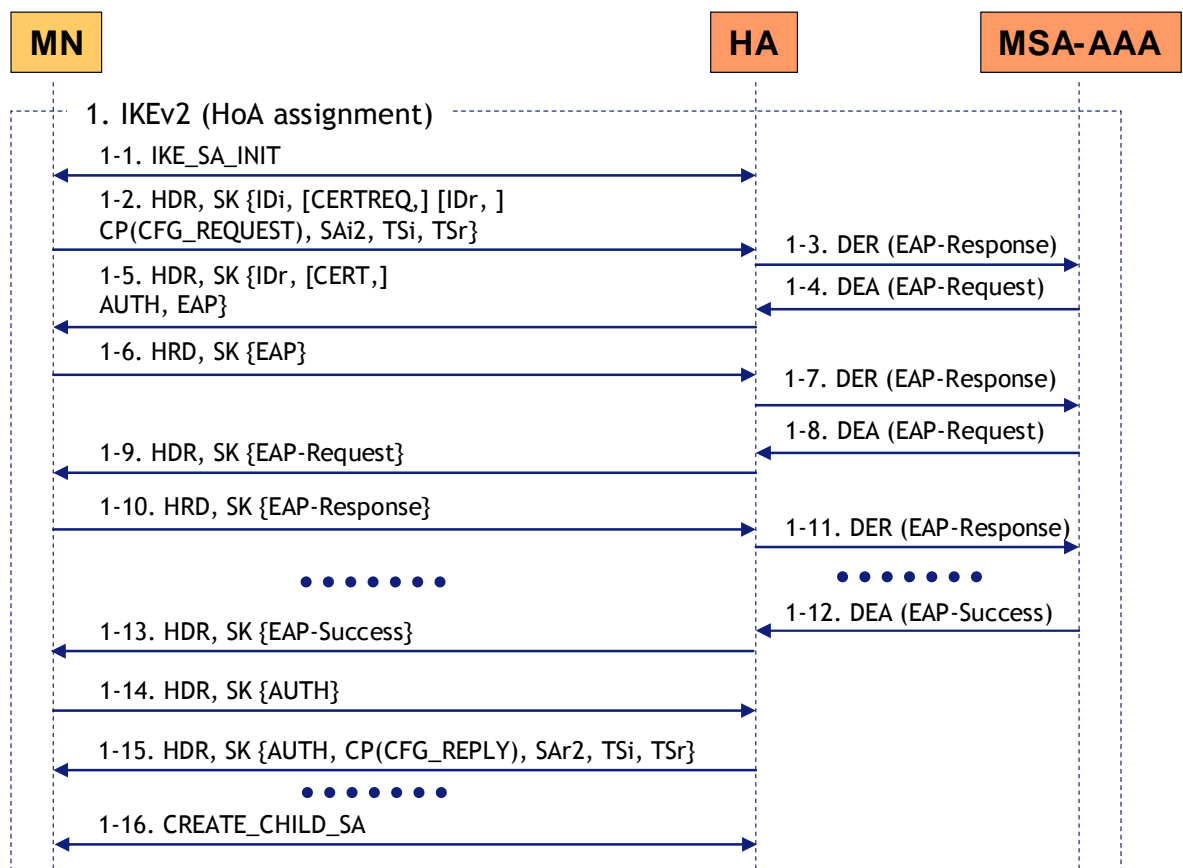


Figure 3-8: IKEv2 and Diameter message flow

- **Step 1-1.** The IKE_SA_INIT exchange is performed by the MN and the HA in order to generate the IKE SA cryptographic material.

- **Step 1-2.** The MN starts an IKE_AUTH exchange, indicating its identity (IDi payload) and the desire to use EAP authentication (omitting the AUTH payload), requesting the assignment of a HoA (including the CP payload) and negotiating the creation of the first IPsec SA (SAi2, TSi and TSr payloads).
- **Step 1-3.** The HA detects that the MN wants to use EAP authentication and starts a Diameter EAP session with the MSA-AAA. The HA sends to the MSA-AAA a new Diameter-EAP-Request message containing an EAP response packet with the MN identity. All the EAP packets are transported over Diameter by using EAP-Payload AVPs.
- **Step 1-4.** The MSA-AAA creates an EAP request packet and sends it to the HA into a new Diameter-EAP-Response.
- **Step 1-5.** The HA responds to the MN with an IKE_AUTH message containing its identity (IDr payload), information to authenticate the HA (AUTH payload and optionally a CERT payload) and the MSA-AAA EAP request packet. All the EAP packets are transported over IKEv2 by using EAP payloads.
- **Step 1-6 and 1-10.** The MN creates an EAP response packet and sends it to the HA using an EAP Payload inside a new IKE_AUTH message.
- **Step 1-7 and 1-11.** The HA forwards the EAP response packet to the MSA-AAA using an EAP-Payload AVP into a new Diameter-EAP-Request message.
- **Step 1-8 and 1-12.** The MSA-AAA creates a new EAP request packet or an EAP success packet (depending on the roundtrips needed by the EAP method) and sends it to the HA using a new Diameter-EAP-Response message. If the authentication has been successful and the EAP method creates a shared key as a side effect, that key is also transported to the HA using an EAP-Master-Session-Key AVP.
- **Step 1-9 and 1-13.** The HA forwards the EAP request (or success) packet to the MN using an EAP payload in a new IKE_AUTH message. If the EAP method needs more roundtrips, repeat from Step 6.
- **Step 1-14.** The MN sends a new IKE_AUTH message to the HA, including information to prove its identity (AUTH payload).
- **Step 1-15.** The HA completes the IKE_AUTH exchange by probing its identity again (AUTH payload), providing the HoA to the MN (CP payload), and completing the first IPsec SA negotiation (SAr2, TSi and TSr payloads).

- **Step 1-16.** The MN and the HA may perform several CREATE_CHILD_SA exchanges in order to create additional IPsec SAs needed to protect MIPv6 traffic.

The MN can then send a Binding Update to the HA as shown in Figure 3-9. The Diameter EAP exchange that takes place during the IKE SA establishment only authenticates the MN, so the mobility service must be explicitly authorized by the MSA upon reception of this first BU (a new experimental authorization application is defined for this purpose). In order to perform the authorization of the mobility service, the HA sends a Diameter MIP6-Authorization-Request (MAR) to the MSA AAA server. This message has the Auth-Request-Type AVP set to AUTHORIZE_ONLY, and includes at least a User-Name AVP containing the identity that will be used to authorize the mobility service on behalf of the MN. Note that this identity was conveyed to the HA within the last DEA message (step 1-12), so it might not be the same one that was used by the MN for the IKEv2 authentication.

After checking the MN's profile, the MSA-AAA replies with a Diameter MIP6-Authorization-Answer that contains a Result-Code AVP with the authorization decision. This message may also contain additional AVPs to enforce specific policies for the mobility service. The HA can then send a BA to the MN (step 2-5), completing the bootstrapping process.

The following Figure 3-9 shows the message flow regarding the second part of the process:

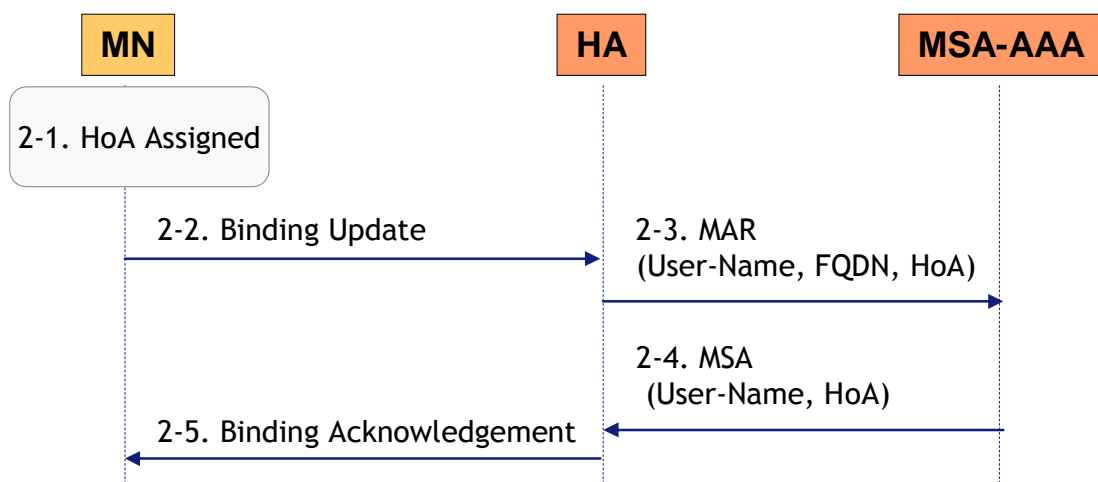


Figure 3-9: Mobility service authorization

- **Step 2-1.** The MN has authenticated with the HA, and has obtained a HoA.
- **Step 2-2.** The MN sends the first BU to the HA.
- **Step 2-3.** The HA receives the first BU and request authorization on behalf of the MN by sending a Diameter MIP6-Authorization-Request (MAR) to the MSA-AAA server.

- **Step 2-4.** The MSA-AAA server replies with a Diameter MIP6-Authorization-Answer (MSA) including the authorization response.
- **Step 2-4.** The HA sends a BA to the MN, finalizing the bootstrapping process.

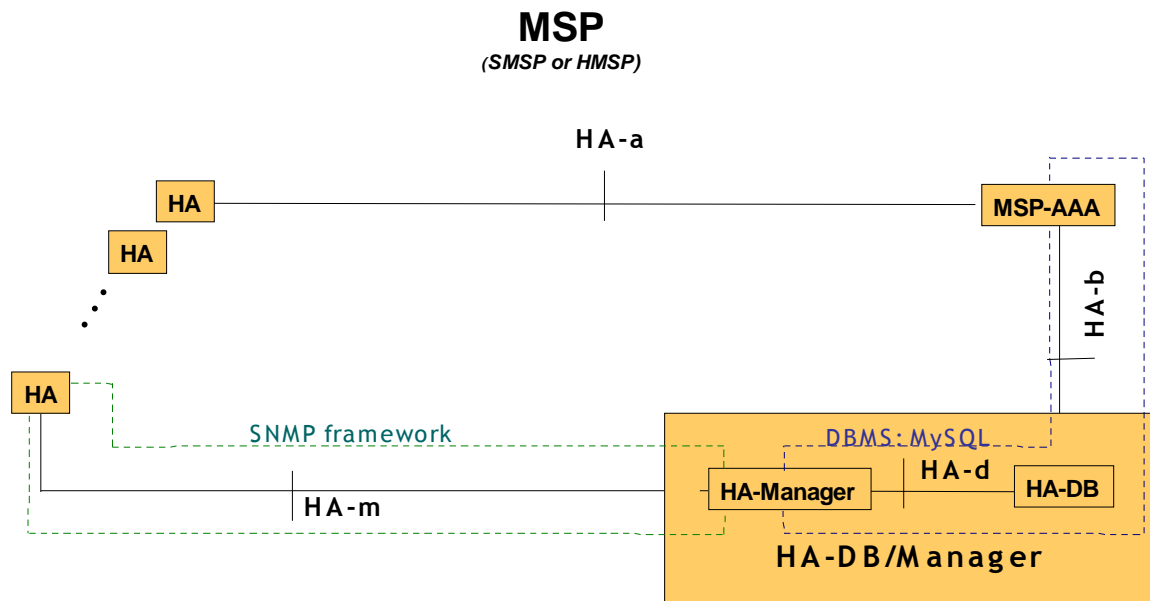
3.2.3 Home Agent load sharing

3.2.3.1 Architecture and overview

One component of the ENABLE prototype will be a HA load sharing mechanism. The architecture of this mechanism has been presented and described in detail in Deliverable D1.1 Section 5.4 of ENABLE [ENABLE-D1.1], but we illustrate this architecture here once again for reference (see Figure 3-10).

In order to provide for load sharing and reliability, the MSP operates several HAs. Each mobile node that requests mobility service gets assigned one HA. For performing the selection, the MSP assesses the current situation among the HAs by evaluating several pre-defined selection parameters, which have been outlined already in [ENABLE-D1.1] and are explained in more detail in section 3.2.3.2. Some selection parameters are available on the HAs. These parameters are collected by the HA Manager periodically and stored in a database denoted as HA-DB. For our prototype we have decided to perform this collection via SNMP since SNMP is the de facto standard network management protocol in the Internet. Beside the parameters collected from the HAs, we have parameters set by the HA administrator that have also relevance for load sharing and those are stored in HA-DB as well. In our prototype we will realized the database and the respective interfaces in MySQL.

In order to perform the evaluation, the MSP-AAA periodically queries the HA-DB for its content. On demand, the current load of each HA is calculated and the most appropriate HA is selected (see section 3.2.3.3 for more details about the selection process). In the integrated scenario (MSA = ASA), after selection the address of the selected HA is forwarded to the MASA entity. Since in our case MSP = MSA, in the integrated scenario MSP and MASA are the same entity. In the split scenario, HA load sharing is realized via HA relocation. After selection of the most appropriate HA the MSP-AAA triggers HA relocation with the selected HA as new designated HA.



HA-m: Collection of HA selection parameters per SNMPv3 (RFC 3410, RFC 3411, ...)

HA-b: Reading/Querying of HA selection parameters per SQL

HA-d: Storage of HA selection parameters per SQL

Figure 3-10: HA load sharing architecture

3.2.3.2 Selection parameters

The selection parameters for determining the "best" HA can be divided into selection parameters obtained from the HAs and selection parameters that are preconfigured and stored in the HA-DB. In order to have comparable selection parameters, all parameters will be normalised to have values between 0.0 and 1.0.

Selection parameters obtained from Home Agent:

- Number of home registrations (**Registrations**): Currently no MIPv6 MIB implementation exists and currently no suitable MIB object is specified in [RFC 4295] that stores the current number of home registrations at the home agent. Although in [RFC 4295] an object `mip6HaCounterTable` with elements of type `mip6HaCounterEntry` is available, none of these objects can be used for retrieving directly the number of home registrations via SNMP. The number of home registrations could be derived implicitly by counting the number of elements in `mip6HaCounterTable` but this method would require the transfer of the complete object `mip6HaCounterTable` to the HA-Manager for obtaining a single value. Therefore we decided to define an appropriate new MIB object, denoted as

REGISTRATION object below the enterprises.netSNMP.netSnmpExamples subtree with OID 1.3.6.1.4.1.8072.2.5.1. Normalisation of this parameter is achieved by dividing the number of registrations (obtained from REGISTRATION MIB object) by the maximum number of possible home registrations for the respective HA (Max_Reg), which is set by the administrator:

$$\text{Registrations} = \frac{\text{MIB value } \square \text{REGISTRATION} \square}{\text{Max_Reg}}$$

- Currently consumed bandwidth on home link (**Bandwidth**): We will calculate the average consumed bandwidth by using several available MIB objects defined in [RFC 1213]:
 - IfInOctets (OID: 1.3.6.1.2.1.2.2.1.10): This MIB object represents the count of the inbound octets of traffic pertaining to the chosen Interface.
 - IfOutOctets (OID: 1.3.6.1.2.1.2.2.1.16): The MIB object gives the total number of bytes sent on the chosen interface.

From these MIB objects the HA-Manager can calculate the average, currently consumed bandwidth using the following algorithm:

$$\text{Bandwidth} = \frac{\max[\Delta \text{ifInOctets} + \Delta \text{ifOutOctets}] * 8}{\Delta \text{time} * \text{Max_Band}}$$

with

$$\Delta \text{ifInOctets} = \text{ifInOctets}(t) - \text{ifInOctets}(t-1)$$

$$\Delta \text{ifOutOctets} = \text{ifOutOctets}(t) - \text{ifOutOctets}(t-1)$$

$$\Delta \text{time} = \text{time}(t) - \text{time}(t-1)$$

Additionally, the ifDescr MIB object values from the HAs have to be retrieved within the get-bulk request since they are needed to associate the requested bandwidth object values to the respective interfaces.

Selection parameters available on HA-DB

The following selection parameters are set by the HA administrator and pre-configured during HA-DB initialization but may be changed by the administrator during operation via an interface to the HA-DB provided by MySQL.

- Announcement of upcoming maintenance (M_Flag): This selection parameter is set by the HA administrator to 1 in case of an upcoming HA maintenance service, otherwise it is set to 0.
- HA Location (Region_ID): The MSP might be interested to assign a HA located in a specific region of its network, e.g. as close as possible to the MN. In this case the MSP would collect only the selection parameters from those HAs, which are located in the same region as the MN. The location of the HA is specified in the Region_ID.

Additional parameters required for Home Agent load sharing

Additionally to the selection parameters, the following parameters are required for the HA load sharing implementation and will be stored in the database as well:

- HA interface address (HA_IP): This is the global IP address of the interface, on which the HA is providing MIPv6 services. In case a HA provides MIPv6 services on multiple interfaces, it will have a separate HA-DB entry for each of these interfaces. This parameter is represented as AAAA-record.
- Maximum number of possible home registrations (Max_Reg): The maximal foreseen number of active home registrations (Integer) for a HA. This value is required in order to normalize the number of home registrations.
- Maximum bandwidth availability at home link (Max_Band): The maximum possible interface bandwidth (Integer) for the link a HA is serving as home link. This value is required in order to normalize the bandwidth.
- HA polling interval (HA_Ptime): The polling interval decides how often the HA selection parameters are collected by the HA-DB/Manager. Note, this polling interval implicitly also sets the delta time value used for the calculation of the (Bandwidth) parameter.

3.2.3.3 HA selection

For HA selection, the MSP-AAA has to compute the HA load for each of the HAs and to determine on this basis the most suitable one. Therefore, the MSP-AAA queries the HA database (HA-DB) for its content by using standard SQL Select messages. In order to minimize the signalling overhead when reading the HA-DB, the MSP-AAA by default reads the HA-DB periodically and stores the obtained data in a local vector array called Home-Agent-Parameter-Matrix. It comprehends for each HA of the MSP a separate line with all the selection parameters

of the respective HA, as well as with additional parameters required for performing HA load sharing, such as the HA's IP address.

The time between the polling cycles (HA-DB_Ptime) can be adjusted to the needs of the administrator on the MSP-AAA. The respective timer value HA-DB_Ptime is defined in seconds. A candidate for a default value would be 180s (3 minutes) since evaluation parameters are not expected to change very significantly during this period.

HA selection is performed in a two step approach. First, if the parameter Region_ID is specified, only the HAs with an appropriate Region_ID are pre-selected; if this parameter is not specified, all HAs are pre-selected. HAs with the M_Flag set are removed from this pre-selection since a HA that will be shutdown for maintenance soon will not be considered as candidate. Second, the load of each of those pre-selected HAs is evaluated. For calculating the load, the selection parameters will be considered in a weighted way:

$$\text{load}_{\text{HA}_i} = W_1 * P_{i1} + W_2 * P_{i2} + \dots + W_n * P_{in}$$

P_{i1}, \dots, P_{in} are the HA selection parameters for HA_i

W_1, \dots, W_n are the weighting factors.

The default weighting factors are set by the MSP. Thereby the selection parameters can be implemented in a mobility service provider independent way, allowing with the setting of the weighting factors each MSP to select its own HA load sharing policy, which best fits its operational needs. For example, if a MSP has no interest on considering the selection parameter P_{im} for HA load sharing, it just has to set the respective weighting factor W_m to zero when calculating the load for each of its HAs. Weighting factors are integers within the range from 0 to 100, with the higher weighting factor giving more relevance to the corresponding HA selection parameter.

The selection process can be described as function denoted as Select_HA() that takes the selection parameters as input and returns as result the dedicated IPv6-Address of the selected HA:

Selected HA = Select_HA (Reg_ID; On_Demand_Flag; W_Flag; W₁...W_n)

The following input parameters are given:

- **Reg_ID:** In case the MSP wants to assign a HA located in a specific region it would set this parameter and the SELECT_HA function would only evaluate HAs with a matching Reg_ID. The Reg_ID is stored as Integer. In case the Reg_ID is set to zero, all stored HAs should be considered for the HA selection task.

- **On_Demand_Flag:** The MSP-AAA usually reads periodically the HA-DB, and updates with this information its own Home-Agent-Parameter-Matrix (On_Demand_Flag set to 0). This periodic interval has to be configured on the MSP-AAA. However, optionally the call for the Select_HA() function also allows setting an On_Demand_Flag to 1, which triggers the MSP-AAA to update its Home-Agent-Parameter-Matrix by collecting the most recent information from the HA-DB. By default the On_Demand_Flag is set to 0.
- **W_Flag:** Only if this bit is set, the weighting factors specified in the SELECT_HA function will be considered, otherwise default weighting factors to be configured on the MSP-AAA are being used. These default weighting factors are set by the MSP-AAA, reflecting its specific HA load sharing policy.
- **W1...Wn:** The weighting factors range from 0 to 100 whereas the importance increases by the value.

3.2.3.4 Message flow for HA load sharing

In this section we would like to illustrate the message flow diagram of the HA load sharing mechanism (see Figure 3-11).

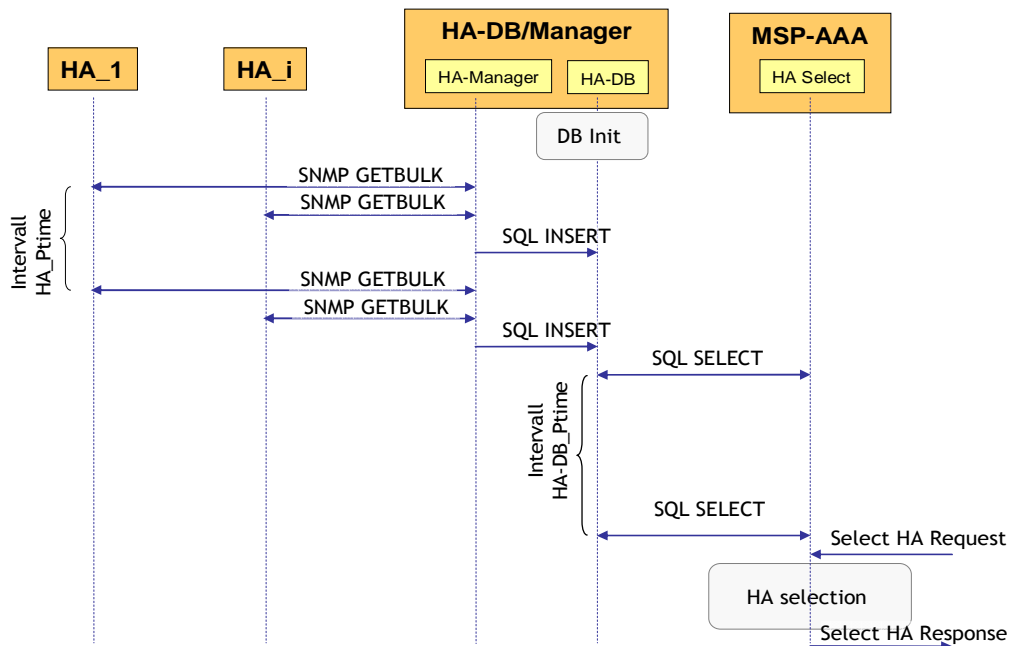


Figure 3-11: Message flow for HA load sharing

After start, the HA-DB is initialized, and the static parameters are set. With an interval of HA_Ptime, the HA-Manager process periodically queries each HA (HA_1, .. HA_i) via an

SNMP-GetBulk() request for the current parameters, the HA-Manager normalizes the parameters, and stores the parameters in the database HA-DB using the SQL-Insert() function.

With an interval of HA-DB_Ptime, the HA Select process on the MSP-AAA entity queries the HA-DB for the current normalized parameters and stores them in the local Home-Agent-Parameter-Matrix.

The HA Select process provides an interface for other processes in order to invoke the selection mechanism. The requesting process sends a Select HA Request message towards the HA Select process, containing several parameters that specify and adjust the selection process, and the HA Select process returns a message Select HA Response that contains the IPv6 address of the selected HA. In the integrated scenario with the MSP being the home provider (MASA=MSP) the Freeradius server component of the MASA will invoke this interface.

3.2.4 Interworking with IPv4 networks

Mobile IPv6 allows Mobile Nodes (MN) to move among different IPv6 subnets while maintaining live ongoing sessions. In order to maintain the sessions also when moving into IPv4 networks, new functionalities must be added to MIPv6.

The functionality considered in this software development which allows a MN to keep alive the ongoing IPv6 sessions, while moving within IPv4 sub-networks are:

- **Support of MIPv6 signalling messages within IPv4 networks:** in order to be able to maintain the reachability within IPv4 networks MIPv6 has to be able to register at the HA both IPv6 and IPv4 Care of Addresses (CoA), moreover it has to be able to send the MIPv6 mobility management messages when the MN is located within IPv4 networks.
- **IPv6 data transportation within IPv4 networks:** the MN has to be able to send/receive IPv6 data packets when it is located within IPv4 networks.
- **Movement detection for IPv4 networks:** it is a fundamental feature since it is needed to trigger the mobility management procedure.

The solution adopted to add the support of MIPv6 signalling messages and the support of the IPv6 data transportation within IPv4 networks is based on the draft [DSMIP].

The movement detection algorithm proposed for IPv4 networks is mainly based on the DHCP protocol and the procedure described in [DNAv4].

3.2.4.1 Dual Stack Mobile IPv6 (DSMIP)

The solution described within [DSMIP] gives to a dual stacked MN the ability to send and receive signalling messages and data packets while it is attached within an IPv4 network.

This approach requires the presence of a HA and a Home network dual stacked. This assumption is needed since this solution is based on the tunnelling of IPv6 packets, both signalling or data, within an IPv6 in an IPv4 tunnel, the end points of the tunnel being the MN and the HA.

The format of the Binding Update (BU) and Binding Acknowledge (BA) messages will depend on the type of access network the MN is attached to. Three different cases may be detected:

- MN attached to an IPv6 network.
- MN attached to an IPv4 network and configure an IPv4 public address.
- MN attached to an IPv4 network and configure an IPv4 private address (NAT).

The DSMIP draft describes a solution for all these three cases but in our development work we have not considered the third scenario: when a MN is attached to an IPv4 only network we have made the assumption that it is always able to configure an IPv4 public address

When the MN is attached to IPv6 networks, since this scenario can be handled with the Mobile IPv6 protocol, the BU / BA messages follow the standard [RFC3775] therefore no extensions are required.

When the MN is attached to IPv4 networks and it configures an IPv4 public address, the IPv6 BU / BA messages need to be tunnelled. The format of these messages is as follows.

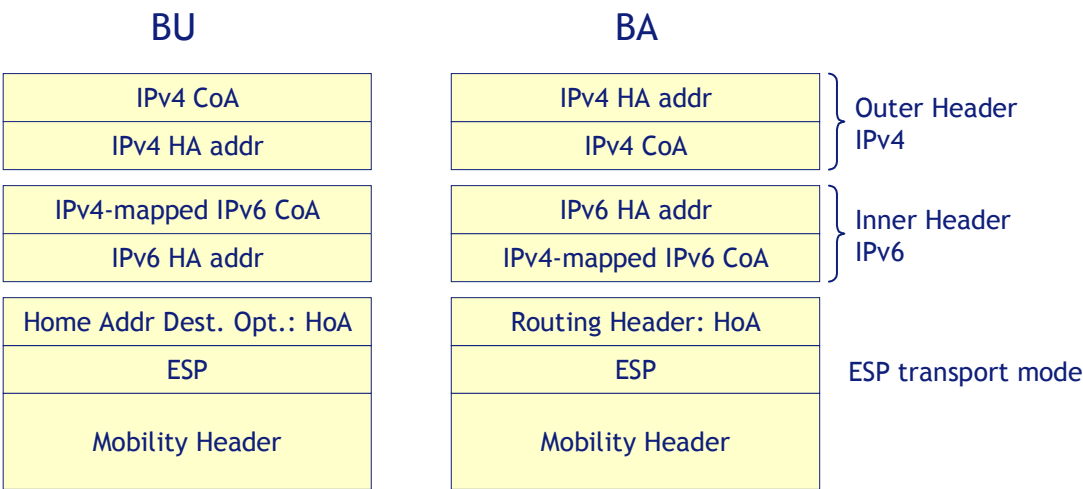


Figure 3-12: BU/BA messages within IPv4 only public network

As shown in Figure 3-12 the destination address of the outer header is the IPv4 address of the HA and the source address is the IPv4 public address that the MN has. In the inner packet there is a standard IPv6 BU/BA message in which an IPv6 CoA is announced. Since the MN owns only a IPv4 CoA, the IPv6 CoA field is filled with the IPv4 address represented in the IPv4-mapped IPv6 form.

The processing of the BU containing the registration of the IPv4-mapped IPv6 CoA is the same as specified within [RFC3775]. If the HA accepts the registration creates a new Binding Cache Entry (BCE) where the CoA address (IPv4-mapped IPv6) is inserted in association with the IPv6 HoA. Following the standard [RFC3775], once the BCE is inserted, the HA replies to the MN sending a BA. The destination address of the BA is the IPv4-mapped IPv6 address. The HA, recognising these types of addresses, sends the BA through the IPv6 in IPv4 tunnel using as destination address the IPv4 address of the MN and as source address his IPv4 address. Upon receiving this BA, the MN updates the Binding Update List (BUL) entry referring to the CoA IPv4-mapped IPv6 contained within the BA.

Hence, all packets addressed to the mobile node's IPv6 home address will be encapsulated in an IPv4 header that includes the home agent's IPv4 address in the source address field and the mobile node's IPv4 care-of address in the destination address field.

3.2.4.2 Movement detection

The Movement Detection (MD) algorithm plays a key role in terms of handover latency since it has to trigger the mobility signalling procedure in order to update the registration within the HA, announcing the new position of the MN (new CoA).

Since in the considered scenario the MN is able to move among sub networks both IPv6 and IPv4, the movement detection procedure has to be able, in both cases, to understand when the MN changes IP sub-network in order to trigger the binding management procedure.

In order to define an algorithm we have done some assumptions:

- When an interface of the MN is attached to an IPv6 network, the stateless auto-configuration is used [RFC2462] to obtain a global IPv6 address and the Neighbor Discovery procedure [RFC2461] to probe the reachability of the default router.
- When an interface of the MN is attached to an IPv4 network, the DHCP protocol [RFC2131] is used to configure a public IPv4 address and the ARP protocol to probe the reachability of the default router.

This algorithm has been designed to support MNs that own multiple interfaces. A preference value is assigned to each interface to select which interface must be used when two or more interfaces have connectivity. If an interface has both IPv4 and IPv6 connectivity the IPv6 protocol is the preferred one.

The representation of the algorithm is done using a state diagram. The following list defines the states of the interfaces (blue ovals):

- **Start:** when the MIPv6 protocol is activated all the active interfaces are in this state.
- **No router:** the interface is active but cannot be used since it is not globally reachable (i.e. no IP address or/and no access router).
- **OK IPv6:** the interface has an IPv6 default router and an IPv6 global address that is registered with the HA.
- **OK IPv4:** the interface has an IPv4 default router and an IPv4 public address that is registered with the HA.
- **Suspect IPv6:** some movement hints have been received while the interface was in the state OK IPv6; therefore the IPv6 reachability has to be checked.
- **Suspect IPv4:** some movement hints have been received while the interface was in the state OK IPv4; therefore the IPv4 reachability has to be checked.

There are two types of events that may cause the movement from one state to another, the reception of a network message or a timer expiration associated to the state. Except for Start, a timer is associated to every state.

The following list describes the main transitions that may occur between the above described states:

- **No router → No router:** the timer (**Timeout**) expires, the algorithm verifies if the interface is the one registered within the HA. If the verification is positive it tries to register another interface that is in the state OK IPvx (IPv6 or IPv4). From the interface, on which the timer is expired, are sent two messages Router Solicitation (RS) and DCP Discovery.
- **No router → OK IPv6:** a RA is received, the interface configures an IPv6 default router and a global IPv6 address. The MD algorithm verifies if the priority assigned to that interface is higher or equal in comparison to the one of the registered interface. If the

answer is “yes” the MN updates the registration with this new interface, otherwise it changes only state without performing any action.

- **No router → OK IPv4:** a DHCP acknowledgement message is received, the interface configures an IPv4 default router and a public IPv4 address. The MD algorithm verifies if the priority assigned to that interface is higher or equal in comparison to the one of the registered interface. If the answer is “yes” the MN updates the registration with this new interface, otherwise it changes only state without performing any action.
- **OK IPv6 → Suspect IPv6:** the timer (**Timeout Rtr**), associated to the IPv6 default router, expires. The MN has to verify if the default router is still reachable, this verification is performed sending a NS to the IPv6 address of the router.
- **OK IPv4 → Suspect IPv4:** the timer (**Timeout ARP**) of the ARP entry, associated to the IPv4 default router, expires. The MN has to verify if the default router is still reachable, this verification is performed sending an ARP request for the IPv4 address of the router. Contemporaneously is sent a DHCP request.
- **Suspect IPv6 → OK IPv6:** a NA message is received, this means that the default router is still reachable therefore the MN guesses that it still attached to the same IPv6 sub-network.
- **Suspect IPv4 → OK IPv4:** a ARP reply or a DHCP acknowledgement is received therefore the MN guess that it still attached to the same IPv4 sub-network.
- **Suspect IPv6 → No router:** the timer (**Timeout NUD**) expires. The MN before to move to the state **No router** sends from the interface a RS and a DHCP discovery.
- **Suspect IPv4 → No router:** the timer (**Timeout ARP**) expires, this means that an ARP request was sent but the ARP reply is never received, the ARP entry was deleted. Another event that may cause this transition is the reception of a DHCP NACK message. The MN before to move to the state **No router** sends from the interface a RS and a DHCP discovery.
- **OK IPv4 → OK IPv6:** the MN moving may enter at any time within an IPv6 network and receives a RA message unsolicited. If the RA message is received the MN sets up: the IPv6 CoA and the IPv6 default router. Then the MN checks if the priority assigned to the interface is higher or equal to the one registered. If the answer is “yes” the MN updates the registration, otherwise it changes only state without performing any action.

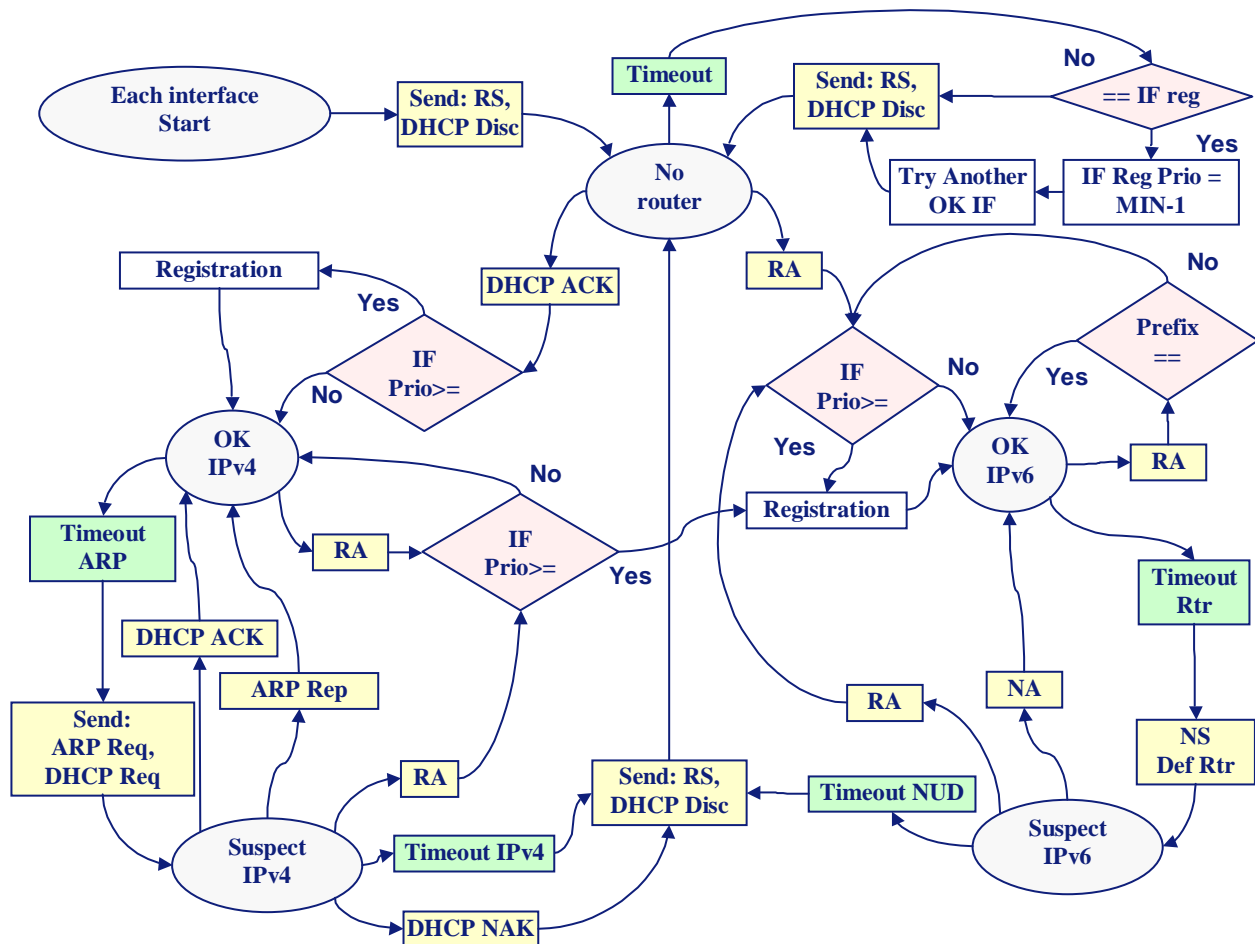


Figure 3-13: Movement Detection state diagram

3.2.5 MIPv6 firewall traversal

There are problems and impacts caused by having firewalls in Mobile IPv6 environments. When a user moves to a visited network, a firewall located in the home network, the visited network or the access network of the corresponding node may have impacts on the Mobile IPv6 data and signalling messages. For instance, route optimization, an integral part of Mobile IPv6 specification, does not work with state of the art firewalls that utilize stateful packet filtering. This set of extensions is a fundamental part of the protocol, enabling optimized routing of packets between a mobile node and its correspondent node and therefore providing optimized communication performance. However, in most cases, current firewall technologies do not support Mobile IPv6 or are not even aware of Mobile IPv6 mobility extension headers. Since most networks in the current business environment deploy firewalls, this may prevent future large-scale deployment of the Mobile IPv6 protocol.

In ENABLE there are a number of middlebox traversal solutions being studied such as Universal Plug and Play, STUN/TURN/ICE, Application Layer Gateways, Middlebox Communication,

Simple Middlebox Control, Policy Based Networks and the VPN approach, but as a prototype ENABLE is going to develop a specific firewall pinhole creation and authorization framework based on an extended NSIS [RFC4080] NAT/FW NSLP [NATFW] approach.

The following Figure 3-14 shows the overall reference scenario in which a NSIS for Mobile IPv6 middlebox traversal implementation has to work.

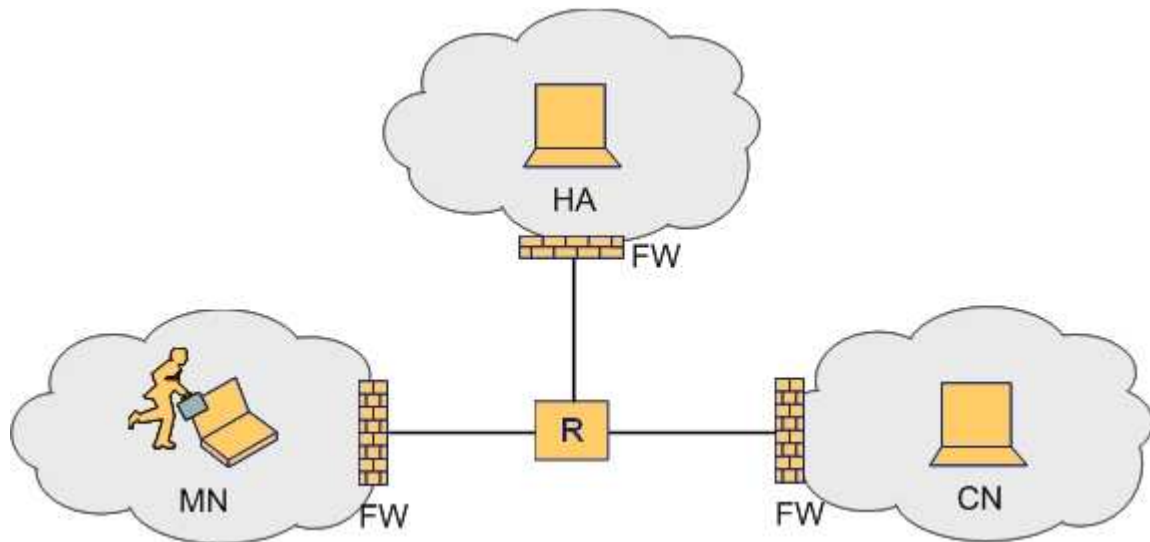


Figure 3-14: Reference network scenario for middlebox traversal

The three scenarios selected for the prototype implementation are:

- Scenario 1 – Firewall located in Mobile Node’s ASP.
- Scenario 2 – Firewall located in Correspondent Node’s ASP.
- Scenario 3 – Firewall located at the edge of the Mobile Node’s MSP.

The goal of the firewall traversal implementation work in WP6 is to develop an ‘NSIS for Mobile IPv6 firewall traversal’ prototype. Therefore, we will develop a prototype implementation that is able to signal all pinholes which are necessary to run Mobile IPv6 and its features in the presents of firewalls. In the first step the prototype should be able to overcome the problems which occur when a firewall is in the MN’s ASP (Scenario 1). In the second step, the prototype implementation should be able to handle all three possible scenarios. However, the prototype which will be developed inside ENABLE will not include AAA interaction

3.2.6 Mobility optimizations

In order to support seamless inter-technology handover when a Mobile Node (MN) moves across heterogeneous access networks, we have to take into consideration many important aspects, such

as application classes, service continuity and quality, network discovery, network selection, security, to ensure the overall Quality of Service (QoS) for users. In Mobile IPv6 (MIPv6), the handover delay introduced by the neighbour network discovery, address configuration, mobility Binding Updates (BU), as well as the network specific authentication and authorization would be intolerable for many services. This could be an extremely serious problem for real-time services, such as VoIP and streaming media, which have stringent performance requirement on end-to-end packet latency and loss.

Lots of effort has been put on optimising MIPv6 in the IETF MIPSHOP working group. Hierarchical MIPv6 [RFC4140] and Fast handover for MIPv6 [RFC4068], [RFC4260] are two standardised schemes that are designed to enhance MIPv6 in the signalling and handover aspects so as to reduce the handover delay. HMIPv6 reduces the signalling overhead and binding update delay by using a hierarchical architecture - the Mobility Anchor Point (MAP) is integrated in this architecture for this purpose. FMIPv6, on the other hand, minimize the duration of service disruption during the handover by exploiting various L2 triggers to anticipate handover - a new Care-of-Address (NCoA) is prepared at the new access router (NAR) for handover to allow a bi-directional tunnel between previous and new access routers to be established. The handover decision within a heterogeneous network plays a vital role in the seamless mobility service provisioning.

By the time we write this document, our study has been focus on optimising MIPv6 with the FMIPv6. We will implement a prototype system to demonstrate the outcome of this study.

3.2.6.1 Overview of FMIPv6 Protocol

The latencies involved in the movement detection, CoA configuration, and CoA testing (i.e. DAD) in MIPv6 could be reduced if we have the knowledge of neighbouring ARs and its subnet affiliations prior to handover. Such mechanism allows a MN to anticipate its attachment with a new router (behind a new link), by querying its default router to provide the subnet prefix, IP address and MAC address of a target router attached to a neighbouring access point.

The FMIPv6 is designed to allow MNs to anticipate and initiate an IP layer Handover through the usage of link layer triggers. These link layer triggers are delivered to the network layer modules as events for reporting changes with regard to the link and physical layer conditions. For instance, when the MN detects that its signal quality with its attached AP is going down or about to go down, the link layer sends a trigger to the network layer which in turn starts the IP layer movement anticipation and initiation.

During the anticipation phase, the MN sends Router Solicitation for Proxy Advertisement (RtSolPr) message to the default access router to resolve one or more AP Identifier (AP-ID) to

subnet-specific information. This router known as oAR (Old Access Router) replies with a Proxy Router Advertisement (PrRtAdv) message which contains the neighbouring router's advertisement which includes one or more [AP-ID, AR-Info] tuples. Using this information the MN formulates a new CoA (NCoA) while it is still present on the oAR's link. Hence, the latency due to new prefix discovery is eliminated. A Fast Binding Update with the NCoA is sent to the oAR upon which the oAR sends a Handover Initiate (HI) message to the new AR (nAR). This HI message serves two purposes: firstly it identifies whether the nAR is aware of any address duplication and secondly, it triggers the set up of a tunnel between the oAR and nAR. After the tunnel is set up, the nAR replies with a Handover Acknowledgement (HACK) message. Upon receiving the HACK message, the oAR sends a Fast Binding Acknowledgement (FBack) to the MN.

In the event the MN moves without receiving an FBack, the MN mobile can start using its NCoA after sending a Fast Neighbour Advertisement (FNA) to the nAR. In this case, the FBU is sent from the nAR's link and encapsulated within the FNA message. Essentially, the FNA message is used to inform the oAR that the MN is now attached to its new link and requests two things: first, it requests that the nAR forwards any buffered packets to the MN (including the FBack if it did not receive it on the oAR's link), and secondly, it requests that the nAR tells the MN if the NCoA is invalid. The latter is delivered to the MN in a Neighbour Advertisement Acknowledgement (NAACK), which is included in a router advertisement unicast to the MN's previous Care-of-Address (PCoA).

There are two types of operational modes in FMIPv6: the predictive mode and the reactive mode. In the predictive mode, the FBU is sent from the oAR's link and the FBack is also received in the oAR's link before it moves to the new link. In the reactive mode, the FBU is sent from the nAR's link. The reactive mode also includes the case when FBU is sent from the oAR's link but Fast Binding Acknowledgement (FBack) has not been received yet. Therefore, in the reactive mode, long handover delays would be induced due to NCoA configuration and Binding Updates. Due to this reason, it is desirable that the FMIPv6 protocol always operates in the predictive mode. Given below is a diagram which illustrates the operations of the FMIPv6 protocol.

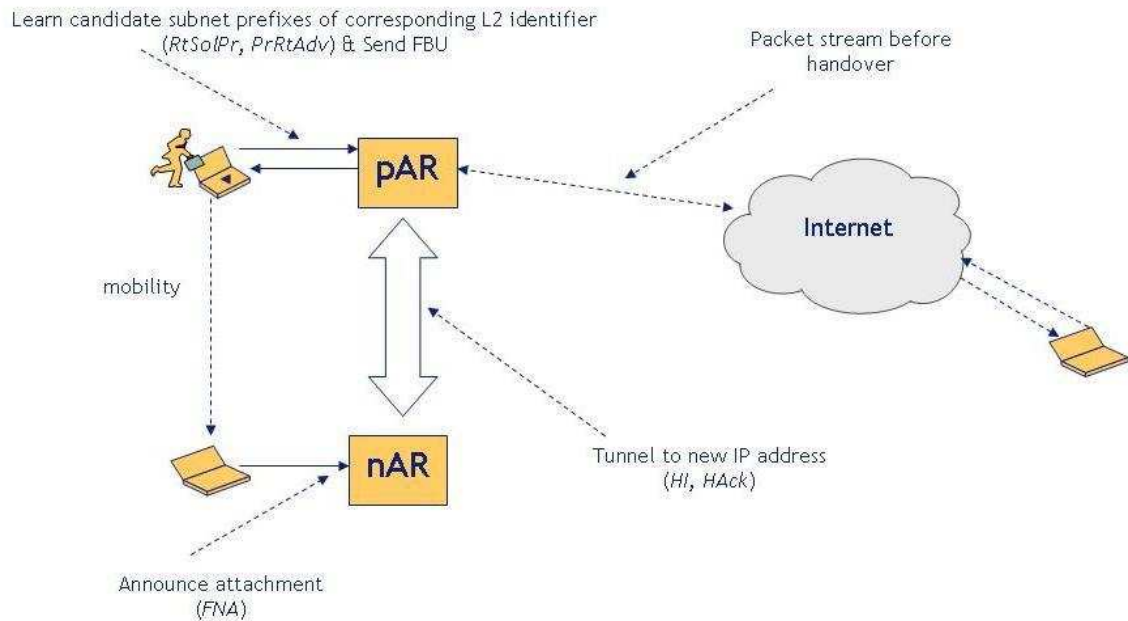


Figure 3-15 FMIPv6 Protocol Flow Scheme

3.2.6.2 Description of the Reference Architecture

In WP6 we will first implement a prototype system to demonstrate FMIPv6 based MIPv6 handover service optimization scheme.

The reference architecture of this prototype system is shown below. In this architecture, the components of service authorisation for MIPv6 service and FMIPv6 - the BCA, BAA proxy and BAA - are included. However, at the first stage of system prototyping, we will not implement the authorization components, BCA, BAA proxy and BAA functions that collated in the MSA.

4. INITIAL PROTOTYPING

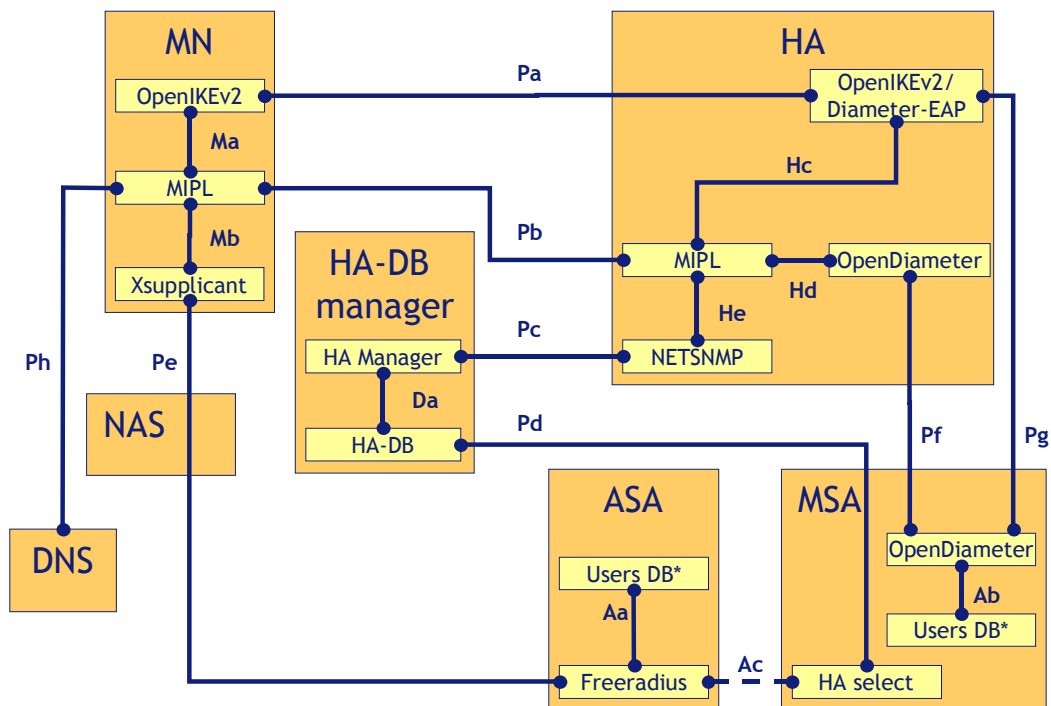
This section provides the software development plans for the initial prototyping of six functional components from the ENABLE project. The section also gives an explanation of how four of the six components (EAP-based MIPv6 bootstrapping, AAA for MIPv6 bootstrapping, Interworking with IPv4 networks and HA load sharing) are being integrated at an early stage and provides descriptions of the software architecture, interface descriptions and software modules to be created.

An initial description of additional on-going developments on NSIS and FMIPv6 is also given while descriptions of the common software development platform and development tools being used during the prototyping phase are here also.

4.1 Integrated Software architecture

4.1.1 Reference Architecture

It is to be noted that this reference architecture is based on the assumption that the MSA and the MSP are co-located.



* In an integrated scenario this two User DB could be the same one and the interface Ac is present

Figure 4-1: Integrated Software architecture

In the Figure 4-1 is drawn the Integrated Software architecture. The functional elements (orange rectangle) that compose this architecture are:

- Mobile Mode (MN).
- Home Agent (HA).
- HA-DB Manager.
- Network Access Server (NAS).
- DNS Server (DNSS).
- Access Service Authorizer (ASA) server.
- Mobility Service Authorizer (MSA) server.

The yellow rectangles represent the software modules, meanwhile the main interfaces are represented with blue connectors. Both software modules and interfaces are described in the following subsections.

4.1.2 Interface descriptions

4.1.2.1 Mobile Node

4.1.2.1.1 Ma (MIPL – OpenIkev2)

Ma is an inter process interface implemented using a **UDP socket**. This interface is used to trigger the OpenIkev2 daemon in order to establish a Security Association (SA) between the MN and the HA. Within this trigger message the MIPL daemon communicates to OpenIkev2 the Home Agent address previously obtained (through EAP or DNS) with which the MN has to perform the SA negotiation. Within the response the OpenIkev2 daemon communicates to MIPL the Home Address (HoA) assigned to the MN during the IKEv2 exchange. The hypothesis is that the IKE daemon is started before MIPL.

The MIPL Configuration message exchanged between MIPL and OpenIkev2 has the following format (Figure 4-2).

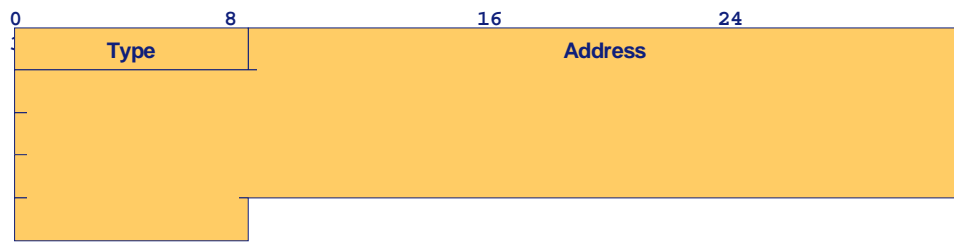


Figure 4-2: MIPL Configuration messages

The Type field is one octet, and indicates the type of MIPL Configuration message. The types used by this interface are:

- **4 - HoA Request:** sent by MIPL to OpenIkev2; it triggers the IKEv2 exchange with the specified HA.
 - Address = HA address.
- **5 - HoA Reply:** sent by OpenIkev2 to MIPL and used to deliver the HoA.
 - Address = HoA.

The rules that the MIPL and the OpenIkev2 modules have to observe are:

- As soon as MIPL daemon knows the HA address, it sends to OpenIkev2 the HoA Request inserting the just obtained HA address.
- When OpenIkev2 daemon receives the HoA Request it starts the IKEv2 exchange with the HA address specified in the Request.
- When OpenIkev2 daemon obtains the HoA it sends the HoA Reply message to the MIPL daemon.
- When MIPL daemon receives the HoA Reply it sets up the Secure Policy Database (SPD) using the HoA received and sends the first BU.

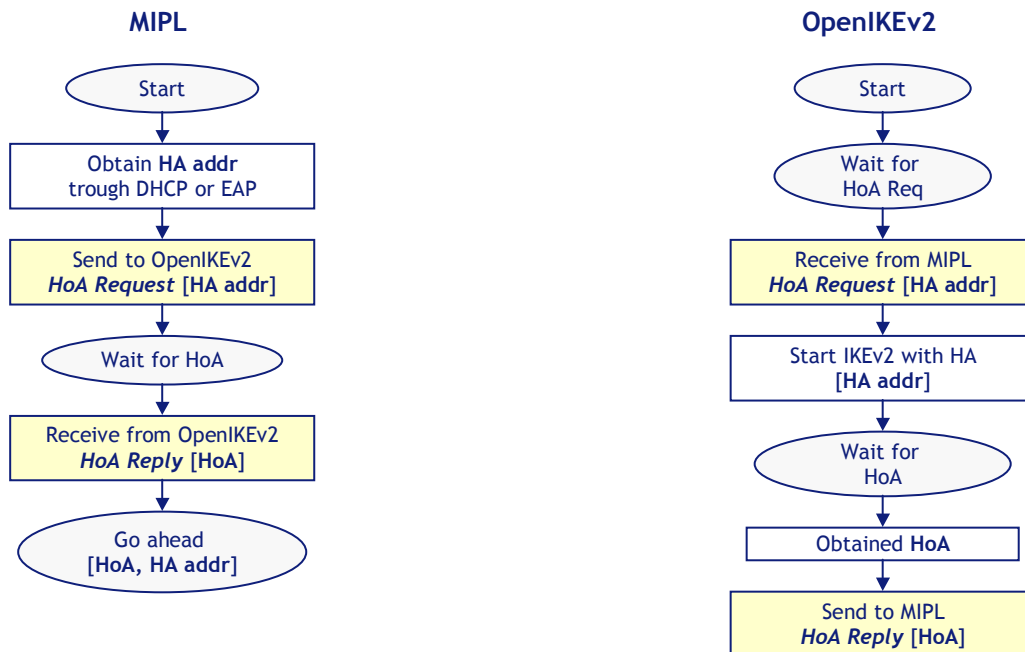


Figure 4-3: Interaction between MIPL and OpenIkev2

4.1.2.1.2 Mb (MIPL – Xsupplicant)

Mb is an inter process interface implemented using a UDP socket. This interface is used to communicate to the MIPL daemon the result of the network access procedure and eventually, in the case of integrated scenario (MSA and ASA co-located), the obtained HA address.

The hypothesis is that the Xsupplicant [Xsupplicant] daemon is started before the MIPL one. Even if this hypothesis is done there is no knowledge on when and if the EAP session, handled by the Xsupplicant daemon, is performed. For instance if the MN try to access in a network that requires an 802.1x authentication, the EAP session starts when the Xsupplicant daemon receives an EAP Request Id. message from an AP. Otherwise, accessing in a free network the connectivity is obtained as soon as the MN get in touch with an AP. Moreover the MIPL daemon may be started at any moment, when the MN is switched on or when the user requires the IP mobility service. Consequently MIPL initialisation and network access, handled by Xsupplicant daemon, reasonably are performed asynchronously. The MN may face two different scenarios, in the first one the MIPL daemon is started after the accomplishment of the network access procedure, in the second one MIPL is started before.

The messages exchanged between MIPL and Xsupplicant has the format reported in Figure 4-4. In order to handle both scenarios the three type of messages defined are:

- 1 - HA addr Request: sent by MIPL to Xsupplicant asking for the HA address.
 - Address = 0...000.

- 2 - HA addr Reply: sent by Xsupplicant to MIPL, it is used to communicate to the MIPL daemon the HA address bootstrapped, if the HA is never obtained the message is sent inserting 0...00 instead of an IPv6 address.
 - Address = 0...000 or HA address
- 3 - HA addr Acknowledgement: sent by MIPL to EAP, it confirms HA addr Reply reception.
 - Address = 0...000.

The rules that the MIPL and the Xsupplicant modules have to observe are:

- when the MIPL daemon is started it sends a HA addr Request and wait for a HA addr Reply.
- when a HA addr Reply is received the MIPL confirms the reception sending an HA addr Acknowledgement.
- when the access procedure, handled by the Xsupplicant daemon, terminates an HA addr Reply is sent by Xsupplicant to the MIPL, if an HA address is obtained this address is inserted within the message elsewhere it is sent 0...00 instead of the IPv6 address, after this Xsupplicant daemon waits for a message from MIPL.
- if Xsupplicant daemon receives an HA addr Request it repeats the procedure described at the previous point.
- if an HA addr Acknowledgement is received the Xsupplicant daemon deduces that the HA addr Reply was received successfully.

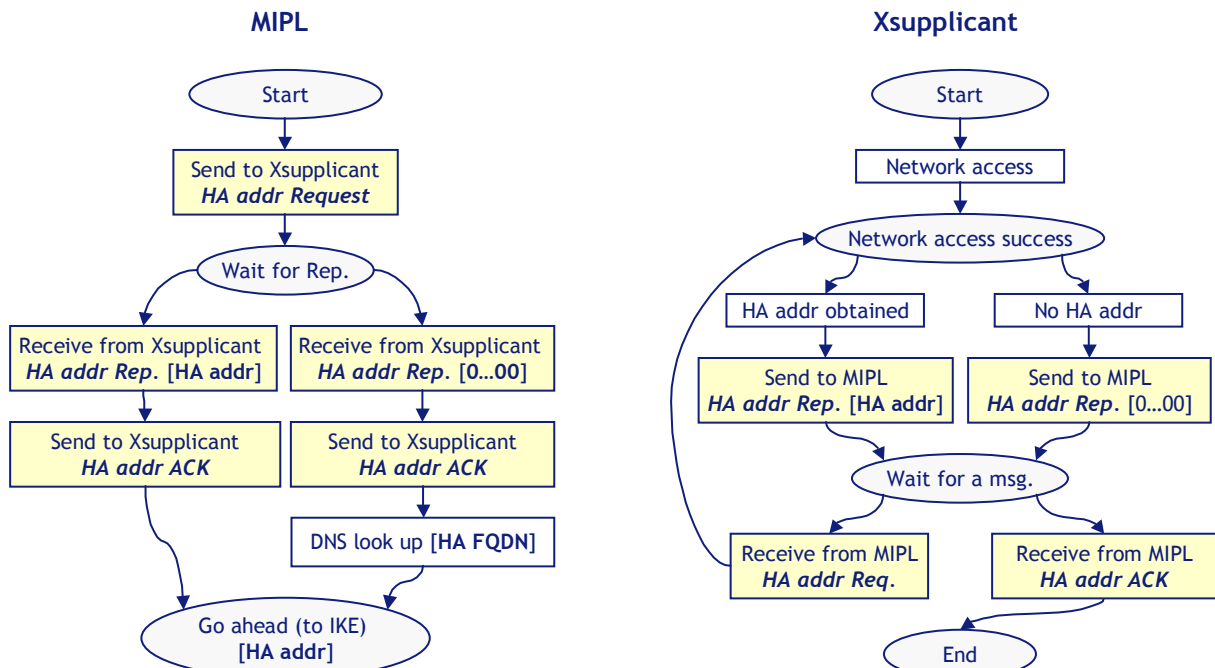


Figure 4-4: Interaction between MIPL and Xsupplicant

4.1.2.2 Home Agent

4.1.2.2.1 Hc (MIPL – OpenIKEv2)

Hc is the interface between the MIPL daemon and OpenIKEv2, it is implemented using a shared file (NAI-HoAs).

The file is a text file composed by two columns one containing the HoA and one that may contain the user's NAI, each row represent an assignable HoA. If a NAI is present in the second column it means that the HoA has been assigned.

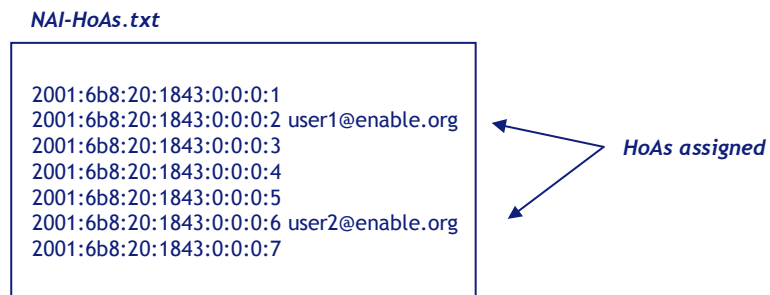


Figure 4-5: NAI-HoAs file example

When the MIPL is started at the HA it reads the standard configuration file where is specified the location of the NAI-HoAs file. After the reading operation, of the configuration file, the MIPL daemon reads the NAI-HoAs file in order to initialise the SPD. MIPL has to insert in the SPD an entry for each assignable HoA.

The same file (NAI-HoAs) is used by MIPL to obtain the NAI associated to the HoA received within a BU. The MIPL module search into the NAI-HoA file which NAI is associated to the HoA received.

When OpenIKEv2 needs to assign a HoA, it reads the NAI-HoAs file looking for an unassigned HoA (a HoA is unassigned if its second column is empty). If any free HoA is found, OpenIKEv2 assigns the HoA and writes the MN NAI in its second column to ensure that MIPL can map the HoA with the NAI in order to authorise the mobility service for that MN.

Rules for the access to the file:

- When OpenIKEv2 needs to assign a HoA, it reads the NAI-HoAs file looking for an unassigned HoA. If any free HoA is found, OpenIKEv2 writes the MN NAI in its second column.
- When MIPL receives the first Binding Update from a new user, it search the NAI associated to the received HoA within the NAI-HoA file.
- When the MIPL recovers the NAI, it starts the user authorization using that NAI.
- The NAI is stored within the new BC entry created in order to be able to re-authenticate the user without accessing again to the NAI-HoA file.

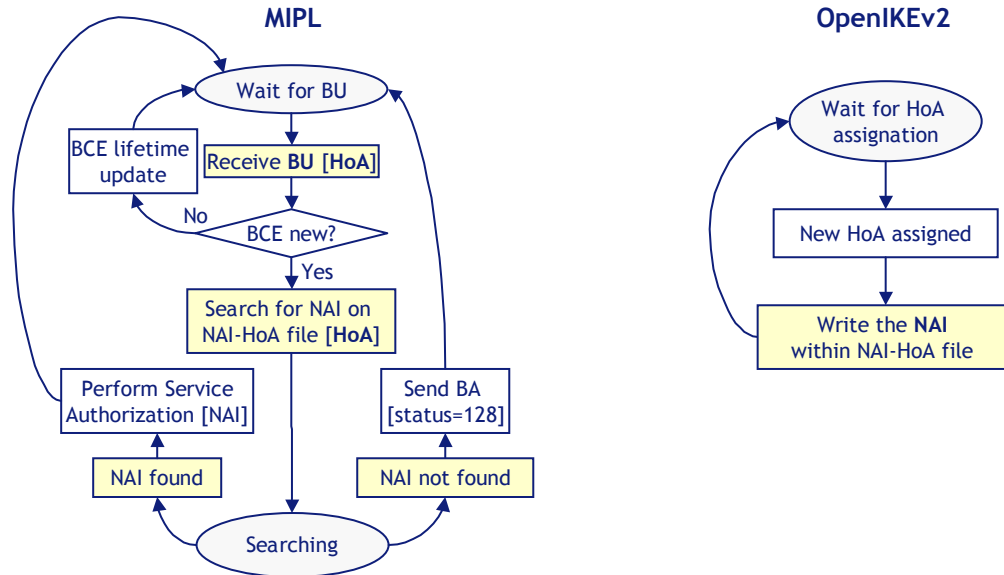


Figure 4-6: Interaction between MIPL and OpenIKEv2

4.1.2.2.2 Hd (MIPL – OpenDiameter)

Hd is an inter process interface implemented using a **UDP socket**. This interface connects MIPL and OpenDiameter to achieve authorization and manage authorization functionalities for the MIPv6 service, as better explained below.

Messages defined for this interface are:

- **User Authorization Request** (Figure 4-7): message sent by MIPL to OpenDiameter client for Mobile User's authorization of MIP6 service.



Figure 4-7: User Authorization Request

- MN Identifier Type = 1 (NAI).
- Identifier = NAI.
- **User Authorization Reply** (Figure 4-8): message sent by OpenDiameter client to MIPL, communicate the result of the User's authorization of MIP6 service

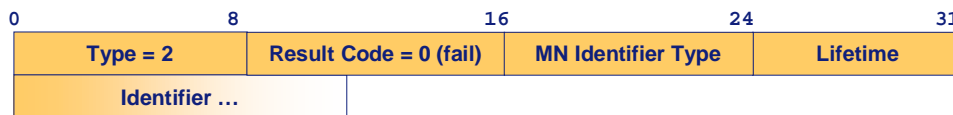


Figure 4-8: User Authorization Reply

- Result Code = 1 (on success), 0 (on failure).
- MN Identifier Type = 1 (NAI).
- Identifier = NAI.
- Lifetime = user authorization lifetime.

Rules for messages exchange:

- When MIPL receives the first Binding Update from a new user, MIPL obtain the NAI associated to the received HoA from the NAI-HoA file. MIPL sends a User Auth. Req. to OpenDiameter inserting the obtained mobile node's NAI into MN_ID_NAI field.

- When OpenDiameter receives a User Auth. Req., it performs user authorization contacting a backend AAA server:
 - 1. in case of success OpenDiameter sends to MIPL a User Auth. Rep. with result_code set to 1 and the NAI of authorized user into MN_ID_NAI field to let MIPL match this success message with the pending request. The MIPL stores the NAI within the BC.
 - 2. in case of authorization failure, OpenDiameter sends back to MIPL a User Auth. Rep. message with result_code 0.
- When the authorization timer for a user's Binding Cache Entry (BCE) expires, MIPL sends a User Auth. Req. to OpenDiameter with user's NAI, previously obtained and stored within the BCE, into MN_ID_NAI field to re-authorise the user; then Open Diameter sends back a User Auth. Rep. message with the answer of the Diameter server.
- When BCE lifetime expires the MIPL delete the BCE.

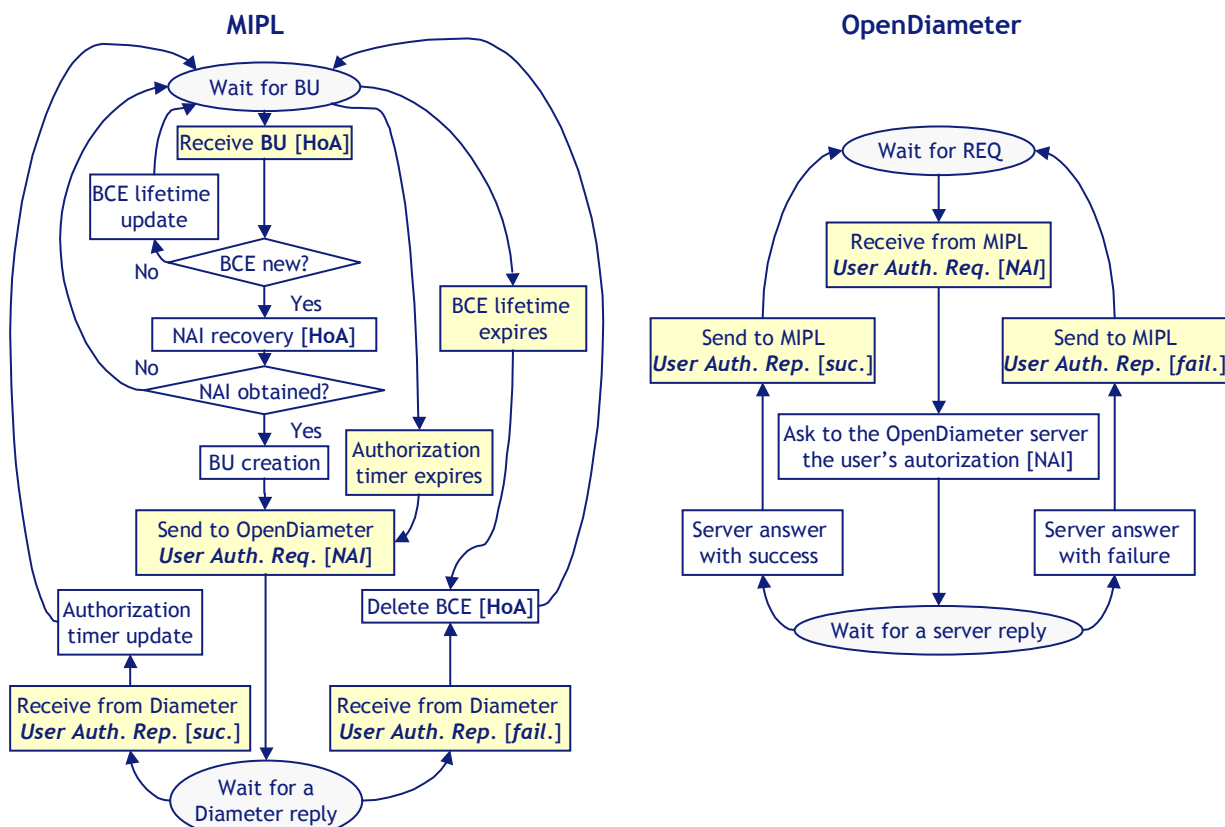


Figure 4-9: Interaction between MIPL and OpenDiameter

4.1.2.2.3 He

He is the interface between the MIPL process and the SNMP agent both running on the HA. One selection parameter that has to be retrieved from MIPL is the number of current home registrations (**Registrations**). Since SNMP agents read from MIBs and neither the MIPv6 MIB is implemented nor the required object is present in the current MIPv6 MIB specification, we need to define and implement the required MIB object on our own. The object denoted as **REGISTRATION** object will be linked within the enterprise subtree to the OID 1.3.6.1.4.1.8072.2.5.1.

A variable that stores the current number of home registrations is available in the MIPL file `bcache.c`, denoted as **bcache_count**. We will modify the file `bcache.c` in order to store the variable `bcache_count` in the **MIB object REGISTRATION**.

4.1.2.3 Home Agent-DB manager

4.1.2.3.1 Da

Da is the interface between HA Manager and HA-DB. The interface is realised in SQL and **SQL commands** are used to exchange data. The HA Manager process uses SQL INSERT and UPDATE commands, respectively, to insert and update the parameters of the database HA-DB. Since HA Manager will be realised as a Java process, a HA-DB JDBC interface is used, which is a MySQL compatible API. (see <http://java.sun.com/javase/technologies/database/index.jsp>).

4.1.2.4 ASA / MSA

4.1.2.4.1 Aa (FreeRadius – SQL users database)

Aa interface is used to retrieve users' data used to authenticate/authorise the access and to deliver MIPv6 bootstrapping parameters. This interface is a standard TCP socket used to deliver SQL queries. The default port used by MySQL is 3306.

4.1.2.4.2 Ab

This interface is used to retrieve user authentication and authorization data. Currently, this interface is based on direct file access to the `opendiameter`'s configuration files, although it may be modified in the future to use a TCP socket in order to fetch the data from a MySQL database via SQL queries.

4.1.2.4.3 Ac

The interface Ac is present in case of the integrated scenario when the ASA and the MSA are the same entity, the MASA, and this interface is not present in case of the split scenario. Hence, this interface is a local interface between the Freeradius instance and the HA Select instance on the MASA entity. The interface will be realized as UDP socket, using port 5100, which has been chosen arbitrarily and should not conflict with other applications. The Freeradius instance sends a message Select HA Request to the HA Select instance, containing the following parameters:

Reg_ID; On_Demand_Flag; W_Flag; W1;...Wn

The HA Select instance selects a HA and sends back a message Select HA Response that contains the IPv6 address of the selected HA. This way, the HA Select instance realizes the function Select_HA() as described in section 3.2.3.3.

Messages defined for this interface are:

- **Select HA Request** (Figure 4-10): message sent by the invoking process (e.g. the Freeradius instance) to the HA Select instance:

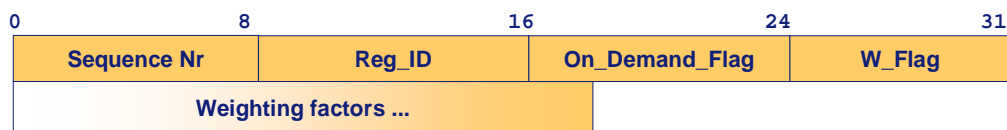


Figure 4-10: Select HA Request

- Sequence Nr (1 byte): The Sequence Nr is replied back in the Select HA Response message in order to relate request and response message.
- Reg_ID (1 byte): Region ID of the region within scope. A value of 0 signals that all regions are relevant.
- On_Demand_Flag (1 byte): If this value is set to 1, this signals to the HA Select process to update the parameter matrix. The default value is 0.
- W_Flag (1 byte): This value signals how many weighting parameters are following. A value of 0 indicates that no weighting factor follows and that the load calculation should be done by using default weighting factors configured on the MSP-AAA.
- Weighting factors: This field contains a sequence of W_Flag weighting factors, each 2 bytes long.

- **Select HA Response** (Figure 4-11): message sent by the HA Select process to the invoking process (e.g. the Freeradius instance):

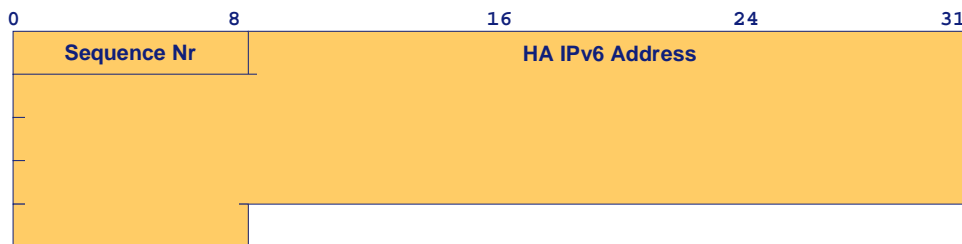


Figure 4-11: Select HA Response

- Sequence Nr (1 byte): Sequence number that is taken from the respective Select HA Request message in order to relate request and response message.
- HA IPv6 address (16 byte): IPv6 address of the selected HA.

4.1.2.5 Network protocols

4.1.2.5.1 Pa IKEv2

Pa is a network interface between the MN and the HA using the IKEv2 protocol. This interface is used to perform the MN authentication by using the Diameter EAP application (note that the authorisation is performed with a different Diameter application, upon reception of the first BU from the MN), to establish the MIPv6 IPsec SAs and to assign the HoA (Home Address) from the HA to the MN.

The IKEv2 exchanges starts when OpenIKEv2 receives the “HoA Request” message though the interface **Ma**. The process has the following steps, as illustrated in Figure 4-12:

- The MN’s OpenIKEv2 client performs an IKE_SA_INIT exchange with the HA, to derive cryptographic material for the IKE SA.
- The MN’s OpenIKEv2 client sends the IKE_AUTH request message including its identity, the desire to use EAP authentication (by omitting the AUTH payload) and requesting the assignation of a remote internal address (in the CP payload). The EAP state machine used for authentication is based on opendiameter.
- The HA’s OpenIKEv2 server acts as an EAP pass-through, forwarding the EAP packets between the MN and the AAA server and vice-versa, using the interface **Pg** and IKE_AUTH messages for EAP transport.

- Once the MN is authenticated, the HA's OpenIKEv2 server obtains a HoA from the internal address pool, stores it in the HoAs file using the Hc interface and sends it to the MN (using the CP payload) in the last IKE_AUTH message. As result of the whole IKE_AUTH exchange, the first IPsec SA is established.
- After that, OpenIKEv2 creates all the remaining IPsec SAs needed to protect the MIPv6 signaling and the traffic between the MN and the HA. In order to do this, the MN OpenIKEv2 client initiates all the needed CREATE_CHILD_SA exchanges with the HA.
- When all the needed IPsec SAs have been created, the MN OpenIKEv2 client sends a "HoA Response" message using the **Ma** interface, in order to notify the new received HoA from the HA.
- The two OpenIKEv2 daemons keep working in order to maintain the IPsec SAs (rekeyings, deletions...).

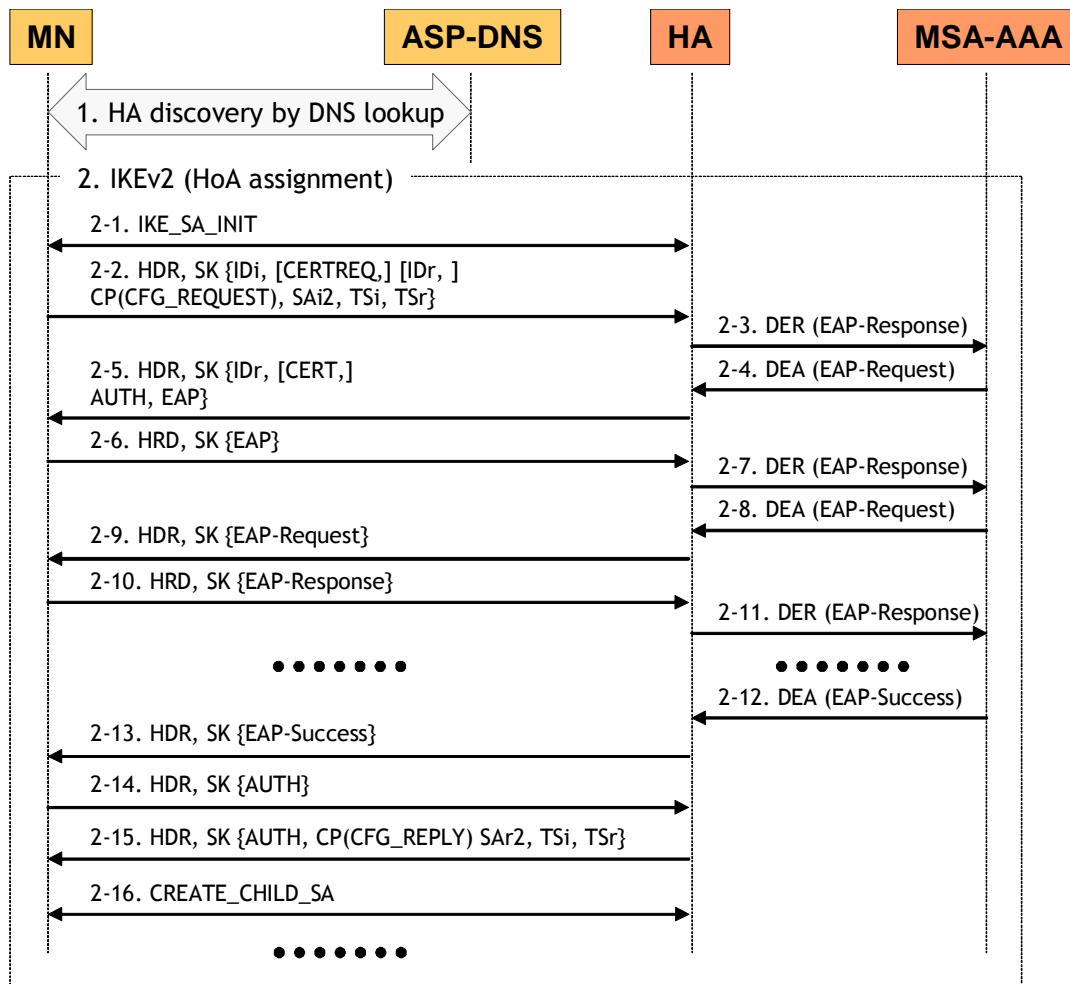


Figure 4-12: MIPv6 authentication and HoA configuration message flow

4.1.2.5.2 Pb MIPv6

The interface **Pb** implements the MIPv6 protocol [RFC3775]. In the first phase of the integration no modifications are foreseen, while for the final integration some features will be added to support IPv4 interworking as described in Section 3.2.4.

4.1.2.5.3 Pc SNMP

Pc is the interface between the SNMP Agent on the HA and the SNMP Manager on the HA Manager. The selection parameters are collected by the HA-Manager from the respective HAs using a **SNMPv2/v3 compatible GETBULK request** that reduces the message overhead by obtaining a bundle of information with only one request message instead of many individual get request messages. If required, this management traffic can be secured using the security features of SNMPv3.

4.1.2.5.4 Pd MySQL

The interface Pd between HA-DB and the HA Select process on MSP-AAA is realized via **SQL commands**, especially the HA Select process uses SQL SELECT queries to retrieve the data from the database. Since the HA Select process will be realised in Java, a HA-DB JDBC interface is used, which is a MySQL compatible API.

4.1.2.5.5 Pe EAP over RADIUS and EAP over 802.1X

The EAP channel is used to perform users' authentication and the bootstrapping of MIPv6 parameters. IEEE 802.1X is used to exchange EAP packets between the MN and the AP, while RADIUS [RFC2138] is used between the AP and the MASA-AAA. The AP (authenticator) is a RADIUS client on the NAS, as depicted on Figure 4-1 which forwards the EAP messages coming from the supplicant (MN) to the AAA server.

4.1.2.5.6 Pf MIPv6 Diameter Application (Authorisation)

The MIPv6 Diameter Application is used only to authorise a user for the mobility service. User authentication is supposed to be already carried out (e.g. EAPoIKEv2).

The two peers are the Home Agent, which acts as Diameter client, and the Home AAA Server (MSA that acts as Diameter server. On the Home Agent, the Diameter Application is triggered by the reception of the first Binding Update message received from a MN: when the BU arrives, the Diameter client asks the MSA server whether the user is authorised for the MIPv6 service or not. If authorisation succeeds, the Home Agent sends a Binding Acknowledgement with status 0

(Binding Update accepted) to the mobile user; otherwise, if authorisation fails, the Home Agent issues a Binding Acknowledgement message with status code 129 (Administratively prohibited).

The Diameter Application requests and answers have the Auth-Request-Type AVP set to AUTHORIZE_ONLY. Some mandatory AVPs have been defined for request messages:

- UserName AVP to carry the user identity (NAIs)
- MN-HomeAddress AVP to inform the AAAH server of the binding Home Address of the user to allow DNS updates (made by AAAH on behalf of HA to fit also scenarios in which HA doesn't have a security association with user's home domain DNS).

Answer messages contain the result code (success or failure) and authorisation AVPs (at least the authorisation lifetime).

4.1.2.5.7 Pg EAP Diameter Application (Authentication)

Pg is a network interface between the HA and the MSA-AAA using the Diameter EAP Application. This interface is used to perform the MN authentication only (since the authorization is initiated by MIPL upon receiving a Binding Update). There are three involved roles: the User (MN OpenIKEv2 EAP client controller), the NAS (HA OpenIKEv2 EAP server controller) and the Server (MSA-AAA, based on opendiameter).

The HA starts the application by sending a Diameter-EAP-Request (DER) message containing an Identity EAP-Response with the MN identity (extracted from the IKEv2 IDi payload). After that, the HA acts as an EAP pass-through, forwarding the EAP packets from the MN to the MSA-AAA and the other way around. The EAP packets between the MN and the HA are transported using EAP payloads into IKE_AUTH messages. The EAP packets between the MN and the MSA-AAA are transported using EAP-Payload AVPs into Diameter-EAP-Request/Diameter-EAP-Response messages.

The EAP authentication may take more than one roundtrip, depending on the used EAP method. In addition, the selected EAP method might generate an EAP-Master-Session-Key as result of a successful authentication. When this key is generated, OpenIKEv2 (in both MN and HA) will use this key to sign the AUTH payload in the last messages of the IKE_AUTH exchange.

4.1.2.5.8 Ph DNS

Ph is a very simple UDP-based interface that implements the DNS protocol. This interface is used by MIPL to obtain the address of the HA via DNS in case the Xsupplicant module does not provide an HA address.

4.1.3 Software modules

4.1.3.1 OpenIKEv2 (MN)

Since the MN has a special behaviour, the standard “openikev2” daemon cannot be used and a new application called “mnikev2” has been developed. This application makes use of the “libopenikev2” and “libopenikev2_impl” libraries. Some additions to these libraries have been implemented to support the proposed scenario:

- A new EAP client controller has been developed in order to support client authentication via Diameter EAP TLS. This client controller uses an initiator EAP state machine from the “libdiameterreap” library of opendiameter 1.0.7-h.
- The XFRM IPsec controller has been improved in order to correctly manage the selectors including IPv6 Mobility Header and to capture the MIGRATE messages from the MIPL kernel part.

This new daemon has the following behaviour:

- When “mnikev2” starts, it keeps waiting on interface **Ma** until a “HoA Request” message is received from MIPL daemon. It takes the MN identity as parameter.
- When a “HoA Request” is received, the needed IKEv2 exchanges are performed in order to authenticate the MN, obtain the HoA and create the needed security associations.
- Once all the IKEv2 exchanges have been performed, “mnikev2” sends a “HoA response” response to the MIPL daemon containing the new assigned address.

4.1.3.2 MIPL (MN)

The MIPL module implements the Mobile Node functionalities. The release used is the MIPL 2.0.2 [MIPL] developed by GO-Core in co-operation with the USAGI/WIDE Project [USAGI].

The original MIPL module requires as input a static configuration file where all the parameters needed to start the mobility service (e.g. Home Agent IPv6 address, Home Address) are stored. Since in our platform these parameters are provided dynamically during the bootstrapping phase, the MIPL module needs to be able to dynamically configure these parameters. In order to allow this dynamical setup, some modifications to the code have been done:

- Implementation of the **Mb** socket interface to exchange messages towards/from the Xsupplicant module.

- A decision algorithm to evaluate the bootstrapping procedure that has to be performed (split or integrated).
- Implementation the **Ma** socket interface for exchanging messages towards/from the OpenIKEv2 module.

Furthermore, the modified MIPL module implements this new logic:

- The MIPL starts reading the configuration file in order to set up some default parameters (e.g. FQDN of the HA to perform the split bootstrapping, IPsec preferences in order to enable/disable the IPsec protection).
- Before starting the movement detection algorithm the MIPL asks for an HA address to the Xsupplicant module.
- If the Xsupplicant module does not provide an HA address, the MIPL module try to obtain an HA address trough a DNS query.
- Once the MIPL module has obtained an HA address, it communicates it to the OpenIKEv2 to trigger the authentication and obtain an HoA.
- When the OpenIKEv2 module returns the HoA, the MIPL performs the movement detection and sends the first BU to HA address using the obtained HoA.

4.1.3.3 Xsupplicant (MN)

The Xsupplicant module realizes the role of IEEE 802.1x supplicant. The distribution used for development has been the v1.0 of the open source Xsupplicant application.

In order to be able to perform the operation required to obtain and process the information required to bootstrap the MIPv6 service, the following modification have been done:

- Implementation of the PEAPv2 module (starting from an already implemented PEAP v0/v1 module).
- Implementation of the code to handle MIPv6 service activation.
- Interface with the MIPL module.

4.1.3.4 NAS (AP)

In the ENABLE implementation there will be two physical devices working as a NAS. Firstly the NAS is in charge of authenticating the users and starting the MIPv6 bootstrapping based on EAP signalling before providing network access.

The first device used will be a Cisco Aironet 1200. The firmware/operating system of this AP does not need any modification to implement the EAP/RADIUS client since this feature is built-in.

The second device used will be a Wireless-G BroadBand Router WRT54G by Linksys. In order for it to perform as a NAS, the firmware of the WRT54G is to be modified. This will enable the NAS to interact successfully with the MN and also with the ASA. The device is based on Linux and the source code for building the firmware is available as GNU [WRTG54G Code].

4.1.3.5 OpenIKEv2 / Diameter-EAP (HA)

In the HA the standard “openikev2” daemon can be used with only some slightly differences:

- A new general configuration option has been added to indicate the location of the NAI_HoAs file.
- A new address configuration method has been implemented to make use of the NAI_HoAs file as seen in the Hc interface in the IKE_AUTH exchange.
- A new EAP server controller has been developed in order to implement the interface **Pg** with the MSA-AAA along the IKE_AUTH exchange.
- The XFRM IPsec controller has been improved in order to correctly manage the selectors including IPv6 Mobility Header and to capture the MIGRATE messages from the MIPL kernel part.
- The standard behaviour when assigning a new address has been modified since there is no need to install any dynamic policy in the SPD (MIPL did it before).
- The OpenIKEv2 HA code uses an EAP pass-through state machine from the “libdiameter-eap” library of opendiameter, as well as an opendiameter authentication session state machine.

4.1.3.6 MIPL (HA)

This MIPL module implements the Home Agent functionalities. The same release used for the MN (MIPL 2.0.2) has been used as a starting point for the development on the HA.

User authorization has required some extensions to the MIPL code:

- Implementation of the **Hd** socket interface for exchanging messages towards/from the OpenDiameter module.
- Parsing of a new configuration file (NAI-HoAs) during the HA start-up phase to initialize the IPsec SPD with the HoA found in the file.
- Implementation of the code to parse the NAI-HoAs file to initialize the IPsec SPD and to obtain the NAI associated to a certain HoA.
- Addition of a new parameter and a new timer to store in the Binding Cache entry the NAI of the user and the associated authentication lifetime.

Furthermore, the modified MIPL module implements this new logic:

- When a Binding Update (BU) with a new HoA (a new user) the MIPL module recover the NAI from the NAI-HoAs file, using as search key the HoA of the user.
- Once the NAI is recovered, MIPL asks to the OpenDiameter server if the user is authorized to use the IPv6 Mobility service.
- If the reply is negative, it sends back to the MN a BA with status 129 (Administratively Prohibited), otherwise it accepts the BU and sends back the BA with status 0 (BU accepted).
- In the new BC entry the HA inserts also the Authorization lifetime and the NAI.
- When the authorization lifetime expires, MIPL asks again the OpenDiameter module to re authorise the user.

4.1.3.7 NETSNMP (HA)

As a SNMP agent on the HA, the NETSNMP package from <http://net-snmp.sourceforge.net/> will be used (net-snmp-5.3.1). The functionality has to be extended by a program that reads the number of home registrations from MIPL and stores it in the respective MIB object REGISTRATIONS (see interface He).

4.1.3.8 OpenDiameter (HA)

This module is used by MIPL to authorise the MIPv6 service. A socket was implemented between MIPL and OpenDiameter to trigger the Diameter Authorization session:

- At the arrival of the first BU from a user.
- To perform a reauthorization of a previously authorized user.

For this socket, we defined also the data structure of the messages (see the Hd Interface in Section 4.1.2.2.2).

Diameter NASREQ Application implemented in OpenDiameter v1.0.7-h has been the starting point for the software development.

Inside Diameter application's state machine, we changed authentication features into authorisation ones, following the original structure and event sequence. For example, the function that prepares the request message has been modified to build a message without authentication AVPs (like User-Password AVP) but with authorization one (UserName AVP). As a consequence, also the part of parsing answer message has been modified accordingly.

4.1.3.9 HA manager (HA-DB manager)

The HA Manager module will be realised in Java and will run on an entity called HA-DB manager. We will use a Debian Gnu/Linux SARGE operating system on this entity, updated with the Linux kernel 2.6.16.20. The kernel has to be patched with the "mipv6-2.0.2-linux-2.6.16"-patch written by Benjamin Thery. As user space code "mipv6-2.0.2" and an appropriate HA configuration file is needed.

For SNMPv3 support, we use NETSNMP from the sourceforge.net webpage (net-snmp-5.3.1) and we add the Westhawk's Java SNMP 5.1 stack to query SNMP information per Java code.

This module used the SNMP-based interface Pc to query the NETSNMP instance on the HAs for dynamic selection parameters and it uses the SQL-based interface Da to communicate with HA-DB (see below).

4.1.3.10 HA-DB (HA-DB manager)

The HA-DB entity is implemented on the same device as the HA Manager (see above). The HA database (HA-DB) is realized in MySQL (Debian package "mysql-5.0" combined with the

“mysql-connector-java-3.1.13” software). It almost works on any operational platform and is distributed as Open Source software. MySQL already supports the standard computer language, SQL to provide data handling. As interface Da to HA-DB JDBC is used, which is a MySQL compatible API to interact with Java applications. The JDBC function InitDB() is used to create the database table structure and to import HA parameters to the database that are needed for the load sharing mechanism. The function is executable in a Java Runtime Environment and extracts line by line a configuration file (mipl-lh.conf). An example of the configuration file is displayed below:

```
# /etc/mipl-lh.conf

# The home agent list:

hip 2001:1b10:1001:2000:0000:0000:0000:0100 mreg 50 mband 10 region 1

hip 2001:1b10:1001:2000:0000:0000:0000:0101 mreg 10 mband 50 region 2

#ENDE
```

Figure 4-13: mipl-lh.conf

Each relevant row, indexed by the string “hip”, determines at least the IP address, and if required also the Max_Reg, the Max_Band and the Region_id value, of an HA that has to be monitored. If a variable parameter is not set, a default value is being imported.

Figure 4-14 outlines the content stored in this HA-DB. The content contains for each HA the selection parameters collected on the HA, the selection parameters already available on the HA-DB as well as additional parameters required for HA load sharing. The database is divided into four different tables. The table SELECTION PARAMETER STATIC contains the parameters available on HA-DB and configured by the administrator. The table SELECTION PARAMETER DYNAMIC contains all dynamic selection parameters collected periodically from the respective HAs. The table ADDITIONAL PARAMETER stores parameters that are not specifically used as selection parameters but are required for performing load sharing. These parameters are only relevant for the HA-DB/Manager and therefore need not be accessed by the MSP-AAA. The last table CACHE PARAMETER contains the last interface values needed to calculate the current average consumed bandwidth.

Each data set within these tables is indexed by the unique HA interface address (IP_HA).

Database tables

SELECTION PARAMETER STATIC		
IP_HA	Region_ID	M_Flag
⋮	⋮	⋮

SELECTION PARAMETER DYNAMIC		
IP_HA	Registrations	Bandwidth
⋮	⋮	⋮

ADDITIONAL PARAMETER		
IP_HA	Max_Reg	Max_Band
⋮	⋮	⋮

CACHE PARAMETER			
IP_HA	OutOctets (t-1)	InOctets (t-1)	Systime (t-1)
⋮	⋮	⋮	⋮

Figure 4-14: Database tables on HA-DB

4.1.3.11 FreeRADIUS (ASA)

The ASA functionality has been implemented with FreeRADIUS which is the premiere open source RADIUS server (released under the GNU General Public License). The server natively ships with libraries for interfacing with LDAP, MySQL, PostgreSQL and Oracle databases and it supports EAP with EAP-MD5, EAP-SIM, EAP-TLS, EAP-TTLS, EAP-PEAP, and Cisco LEAP sub-types.

TI maintains an internal version (forked as of Apr 2004 from the CVS repository) of the program to develop custom extensions, this version has been used as the starting point for the developments in ENABLE. FreeRADIUS is developed in the C language and it is compound by a core module, the radiusd daemon, and a set of modules which are linked dynamically at run time (dlopen): the core is in charge of handle the RADIUS protocol, while all other functions (authentication protocols, interaction with external databases, etc.) are implemented within modules.

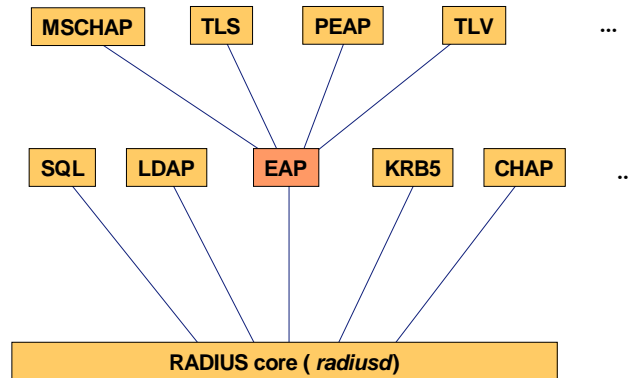


Figure 4-15: FreeRADIUS Software Architecture

Complex functionalities are realised through the usage of a cascade of modules; for example each EAP method is implemented in a separate module. The interface between the RADIUS core and its modules is standardised; the same is true for the EAP module and the modules which implement EAP methods.

In order to implement the required ENABLE functionalities, the PEAP module has been modified to support PEAPv2 and another ad hoc module (TLV) has been created to process the TLVs. Another module linked to the TLV has been implemented for the MIPv6 bootstrapping. Care has been taken in order to realise a modular framework: new services can be bootstrapped adding new modules and using the same interface defined for the interaction between the TLV and MIPv6 module.

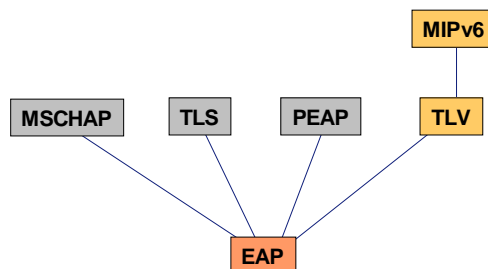


Figure 4-16: MIPv6 bootstrapping module

The SQL module has been also modified to let the MIPv6 module access a SQL database to retrieve users' information for the MIPv6 bootstrapping. Indeed, the original SQL module was only able to perform SQL queries specified in a configuration file and consequently has been modified to perform dynamic queries using as input the SQL requests passed by the MIPv6 module.

4.1.3.12 HA Select (MSA)

The HA Select process is a Java process that runs on MSP (here MSP = MSA). This process periodically queries HA-DB for the current selection parameters via the Pd interface, i.e. it uses JDBC SQL SELECT commands. Upon request, e.g. via the Ac interface, the HA Select process evaluates the load on the HAs, chooses the HA with the lowest load, and returns the IP address of the selected HA. Thereby, it does not take into account HAs that have the maintenance flag (M_Flag) set and, in case the parameter Region_ID is specified, considers only HAs with an appropriate Region_ID. As result, the IP address of the selected HA is returned. More details about the functionality of this process is given in Sections 3.2.3.3 and 3.2.3.4.

4.1.3.13 OpenDiameter (MSA)

All the Open Diameter code used in ENABLE is based on Open Diameter 1.0.7-h. This release contains the following libraries:

- **Framework.** Contains the general state machine framework classes, as well as tasks, jobs, queues and other classes used by the base protocol.
- **Libdiameterparser.** This library contains the classes used to parse diameter messages, including headers and AVPs.
- **Libdiameter.** This is the library that implements the Diameter Base Protocol.
- **Libdiameterenasreq.** Implements the Diameter NAS application. This library is used together with libdiameterereap and libeap to perform the MIPv6 authentication and authorization and to integrate Open Diameter within Openikev2.
- **Libdiameterereap.** Implements the Diameter EAP application.
- **Libeap.** Implements the EAP state machines and EAP messages, using functionality from the OpenSSL library.

Open Diameter contains other libraries (e.g. libpana, libradius, etc.) but they are not used in the implementation of the ENABLE scenarios. Also, instead of using the included daemons (nasd and aaad), custom clients and servers have been developed in order to integrate Open Diameter with OpenIKEv2. The MSA Open Diameter server is currently a custom EAP-based authentication server developed from the libdiameterereap library sample files.

Three different database are deployed:

- Radius DB: used by the FreeRADIUS server to store the users' credential to authorise the network access.
- OpenDiameter IKEv2 DB: used to authenticate the user and to authorise the SA creation between the HA and the MN.
- OpenDiameter MIPv6 DB: used to authorise the MIPv6 service.

The Radius DB is a SQL database which is compound of two main tables:

- access_profiles: it holds the data for network access authorization, plus the information on which services to be bootstrapped.

customer_id	disabled	username	eap_method	credential_type	credential_data	services
00001	no	generic_peap_user	peap	NULL	NULL	NULL
00002	no	generic_tlv_user	tlv	NULL	NULL	NULL
00005	no	zidane	mschapv2	password	coupdeboule	mipv6
00004	no	cassano	mschapv2	password	elgordo	mipv6
00006	no	gattuso	mschapv2	password	ringhio	mipv6

Figure 4-17: access_profiles table

- mipv6_profiles: it holds the data for bootstrapping the MIPv6 service.

username	auth_type	psk_key_length	keys_lifetime
zidane	EAP-IKEv2	NULL	NULL
cassano	PSK-IKEv2	128	NULL
gattuso	rfc4285	NULL	NULL

Figure 4-18: mipv6_profiles table

Furthermore, FreeRADIUS SQL queries requires the presence of an additional table for each main table containing the name of the attribute to be transmitted and an operator (always “:=”).

attr_name	operator
eap_method	:=
credential_type	:=
credential_data	:=
services	:=

Figure 4-19: access_profiles_attributes table

attr_name	operator
auth_type	:=
psk_key_length	:=
keys_lifetime	:=

Figure 4-20: mipv6_profiles_attributes table

The OpenDiameter MIPv6 DB and OpenDiameter IKEv2 DB are implemented through the same XML file containing users' profile for MIPv6 authentication and authorization. This file is named "aaa_user_db.xml" and it is compliant with XML version 1.0 W3C recommendation [W3C]. An XSD file is associated to that file, named "aaa_user_db.xsd", containing the schema of the XML file.

Below there is an example of user DB XML file including only the sub elements used by the authorization application since the entries for EAP authentication are the same ones used by the standard OpenDiameter EAP application.

aaa_user_db.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<user_db xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation='aaa_user_db.xsd'>

  <user_entry>
    <name_match>luca.battistoni@telecomitalia.it</name_match>
    <authz_lifetime>60</authz_lifetime>
  </user_entry>

  <user_entry>
    <name_match>michele.lamonaca@telecomitalia.it</name_match>
    <authz_lifetime>100</authz_lifetime>
  </user_entry>

</user_db>
```

Figure 4-21: Example of user DB XML file

The main element (users' database) is bounded by tags <user_db> and </user_db>. The users DB is divided into sub elements, one for each the user's entry (<user_entry> ... </user_entry>). The user entry contains two sub elements:

- name_match: this element contains the NAI of the user.
- authz_lifetime: this element stores the lifetime associated to the user authorization, the lifetime is expressed in second.

The XSD file associated to the XML user DB file is shown below.

aaa_user_db.xsd

```
<?xml version="1.0" encoding="UTF-8" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">

  <xs:complexType name="user_entry_type">
    <xs:sequence>
      <xs:element name="name_match" type="xs:string" />
      <xs:element name="authz_lifetime" type="xs:integer" />
    </xs:sequence>
  </xs:complexType>

  <xs:element name="user_db">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="user_entry"
          type="user_entry_type"
          minOccurs="0" maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

</xs:schema>
```

Figure 4-22: Example of XML schema file

4.2 Additional ongoing developments

4.2.1 Firewall traversal based on NSIS

4.2.1.1 MN firewall traversal process

In this section, an overview of the functionality and the message flow for the Mobile Node firewall traversal process without regard to AAA-server interaction will be given. Figure 4-23 and Figure 4-24 also depict this message flow sequence.

The firewall traversal process and the message flow for creating pinholes as following:

- When the MN moves to a new network and wants to perform a Binding Update, the MIPL implementation is halted and the local firewall-traversal-application is triggered.
- The firewall-traversal-application at the MN triggers the local NATFW NSLP to install pinholes for the Binding Update message.
- The NATFW NSLP sends a CREATE message (src: CoA, dest: HA; SPIx, for IPsec ESP in transport mode) to create pinholes for the BU message. This pinhole also allows the BA from the HA the MN.
 - The ASP-firewall intercepts this message, processes it, checks authentication and authorization and forwards it to towards the HA.

- The MSP-firewall intercepts this message, processes it, checks authentication and authorization and forwards it to the HA.
- The Home Agent checks authentication and authorization and response to the CREATE message with a RESPONSE message.
- The MSP-firewall processes the message, opens the requested pinholes and forwards the RESPONSE to MN.
- The ASP-firewall also processes the RESPONSE, opens the requested pinholes and forwards the message to the MN.
- When the RESPONSE successfully reaches the MN, the pinholes between MN and HA are established.
- NATFW NSLP informs the firewall-traversal-application about successful pinhole creating.
- At this point, the MN is able to perform the Binding Update to the Home Agent. For the return routability test (RRT) the firewall-traversal-application needs to open additional pinholes. Therefore, the next two steps, the resumption of the MIPL application and the signalling for further pinholes (for HoTI, HoT and CoTI) can be performed at the same time.
- The firewall-traversal-application resumes the MIPL implementation.
 - MIPv6 sends the binding update to the Home Agent.
- The firewall-traversal-application triggers the NATFW NSLP (trigger 1-2).
 - The NATFW NSLP sends CREATE messages for the HoTI (src: CoA, dest: HA; SPIx, for IPsec ESP in tunnel mode) and for the CoTI (src: CoA, dest: CN) to create pinholes. This pinhole creation allows the HoTI and HoT to pass between MN and HA, but not between HA and CN. This pinhole creation falls on the responsibility of the HA (compare section 4.2.1.2). Additionally, the pinhole creation above allows CoTI and CoT and also the BU from MN to CN and BA from CN to MN.
 - The ASP-firewall intercepts this message, processes it, checks authentication and authorization and forwards it toward HA.

- The MSP-firewall intercepts this message, processes it, checks authentication and authorization and forwards it to the HA.
- The Home Agent checks authentication and authorization and response to the CREATE message with a RESPONSE message.
- The MSP-firewall processes the message, opens pinholes and forwards the RESPONSE to MN.
- The ASP-firewall also processes the RESPONSE, opens pinholes and forwards the message to the MN.
- When the RESPONSE successfully reaches the MN, the pinholes between MN and HA are established.
- Note that this signalling has to be performed multiple times, namely for:
 - HoTI messages..
 - CoTI messages.
 - and after reception of HoT and CoT the firewall-traversal-application triggers the NATFW NSLP (trigger 3+4). Trigger 3 starts the pinhole creation for data traffic from MN to CN (src: CoA, dest: CN). This avoid changes of the MIPL implementation at the CN.
- The NATFW NSLP informs the firewall-traversal-application about successful pinhole creation.
- If the pinholes for the HoTI and HoT messages are installed at this time, the firewall-traversal-application resumes the MIPL implementation, which is now able to send the HoTI message. If the pinholes for CoTI are installed, the firewall-traversal-application resumes the MIPL implementation which can now send the CoTI message.
- The NATFW NSLP sends an EXT message to the HA-address to install a pinhole for the HoT message.
 - The ASP-firewall processes this message, checks authentication and authorization and sends a RESPONSE to MN.

- When the RESPONSE reaches the MN, the pinholes at the ASP-firewall are established.
- The NATFW NSLP informs the application about successful pinhole creation.
- Note that this signalling has to be performed several times for:
 - HoT messages.
 - after receiving HoT and CoT, for data traffic from HA to MN.
 - and after receiving HoT and CoT, for data traffic from CN to MN.
- The NATFW NSLP informs the firewall-traversal-application about successful pinhole creating.
- Now, all required pinholes have been created and the MIPL implementation already has performed return routability test and is able to use all modes of communication in Mobile IPv6. The different types of data traffic between MN, HA and CN should also be **enabled**.

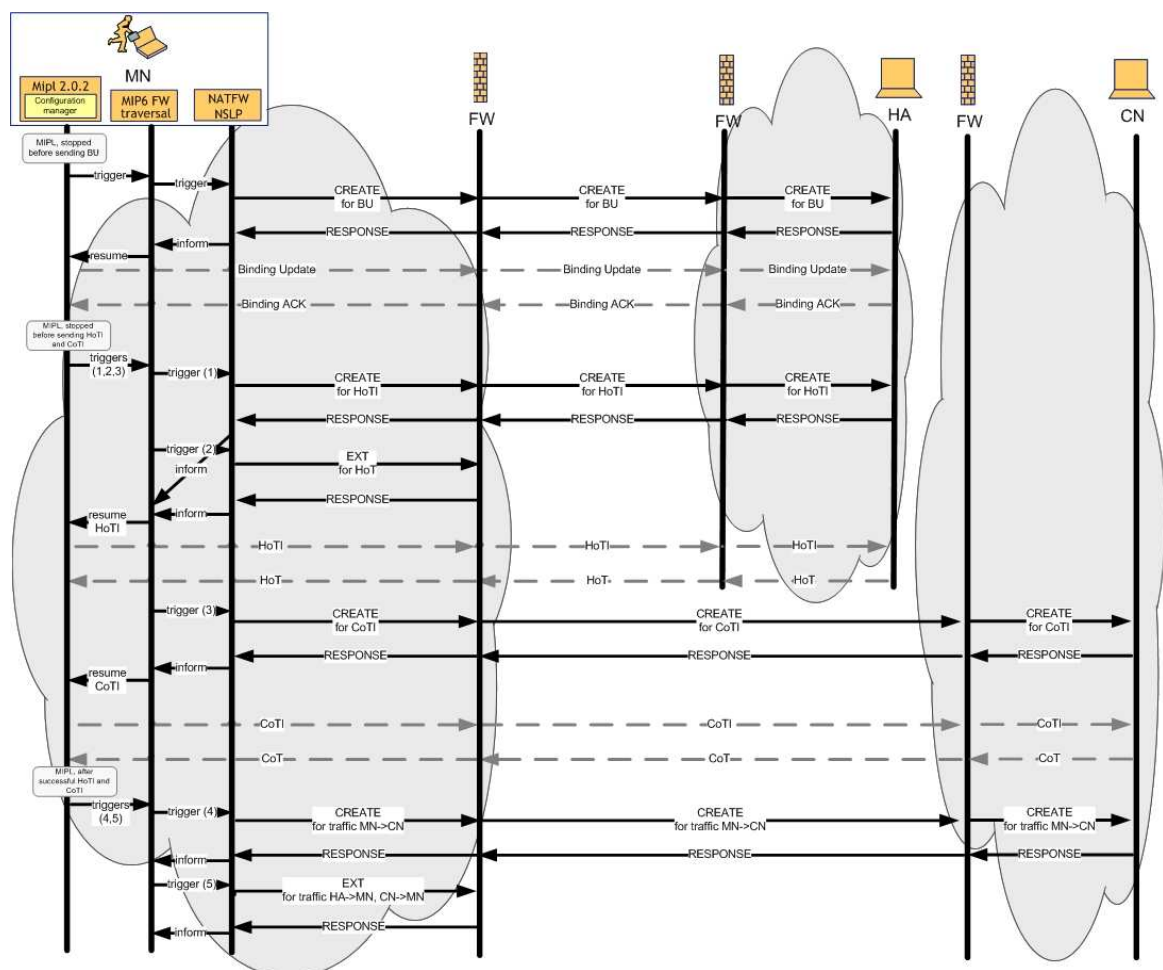


Figure 4-23: MN firewall traversal message sequence

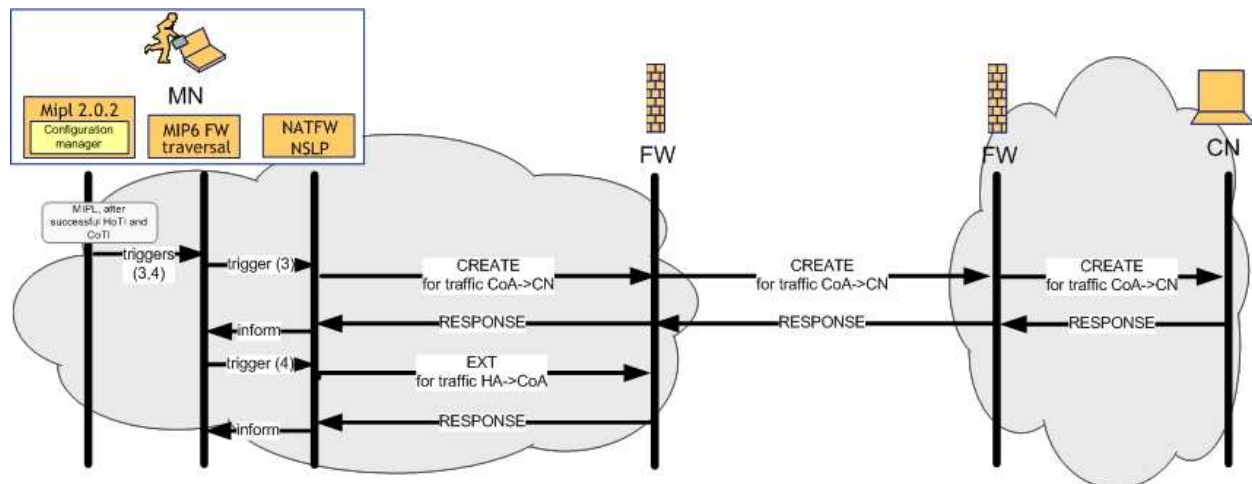


Figure 4-24: MN firewall traversal message sequence for data traffic

4.2.1.2 HA firewall traversal process

In this section, we give an overview about the functionality and the message flow of the Home Agent firewall traversal process without regard for the AAA-server interaction. Figure 4-25 also depicts the message flow sequence.

The firewall traversal process and the message flow for creating pinholes as following:

- When the MN moves to a new network it performs the Binding Update. The MIPL implementation at the HA will be changed accordingly to trigger the local firewall-traversal-application and hand over the required parameters when it receives a Binding Update.
- The local firewall-traversal-application triggers the local NATFW NSLP.
- The NATFW NSLP sends a CREATE message (src: HoA, dest: CN) to create pinholes for the HoTI between HA and CN. This pinhole also allows the HoT to traverse the firewalls between HA and CN.
 - The MSP-firewall intercepts this message, processes it, checks authentication and authorization and forwards it toward CN.
 - The ASP-firewall intercepts this message, processes it, checks authentication and authorization and forwards it to the CN.

- The Correspondent Node checks authentication and authorization and response to the CREATE message with a RESPONSE message.
- The ASP-firewall processes the message, open pinholes and forwards the RESPONSE to HA.
- The MSP-firewall also processes the RESPONSE, open pinholes and forwards the message to the HA.
- When the RESPONSE successfully reaches the HA, the pinholes between HA and CN are established.
- Note that this signalling has to be performed for:
 - HoTI messages.
 - and for data traffic from HA to CN.
- The NATFW NSLP informs the firewall-traversal-application about successful pinhole creation.
- The NATFW NSLP sends an EXT message (src: CN, dest: HoA) to the CN-address to allow data traffic from the CN to HoA.
 - The MSP-firewall processes this message, checks authentication and authorization and sends a RESPONSE to HoA.
 - When the RESPONSE reaches the HA, the pinholes at the MSP-firewall for data traffic from the CN are established.
 - Note that this signalling has to be performed for:
 - data traffic from the CN to HoA (src: CN, dest: HoA),
 - and data traffic from the MN to HA (src: CoA, dest: HA).
 - NATFW NSLP informs application about successful pinhole creating.
- If we want to avoid changes of the MIPL implementation at the CN we can also let the HA open pinholes on behave of the CN. This approach is more easier and avoid changes of the MIPL implementation on the CN, but might not be perfect from authentication and authorization point of view.
- The firewall-traversal-application triggers the NATFW NSLP.

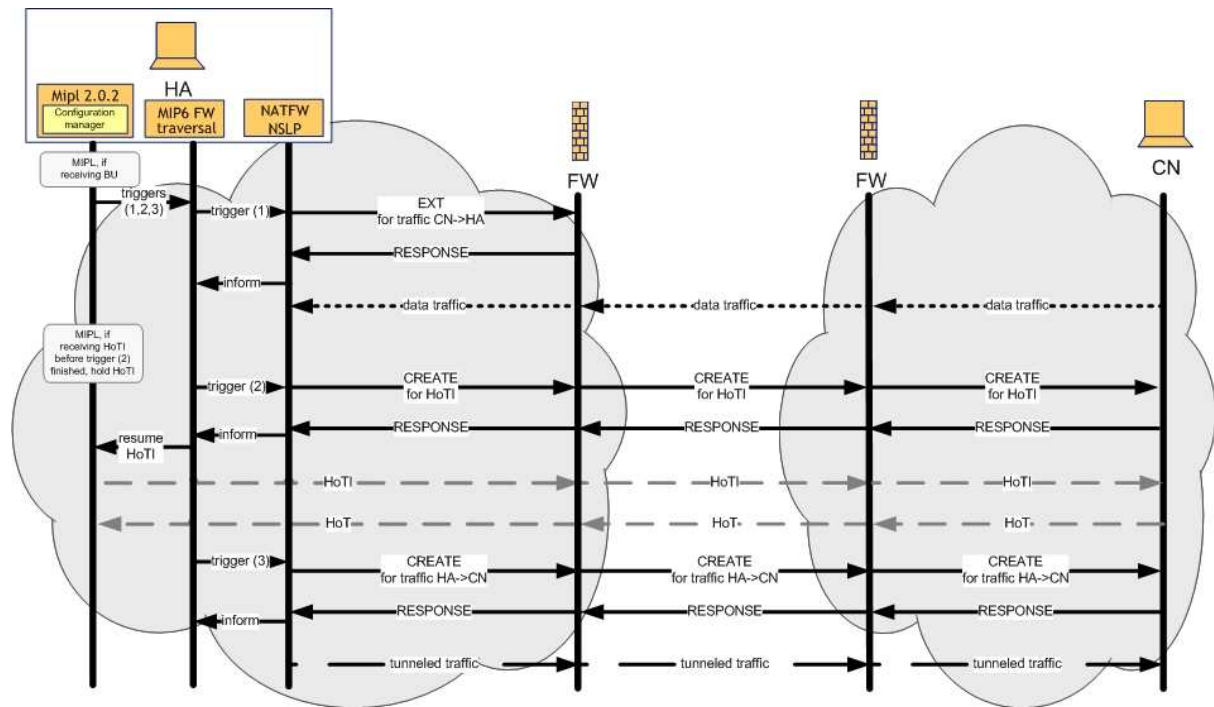


Figure 4-25: HA firewall traversal message sequence

4.2.1.3 CN firewall traversal process

In this section, we give an overview about the functionality and the message flow to and from the CN. The firewall traversal process for the CN could be either done by the HA or by the CN itself. If the responsibility falls on the CN, the procedure requires more intrusive changes into the MIPL implementation but authentication and authorization aspects are simpler. Figure 4-26 depicts this message flow. If the HA is taking care of the signalling, the CN does not need to run a local firewall-traversal-application and no modifications to the MIPL implementation running on the CN necessary. The following shows the signalling if the responsibility falls to the CN. However, as we plan to let MN and HA taking care of the signalling for the pinholes in the CN's ASP-FW, that is not how it would be implemented. It is only shown here for the sake of completeness.

The firewall traversal process and the message flow for creating pinholes (provided the CN is responsible for creating pinholes) is as follows:

- When the MN moves to a new network it performs the Binding Update and later starts to perform the return routability test. At this point, a HoTI is send to the CN via the HA and a CoTI is send directly to the CN. The pinholes for this message to the HA and directly to

the CN are already signalled but the message from the HA to the CN will be blocked by the firewall.

- Therefore, the MIPL implementation at the CN will be changed that it triggers the local firewall-traversal-application to create pinholes for this message in the event of receiving a CoTI message from the MN.
- The NATFW NSLP sends an EXT message (src: HoA, dest: CN) to allow the HoTI message from the HoA to the CN.
 - The ASP-firewall processes this message, checks authentication and authorization and sends a RESPONSE to CN.
 - When the RESPONSE reaches the CN, the pinholes at the MSP-firewall are established.
 - Note that this signalling has to be done several times for:
 - HoTI messages.
 - for data traffic from HA to CN
 - NATFW NSLP informs application about successful pinhole creating.
- Since the HoTI and CoTI can be sent at the same time, it might be possible that the HoTI reach the ASP-firewall before the pinholes are open.
- Hence, the MIPL implementation at the HA must be changed in such a way, that the Home Agent waits a short time before forwarding the HoTI toward the CN.

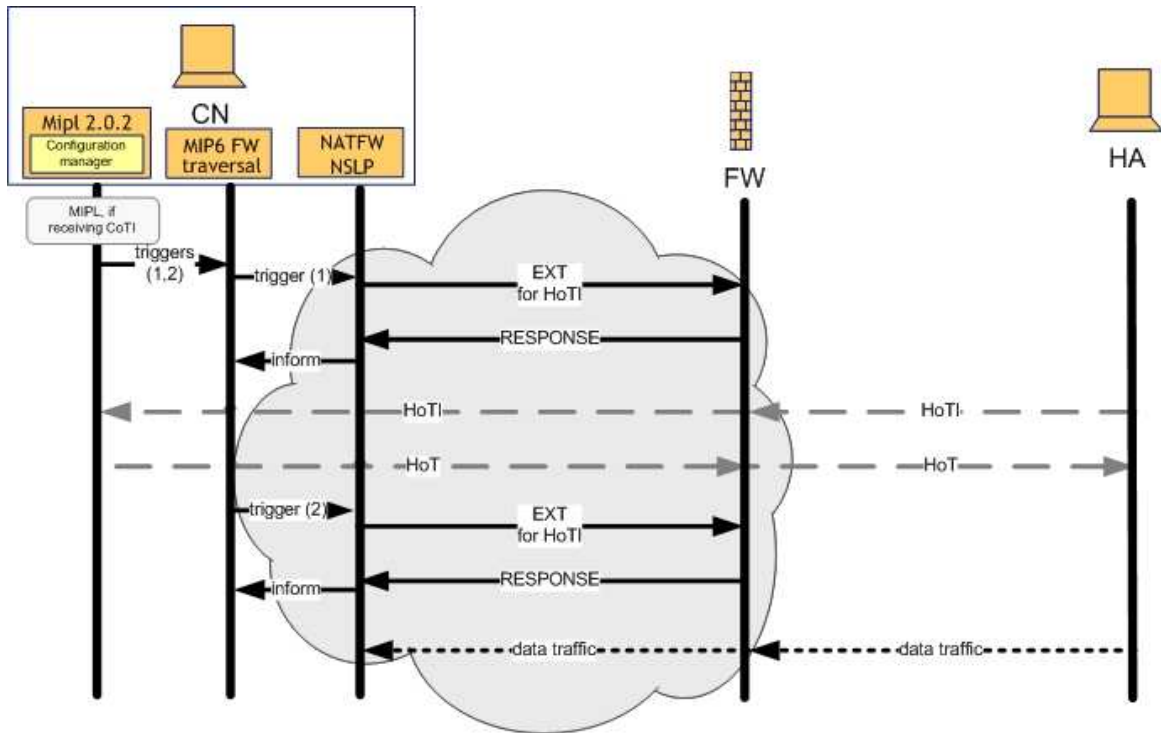


Figure 4-26: CN firewall traversal message sequence

Table 4-1 gives an overview about the required modules and their changes to other software.

Table 4-1: Required modules and software changes for Mobile IPv6 firewall traversal

Network element	Software module	Changes
MN	MIPL	to be modified
	NSIS	needed
	NATFW NSLP	needed
	MIP6 Firewall traversal application	to be developed
HA	MIPL	to be modified
	NSIS	needed
	NATFW NSLP	needed
	MIP6 Firewall traversal application	to be developed
CN	MIPL	needed / (to be modified)
	NSIS	needed
	NATFW NSLP	needed
	MIP6 Firewall traversal application	- / (to be developed)

FWs	NSIS	needed
	NATFW NSLP	needed
	MIP6 Firewall traversal application	-
	Diameter/FreeRadius Client	needed

4.2.1.4 Implementation Overview

4.2.1.4.1 NTLP Implementation Overview

The University of Göttingen has implemented the GIST protocol in C++, using Linux 2.6 kernel. The implementation is fully conformant to the GIST protocol and it's API [NSIS-NTLP], except for some open issues like NAT, tunnelling and detailed mobility support. The code is publicly available in [<http://user.informatik.uni-goettingen.de/~nsis>].

The implementation architecture is shown in Figure 4-27. It is currently based on a single process approach using a main event loop based on [XORP] library, which is used to implement socket maintenance and callbacks as well as timer callbacks. This design has no additional overhead for maintaining and synchronizing multiple threads, which results in a high throughput and a rather simple implementation.

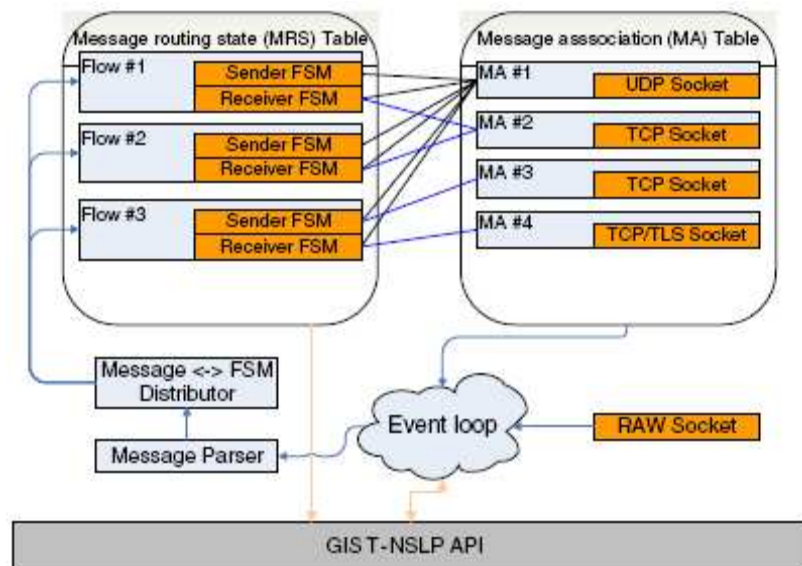


Figure 4-27: Implementation Architecture

Besides the event loop, a key component in GIST implementation is state management. In order to support tens of thousands of signalling sessions efficiently, we used a hash table to manage the MRSs, associated with linked lists to resolve conflicts. A standard lookup takes constant time, however in the worst case, all table entries would be compared to find a given MRS.

To search the MRS table, one needs to know the associated key information, namely the session ID, the NSLP ID and message routing information (MRI). This is nevertheless subject to some limitations, e.g., it is not possible to search for all MRSs using a specific MRI. Such a search feature may be useful to find MRSs that are affected by a detected link failure. A possible solution is to maintain specialized hash tables for link failures, which would allow for quick searches. However, this approach would add maintenance overhead to every MRS table (which usually comprise a number of tables) operation.

In addition to managing MRSs, a GIST implementation has to manage MAs for C-mode operations. If two peers already have an MA and a new session is being established on the same path, the MA should be reused to minimize resource usage. This feature implies that there should be a way to search the MA table for an MA that can be reused for a certain session. Our implementation uses a second hash table to accomplish that goal. The upstream peer information (PI) serves as the key information. The UDP socket is treated as a "virtual" MA for the convenience of unifying the socket interface module.

Another important component of the GIST implementation is the finite state machine (FSM) to maintain states for each session. The GIST finite state machine [GIST FSM] is implemented based on a combination of the XORP timer class and an FSM framework that was originally written for the Linux ISDN device driver.

Every MRS is associated with two FSMs, one for the upstream peer and another one for the downstream peer. There is no need for a global table of FSMs, because every MRS provides pointers to the associated FSMs. In addition, every MA has a list of FSMs which it is associated with, so that the state machines can be informed e.g., when a loss of connectivity with its current peer takes place.

4.2.1.4.2 NAT/FW NSLP Implementation Overview

The NAT/FW NSLP daemon is implemented in userspace using C++. The code builds upon a GIST daemon that was developed at the University of Göttingen, both implementations are freely available in a single release [NSIS_IMPL] for Linux. GIST daemon offers an API for NSLPs to use its generic transport services via UNIX sockets. NAT/FW NSLP daemon itself also offers an API to upper layers to allow applications to trigger signaling flows, such as accepting inbound connections at an edge firewall. As depicted in Figure 4-28, the implementation consists of six main parts:

- server core connecting to GIST-API and delegating callbacks to the other components.
- NAT/FW engine API.

- protocol behaviour defined in a finite state machine.
- message parsing and construction.
- security policy table and
- APIs to firewall and NAT.
- NAT/FW engines, which enforce the NAT bindings and FW policy rules for corresponding data traffic, are connected to the NAT/FW NSLP daemon via the NAT/FW engine-API.

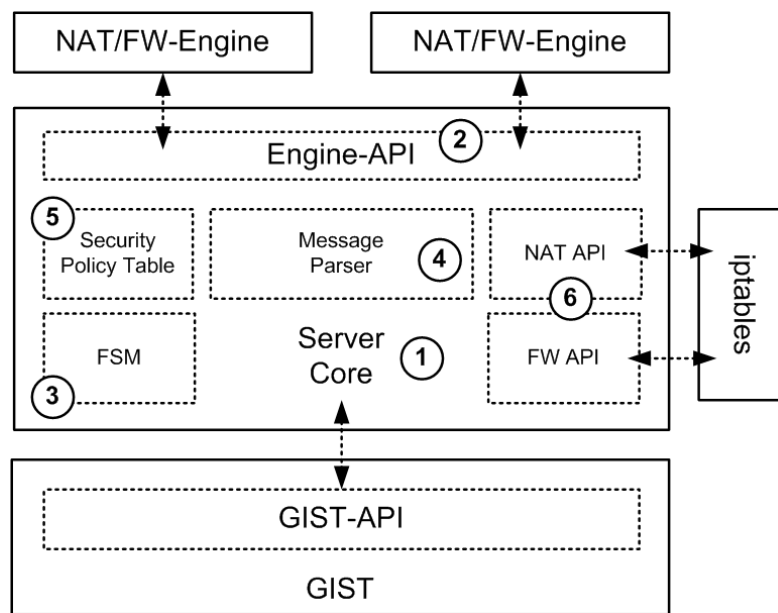


Figure 4-28: NAT/FW NSLP Architecture

We chose to use Linux kernel netfilter [NETFILTER] module and its iptables front end as NAT and firewall, because of its availability and complete coverage of needed features. The use of the low-level iptables API libiptc is still discouraged by its developers because lack of robustness and missing documentation. To avoid problems and incompatibility with different iptables versions we chose to use system() call to invoke an iptables process with according parameters although this approach is known to be inefficient. NAT/FW NSLP imposes only a small set of requirements on the used firewall and NAT, as NAT/FW NSLP supports only adds a small subset of functionality to any possible firewall or NAT implementation and thus replacing the currently used firewall and NAT can be done easily.

It was shown during GIST development that having an efficient finite state machine in source code that represents similar sets of states, transitions and actions as in state machine specification simplifies the understanding of the code without sacrificing performance. A C++ template was

written to allow reusability among GIST and NSLP daemon development, enabling a mapping between the definition of a finite state machine, including states, transitions and actions to corresponding variables, function pointers and executable code.

The NAT/FW NSLP state machine [NAT/FW_NSLP] lists three possible initial states, a host being in an initiator, a forwarder or a receiver idle state. The decision whether a message has to be forwarded or delivered can not be made solely on the destination address in GIST Message Routing Information (MRI) as in NAT case it is being rewritten, similarly as IP headers in NATs. Moreover, it depends on the current NAT configuration (or alternatively, often called reservation) status at the host where a message is received. The idea of the state machine giving an high-level overview for protocol understanding misses some aspects, such as locator rewrite and reservation dependency, that were fundamental aspects during implementation. Incoming message are processed by GIST and delivered to an NSLP if the NSLP is supported on that node. The NSLP decides whether to accept the message or forward it. In the NAT/FW NSLP there are messages that are meaningful either just for NAT or just for firewall. The daemon configuration allows to set flags whether a NAT/FW NSLP host is running a firewall, a NAT or both. After a message is accepted, basic validity checks are performed and the daemon will try to associate an existing state with the incoming message based on the session ID carried in NSLP payload. If there is no state installed yet, a new state machine object with a new session ID needs to be created. As mentioned above, it must be distinguished whether the host is the NR of the signalling path or whether it is a NF. Depending on the initial state, the protocol behaviour for this session is different. In contrast, if the action is triggered via API the host is always the NI and a new state machine object with a new session ID is created. Now, as the state machine object is created and the initial state is determined, the transition is applied on it. Transitions are modelled as a set of states, events and function pointers on the state machine object. According to a given state and an incoming event, a function is called. This keeps the function call overhead very small. The function bodies contain all relevant code that defines the protocol behaviour, such as state manipulation, NAT and firewall interaction, message parsing and construction.

4.2.1.4.3 NSIS and NAT/FW NSLP for Mobile IPv6 firewall traversal - Implementation Overview

The firewall-traversal-application which enables NSIS and NAT/FW NSLP for Mobile IPv6 firewall traversal will be implemented in userspace using C++. The code builds upon the University of Göttingen GIST and NAT/FW NSLP daemons for Linux. NAT/FW NSLP daemon itself offers an API via UNIX sockets to upper layers to allow applications to trigger signalling flows. The firewall-traversal-application will use this API to trigger the creation for the required firewall pinholes. Besides this, an inter-process communication with the MIPL implementation will be implemented, also using UNIX sockets. The inter-process communication, the

modifications to the MIPL implementation and the detailed message flow for the different Mobile IPv6 entities is described in the sections 4.2.1.6 – 4.2.1.9.

4.2.2 Architecture of FMIP6

The goal of the first stage, also the current stage of FMIP6 system prototyping is to provide an implementation of the FMIPv6 protocol fully compliant with RFC4068 and thus to allow improving the handover latency in MIPv6. The implementation of FMIPv6 protocol could be thought as an add-on feature that could be integrated with the 'Integrated Software Architecture' explained in section 4.1.

At the second stage prototyping that will start in 2007, a system which can demonstrate the FMIP6 and GSABA optimized mobility service (designed and studied in WP4) will be developed.

The software architecture of the initial FMIPv6 prototype contains the followings components: - The PAR, NAR, MN, HA and CN. The architecture along with the required interfaces are given in the following diagram.

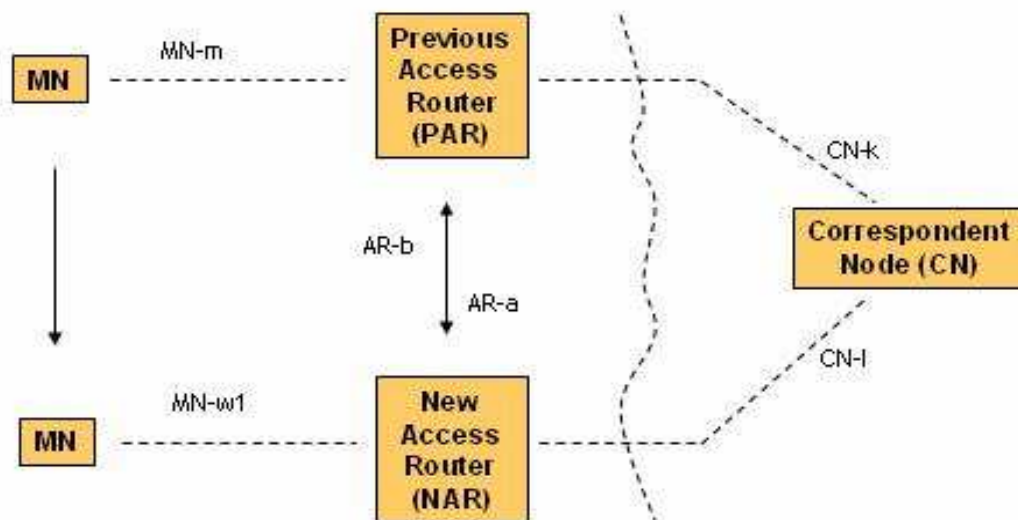


Figure 4-29: FMIPv6 Reference Architecture

The involved interfaces are described here below:

- MN-m: This is the interface between the MN and PAR. The interface has to be built on top the MIPL2.0.2 implementation in order allow FMIPv6 signalling to take place. The interface would allow the MN and PAR to exchange the signalling messages during the 'anticipation' phase in FMIPv6. The signalling would consist of RtSolPr, PrRtAdv and

FBU messages. The MIPL 2.0.2 have to be modified to act as a FMIPv6 MN daemon in this case.

- AR-a: Through this interface the PAR would sent the Handover Initiate (HI), message to the NAR. This message allows the tunnel between the PAR-NAR to be set. It allows checks for the DAD for the nCoA in the NAR's link.
- AR-b: The Handover Acknowledgement (HACK) message would be sent through this interface. The MIPL in the both the AR's have to be modified to act as an FMIPv6 AR daemon.
- MN- w1: Once the MN would connect to the NAR's link, The MN would send the FNA message to declare it's presence in the new link. During the Predictive mode of operation, the MN would be allowed to immediately use the new link to transmit and receive packets. If operated in the reactive mode, the FBU would be encapsulated in the FNA message.
- CN-k: The Interface between the CN and PAR.
- CN-I: The Interface between the CN and NAR.

4.2.2.1 Network Topology of the Test-bed

Given below in Figure 4-30 is the topological view of the FMIPv6 test bed we plan to implement. Note that the authorization aspects of the FMIPv6 such BAA, BAA proxy and BCA functionalities are not included in the first stage of prototyping.

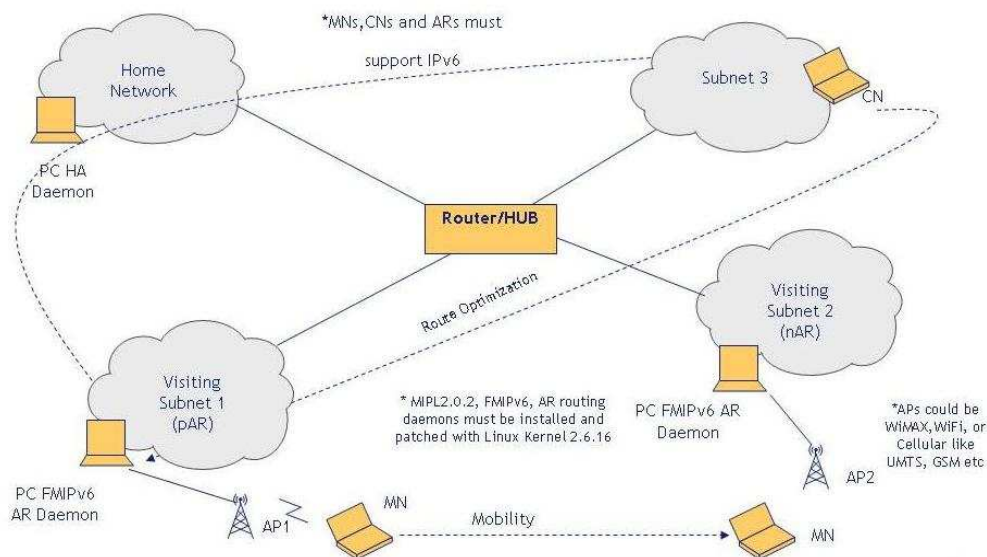


Figure 4-30: Topological view of the FMIPv6 test bed

In the Figure 4-30 above, the Home Agent is a PC acting as a router in the MN's Home Network. It has four interfaces:

- the uplink, in the above scenario it could be called eth0.
- one link connected to the backbone, in this context it could be called eth1.
- an interface, eth2, used by the Corresponding Node (CN) to connect.
- and an interface, which could be set as the home network. This could be called eth3.

Both ARs are connected to the backbone via the eth0 interfaces and provides wireless connectivity through eth1 which is either a wireless card or a wired interface which is connected to an 802.11 dedicated Access Point. For both, eth2 and eth3 on the Home Agent, the same is also valid.

4.2.2.2 High level of the FMIPv6 Implementation process

The high level view of the FMIPv6 process is the following:

- The MN bootstraps and responds to the (optional) authentication request from the (commercial) AP1 and AR1, obtains HA/HoA/CoA information, etc.
- The MN executes mobile IPv6 binding update with HA.
- The MN moves to the overlapped coverage of both AP1's and AP2's, the later one is connected to AR2.
- The MN detects the lower layer L2 (e.g. triggers for handover initialization and preparation from AP1 to AP2, then launches the fmipv6 process.
- Interacting with AR, the MN executes seamless predictive/reactive handover executes.
- The MN executes mobile IPv6 binding update with HA.

The operational flow of the FMIPv6 protocol is given below in the following diagram.

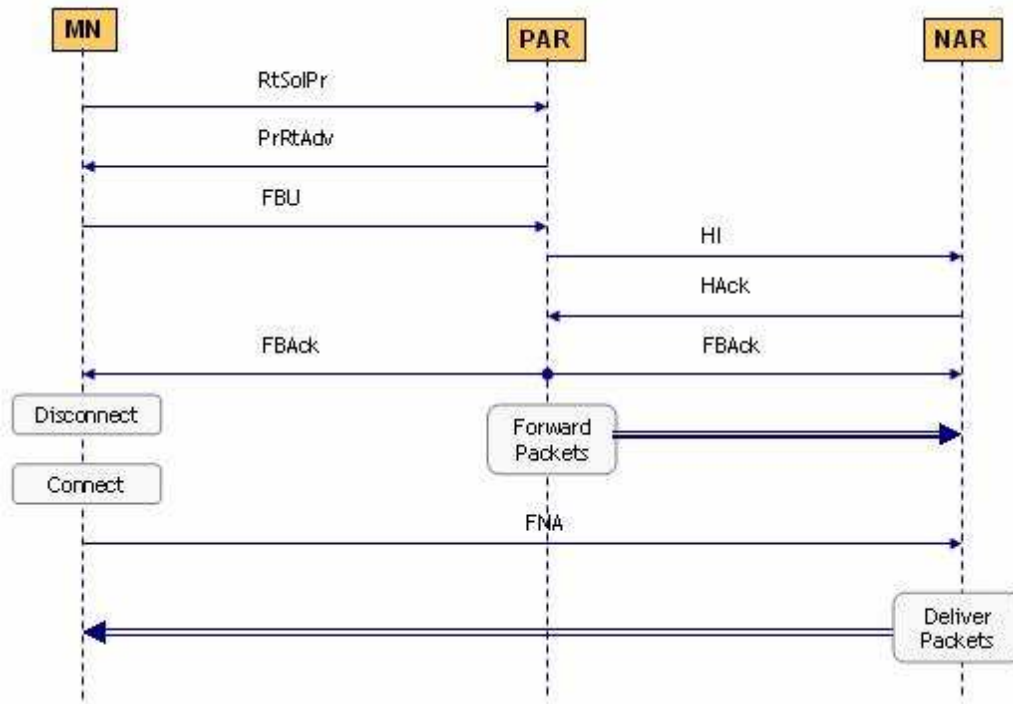


Figure 4-31: Predictive Mode

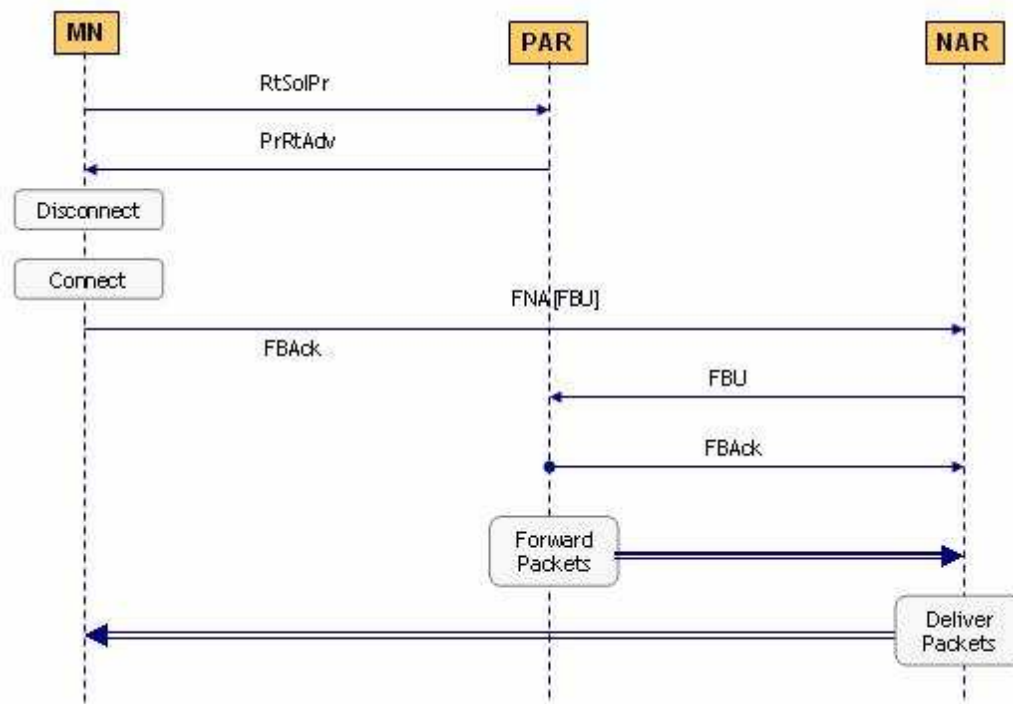


Figure 4-32: Reactive Mode

4.2.2.2.1 Functionalities of the ARs

In the FMIPv6 protocol, there are two types of HOs: MN initiated HO and Network initiated HO. To the MN initiated HO, it includes the predictive mode and reactive mode. In this section, we just describe the AR functions of the MN initiated HO.

The following are the Finite State Machines of pAR and nAR.

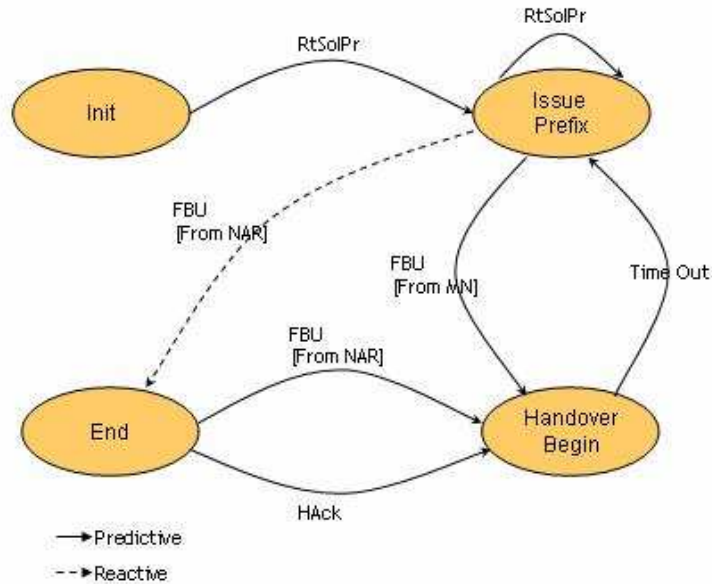


Figure 4-33: pAR Finite State Machine

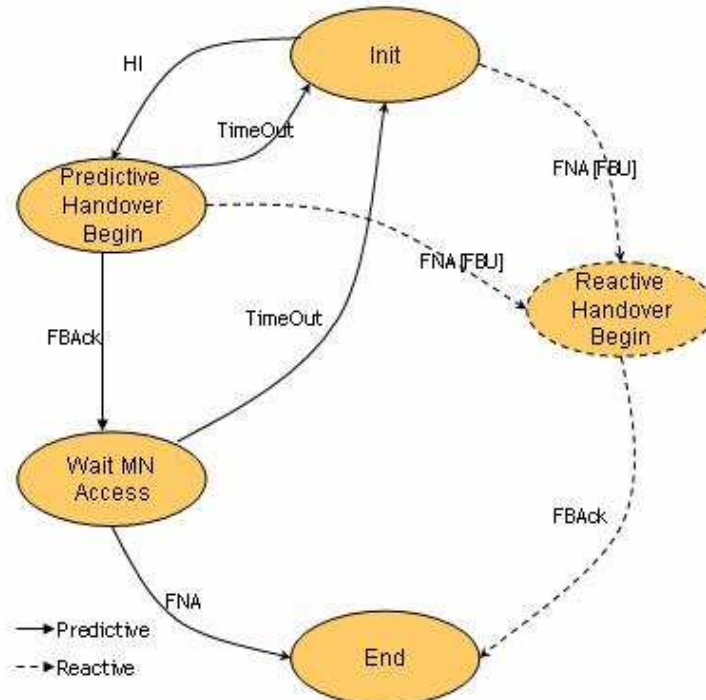


Figure 4-34: nAR Finite State Machine

Detailed functions of pAR and nAR are shown as follows.

- pAR provides the relative information of neighbour networks to MN.
 - At the Init state, the pAR receives the RtSolPr message from the MN. According to the information of the RtSolPr, the pAR sends PrRtAdv message containing one or more [AP-ID, AR-Info] tuples in response. The pAR state changes from Init state to Issue Prefix state.
- pAR launches the HO for MN.
 - For the predictive HO mode, at the Issue Prefix state, when pAR receives the FBU message from MN, it will get the nCoA from the FBU and send the HI message to nAR to determine the validity of the nCoA. To determine the nAR's address for the HI message, the pAR can perform the longest prefix match of nCoA (in FBU) with the prefix list of neighbouring access routers. The pAR state changes from Issue Prefix state to Predictive Handover Begin state. Also, at the Issue Prefix state, the pAR may receive the FBU message which comes from nAR link, and then the HO will enter the reactive mode.
- nAR prepares for the HO of MN.
 - At the Init state, when nAR receives the HI message, it will validate the nCoA included in the HI and construct the HAck message which will be sent to pAR. nAR state changes from Init state to Predictive Handover Begin state.
- pAR confirms the HO of MN.
 - At the Predictive Handover state, the pAR receives the HAck from nAR and establishes the binding between pCoA and nCoA and sends the FBAck message. If a HAck message is not received as a response in a short time period (no less than twice the typical RTT between source and destination, or 100 milliseconds if RTT is not known), the HI SHOULD be resent. Subsequent retransmissions can be up to HI_RETRIES, but MUST use exponential backoff in which the timeout period (i.e., 2xRTT or 100 milliseconds) is doubled during each instance of retransmission. If there are some abnormal reasons, the pAR state will go back to the Issue Prefix state.
- nAR handles the MN attachment.
 - The nAR receives the FNA from MN and do the relative processes. If there is the encapsulated FBU in the FNA, nAR should pick-up the encapsulated FBU and

deliver it to the pAR. If the nCoA corresponding to the FNA message is not acceptable, the nAR must discard the inner (FBU) packet and send a RA with NAACK option in which it may include an alternate nCoA for use.

- pAR handles the FBU from nAR link.
 - In reactive HO mode, pAR may receive the FBU from nAR link. The pAR processes the message and responses the FBAck.

4.2.2.2.2 The Functions of the MN

- Mobile Node Required Functionalities (MN -> PAR (BT)).
 - SP specific Functions.
 - L2 Trigger Function.
 - Wait for L2 triggers or RAs (Function would wait for interrupts). When the signal strength of the MN wireless card detects that the signal with the attached AP is going down or below a certain acceptable threshold, then FMIPv6 protocol would be notified of an imminent Handover (i.e. start L2 handover and anticipate for the L3 Handover).
 - RtSolPr Function.
 - After L2 triggers or RAs, a Function will implemented to immediately send RtSolPr.
 - Binding Update Function /FBU Function.
 - Function to wait for PrRtAdv, and once received, sent FBU (i.e Predictive Mode). For the reactive this a Function will be implemented so that the FBU is encapsulated in the FNA message and sent on the nAR's link.
 - FNA Function.
 - Function to wait for FBack, and once received with error code, close FMIPv6; if success, it is predictive mode, otherwise, it is reactive mode if the MN does not receive the FBack before it leaves the pAR.

- Authorization Function : (PAR (BT) -> (BCA AAA Proxy).
- Function to Service Information Request (Service ID etc) to BCA (AAA proxy)_ and wait Service Information response from BT.

4.2.2.2.3 MN Process Flow

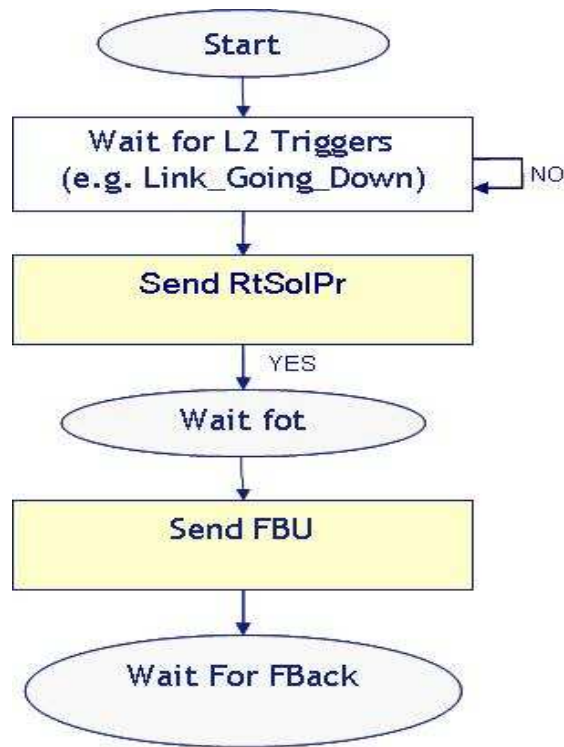


Figure 4-35: Predictive Mode

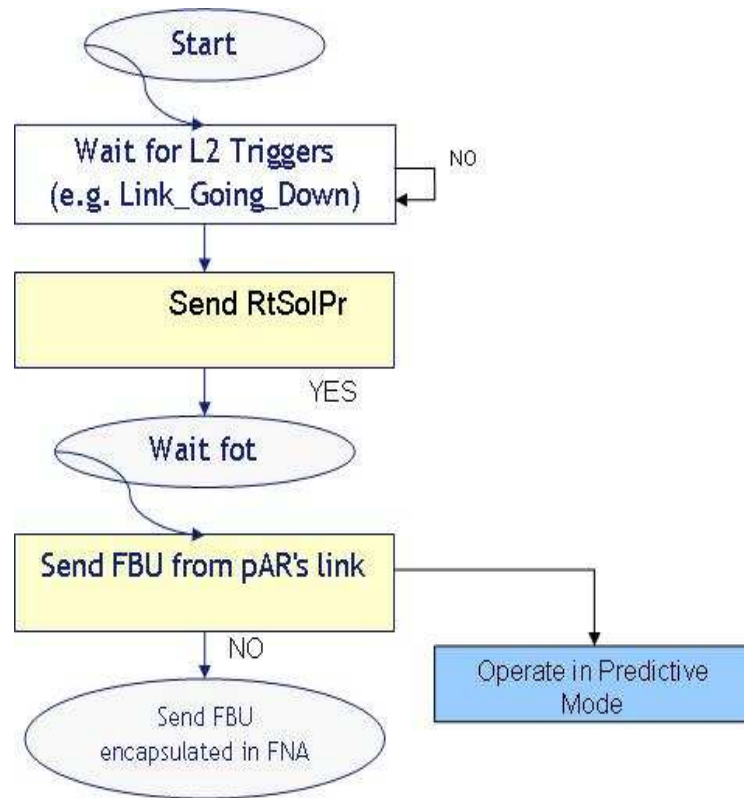


Figure 4-36: Reactive Mode

4.3 Development Platform

The base common platform supported by the consortium and used for software development will be as follows.

4.3.1 Linux distribution

Ubuntu 6.06 (Dapper)

<http://www.ubuntu.com>

Debian 3.1 (Sarge)

<http://www.debian.org/>

4.3.2 MIPL Mobile IPv6 for Linux

MIPL v2.0.2 (requires kernel version 2.6.16.20).

<http://mobile-ipv6.org>

4.3.3 IEEE 802.1X

Xsupplicant 1.2.4.

<http://open1x.sourceforge.net/>

4.3.4 AAA

FreeRadius 1.1.1

<http://www.freeradius.org/>

Opendiameter 1.0.7-h

(PANA is included with Opendiameter)

<http://www.opendiameter.org/>

OpenSAML 1.1

(Apache Tomcat >= 4, and Sunxacml 1.2 also required here)

<http://www.opensaml.org/>

<http://tomcat.apache.org/>

<http://sunxacml.sourceforge.net/>

4.3.5 SNMP

netsnmp 5.3.1

<http://net-snmp.sourceforge.net/>

4.3.6 Database

mysql-5.0

<http://www.mysql.com/>

4.4 Software development tools

In the modern software development lifecycle there are numerous artefact types like error reports, source code, schedules, test cases and documentation that has to be managed. There are a number of development tools that aid the creation and management of these software engineering artefacts, but for the ENABLE project there are two items which are essential to the software development process, a version controlled archive for the source code and bugtracking. In the following section of this document an outline of ENABLE's implementation of Subversion and Trac to server these purposes, is given.

4.4.1 Source code management

A file repository with version control based on Subversion [SUBV] has been put in place, in order to make the coding work of the distributed partner developers easier. Subversion is used to store all versions (stable and unstable) of the software modules developed and provides help to control changes between different uploads, and to insert comments/user information inside the file server's changelog. The basic operations that it supports are: update, add, remove, see history and commit changes of a file.

The subversion server is hosted on

<http://repos.ist-enable.org/repos>

In Appendix B. an explanation of how to configure and use the Subversion client is given.

The subversion repository structure is divided into three distinct areas

- coderepos/
 - Here the live, ever changing src code, build files for ENABLE developments are stored
- tagged/
 - Here a stable distribution for everyone to use is tagged from coderepos and stored in this folder.
- distribution/
 - Here original platform code (e.g.: kernel-2.6.16, mipl-2.0.2, opendiameter-1.07-h) is stored

The overall directory structure under /coderepos/ is

/coderepos/mipv6boot

/coderepos/mipv6eap

/coderepos/mipv6extensv4

/coderepos/dsmip

/coderepos/v6-v4accessrouter

/coderepos/haloadshare

/coderepos/haloadrel

/coderepos/eap_pana

/coderepos/qos

/coderepos/mipv6nsis

/coderepos/fhmipv6

The overall directory structure under /tagged/ is

/tagged/mipv6boot

/tagged/mipv6eap

/tagged/mipv6extensv4

/tagged/dsmip

/tagged/v6-v4accessrouter

/tagged/haloadshare

/tagged/haloadrel

/tagged/eap_pana

/tagged/qos

/tagged/mipv6nsis

/tagged/fhmipv6

/tagged/release-x.x.x

The overall directory structure under /distribution/ is

/distribution/debian/

/distribution/ubuntu/

Further to these directory structures under each software development tree (i.e. /coderepos/mipv6boot) the following sub-structure is in place:

/lib/ - This directory will contain all third party libraries.

/doc/ - This directory will contain all generated documents for the software.

/etc/ - This directory will contain any required miscellaneous files (property files, extra MIBs..etc) that are not handled by any other directory.

/src/ - This directory is the main source code directory for all the software.

/test/src/ - This directory contains the source code for the unit test cases that are used to verify the code. The package structure will mimic that of the main source folder to allow overridden stubs to be used in place of full classes during testing if necessary.

So for example '/coderepos/mipv6boot' would be

/coderepos/mipv6boot/lib/

/coderepos/mipv6boot/doc/

/coderepos/mipv6boot/etc/

/coderepos/mipv6boot/src/

/coderepos/mipv6boot/test/src/

4.4.2 Bugtracking

Trac [TRAC] is an open-source “defect tracking” system. Trac allows developers to gain quick access to the code repository (Repository Browse) and to keep track of current faults on their prototypes (Ticket System) while providing testers with a co-ordinated approach to reporting faults.

It also provides a milestone reporting tool (Roadmap) which provides a view on the ticket system that helps planning and managing the future development of the software components.

Trac has a built-in wiki engine, used to easily add, remove, edit and change some available textual content of the software developments of ENABLE.

For ENABLE, Trac is hosted at

<http://repos.ist-enable.org/>

5. INITIAL TEST-BED DESIGN

The development of the integrated software architecture has been split into two steps. Since the development has to be carried out in parallel by different partners, a first intermediate step for checking the compatibility and the functionality of the software modules was considered necessary (not all the programmed functionalities have been included in the first phase). Two minimal test-bed setups, described within the two following sub section, have been defined to demonstrate each step.

5.1 Test-bed for the initial integration

The first test-bed (Figure 5-1) is designed to demonstrate the following functionalities:

- EAP-based interface for network access and MIPv6 bootstrapping.
- HA discovery through DNS.
- MIPv6 authentication (and HoA provisioning) based on IKEv2.
- Diameter interface between HA and MASA AAA server.
- Realization of the interface between NETSNMP and the MIPL daemon on the HA (i.e. interface He needed for reading the number of active registrations from MIPL).

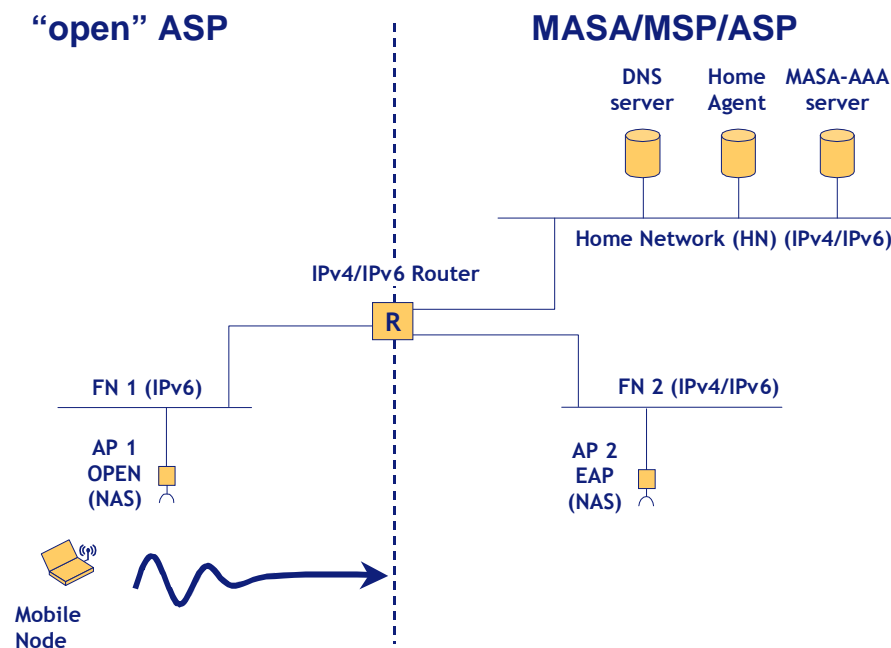


Figure 5-1: Initial Test-Bed (minimal setup)

The Mobile Node is not supposed to enter the Home Network. This assumption is reasonable since it is likely that it will be implemented as a virtual network by mobile operators.

In the initial integration the load sharing functionality is not included therefore only one Home Agent is needed.

Another functionality not included within the first integration step is the IPv4 interworking. These are the minimal number of networks that needs to be deployed:

- FN1 IPv6 only.
- HN, FN2 IPv4/IPv6.

Even if IPv4 interworking is not included and demonstrated here, FN2 needs IPv4 support to provide connectivity between the AP2 and the MASA-AAA server (most commercial AP and FreeRADIUS do not support IPv6).

The bootstrapping scenarios that can be demonstrated in this test-bed are:

- Integrated bootstrapping: when the MN bootstraps in FN2, it is authenticated and the MIPv6 bootstrapped. The HA address is provided to the MN during the network access phase while the HoA is assigned by the HA.
- Split bootstrapping from an open network: the HA address is obtained by the MN through a DNS query using a pre-configured FQDN; the HoA is assigned by the HA.

The minimal number of components needed in this initial test-bed are:

- n.1 IPv4/IPv6 Router.
- n.1 HA.
- n.1 MN.
- n.1 AAA server.
- n.1 DNS server.
- n.2 Access Points (n.1 EAP authentication, n.1 open).

5.2 Test-bed for the final integration

The final test-bed (Figure 5-2) will demonstrate these additional functionalities along with the ones described in the previous section:

- ASP AAA server to demonstrate support for roaming scenarios.
- IPv4 extensions for MIPv6 including support for IPv6-IPv4, IPv4-IPv6 and IPv4-IPv4 movement detection.
- HA load-sharing.

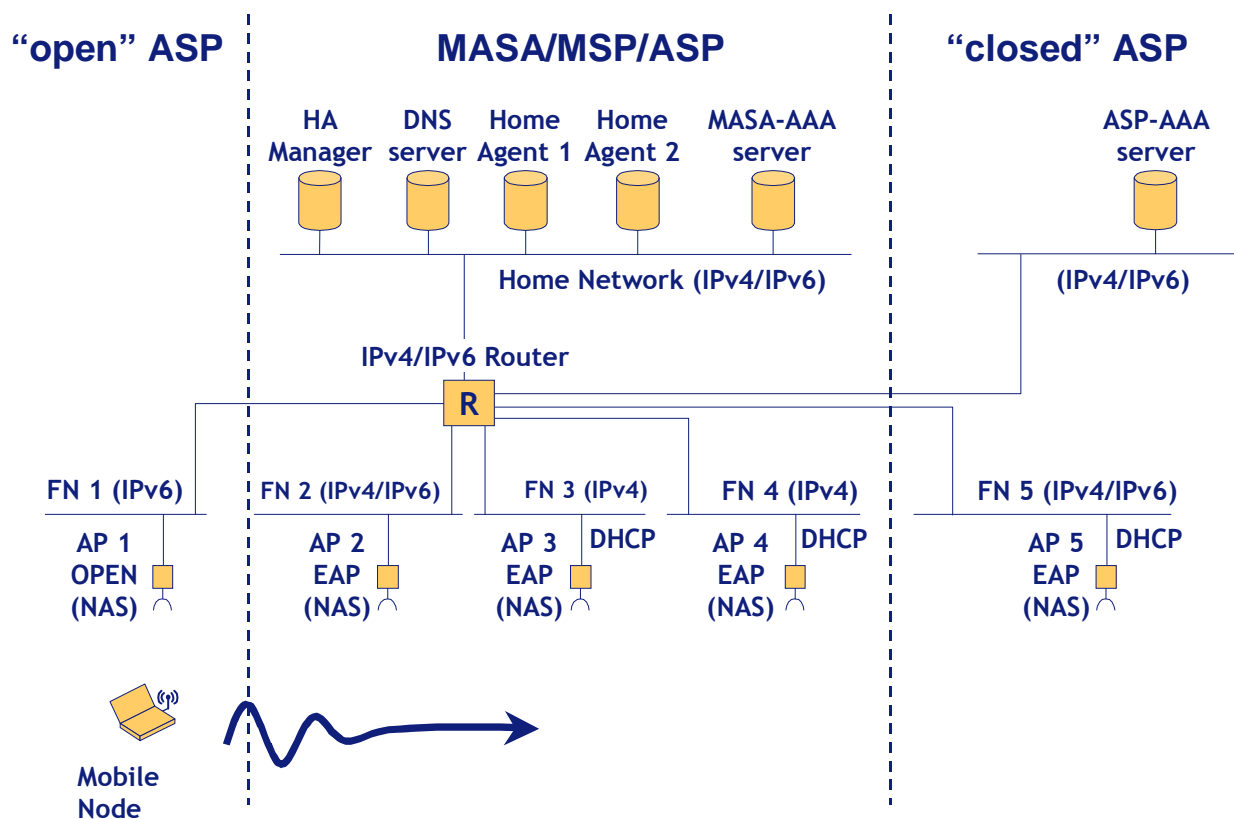


Figure 5-2: Final Test-Bed (minimal setup)

As explained in the previous section MN could never access the Home Network.

Due to the addition of the new functionalities, some network elements have to be added. In order to demonstrate the HA load sharing, an HA manager is needed to collect and process the information received from the HAs. Obviously, at least two HAs are needed.

For the IPv4 interworking functionality (IPv6-IPv4, IPv4-IPv6 and IPv4-IPv4 mobility) these networks must be present:

- n.1 network IPv6 only.
- n.2 IPv4/IPv6 networks (n.1 DHCP-enabled, n.1 without DHCP).
- n.2 IPv4 only networks (DHCP-enabled).

The access router is supposed to be also the DHCP server but is not mandatory. DHCP support in the access networks is very important for the movement detection algorithm implemented for the IPv4 interworking. If a network does not support DHCP, the MN is not able to recognize that this network is an IPv4 one. Therefore, FN2 is considered by the MN as an IPv6 only network though it is actually a dual stack (IPv4/IPv6) one.

New bootstrapping scenarios can be encompassed within this test-bed:

- Integrated bootstrapping in roaming: when the MN bootstraps in FN5, it is authenticated and MIPv6 is bootstrapped through the ASP-AAA acting as a proxy.
- IPv4 interworking: MN is able to bootstrap and roam in IPv4 and IPv4/IPv6 networks.

The minimal number of components needed for the final test-bed is:

- n.1 IPv4/IPv6 Router (due to the large number of hosts and subnets a switched solution with VLANs is preferable).
- n.1 MN.
- n.2 HAs.
- n.2 AAA server (n.1 MASA-AAA, n.1 ASP-AAA).
- n.1 DNS server.
- n.5 Access Points: (n.1 open, n.4 EAP authentication).

5.3 Test bed for NSIS / Mobile IPv6 firewall traversal

Figure 5-3 shows an overview about the ‘NSIS for Mobile IPv6 firewall traversal’ test-bed topology and the involved network components. Note that the prototype implementation will not perform any authentication or authorization functionality.

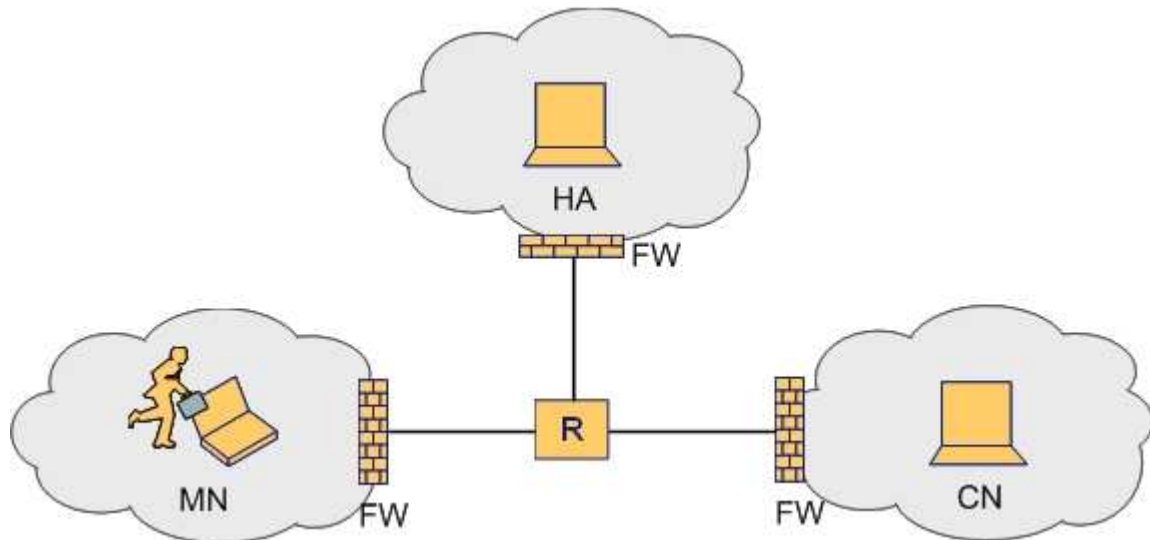


Figure 5-3: 'NSIS for Mobile IPv6 firewall traversal' test-bed topology

5.3.1 Hardware Requirements

The MN, HA and CN could be any type of PC, e.g. a laptop which runs Linux. The firewalls could be any type of PC which runs Linux and netfilter/iptables. In our test-bed we use Ubuntu Dapper for all machines. Possibly the firewall could be also a Linksys Router [<http://www.linksys.com>] which runs OpenWRT [<http://openwrt.org>].

5.3.2 Software Requirements

- OS.
 - Linux Distribution (e.g. Ubuntu, Debian, etc.) with Linux kernel 2.6.16.
- IPv6/DHCPv6.
 - Embedded in Linux kernel 2.6.16.
- Netfilter/iptables.
 - The firewall machines must run netfilter/iptables.
- MIPv6.
 - The MN, HA and CN need to run a MIPL 2.0.2 modified kernel and userspace daemon (with minor modifications to trigger the local firewall-traversal-application to initiate the firewall traversal progress). The modifications are:

- Before a Binding Update to the Home Agent will be performed, the MIPL implementation is halted and the local firewall-traversal-application is triggered. The firewall-traversal-application resumes the MIPL application after a successful pinhole creation for the Binding Update messages and the MIPL implementation can perform the Binding Update.
 - For the return routability test (RRT) the firewall-traversal-application signals further pinholes at the same time. Therefore, the MIPL implementation has to be halted before sending the HoTI and CoTI message. The firewall-traversal-application triggers the creation of several pinholes and if the pinholes for HoTI and HoT are installed, MIPL can send the HoTI. If the pinholes for the CoTI are also installed, MIPL can also send the CoTI.
 - After receiving successful HoT and CoT, MIPL triggers the firewall-traversal-application to install pinholes for following data traffic.
 - When a MIPL application running on a Home Agent receives a Binding Update, it triggers the local firewall-traversal-application and hands over the required parameters.
- NSIS and NAT/FW NSLP.
 - The MN, HA, CN and all firewalls need to run two separated processes, the NSIS NTLP and NAT/FW NSLP software.
 - Mobile IPv6 firewall traversal application.
 - This is the main application to be developed. It will operate on MN, HA and potentially on the CN and interacts with the local NAT/FW NSLP and the local MIPL implementation. Therefore, it is required to modify the MIPL implementation. Then, the firewall-traversal-application is triggered by the MIPL 2.0.2 implementation and instructs the NAT/FW NSLP to create the required pinholes. The triggers for initiate the firewall-traversal-application differ between MN, HA and CN.

5.4 Test bed for Mobility Optimisations

The FMIPv6 protocol prototype will be initially developed as a mobility optimisation feature to be built around overall functional components of the ENABLE architecture and also the

integrated software architecture. The Hardware and Software requirements for the subcomponents of the initial prototype is given below.

5.4.1 AR Sub-System

For the test-bed implementation, the ARs are considered to be Linux based PCs. The detailed hardware and software requirements are as follow.

5.4.1.1 Hardware and Software Requirements

- two PCs act as pAR and nAR respectively. Each AR is connected with one AP.
- Linux OS with kernel version 2.6.16.
- Patching MIPL-2.0.2 and installing the corresponding UserSpace.
- FMIPv6 AR daemon.

Given below in the following section, we have pinpointed all the hardware and software requirements for the MN in the test-bed scenario.

5.4.2 MN System

5.4.2.1 Hardware Requirements

For the test-bed implementation, the MN could be a laptop which can support Linux and can communication with wireless LANs. Any Linux supported wireless card should work with the FMIPv6 implementation. In our system, we choose the cards based on an Atheros chipset with a few improvements of the Madwifi drivers.

5.4.2.2 Software Requirements

- OS.
 - Any Linux Distribution (e.g. Red Hat, Ubuntu, Debian etc) with Linux kernel 2.6.16 only
- IEEE802.11 WLAN Terminal.
 - Interaction with AP to obtain WLAN access.
- IPv6/DHCPv6.

- Embedded in Linux kernel 2.6.16.
- MIPv6.
 - MIPL 2.0.2 modified kernel (with a minor modification for better fmipv6.org performance) and userspace daemon (also slightly modified).
- IKEv2 (optional).
 - Establish SA with HA; Optional for FMIPv6, TBD.
- FMIPv6-MN Daemon.
 - FMIPv6 source code from fmipv6.org to be patched with Linux kernel 2.6.16 and MIPL 2.0.2.

6. CONCLUSION

The scope of this document was to report on case studies and initial prototypes which would highlight and eventually facilitate efficient and operational mobility in large heterogeneous IP networks.

In approaching the case studies, this document starts with a review of four other IST projects, namely IST Ambient Networks, IST Daidalos, IST ePerSpace and IST Simplicity. While there are in excess of seventeen high level scenarios to choose from, it was found that all the best mobility related scenarios were already being implemented by the original project. From this point of view the consortium decided to continue evaluating the application scenarios as mentioned in the original ENABLE description of work, which included Location Based Services (LBS), Search and Rescue scene management (emergency applications), and VoIP Application with HA failover & Middlebox traversal.

A positive aspect of reviewing the IST project scenarios was that two keys methodologies became clear. Firstly, the use of UML was not advisable for the ENABLE project. This is mainly because ENABLE was taking a technological bottom up approach to its' research activities, and the use of UML scenario based development is more suitable when a top down approach is being employed.

Secondly, when it came to defining the 'per scene' template layout a combination of the best aspects from the IST projects was taken. This lead us to having sub-section headings 'Scene Challenge', 'Supported Services', 'Mobility Issues', 'User Experience' and 'Domain specific Issues' in each scene, which helped greatly in each scene definition.

Having completed a story board of six scenes for a search and rescue scenario, two of the scenes really stood out, Scene 3 were not enough assets are on site and volunteers are called in, and Scene 6 were the Ambulance picks up the victim and is returning to hospital location. These two scenes provide specific application case studies which we believe are flexible enough to support the verification of the technical and business requirements of a Mobile IPv6 service environment.

It is for this reason that Scene 3 & Scene 6 have more detailed descriptions, are mapped to the physical nodes in the test-bed infrastructure and will be the ones that will be used to demonstrate the ENABLE project technological achievements.

The concluding part of the application scenario is a very preliminary review of the specialised business entities (ASA, ASP, MSA, and MSP) in the scenario and the part they play during the search and rescue.

In approaching the initial prototype(s) each ENABLE work package identified specific technological components that were going to be researched during the project lifetime.

From here the consortium chose six functional components to develop further into working prototypes, four of which integrated quite seamlessly, EAP-based MIPv6 bootstrapping, AAA for MIPv6 bootstrapping, Interworking with IPv4 networks, and HA load sharing.

MIPv6 firewall traversal and Fast Mobile IPv6 (FMIPv6) stand out as separate mobility solutions and further investigation in the second year of the project will be made, to ascertain their possible inclusion to the first year integrated software.

Section 3 of the document gives an overview of each of these six technological components and Section 4 goes on to provide the software architecture, interface descriptions and software modules to be completed by the ENABLE partners on each component.

Combining the initial test-bed description from Section 5 with the two previous sections (3 & 4) provides the reader with an overall plan for the technological developments and specific functionalities that will be included in the final demonstration of the ENABLE project.

In conclusion this document is a comprehensive summary of a search & rescue scene management application scenario and provides the planning of software developments and test-bed integration effort for functionalities being prototyped from WP 1, 2, 3, 4 and 5 of ENABLE.

7. REFERENCES

- [DNav4] B. Aboba, J. Carlson, S. Cheshire: Detecting Network Attachment in IPv4 (DNav4), IETF RFC4436, March 2006
- [DSMIP] H. Soliman, “Mobile IPv6 support for dual stack Hosts and Routers (DSMIPv6)”, draft-ietf-mip6-nemo-v4traversal-03 (work in progress), October 2006
- [EAP-AKA] Arkko, J. and H. Haverinen, "EAP-AKA Authentication", RFC 4187, January 2006.
- [ENABLE-D1.1] IST ENABLE, Deliverable D1.1 “Requirements, scenarios and initial architecture”, June, 2006
- [GIST_FSM] T. Tsenov, H. Tschofenig, X. Fu, C. Aoun, E. Davies, “GIST State Machine”, draft-ietf-nsis-ntlp-statemachine-02.txt (work in progress), June 2006
- [MIP4-PPP] J. Solomon, S. Glass, “Mobile-IPv4 Configuration Option for PPP IPCP”, RFC 2290, February 1998.
- [MIPL] MIPL Mobile IPv6 for Linux, <http://www.mobile-ipv6.org/>
- [NATFW] M. Stiernerling, H. Tschofenig, C. Aoun, E. Davies, NAT/Firewall NSIS Signaling Layer Protocol (NSLP), Internet draft (draft-ietf-nsis-nslp-natfw-11), work in progress, April 2006.
- [NAT/FW_NSLP] C. Werner, X. Fu, H. Tschofenig, C. Aoun, N. Steinleitner, Ed., “NAT/FW NSLP State Machine”, draft-werner-nsis-natfw-nslp-statemachine-03.txt (work in progress), June 2006
- [NETFILTER] <http://www.netfilter.org>
- [NSIS_IMPL] <http://user.informatik.uni-goettingen.de/~nsis/download.html>, “Next Steps in Signaling (NSIS) Implementation by University of Goettingen”, (work in progress), June 2006
- [NSIS_NTLP] H. Schulzrinne, R. Hancock, “General Internet Signaling Transport”, draft-ietf-nsis-ntlp-11.txt (work in progress), August 2006

- [PEAPv2] Palekar, A. et al., "Protected EAP Protocol (PEAP) Version 2", draft-josefsson-pppext-eap-tls-eap-10 (work in progress), October 2004.
- [PPP] W. Simpson, "The Point-to-Point Protocol (PPP)", RFC 1661, July 1994.
- [RFC1213] K. McCloghrie, M. Rose: Management Information Base for Network Management of TCP/IP-based internets: MIB-II, IETF RFC1213, March 1991.
- [RFC2131] R. Droms: Dynamic Host Configuration Protocol, IETF RFC2461, March 1997
- [RFC2138] C. Rigney, A. Rubens, W. Simpson, S. Willens: Remote Authentication Dial In User Service (RADIUS), IETF RFC2138, April 1997.
- [RFC2461] T. Narten, E. Nordmark, W. Simpson: Neighbor Discovery for IP Version 6 (IPv6), IETF RFC2461, December 1998
- [RFC2462] S. Thomson, T. Narten: IPv6 Stateless Address Autoconfiguration, IETF RFC2462, December 1998
- [RFC3775] D. Johnson, C. Perkins, J. Arkko: Mobility Support in IPv6, IETF RFC 3775, June 2004
- [RFC4068] R.Koodli, Ed.: Fast Handovers for Mobile IPv6, IETF RFC4068, July 2005
- [RFC4080] R. Hancock, G. Karagiannis, J. Loughney, S. Van den Bosch, "Next Steps in Signaling (NSIS): Framework", RFC4080, June 2005.
- [RFC4140] H. Soliman Flarion, C. Castelluccia, K. El Malki, L. Bellier: Hierarchial Mobile IPv6 Mobility Management, IETF RFC4140, August 2005
- [RFC4260] P.McCann, Mobile IPv6 Fast Handovers for 802.11 Networks, IETF, RFC4260, November 2005.
- [RFC4295] G. Keeni, K. Koide, K. Nagami, S. Gundavelli: Mobile IPv6 Management Information Base, IETF RFC4295, April 2006.
- [SUBV] <http://subversion.tigris.org/>
- [TRAC] <http://trac.edgewall.org/>

- [USAGI] USAGI(UniverSAI playGround for Ipv6) Project www.linux-ipv6.org
- [WRTG54G Code] www.linksys.com -> Support -> Downloads -> WRTG54G V2.0 --
Wireless-G Broadband Router -> GPL Code
- [W3C] www.w3.org/TR/2000/REC-xml-20001006
- [XORP] www.xorp.org
- [Xsupplicant] Xsupplicant v 1.0 <http://open1x.sourceforge.net/>

APPENDIX A. SUMMARY OF HIGH LEVEL MOBILITY CONCEPTS

D4.1, V1.0, Ambient Network Mobility Scenarios & Requirements, July 2004

1. **Traditional mobility:** Traditional mobility is associated with a mobile terminal which connects to a different wireless station either related to a different or the same access network. This actually is not a new concept, but of course the new architecture also supports traditional requirements/concepts.
2. **Session mobility:** Change of interface in a multi-homed device, where the communication endpoints remain the same.
3. **Independent flow mobility:** This is similar to the previous approach, the difference is that flow endpoints may be moved between different interfaces, that do not necessarily belong to the same device.
4. **Distributed flow (application) mobility:** An application may combine several flows. There are different possible scenarios for mobility in this situation. Either some or all of the flows may be moved in a joint action.
5. **Managing multiple triggering sources for handovers:** Intelligent triggering logic in the mobile device and AN enables to optimize handover from different perspectives based on a series of triggering events. Especially in the situation of multiple triggering events, some of these may lead to different (even contradicting) handover proposals, which should be resolved by a triggering logic.
6. **Multi-homed mobility:** A node / device which, may connect to different services providers through one or even more interfaces changes its location. This has impacts on running flows. The complexity comes from the versatile configuration setup (e.g. how the node is connected to one more ANs or service providers through one or more interfaces and gateways, whether connections are used simultaneously or not etc.) and how this is handled in the case of mobility.
7. **Cluster mobility:** A cluster consists of several usually temporarily co-located nodes. The whole cluster may move as a single unit.
8. **Inter-technology mobility (vertical handover):** This topic addresses the change of access network type for a mobile endpoint, which moves location. For example a multimode terminal changes between WLAN and cellular access points.

9. **Inter-address space mobility:** (IPv4/IPv6/something else) Handover takes place between address spaces or more generally a mobile endpoint obtains a new location identifier, which may not be compatible with the previous one.
10. **Inter-trustdomain mobility:** The change of access point / network and the offered services may be defined by the security association between mobile endpoint and AN. Different parties may be involved in the authorization process, i.e. some level of trust might be achieved even without direct security association between mobile endpoint and AN.
11. **Inter-provider domain mobility:** With the change of the access network a mobile endpoint may experience a different treatment, e.g. more available bandwidth expanded and a middlebox for enriched media delivery may be needed.
12. **Predictable mobility:** This concept is about utilising possible information [like velocity, available PAN entities, device battery level,...] about future needs for mobility operations. Basically this applies to all kinds of mobility.
13. **Hierarchy considerations with mobility:** This mostly describes hierarchical clusters and independent movement behaviour on different levels.
14. **Multicast mobility:** One or several mobile entities participating in a multicast session move location.
15. **Cluster organisation and management:** This is about criteria for defining membership of moving clusters, maintenance while all nodes of the cluster are mobile.
16. **Cluster reachability:** Cluster reachability addresses methods to make mobile entities within a cluster reach-able by any correspondent nodes.
17. **Multihomed cluster mobility:** In this situation a cluster of nodes has multiple peering points to other networks.
18. **Physical to virtual link handover:** For example, where a cluster splits in two, but wishes to remain as a single AN, and switches from a physical connection to a virtual one.
19. **Optimization across different mobility events:** We cannot consider different handover events in isolation, one mobility event will have impacts on the network that may cause further mobility events to occur. This is not a different sort of mobility, but a statement about scope allowing us to talk about different mobility types in a uniform way, whereas often in current research different types of mobility are usually considered in a decoupled fashion.

20. **Advanced location management:** A location management concept, which would consist of location updating and paging. Advanced location management techniques would be necessary especially in the heterogeneous wireless access environment, etc. Different aspects to consider here may include issues like how the current location would influence the handover decision.
21. **Name identifier concept for handover support:** A concept of identifier, which is used for user/terminal/cluster identification and key information of mobility management control in the network. It may be investigated how the name identifier concept may support advanced handover mechanisms.
22. **Mobility control space support for network connectivity:** How do different ANs agree on mobility mechanisms / protocols to use after they merge to a single AN ? The mobility control space (MCS), which is part of the AN control space (ACS) should be the answer. There might be different services for mobility support when two ANs join together, e.g. signalling, routing encapsulation techniques to be used etc., which have to be negotiated between the merging networks.
23. **Managing middleboxes for control of routing streams:** Splicing and merging of communications via “middleboxes” is provided as a generic tool for applications to facilitate resolving heterogeneity on “upper layers” as well as dynamic service composition for ongoing communications. Media adaptation may require to route packets along different node (i.e. middleboxes), which can be considered as some kind of ‘middlebox handover’.
24. **Contextual predictable mobility:** Contextual information could be seen as a way to pre-arrange resources, by using the information already known by the user. In this sense, this could be seen as a foreknown mobility necessity (in this case, the user plays a fundamental role, as he /she has to give complete information regarding the contextual arrangements). The difference compared to concept #12 ‘predictable mobility’ is that here some context information is considered for the prediction, while for concept #12 this is based on the mobility pattern (i.e. the movement behaviour) of the mobile entity.
25. **Combined mobility management:** Provision of combined mobility management technique is meant here(including addition and deletion of techniques during the communication) for one or more mobile entities, e.g. performing middlebox insertion and interface change in a combined action. Different atomic mobility events are related to each other and some state information is required to manage such a compound action.

26. **Unresolved mobility:** A mobile endpoint is handed over, but actually, when the handover is initiated the new location yet is not clear. For example a user specifies some criteria and preferences and this information is used to search candidate locations and finally determine the new location to handover the mobile endpoint. The handover decision in this case is delegated to the network, since the mobile endpoint for some reason is not capable to make this decision.
27. **Handover redirection:** A node, defining the new location, where a mobile endpoint should be placed at can't accommodate the mobile endpoint for some reason. The most obvious reaction is that this node rejects the handover. But it may be possible that instead the mobile endpoint is redirected to a different location. However this introduces the potential risk of infinite loops in the handover process.

APPENDIX B. INSTALLING AND USING SUBVERSION

In this appendix an explanation of how to configure and use the Subversion client is given. As this service can also be used over the Windows platform, it will also show here some software for windows environment.

Ubuntu LINUX

Installation: first of all you have to install the Subversion client package. You can obtain it from the installation CDs or from internet repositories. If the installation is not automatic you have to type as root:

```
apt-get install subversion subversion-tools    (version may differ)
```

Configuration: once the package is installed, return to your private account and go to the directory from which the subversion directory structure should begin, and type:

```
svn checkout http://repos.ist-enable.org/repos
```

In the previous command you have to change the <login> sequence for your account login details.

Once you execute this command, the system connects with subversion server and asks for your password. This password is the fixed assigned password to your account.

This will download the whole ENABLE structure in your current directory.

If you only want a part of the repository, and you know where it is located you can modify the source directory this way:

```
svn checkout http://repos.ist-enable.org/repos/coderepos/FMIPv6/
```

This is an example of downloading only fmipv6 files.

Adding a file: if you create a new file into the repository structure, you have to do in the directory where it is:

```
svn add <file.ext>
```

```
svn commit
```

The commit command stores the changes in server. This command allows you to add a comment/log to the added file.

Removing a file: if you want to delete a file of the repository, in the directory where it is located you have to do:

```
svn remove <file.ext>
```

```
svn commit
```

Updating local files: if you want to download the last version of the code in the server you have to type:

```
svn update
```

Committing local changes to server: to upload your locally modified files and combine with other versions from other people, type:

```
svn commit
```

Remember that when you do this, an editor is opened and lets you to add comments to your new code version.

History log: if you want to see the change log of a file, including date of change, reviewer, number of lines changed and comments, you should type:

```
svn log <file.ext>
```

Windows Subversion client tools

There are many software tools over Windows to interoperate with a subversion server. We are not going to detail here the configuration, nor the use of them because it depends on the tool. You should be able to configure in an easy way (based on the linux configuration) a windows subversion tool.

The chosen tool has been ‘Tortoise SVN’ that is completely integrated on Windows file explorer and is easy to configure. Tortoise SVN can be installed from the following location:

<http://tortoisesvn.tigris.org/>.

An informative user guide is available at:

<http://www.cs.plu.edu/~dwolff/svn-tutorial/svn-tutorial.html>