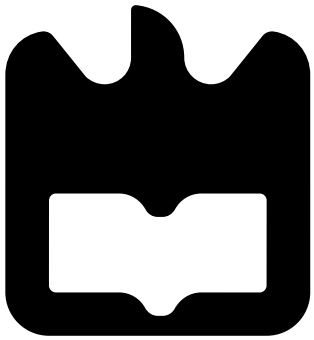
Universidade de Aveiro
Departamento de
Electrónica, Telecomunicações e Informática
2014

Fábio Cristiano
Outeiro de Aleluia
Martins

Separação de Identificação e Localização para Mobilidade de Veículos



Universidade de Aveiro
Departamento de Electrónica, Telecomunicações e Informática
2014

Fábio Cristiano Outeiro de Aleluia Martins

Separação de Identificação e Localização para Mobilidade de Veículos

" What we know is a drop, what we don't know is an ocean. "

— Isaac Newton

Fábio Critiano
Outeiro de Aleluia
Martins

Separação de Identificação e Localização para Mobilidade de Veículos

Dissertação apresentada à Universidade de Aveiro para cumprimento dos requisitos necessários à obtenção do grau de Mestre em Engenharia Electrónica e de Telecomunicações, realizada sob a orientação científica da Doutora Susana Sargento, Professora auxiliar do Departamento de Eletrónica, Telecomunicações e Informática da Universidade de Aveiro e co-orientação do Doutor Ricardo Matos, Engenheiro de sistemas na empresa Veniam'Works.

o júri / the jury

presidente / president

Professor Doutor Paulo Miguel Nepomuceno Pereira Monteiro
Professor Associado do Departamento de Eletrónica, Telecomunicações e Informática da Universidade de Aveiro

vogais / examiners committee

Professor Doutor Manuel Alberto Pereira Ricardo
Professor Associado do Departamento de Engenharia Eletrotécnica e de Computadores da Faculdade de Engenharia da Universidade do Porto (Arguente)

Professora Doutora Susana Isabel Barreto de Miranda Sargento
Professora Auxiliar do Departamento de Eletrónica, Telecomunicações e Informática da Universidade de Aveiro (orientadora)

agradecimentos / acknowledgements

Desde o início desta tese de mestrado, contei com a confiança e o apoio de inúmeras pessoas. Sem estes contributos, esta dissertação não seria de todo possível.

Em primeiro lugar quero agradecer à minha família pelo incentivo recebido ao longo deste ano. Em especial aos meus pais, Helder e Paula, por todo o amor, alegria e atenção sem reservas. Aos meus avós paternos e maternos por todo o valor e compreensão que tiverem. Ao meu primo João por toda a companhia. Ao meu tio Nicolá pelo exemplo e pela pessoa que ele é para mim. À minha namorada Sofia, um especial obrigado também, por me ter aturado, respeitado, compreendido, ouvido sem nunca deixar de me apoiar incansavelmente. A todos os meus amigos que acreditaram em mim e me apoiaram de início ao fim, eles sabem quem são, quero deixar-lhes aqui também uma palavra de apreço.

Agradeço a todos os colaboradores do grupo de investigação NAP,que dia a dia estiveram presentes no decorrer deste percurso, são eles João Aparício, João Mendes, Duarte, João Azevedo, Ferraz, Pardal, Bruno Areias, Luís e a todos aqueles que me ajudaram no desenvolvimento desta Dissertação, em especial ao Diogo Lopes, por toda a paciência e apoio dado e ao Rafael Gomes pela motivação e alegria.

Quero agradecer também à professora Susana Sargento por me ter cativado com este projeto interessante e por toda a disponibilidade e apoio demonstrados durante a realização do mesmo, bem como à empresa Veniam'Works e todos os seus trabalhadores pela ajuda prestada, em especial, sem ser demais referir, ao Diogo Lopes.

O meu profundo e sentido agradecimento a todas as pessoas que contribuiram para a concretização desta dissertação, estimulando-me intelectual e emocionalmente. Um enorme obrigado.

Resumo

Vivemos num mundo tecnológico, onde assistimos a uma evolução progressiva dos dispositivos e de comunicação digitais. Hoje em dia, os *smart-phones* e *smart TV* vieram substituir o telemóvel e a televisão, respetivamente. A internet está cada vez mais rápida, com mais serviços e aplicações, tornando-se num bem essencial e indispensável a nível mundial. Com uma constante interatividade entre utilizadores, as redes sociais são uma das grandes fontes de comunicação, dando-se, por vezes, prioridade à comunicação através do tão conhecido *Facebook* à comunicação pessoal. Estamos perante uma convergência e avanços tecnológicos, um mundo cada vez mais inter-relacionado e complexo. Devido a esta permanente necessidade de comunicação e ligação, as redes veiculares estão a atrair um interesse significativo.

As redes veiculares têm sido desenvolvidas, não só para melhorar o tráfego rodoviário, mas também para proporcionar interligação e entretenimento aos seus utilizadores. A comunicação entre os veículos e o acesso à internet por parte dos passageiros têm sido o principal objetivo na evolução e investigação destas redes.

Todavia, na evolução destas redes, permanecem inúmeros desafios. A grande mobilidade dos veículos durante o seu trajeto tem como consequência a necessidade de uma infinidade de *handovers*. Face a isto, é necessário um protocolo de mobilidade apropriado de forma a evitar a perda de ligação. Este protocolo deverá ser capaz de fornecer mobilidade, não só ao veículo, mas também aos seus passageiros.

O objetivo desta dissertação de mestrado centra-se no estudo do protocolo de mobilidade já existente da *Cisco Systems*, *The Locator/ID Separation Protocol* (LISP), e da sua extensão *LISP-MN* da organização *LISPmob*, de maneira a verificar a possibilidade de o adaptar para redes veiculares.

Através do router virtual da *Cisco* CSR 1000v, criou-se e configurou-se num ambiente privado um servidor capaz de armazenar e monitorizar todos os veículos bem como os seus passageiros. Cada veículo, representado por um identificador, regista-se no servidor indicando a sua localização no momento, sendo esta sempre atualizada quando o veículo muda de rede e já não estiver ao alcance da anterior. Assim, o servidor é a parte central na comunicação entre veículos funcionando como um mapa contendo todas as localizações associadas a cada veículo e fornecendo assim, sempre que requisitada, a localização necessária de um veículo a outro, permitindo a criação de um túnel entre eles e consequente estabelecimento de ligação. Para proporcionar um *handover* mais rápido entre estações fixas e móveis foram feitas alterações a nível de software do *LISP-MN*. Alterou-se a implementação *LISP-MN* de maneira a garantir mobilidade para veículos, ou seja, para handovers com rápidas transições, visto que na implementação da *LISPmob* só é garantida mobilidade para handovers lentos, tornando assim impossível o handover entre veículos e à consequente inutilização da tecnologia WAVE, criada especialmente para tal. Alterou-se também a forma de processamento na atualização das caches dos nós móveis que estão em comunicação, de maneira que, na ocorrência de *handover*, as atualizações das *cache* fossem permitidas, não só na receção de um novo endereço, como também na receção de uma nova *gateway*, evitando assim possíveis problemas de falhas de mensagens de controlo do protocolo essenciais para o estabelecimento de comunicação e transmissão de dados entre veículos. Posteriormente, criou-se um *Connection Manager* capaz de gerir o *handover* de forma automática independente da ligação de acesso bem como da versão do protocolo de internet utilizada, permitindo assim a ligação por parte dos veículos e seus passageiros à rede com melhor sinal. Assim, através do mecanismo de mobilidade referido garantiu-se a mobilidade entre veículos e respetivos passageiros.

Os testes efetuados em laboratório e na estrada incidiram sobre as tecnologias de acesso IEEE 802.11p (WAVE), uma tecnologia desenvolvida especialmente para as redes veiculares, e o IEEE 802.11g (WI-FI), uma das tecnologias mais utilizadas atualmente. Verificou-se através dos resultados obtidos que os tempos de *handover* através da tecnologia WAVE eram significativamente inferiores aos da tecnologia WI-FI, inferindo assim que a tecnologia de acesso IEEE 802.11p é a mais apropriada para as redes veiculares.

Os resultados de *handovers* realizados em vários cenários de laboratório e estrada mostram que os mecanismos desenvolvidos permitem fornecer mobilidade transparente dos veículos e seus passageiros.

Abstract

We live in a technological world, where we witnessed a progressive evolution of devices and digital communication. Nowadays, the smartphones and smart TV have replaced the phone and television, respectively. The internet is getting faster, with more services and applications, making it very essential and indispensable worldwide. With a constant interactivity between users, social networks are a major source of communication, giving up sometimes priority to communication through the well-known "Facebook", instead of personal communication. We are facing a convergence and technological advances, an increasingly complex and interrelated world. Due to this constant need for communication and connection, vehicular networks are attracting significant interest.

Vehicular networks have been developed, not only to improve road traffic, as well as interconnection and to provide entertainment to their users. The communication between vehicles and internet access by passengers have been the main goal in the development and investigation of these networks.

However, in the evolution of these networks, many challenges remain. The high mobility of vehicles during their commute entails the need of a plethora of handovers. Mobility protocol suitable to prevent the connection loss is required. This protocol should be able to provide mobility, not only to the vehicles, but also to the passengers.

The purpose of this dissertation focuses on the study of existing mobility protocol from Cisco Systems, the Locator/ID Separation Protocol (LISP), and its extension LISP-MN from LISPmob organization, in order to verify the possibility to adapt to vehicular networks.

Through the virtual router from Cisco CSR 1000v, it was created and configured in a private environment a server capable to store and monitor all vehicles and their passengers. Each vehicle, represented by an identifier, is recorded on the server indicating its location on the time, and it is always updated when the vehicle changes the network and it is no longer reachable through the other. Thus, the server is the central part in the communication between vehicles functioning as a map containing all locations associated at each vehicle and thus providing, when required, the necessary location of a vehicle to another, allowing the creation of a tunnel between them and consequent establishment of connection. To provide faster handover between fixed and mobile stations, changes were made to the software of LISP-MN. LISP-MN implementation has changed in order to ensure vehicular mobility, with fast handover transitions, which with LISPmob is not guaranteed, it just only ensures mobility in slow handovers case. Thus, it makes impossible handovers between the vehicle and the consequent use of WAVE technology, specially created for these networks. It was also changed the way to update the caches of mobile nodes that are in communication, so that when the handover occurs, cache updates are allowed not only on the reception of a new address, but also on the reception of a new gateway, thereby avoiding potential problems on control messages of the protocol essential to establish the communication and further data transmission between vehicles. Subsequently, a Connection Manager was created capable to manage the handover automatically independently of the access network and of the Internet protocol version used, thus allowing the connection of the vehicle and its passengers to the network with best signal. Given those facts it was guaranteed the mobility of vehicles and their respective passengers.

The tests performed in the laboratory and on the road were focused on the access technology IEEE 802.11p (WAVE), a technology developed especially for vehicular networks, and IEEE 802.11g (WI-FI), one of the most used technologies today. It was verified by the results obtained, that handover times through the WAVE technology were significantly lower than those of WI-FI technology, and thus inferring that the access technology IEEE 802.11p is the most suitable for vehicular networks.

The results of handover performed in various lab and road scenarios show that the developed mechanisms provide transparent mobility of vehicles and their passengers.

Contents

List of Figures

List of Tables

Acronyms

AP Access Point

AR access router

ASU Anchor Set Update

ASA Anchor Set Acknowledgement

BA Binding Acknowledgement

BC Binding cache

BCE Binding Cache Entry

BSS Basic Service Set

BU Binding Update

BUL Binding Update List

BW Bandwidth

CCH Control Channel

CN Correspondent Node

CoA Care-of address

CPU Central Processing Unit

CSR Cisco Cloud Services Router

DHCP Dedicated Host Configuration Protocol

DMAR Data Mobility Access Router

DMIPA Distributed Mobility IP Anchoring

DSRC Dedicated Short-Range Communications

EIDs Endpoint Identifiers

ETR egress tunnel router

FCC Federal Communications Commission

GPRS General Packet Radio Service

HA Home agent

HN Home Network

HoA Home address

HTTP Hypertext Transfer Protocol

IEEE Institute of Electrical and Electronics Engineers

IP Internet Protocol

IPv4 Internet Protocol version 4

IPv6 Internet Protocol version 6

ITR ingress tunnel router

lab laboratory

LISP Locator/ID Separation Protocol

LISP-MN LISP Mobile Node

LL Link-local

LMA Local Mobility Anchor

LMD Local Mobility Domain

LTE Long Term Evolution

MAC Medium Access Control

MAG Mobile Access Gateway

map-and-encap mapping and encapsulation protocol

m meters

MANET Mobile Ah-hoc Network

ms milliseconds

MIPv4 Mobile Internet Protocol version 4

MIPv6 Mobile Internet Protocol version 6

mMAG mobile MAG

MN Mobile Node

MN-ID Mobile Node Identifier

MN-HNP Home Network Prefix

MNN Mobile Network Node

MR Map-Resolver

MS Map-Server

MSF Mobility Support Flag

NAT Network Address Translation

NDP Neighbour Discovery Protocol

NetLMM network-based localized mobility management

OBU On-board Unit

OS operating system

PBA Proxy Binding Acknowledge

PBU Proxy Binding Update

PETR Proxy Ingress Tunnel Router

PITR Proxy Ingress Tunnel Router

PMIPv6 Proxy-Mobile IPv6

proxy-CoA Proxy care of address

QoS Quality of Service

RA Router Advertisement

RLOCs Routing Locators

RS Router Solicitation

RSSI Received Signal Strength Indicator

RSU Road Side Unit

SCH Service Channel

SMR Solicit-Map-Request

SN static node

TCP Transmission Control Protocol

TR tunnel router

TTL Time-To-Live

UDP User Data Protocol

VANET Vehicular ad-hoc Network

VLC VLC media player

V2I Vehicle to infrastructure

V2V Vehicle to Vehicle

WAVE Wireless Access in Vehicular Environments

WBSS WAVE Basic Service Set

WI-FI IEEE 802.11 a/g/n

WLAN Wireless Local Area Network

Chapter 1

Introduction

1.1 Motivation

In a technological world, connection is the watchword. Information technology and communication definitely has entered in our lives. We are increasingly dependent on them, in the private context as well as in the workplace.

There is a constant need to always be connected which has made the technology to develop in a fast pace. Mobile phones have become an integral part of our daily lives, now commonly used for data rather than voice. The internet has become a necessity, increasingly faster with more services and capacity; WI-FI hotspots have been spread around the world, allowing user connections to the internet, although some limitations remain such as the lack of handovers capabilities, short range and the time lost in authentication. On the other hand, we had a strong evolution on cellular networks and technologies.

People need to be connected and the vehicles are not exempt from this trend. The opportunity to be connected to the Internet during a journey would be great for all passengers being able to access to their work tools, such as e-mail and entertainment contents sharing their experiences instantaneously. Vehicular networks can also be an important approach in order to improve the quality for all drivers and pedestrians, such as safety warnings and traffic information. It is often believed that acting as a network could avoid accidents and traffic congestions, than if each vehicle tries to solve these problems individually.

A Vehicular Ad-hoc Network (VANET) turns every participating car into a wireless router or node, allowing cars to connect, creating a network with a wide range. As cars fall out of the signal range and drop out of the network, other cars can join in, connecting vehicles to one another so that a mobile Internet is created. The concept used is similar to the one applied on ad-hoc networks. Cars act as mobile nodes carrying a device called On

Board Unit (OBU), which allows other nearby nodes (vehicles) to connect through several wireless technologies, such as WAVE (IEEE 802.11p), WI-FI (IEEE 802.11a/b/g), LTE (4G). Thus, users are allowed to connect to this OBUs inside the vehicles; besides that, the vehicles should be capable to bind to stationary providers, which are present along the road. These stationary providers can be Road Side Units (RSUs) or WI-FI Access Points (APs), which in turn will provide them access to the Internet.

Thanks to VANETs, users are approaching to the main goal, which is always being connected to the best network available without losing their connection during their journey. To ensure that and taking into account high mobility in vehicular networks it is imperative that, with an appropriate mobility protocol, OBUs provide seamless handover between APs along the road. There are several studies about mobility protocols, but it still remains unsolved in commercial networks. Furthermore, WAVE is a new developed access technology special to vehicular networks, and the mobility protocols lack their evaluation with this technology.

Some mobility protocols have already been evaluated on a vehicular scenario in our group, such as in the work developed by [16] and [34]. According to [16], this work has proven that the Proxy-Mobile IPv6 (PMIPv6) protocol is capable of providing mobility to the cars moving along the road, and changing their attachment points between the available fixed infrastructures or even through a 3G connection. It has also demonstrated that the WAVE protocol is the most suitable access technology to be used in the VANETs, since it provides seamless handover capabilities without loss of packets and with reduced times. However, this protocol cannot support mobility to the entire network, as it was mentioned before, and, as it is an IPv6 mobility protocol, it does not have any support for IPv4 mobility. Regarding the work in [34] developed in our group, it is mainly focused on N-PMIPv6 mobility protocol, which made a significant progress comparing with PMIPv6. The mobility for the entire network was guaranteed, and this means that the mobility for the vehicles and their passengers was ensured regardless of the access technology. Furthermore, cars must be able to connect to a fixed infrastructure, such as AP or RSU through WI-FI or WAVE access technology, and on the other hand, they must allow their passengers (single-hop) or others vehicles (multi-hop) to connect to the network through the same access technologies. In other words, vehicles can act at the same time as mobile nodes and routers capable of providing network connection, not only to its passengers, but also to other vehicles nearby. Thus, a multi-hop connection over vehicular network extends the range of the internet access decreasing the necessity of fixed infrastructures, which in turn translates into lower cost for the development of vehicular networks; however, the more hops you have, the lower is the bandwidth (BW), being a problem in the connection

quality.

Apart from that, both PMIPv6 and N-PMIPv6 bring some limitations, as all the traffic needs to traverse one point known by LMA (server); vehicles can just establish a link with an attachment stationary point. The network and mobility scalability are a strong problem due to the fact that the LMA is a centralized entity. It is commonly recognized that today's Internet routing and addressing system is facing serious scaling problems, which IPv6 is not by itself a solution. Given these facts and in order to avoid a resource overhead as well as much complexity, the Locator/Identifier Separation Protocol (LISP) protocol was developed by *CISCO*. This protocol splits the location from identity, which is a requirement to provide native mobility and multihoming. On the other hand, in order to face the scalability problems, a distributed entity acting as a database anchor is needed. Moreover, it is important to evaluate this mobility protocol, applied for a vehicular network, measure the handover times in order to understand its applicability to these environments.

1.2 Objectives and Contributions

The work in this dissertation will focus on the implementation and evaluation of a mobility protocol for vehicular networks, able to ensure the mobility of vehicles and their passengers during their journey. The LISP-MN, an open-source implementation based on LISP to allow mobility and multihoming natively will be used; in order to face routing scalability problems, a virtual router provided by *CISCO* with LISP mobility features is also used to work as a server.

To reach this goal, the thesis has the following objectives:

- `Study LISP protocol`: understand how it works and discover how it can be adapted to vehicular networks.
- `Protocol scenarios`: define in which scenarios LISP protocol can be tested.
- `Protocol Adaptation`: LISP needs to be adapted to vehicular networks, capable to maintain network session while doing handover over WAVE or WI-FI. So, protocol and software changes need to be performed in LISP-MN to ensure fast mobility into the vehicles.
- `Map-Server and Map-Resolver Implementation`: through the virtual router from Cisco CSR 1000v, it is created and configured in a private environment a server capable to store and monitor all vehicles and their passengers.

- `Protocol Compile`: as the OBUs do not have the compiler inside, a virtual machine with OpenWrt repository (builder) will be used to generate the binary compiled for OpenWrt and send it to the OBUs.
- `Protocol Implementation`: as LISP-MN implementation just allows slow handovers, it was changed in order to ensure mobility for the vehicles which performing fast handovers and consequently, due to the fast handovers implemented, it was also changed the way of updating vehicular caches in order to not fail any LISP control message fundamental to the process. Further, after compiled it was necessary to set files according to Map-Server configurations to run with the binary in each OBU, allowing the vehicles and their passengers to register on Map-Server and to know where are the other passengers.
- `Connection manager implementation`: to automate the handover procedure, it is required an unity capable of monitoring the available networks. It shall identify the best network available and trigger the handover whenever needed.
- `Testing protocol`: evaluate LISP mobility protocol in different scenarios in the laboratory (lab) and in the road, in order to realize weather it is an advantage for vehicular networks.

1.3 Document Organization

This Dissertation is organized as follows:

- `Chapter 1`: presents the Dissertation contextualization, the motivation, the contribution and the objectives.
- `Chapter 2`: presents the state of the art of vehicular networks, which addresses mobility protocols and contains other vehicular features.
- `Chapter 3`: describes deeply the mobility protocol chosen to further implementation.
- `Chapter 4`: shows all implementations, components, tools, optimizations and adaptations performed for further evaluation of this mobility protocol into vehicular networks.
- `Chapter 5`: depicts used testbeds to test the mobility protocol implemented. Further, it presents and discusses the results obtained in the laboratory and on the real road environment.

- `Chapter 6`: summarizes the work that has been performed during this Dissertation and also suggests possible future development to continue and optimize the work already done.

Chapter 2

State of the Art

2.1 Introduction

This chapter describes the vehicular networks and their characteristics relevant for the handover of the vehicles, and the current relevant mobility approaches.

In this context, the topics of this chapter and their organization are presented below.

Section 2.2 describes the meaning and the features of vehicular networks.

Section 2.3 portrays what are the equipments used in vehicular networks.

Section 2.4 illustrates the vehicular network architecture and its features.

Section 2.5 shows several network access technologies and their possible applications or advantages to VANETs.

Section 2.6 describes different mobility protocols, the main features and how they work, and finally they are compared with each other.

In sum up, section 2.7 is the chapter summary.

2.2 Features

Primarily, it is important to be acquainted with the thematic of vehicular networks (VANETs). Above all, VANETs are a group of vehicles interconnected via several technologies, such as WI-FI, IEEE 802.11p (WAVE) or even cellular. Moreover, they are capable of sharing software, hardware, and information between them and many users. Thus, cars and users, both seen as mobiles nodes (MNs), are able to communicate between them and sharing informations. Further, an internet connectivity is possible to all users inside the vehicles, which nowadays it is an essential commodity.

There are some special characteristics [21] exclusive of vehicular networks as follows:

- `Predictability`: predictability is possible thanks to GPS that are in all cars, providing the position, and to the road that limit the movement of vehicles.
- `Higher computational capability`: OBUs can afford significant communication, computing and sensing capabilities.
- `No power constrains`: as the the components inside the cars are powered by them, this should not be an issue.
- `Partitioned network`: vehicular networks are usually fragmented due to the dynamic environment where they are inserted, resulting in some isolated clusters.
- `Rapid topology changes`: vehicles are constantly moving, and they are the networks nodes.
- `Large scale`: each vehicle is going to have two functionalities, acting as a MN and a router. Scaling the network to the number of vehicles in the roads is a major challenge in these networks.

2.3 Equipment

There are several essential elements responsible to the functioning of the vehicular networks, such as Road Side Units (RSUs) and On Board Units (OBUs), which are indubitably important.

The RSUs and OBUs may be similar, with different functionalities in the vehicular network. RSUs act as fixed infrastructures along the road, providing several wireless technologies as well as a physical connection and internet access for vehicles and their passengers. OBUs are inserted inside the vehicle with the functionality to provide wireless technologies to allow the connection with users. To describe what is the main hardware inside of this OBU and at the same time the RSU, Maria Kihl [21] presents their equipment as follows:

- `Central Processing Unit (CPU)`: responsible for the communication protocols operations and the application performance.
- `Wireless transceiver`: needed to provide the way to send and receive data between car's working as Antennas.
- `GPS`: provides location and several metrics which maintain the vehicles synchronized.

- `Sensors`: necessary to analyse some variables needed to be sent between vehicles.
- `Input/Output interface`: is the part that interacts with humans and the board itself.

Regarding the equipment described before, it is illustrated in figure 2.1 the element developed in our group with the required features [4] which are detailed as follows:

- PCEngines Alix3D3 Module with a 500 MHz AMD Geode LX800, 32-bit x86 architecture, 256 MBytes of memory and Ethernet connection.
- DSRC/WAVE Module compliant with IEEE 802.11p.
- WI-FI Module compliant with IEEE 802.11b/g.
- Omnidirectional antenna prepared for frequencies in the range of 2.4 GHz, with a 5dBi gain.
- Omnidirectional L-Com Antenna prepared for frequencies between 5.150 and 5.9 GHz, with a 5dBi gain.
- Linux Debian (squeeze) Operating system, with the 2.6.32 kernel compiled with the options to support mobility protocols.
- Driver ath5k modified to support the IEEE 802.11p/1609.x [4].
- GPS GlobalTop (MediaTek MT3329).

The main difference between both boards units illustrated above is the presence of the WAVE communications, very useful in vehicular mobility, which their features are:

- Wave fast association.
- Support for the WAVE Short Message Protocol.
- Existence of Control Channel (CCH) and Service Channel (SCH) and support for operations with channel switching.

Figure 2.1: On Board Unit (OBU)

2.4 Network Architecture

This section introduces the VANET network architecture. One example is illustrated in figure 2.2 which shows vehicles connecting with each other and with fixed infrastructures. According to Lee and Gerla [47], there are three possible vehicular architectures subdivided in three categories:

- `Hybrid`: Considered an intelligent and at the same time flexible architecture. The hybrid architecture is a non-centralized architecture; this means that it does not have a centralized authority thus the information is passing through the vehicles in a distributed way. Furthermore cars could act as nodes or mobile nodes.

- `Pure cellular/WLAN`: To be connected to the internet, vehicles could choose one of two paths, or by cellular gateways or either by access points (APs) through WLAN interface. Thus, during their journey, internet connection and services are ensured by a link to cellular tower or APs.

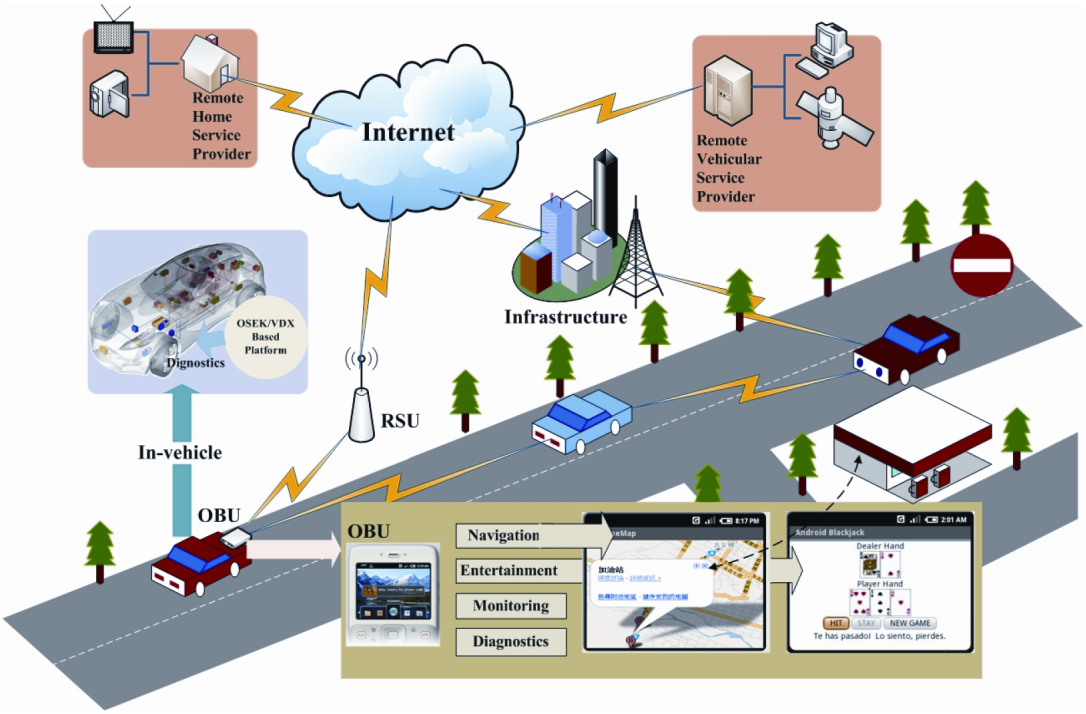


Figure 2.2: VANETs Architecture [9]

- `Pure ad hoc`: In this case the connections are established peer-to-peer between vehicles. Basically this means that whether the vehicles have more than one option to have connection to the internet, such as cellular tower, APs or even vehicles, their priority is the communication between vehicles.

2.5 Network Access Technology

There are multiple technology communication ways to access the network. In this section some of them will be described. Thus, depending on the type of the application, scenario or other effects, there are different advantages and disadvantages between them.

2.5.1 Dedicated Short-Range Communications (DSRC) allocated spectrum

DSRC based on [30] is a type of short wireless communication that allows data transmission between vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) ir order to provide safety to them. It is a reliable type of access network technology for crash preven-

tions and safety applications; a usage case is presented in figure 2.3.

Regarding the spectrum, the Federal Communications Commission (FCC) allocated 75 MHz of spectrum in the 5.9 GHz band used for mobility applications and for vehicle safety. Further the DSRC spectrum is divided into 7 channels, each one with 10 MHz, high data rate, short range radio and half-duplex. The DSRC channel allocation is illustrated in figure 2.4.

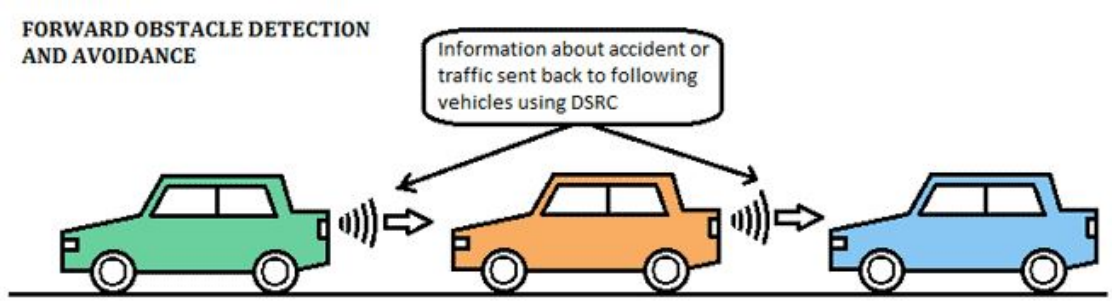


Figure 2.3: Obstacle Detection using DSRC [32]

There are many advantages using DSRC in V2V and V2I communications presented as follows:

- Crash prevention with real time advertisements alerting drivers.
- Obstacle Detection and Avoidance.
- Real-time connectivity to all user services.
- Enable mobility between vehicles and infrastructures.
- Enable fast communication and low latency.

2.5.2 IEEE 802.11p / WAVE

The standard IEEE 802.11p (WAVE) is the most appropriate access wireless technology for the vehicular network.

The vehicles are in constantly position change in a short period of time, establishing several V2V or V2I connections during their way. Thus, it was required to create a standard capable of supporting these fast transitions, providing easy and fast wireless short-range communication between them. Given this fact, it was specified in 2004 Wireless Access in

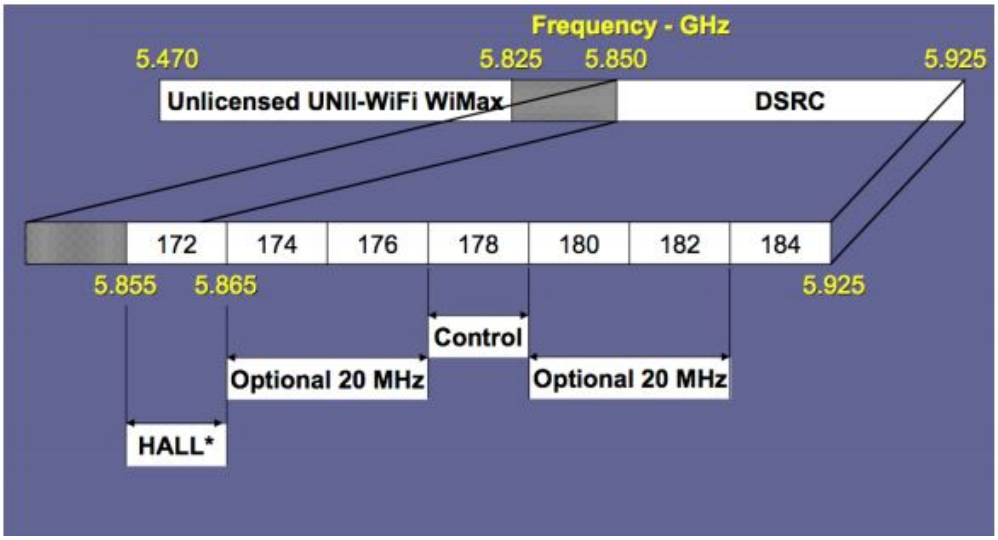


Figure 2.4: DSRC - Channel Allocation[18]

Vehicular Environments (WAVE) the norm IEEE 802.11p created by the task force group, modifying the standard IEEE 802.11a and becoming capable to operate in DSRC band. It is possible to observe in figure 2.5 the use of WAVE applied on vehicular environment.

It is important to refer that WAVE is an evolution of DSRC. DSRC focuses on low overhead operation based on the Wireless Fidelity (WI-FI) architecture [12].

According to [27] the IEEE 802.11p standard is meant to:

- Avoid joining to the Basic Service Set (BSS), as it happens in IEEE 802.11, a set of functions and services required for the WAVE stations in order to answer quickly to the vehicle changing without any drop message.
- In order to control IEEE 802.11 MAC, it was performed an amendment in WAVE interface functions and signalling techniques.

Some changes on MAC for WAVE operations are described below according also to [27]:

- Any wave station is able to send and receive data frames with the destination and source field set to 0, independent if it is or not a member of WAVE BSS (WBSS).
- There are many WBSS which are familiar to WAVE mode operation and with their identification field set to 0. A WBSS is able to communicate, thus start its initialization after receiving the necessary informations from a radio in WAVE mode.

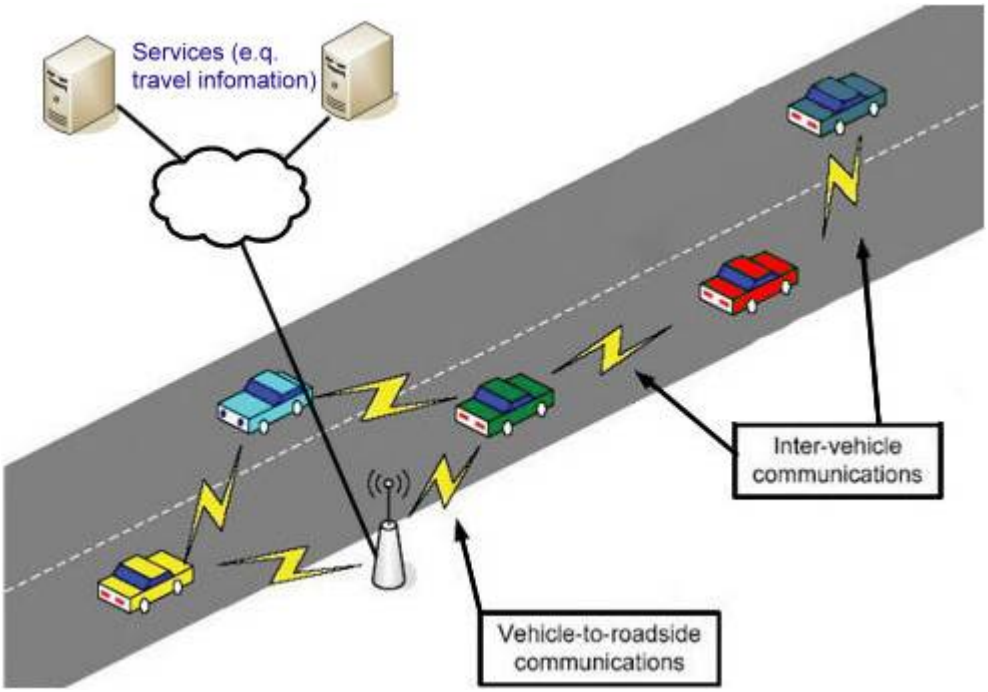


Figure 2.5: WAVE applied in vehicular communication [5]

- The radio is configured to send and receive data frames with the identification field (BSSID) from one WBSS. The node leaves the WBSS when stopping sending or receiving data frames without identification field set from the WBSS.
- One station can just join one WBBS at each time and also, whether it is in WAVE operations mode, it can just join the WBSS and not BSS.
- In case of no member still present on the WBSS, it ceases to exist.

There are also some changes to the level of the PHY layer such as:

- Wave PHY layer is based on the OFDM PHY defined for IEEE 802.11a. The channel wide become 10 MHz instead of 20 MHz presented on IEEE.11a.
- Improved receiver performance requirements.
- Improved transmission mask.

Thus, in figure 2.6 it is presented the WAVE protocol stack. It is possible to observe a division in two standards [18], so WAVE not only contains the standard IEEE 802.11p

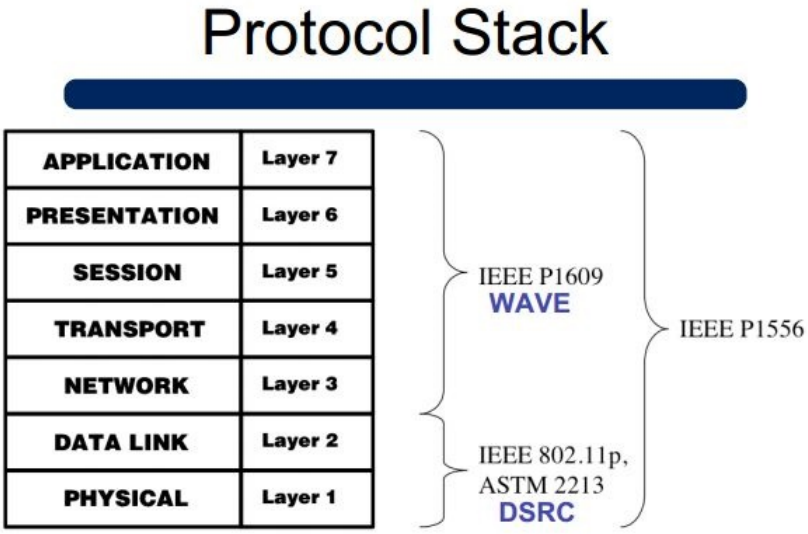


Figure 2.6: WAVE Protocol Stack [18]

---

but also IEEE 1609 which is the upper-layer standard. IEEE 1609 completes WAVE in some details, and it is also divided in some slices. Firstly, IEEE 1609.2 standard focuses on resource manager defining data flows, key components and command messages of WAVE. IEEE 1609.2 standard covers the security communication. IEEE 1609.3 standard is responsible for the WAVE connection setup and management. Lastly, IEEE 1609.4 is liable to Multi-Channels Operations based on the IEEE 802.11p Physical layer and Medium Access Control layer supplies.

2.5.3 Multi-Technology approach

To accelerate vehicular communications, it is needed to use multi-technology systems. Besides the WAVE technology, vehicles should also be able to connect to the already existent WI-FI APs along the road as well as to cellular infrastructures. The decision for one instead of another should take into account the cost for the user and the quality of the connection.

There are some cities which provide free WI-FI APs in order to allow people to connect to the internet. APs spread all over the city, which allows their subscribers to have free internet connection when are in the presence of one of these private routers. However, external users are also able to connect those APs, although they may have some inherent costs per hour.

Nevertheless, there are some issues regarding WI-FI connections, such as, their small range, their slow authentication connection and their location, because mostly of them are located in main area of the cities. Thus, they are not a good solution in vehicular communications unless the vehicle is stopped or in low movement.

Nowadays, the countries are covered by cellular networks, such as UMTS as known by 3G, LTE, known as 4G. In this case all vehicles have the possibility to connect to cellular networks along the road when the RSUs are not in range. However, such networks have a high cost to the user, so the idea is to reduce their use in vehicular communications.

In all due fairness, for the best of the vehicles and their passengers, they must use the WAVE technology. With the increase of RSUs along the road and consequently OBUs inside the vehicle, the vehicles are available to communicate through WAVE. Thus, to the ideal scenario, the vehicles along the road should connect firstly to the RSUs in case they are in range, then to the APs and lastly, just in case there is no other alternative, to the cellular stations.

During this master dissertation, the WAVE technology has been the core technology due the fact that it brings gains in terms of costs and speed to the vehicular communications.

2.6 Mobility Protocols

During the journey the cars are changing their connection to different attachment points, the addresses are being changed and the routing is done through another attachment point. Due to the high vehicle velocity, it is hard to maintain a seamless handover and a stable connectivity to the Internet. Furthermore, several times during the movement, vehicles will gain a new IP address, their appropriate network mask and either the default router; otherwise the packets will be lost and the connection is broken. In order to maintain the connection alive while the vehicle is moving, it is necessary a mobility protocol capable to ensure the session continuity.

There are several mobility protocols, each one with their advantages and disadvantages. To be ideal, the chosen mobility protocol should include the features depicted as following according to Zhu et al. [48]:

- `Mobility without packet loss`: VANETs should be an extension of the Internet and the vehicle mobility should, regardless of the technology used by the car to connect to the Internet, it should be able to maintain its Internet Gateway available in order to not lose any packet.

- `IPv6 support`: IPv6 was developed taking mobility into account, by supporting auto-configuration and routing extension headers.

- `Smooth and fast handover`: in order to support seamless handover between APs of the same or different technology, horizontal or vertical handover, respectively, it is needed a mobility protocol to do the handover fast and smooth without being noticed by the users and its sessions.

- `Efficiency and scalability`: as vehicular networks can have thousands of connections at the same time, a mobility protocol with a highly scalability and efficiency is mandatory.

The next sub-sections describe some of the main approaches for mobility.

2.6.1 Terminology

The following terminology is used according to [28]:

- `Mobile Node (MN)`: is a node capable to roam into different networks changing its location to another point of attachment.

- `Correspondent Node (CN)`: is any node that communicates with the MN; a MN can be a CN and a CN can also be a MN depending on the scenario.

- `Home address (HoA)`: is a permanent address assigned to the MN and is used by the CN to reach MNs because is the only address which is maintained regardless of the point of attachment. Further, as it happens in all IPv6 addresses, this home address has a 64 bit prefix which represents his Home Network and the suffix represents his node identifier. When a packet is sent to the home address, the routing is done through his home network prefix.

- `Home agent (HA)`: a router on the Home Network that enables the MN to roam; this means that this router knows the information about the MN while he is in the visit network.

- `Home Network (HN)`: is a network where the MN belongs when it is not roaming.

- `Foreign Network`: is any network visited by the MN without to be the HN.

- `Care-of address (CoA)`: is an address that corresponds to the location of MNs, representing at which point of attachment it is connected. Further, this address is formed by the prefix of the Home Network or the Foreign Network, depending where they are, combined with the MNs interface identifier.

- `Binding`: is the association of the MNs HoA with a CoA for a certain period of time, between the MNs current location and the stable home address.

- `Binding cache (BC)`: is a volatile memory who stores all the bindings for one or more mobile nodes. It is maintained by the informations provided from the correspondent node and the home agent. Each entry in the BC contains the MNs home address, the corresponding CoA and the lifetime that indicates the validity of that entry.

- `Binding Update (BU)`: is a message with the purpose to inform the HA of the MN's current address (i.e., CoA) [28].

- `Binding Acknowledgement (BA)`: the HA, after receiving the BU and make an association between the home address to the MN and the CoA it received, answers with a binding acknowledgement (BA).

- `Router Solicitation (RS)`: this type of message is used by a host to query information to the local routers which they will answered with a Router Advertisement (RA) containing the current routing location or perform stateless auto-configuration [36].

- `Router Advertisement (RA)`: a Router Advertisement message is used by the routers in order to answer to the RS messages required from the hosts [36].

2.6.2 MIPv6

Firstly, before addressing Mobile Internet Protocol version 6 (MIPv6) it is important to note that Mobile Internet Protocol version 4 (MIPv4) [10], one previous protocol proposed by IETF had some problems, such as short IP addresses, poor security and Quality of Service (QoS); thus the IETF created the MIPv6 [28] to deal with these problems.

MIPv6 is a protocol created as a subset of Internet Protocol version 6 (IPv6) to support mobile connection. MIPv6 is different from the IETF Mobile IP standard [7] and is designed to allow the MN to change its network while keeping the same IP address. Each MN is identified by its home address and its care-of address. The home address (HoA) is a fixed

IP address that identifies the MN independent of its location; otherwise the care-of address (CoA) changes at each new point of attachment and provides information about the Mn's current location and situation.

When the MN is away from its home network, it must acquire a CoA which presents the current location; this is performed through IPv6 Neighbourhood Discovery [36].

MIPv6 uses IPv6 routing header rather than IP encapsulation, and specifies how the MN registers in the home agent, and how the home agent sends the packets through the tunnel to the MN. There is at least one home agent who receives the HoA and the CoA of each MN.

2.6.2.1 Operation method

When a mobile node is away from the HN, it sends a CoA informing his home agent about its current location. A node that wants to communicate with a MN uses the home address of the MN to send packets. The HA intercepts these packets, checks its cache table and tunnels the packets to the MNs CoA. In order to explain deeply, it is first presented the MIPv6 support services:

- `Discovery`: the MN, every time it changes his network, it triggers an ICMP RS message in order to receive the advertisement with the CoA information and then initiate the registration.
- `Registration`: when a MN is away from home, it registers its CoA in its HA. This procedure is done by sending a BU to its HA with the CoA information obtained on Discovery services. The HA stores this information in the BC, in order to always know where this MN is located to forward the packets towards the MN. Finally, the HA sends a BA to the MN in order to validate the association between the home address and the CoA of his MN.
- `Tunneling`: when the HA sends a BA to the MN, it creates a tunnel to the respective CoA. Thus, it can forward, by this tunnel, all packets which have this MN as destination.

In this context, it is illustrated the architecture of MIPv6 in figure 2.7 and the Mobile IPv6 operation is presented the following steps:

- MN performs address auto-configuration to get its care-of address.
- Upon receiving the care-of address, the MN registers it with HA on HN using BU.

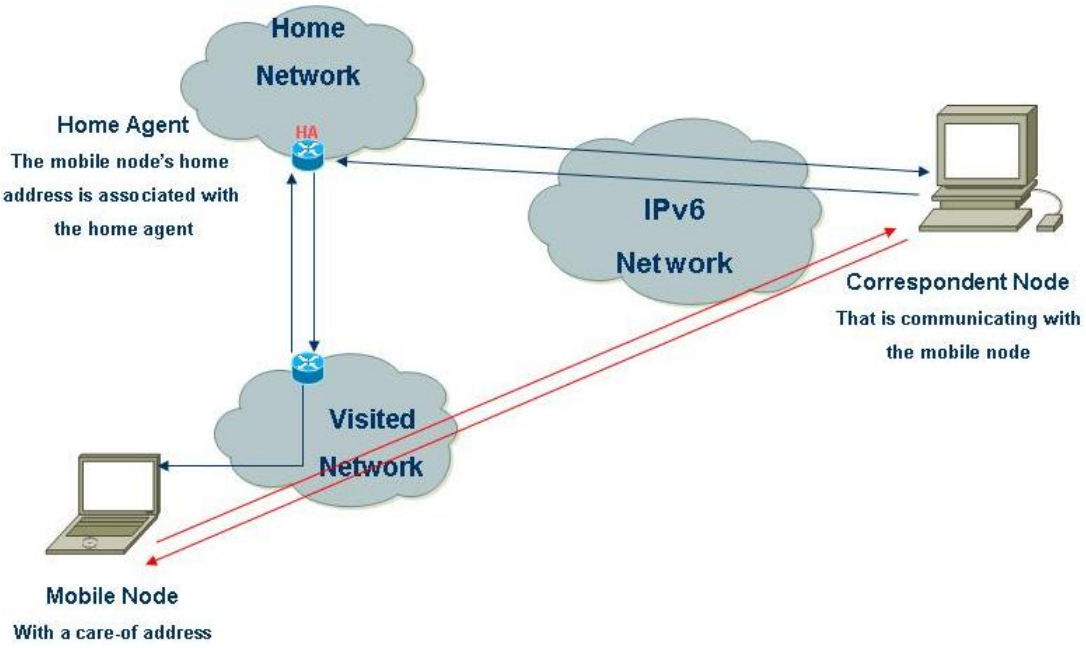


Figure 2.7: MIPv6 Architecture [25]

- The HA, using Neighbour Discovery, answers with RA to the RS required by MN.
- The BA is also sent by the HA to the MN in order to validate his registration and create a tunnel.
- The HA intercepts all packets destined for MN and sends them through the tunnel previous created on the registration.
- When the MN moves, it has to perform again all these steps, to advise his HA and the CN in order to update his new location.

2.6.3 PMIPv6

The Proxy Mobile IPv6 (PMIPv6) [20] is a network-based localized mobility management (NetLMM) protocol standardized by IETF. In order to make a solution that relocates mobility procedures from the mobile device to network components, the NetLMM working group [29] of the IETF allows vehicles with conventional IP's roaming into different APs and belonging to the same local domain.

PMIPv6 enables the same functionalities as MIPv6, but the main difference is encountered in the IP address assignment. While in PMIPv6 the hosts can maintain their IP

address when roaming into different APs, in MIPv6 the network is responsible to implement this functionality, which tracks the movements of the host and begins the required mobility signalling on its behalf; MIPv6 is a "host-based" approach while PMIPv6 is a network-based approach.

Comparing MIPv6 with PMIPv6, according to [2] it is possible to observe that, being a network-based approach, it has the following advantages:

- `Deployment`: MN does not require any modification which allows service providers to give the services to as many MNs as possible.
- `Controllability`: From the network service provider point of view, it allows them to control the network in terms of traffic and quality of service (QoS) such as differentiated services.
- `Performance`: As the network is doing the mobility management on behalf of the MN, the MN does not need to participate. Thus, the number of exchanged messages in the wireless network are reduced as well as the tunnelling overhead.

Thus, this supporting localized mobility management protocol for a MN [44] is detailed below.

2.6.3.1 Terminology

The following terminology is important to better understand how the Proxy Mobile IPv6 works. Below, it is included only the new terminology that was not present on the MIPv6:

- `Local Mobility Domain (LMD)`: Network that is PMIP-enabled. The LMD contains one Local Mobility Anchor (LMA) and multiple Mobile Access Gateways (MAGs).
- `Local Mobility Anchor (LMA)`: All traffic from and to the MN is routed through the LMA. The LMA maintains a set of routes for each MN connected to the LMD.
- `Mobile Access Gateway (MAG)`: The MAG performs the mobility related signalling on behalf of the MNs attached to its access links. The MAG usually is the access router (first hop router) for the MN.
- `Binding Cache Entry (BCE)`: Entry in the LMA's BC. Each entry has the fields MN-ID, MAG proxy-CoA and MN-prefix.

- `Binding Update List (BUL)`: Cache maintained by the MAG which contains information about the attached MNs.

- `Proxy Binding Update (PBU)`: PMIP signalling packet sent by the MAG to the LMA in order to indicate a new MN. The PBU has the fields MN-ID (e.g. MN MAC), MAG address (proxy-CoA) and handoff indicator to signal if the MN-attachment is a new one or a handoff from another MAG.

- `Proxy Binding Acknowledge (PBA)`: Answer to a PBU sent by the LMA to the MAG. The PBA contains the MN-ID, the MAG address and the prefix assigned to the MN.

- `Proxy care of address (proxy-CoA)`: IP address of public interface of MAG. The proxy-CoA is the tunnel endpoint address on the MAG. The LMA encapsulates packets destined to the MN into a tunnel packet with destination address equal to Proxy-CoA.

- `Mobile Node Identifier (MN-ID)`: The only identifier of mobile node, e.g. one of its MAC addresses.

- `Home Network Prefix (MN-HNP)`: Prefix assigned to the MN by the LMA.

2.6.3.2 Operation method

PMIPv6 operation method is represented in figure 2.8. The network-based mobility management support protocol to an MN has two main entities, the LMA and the MAG. The LMA, acting as HA in PMIPv6, is usually the anchor point for the MN prefix assignments with the functionality of maintaining the informations and the state of the MN. MAG is the attachment point between the MN and the network, and is responsible to send informations to the LMA regarding MN movements and consequently registering him there. The following steps show the operation method applied in PMIPv6 protocol according to [42]:

- When the MN attaches to one MAG, the MAG detects the attachment and triggers a Proxy Binding Update Message (PBU) to the LMA. The LMA processes the PBU message, assigns the MN with the home network prefix, stores this entry in the internal cache table, and answers to MAG with a Proxy Binding Acknowledgement (PBA) containing the home network prefix. Moreover, with these two messages, PBU

and PBA, a bidirectional tunnel is created by the LMA with the MAG, that can be used for forwarding traffic.

- As soon as the MAG receives the PBA, it sends a RA message to the MN with the available prefixes for the MN to create his IP address, combining the prefix with his own and permanent address at suffix. Once the MN IP address is formed, it becomes ready to send and receive data packets.

- Thus, when any packet is sent to the MN, the LMA checks his internal cache to know where is this MN. Once it is known, the packets are sent to the respective MAG which removes the outer header, and forwards the packets to the MN. The same can happen in reverse; the MN can send packets to the MAG which forwards them to the LMA using the tunnel. Then, the LMA removes the outer header and routes it to the CN.

- Once the MN leaves or changes the network, the MAG detects it and alerts the LMA sending a deregistration message. Then when the MN attaches again to another network, every step described here is repeated.

As explained in this subsection, PMIPv6 solves most of the issues of the MIPv6 protocol, but it is still not ideal to VANETs due to the fact that it just provides mobility to the MN and not to the entire network: this means that vehicles mobility is ensured, but the passengers mobility is not addressed. Thus, N-PMIPv6 was proposed in [42], which has been extended and developed in our group [34]. The next subsection will describe N-PMIPv6.

2.6.4 N-PMIPv6

N-PMIPv6 extends PMIPv6, which has been previously submitted to real applications evaluations on our group in a previous MSc Dissertation [34], to support network mobility. It introduces the mobile MAG (mMAG) and maintains the two entities, LMA and MAG. Figure 2.9 illustrates the operation of the N-PMIPv6 protocol.

2.6.4.1 Operation method

According to [42] and [26], the registration and handover procedures are executed as follows:

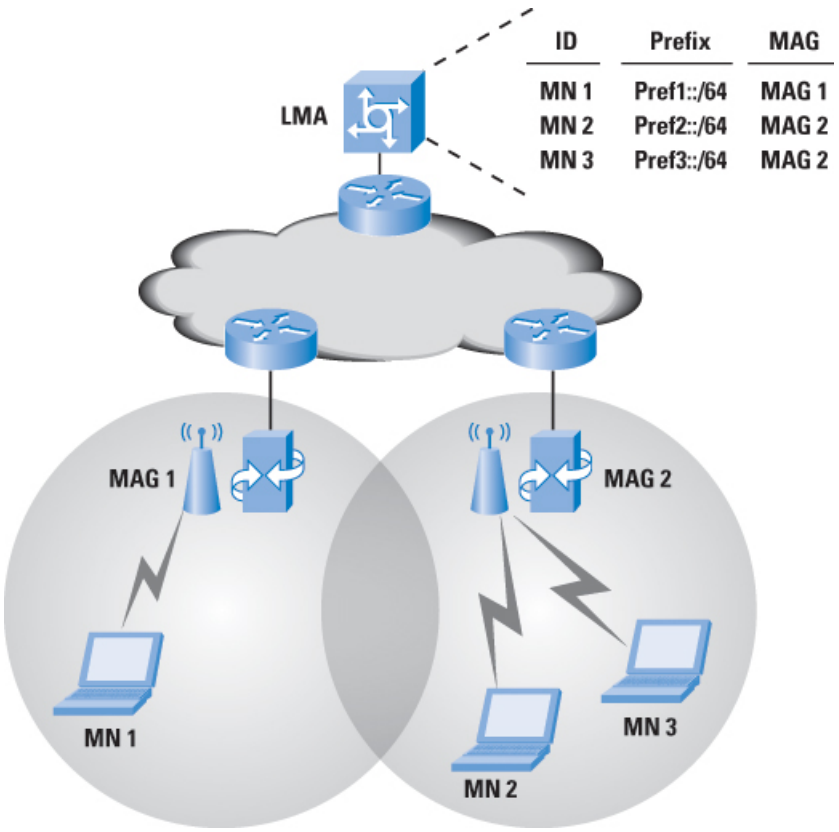


Figure 2.8: PMIPv6 Architecture [22]

- When a mMAG with a MN attaches to the MAG, the MAG sends the PBU message containing the mMAG-ID to the LMA.

- Upon receiving the PBU, the LMA assigns the mMAG the HNP-1 and creates the BCE. Next, the LMA returns the PBA to the MAG.

- Upon receiving the PBA, the MAG sends the RA message containing the HNP-1 to the mMAG.

- Upon receiving the RA message, the mMAG sends the PBU message containing the Mobile Network Node (MNN)-ID to the LMA.

- Upon receiving the PBU message, the LMA assigns the MNN the HNP-2 and creates the BCE. N-PMIPv6 adds a new field, the M flag, to the BCE. The M flag of MNN BCE is set to indicate that the MNN is connected to a mobile network.

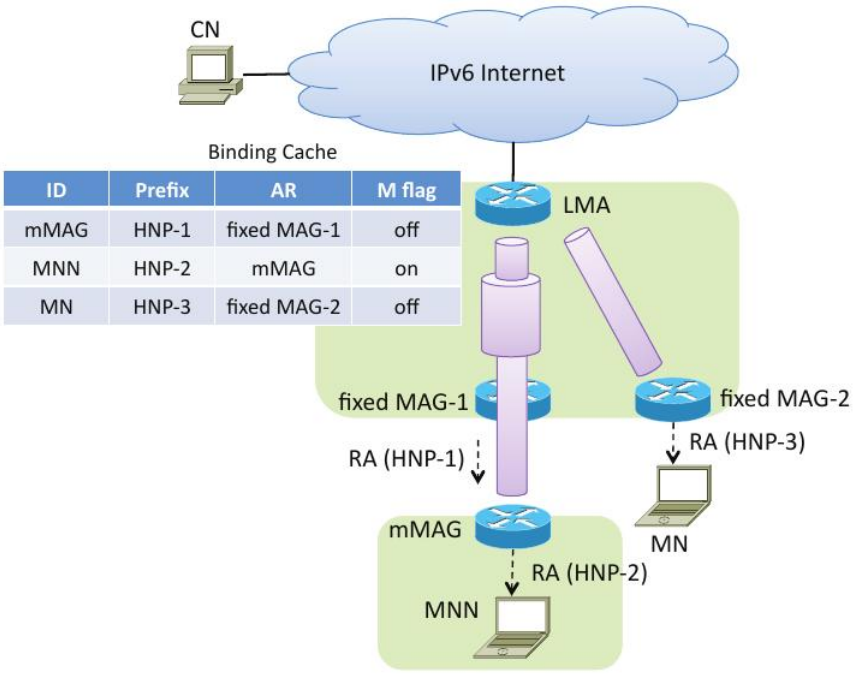


Figure 2.9: N-PMIPv6 Architecture [42]

- Next, the LMA returns the PBA to the mMAG. Upon receiving the PBA, the mMAG sends the RA message containing HNP-2 to the MNN.

- The data packet destined to the MNN first reaches the LMA. The LMA finds the MNN BCE. Since the M flag is "on" in the MNN BCE, the LMA searches for the mMAG BCE. Next, the LMA encapsulates the packet for tunnelling to the mMAG and encapsulates it again for tunnelling to the MAG. The LMA forwards the packet to the fixed MAG. The fixed MAG removes the outer tunnelling header and forwards it to the mMAG. The mMAG retrieves the original packet and forwards it to the MNN.

- When the mMAG moves to the another MAG, the same procedures as in the initial registration are performed. In this procedure, the AR field of the mMAG BCE is updated from MAG to the another MAG. Other fields of mMAG BCE and MNN BCE remain unchanged. Thus, in N-PMIPv6, the signalling messages are not sent on the wireless link when a handover occurs.

In PMIPv6 the MAGs are static entities directly connected to the LMA, and it does not allow chaining MAGs, thus the mobility is not ensured in the whole network.

The PMIPv6 protocol needs to be modified to ensure network mobility, mobility to the OBUs, which represents the vehicles, and to respectively users. For this purpose, it is necessary that the mMAG be capable to configure itself according to the attachment point which it is connected. In addition to these changes, the PMIPv6 MAG must be modified to acquire these features.

Given these facts and according to [34], the necessary changes made to ensure mobility in entire network are:

- LMA must be able to recognize mMAGs and be able to create tunnels to these mMAGs as if they were ordinary MAGs.
- The MAG must be able to identify whether it will intercede as a MAG or as a mMAG.
- In case it operates as mMAG, the mMAG has to be able to identify its IPv6 prefix assigned on the network where it is connected, in order to configure its own IPv6 address, so that it will be able to communicate with the LMA and consequently the Internet.
- As a mMAG, it must also have a RS filtering system.

Despite having guaranteed mobility in the whole network it is still not an ideal protocol with scalability issues as well as issues of resources, since all traffic goes through the LMA. Furthermore, it is a protocol that does not support multihoming which further validates the fact of overload of resources.

The last subsection of this chapter presents a distributed mobility protocol that avoids and overcomes the issues described above.

2.6.5 DMIPA

In order to overcome the issues provided by centralized mobility protocols, such as non-optimal routes, scalability, network bottlenecks, single point of failure and attack, it was proposed Distributed Mobility IP Anchoring (DMIPA) protocol [41]. DMIPA is a protocol for dynamic environment developed by our group, and it is in development and testing phase.

According to [41], DMIPA is a new approach based on the host that aims to provide distributed mobility management in heterogeneous and flat networks. The DMIPA's architecture presented in figure 2.10 is comprised mainly by two entities, the Data Mobility

Access Router (DMAR) and the MN. The DMAR is an access router (AR) with IP mobility management functions, and together with the MN, it is responsible for maintaining the continuity session. The MN can move through the heterogeneous network changing its attachment point while still reachable.

In order to explain the DMIPA protocol, it is needed to add the following messages that were not presented in the previous protocols:

- `Mobility Support Flag (MSF)`: this flag is introduced in the Reserved field of DMIPA'S RA message in order to provide useful information to know if it is a DMAR or an AR. If the MSF is set to zero, then it is a legacy AR; otherwise MSF is equal to one which representing a DMAR.

- `Anchor Set Update (ASU)`: this message, as well as the next one, are exchanged when MN and the current DMAR communicates with each other. ASU message is sent by MN providing its attached DMAR with the IPv6 addresses of the current set of DMARs.

- `Anchor Set Acknowledgement (ASA)`: this is a message from the DMAR to MN in order to answer to the ASU message, which indicates the success of the process.

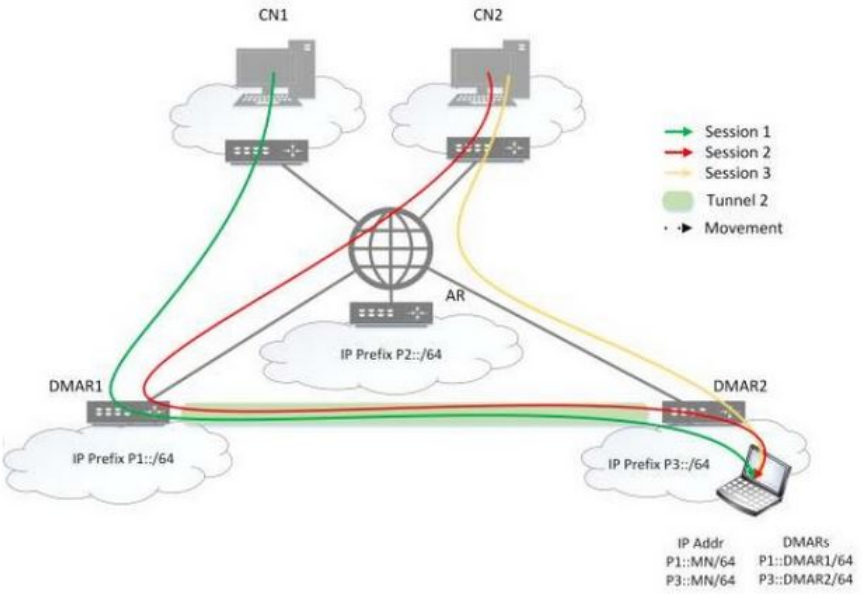


Figure 2.10: DMIPA Architecture [8]

Taking into account [41], DMIPA protocol has the following features:

- MNs and ARs have the IP mobility functionalities with mobility support known as DMARs.
- IP mobility is performed for ongoing sessions while the handovers occurrence.
- A new session is always anchored to the new DMARs while the ongoing session is maintain anchored to the previous DMAR.
- When the MN connects to a DMAR, it is guaranteed the forwarding of the ongoing sessions from the previous DMAR; otherwise these functions are supplied by the MN.
- There are no centralized databases, and MNs keep their mobility context.

Moreover, according to [41], the protocol operation method considers the movement of the MN from DMAR1 to AR, and then to DMAR2 presented in figure 2.11 and described below:

- MN is attached to DMAR1.
- MN requests the network prefix by sending a RS message. Then, DMAR1 replies with a RA message which contains the network prefix P1::/64 and a true MSF value.
- Upon receiving the RA, the MN configures the IPv6 address P1::MN/64 as a preferred address.
- MN adds the IPv6 address of DMAR1 to the database which contains the available DMARs set.
- MN starts data session 1 using P1::MN/64 as IPv6 source address.
- MN attaches to a legacy AR.
- MN sends RS message and receives RA with the network prefix P2::/64 and negative value of MSF.
- MN configures the P2::MN/64 address; however the P1::MN/64IPv6 address is remained as the preferred address.
- DMAR1 receives BU from MN to establish a tunnel (Tun1), and then DMAR1 sends BA to confirm the success.

- It remains active the data session 1, and the traffic flow from/to P1:MN/64 is tunnelled from/to P2::MN/64.
- It is started a new session, the data session 2, using the P1::MN/64 IPv6 address in order to provide session continuity if the MN changes its attachment point. Therefore, data session 2 is tunnelled from the beginning.
- MN attaches to DMAR2.
- To obtain the network prefix, the MN sends a RS to DMAR2 which replies with a RA message containing the IPv6 prefix P3::/64 and a true MSF value.
- MN performs the configuration of the IPv6 address P3::MN/64 as the preferred IPv6 address.
- MN adds the IPv6 Address of DMAR2 to the set of available DMARs IPv6 address list.
- MN sends an ASU message to DMAR2 containing DMAR1 IPv6 address information (P1:DMAR1/64) and with the respective MN IPv6 address (P1::MN/64) which DMAR2 answered with an ASA message to the MN to confirm the success.
- DMAR2 sends BU message to DMAR1 to establish a tunnel which DMAR1 replies with BA message, and then it is created a tunnel (Tun2) between those DMARs.
- Both sessions, data session 1 and data session 2 are maintained through a tunnel between DMAR1 and DMAR2.

In sum, according to [41], this distributed mobility protocol has better results to vehicular networks than MIPv6.

2.6.6 LISP

Nowadays, the Internet architecture is starting to present some problems which could not be foreseen in the past and which are strictly related to its nature. One of the biggest one is regarding to routing scalability.

Further, IPv6 cannot solve this issue, because IPv6 did not change anything regarding the usage of IP addresses; it remains representing the location and the identification of the host at the same time, with no logical division, and so it still suffers the same problems as IPv4.

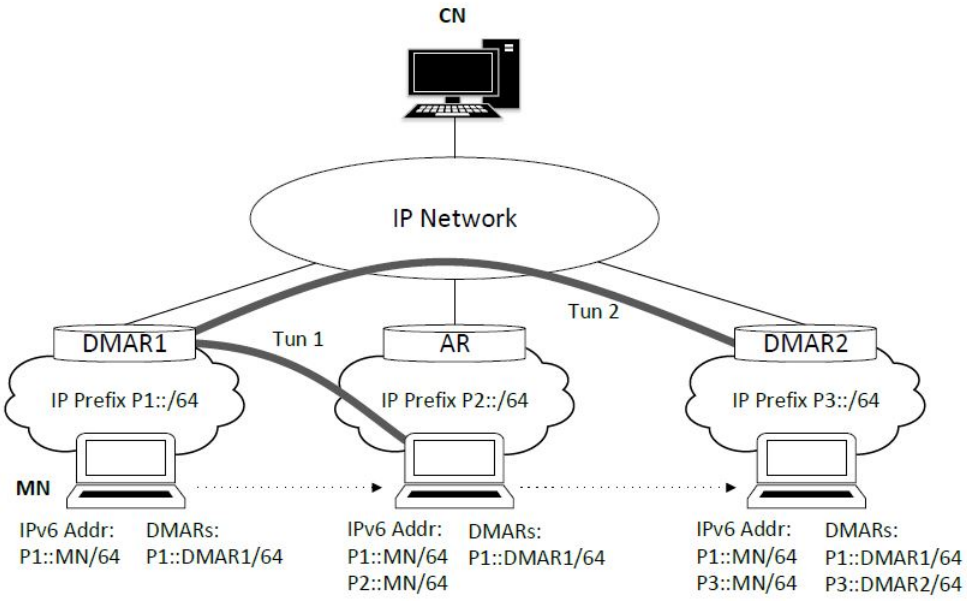


Figure 2.11: DMIPA Operation Method [41]

The Locator/ID Separation Protocol (LISP) is a Cisco protocol which is being developed as a potential solution to the routing scalability problem in the current internet. It splits the traditional IP into two new different name spaces, syntactically indistinguishable from the current internet addresses and compatible to their architecture, which are the Endpoint Identifiers (EIDs) to name hosts in edge networks, and Routing Locators (RLOCs) for the nodes in the transit networks. Further a distributed database, the mapping system, is responsible for maintaining the associations between the RLOCs and EIDs.

Thus, compared to all protocols mentioned above, LISP-MN designed to provide scalable mobility for LISP mobile nodes has a set of advantages according to [3]:

- LISP splits host identity from its location, so it allows LISP multihoming. With multihoming every node can be attached to one more access point; each EID can be mapped and reachable through many RLOCs.

- LISP divides the control plane from the data plane, which enables each part to scale independently. Since LISP-MN does not require Foreign Agent or Home Agent network components in the data plane, it avoids triangle routing at the data plane level for IPv4 and IPv6 addresses. Moreover, the data packets are usually forwarded for the shortest path, and thereof LISP-MN incorporates natively route optimization

support.

- The MIPv4 and MIPv6 protocols supply basic and advanced functionality to MNs with advanced features, such as [31] and [19].
- With the separation of control plane from data plane in LISP-MN, the decoupling of end-point identity from the mobility service provider becomes easier. The only functionality of the control plane is to locate a mobile node. Identical to DNS, LISP control plane has a distributed and federated mapping system nature. This distributed nature, when compared to already existent alternatives, renders LISP-MN as a more transparent and open solution. LISP communication at the data plane level does not depend on a specific mobility service provider.
- No changes are required to the host protocol stacks or to the internet infrastructure.

Since LISP is the protocol being developed in this dissertation, it will be detailed in chapter 3.

2.7 Chapter Considerations

This introductory chapter described several topics concerning the work already done up to date focusing on the subject of this dissertation, in mobility and vehicular networks.

In this context, it presented several features about VANETs, the equipment was detailed as well as their network architecture. Then, the network access technologies were mentioned, emphasizing a new access technology, the IEEE 802.11p (WAVE) which has been specially developed to support the unique features of these networks. However, there are just a few real studies containing this standard, so it is imperative that the current existing protocols are evaluated and adapted to that new access technology in order to accelerate VANETs deployment. It is also necessary to test the existent mobility protocols with WAVE access technology on the vehicular scenarios to find the most suitable for these networks, to ensure mobility for the vehicles and their passengers as well as to obtain reduced handover times. Hereupon, it was summarized and compared several mobility protocols, which the main details are presented in the table 2.1. Note that in chapter 5, LISP is evaluated in the vehicular networks with multi-technology handover according to this MSc Dissertation.

The next chapter focuses on the description of the mobility protocol chosen to adapt to vehicular networks, known as LISP.

Table 2.1: Comparison between mobility protocols

Protocol Criteria	MIPv6	PMIPv6	N-PMIPv6	LISP
Location management	Yes	Yes	Yes	Yes
Mobility Scope	Global	Local	Local	Global
Required elements	Home Agent	LMA,MAG	LMA,MAG,mMAG	MS,MR,xTR,MN
MN modification	Yes	No	Yes	Yes
Localized Routing	Yes	No	No	Yes
Handover latency	Bad	Good	Good	Good

Chapter 3

Locator/Identifier Separation Protocol

In this chapter it will be described the LISP mobility protocol.

Section 3.1 briefly introduces the LISP mobility protocol, and section 3.2 describes the LISP components and their functionalities.

Section 3.3 illustrates how the protocol mainly works, showing which messages are presented and how LISP works.

Section 3.4 describes how mobility is guaranteed, showing how it is processed as well as the possible scenarios that can exist.

Finally section 3.5 summarizes the topics described.

3.1 Overview

This section presents an overview of LISP protocol.

The main drivers of this proposal are the scalability issues of the current Internet's routing infrastructure, as well as the possibility to perform multihoming.

In addition, the idea of using a single IP address for both identifying a device and where this device is located in the whole network topology began to fail, because it required topological address assignment and a limited margin for topology changes. Here is when LISP appeared, solving this necessity of separating the device identifiers and its location in the network.

LISP, according to the authors [13], is based on the idea of splitting the current routing and addressing architecture into non-routable EIDs, which define the endpoint network devices, and routable RLOCs, which describe how a device is attached to the network. As we noticed, RLOCs are addresses used by network elements and define where in the routing topology a destination node is to be found; otherwise EIDs represent the identity

of the node, regardless of its location, and are used as addresses in the endpoint devices. In order to be incrementally deployable and with no changes or problems to end systems, RLOCs and EIDs are both using the IP address space, either IPv4 or IPv6.

In order to reach a host, identified by its EID, one must first find the current location (RLOC) of the host. LISP provides a publicly accessible Mapping System that is responsible to serve the EID-to-RLOC mapping information. Basically, that happens because it is a mapping and encapsulation protocol (map-and-encap). In the map-and-encap scheme, when a source sends a packet to the EID of a destination not found in the source cache, the packet traverses the mapping system infrastructure which has the RLOC of the corresponded EID. Once RLOC associated to an EID is discovered, packets with headers from the EID namespace are encapsulated in a second header from the RLOC space, and are routed to the destination, where the LISP header is removed before delivering packets to the destination device. LISP introduces gateway routers, called Tunnel Routers, that perform the LISP encapsulation, and decapsulation at each site's ingress and egress points. These gateways either act as ingress tunnel router (ITR) or as egress tunnel router (ETR). ITRs tunnel packets to others LISP gateways which then act as ETRs; this means that ITRs make the encapsulation from EID to routing network, unlike ETRs make the decapsulation from routing network to EIDs.

On an ongoing connection, the location of the host can change many times, so splitting the host identity (EID) from its locator (RLOC) enables seamless endpoint mobility by allowing the applications to bind to a permanent address, the host's EID. In case of location changes, the LISP tunnel routers will encapsulate the packets to the new RLOC, preserving the connection session alive.

The basic LISP architecture by itself does not support mobility. Recently, the mobility extension LISP Mobile Node (LISP-MN) [14] was presented in LISPmob group [3]. It describes a mechanism that enables LISP mobile nodes to roam into LISP and non-LISP networks while being reachable under the same identifier address. Indeed, LISP architecture, as described in figure 3.1, allows not only LISP-to-LISP communication, but also LISP-to-non-LISP, as we will following observe.

3.2 LISP Network Elements

The LISP specification bases itself on a few fundamental network elements, described below based on [13]. They are:

- `Ingress Tunnel Router (ITR)`: is a router that accepts IP packets from site end-

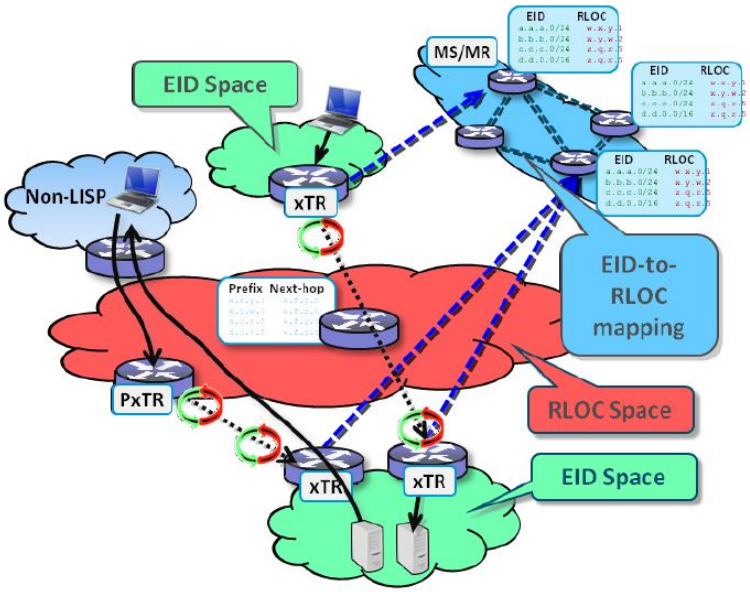


Figure 3.1: LISP Architecture [3]

systems on one side, and sends LISP-encapsulated IP packets towards the routable network to the other side. The ITR treats this "inner" IP destination address as an EID, and performs an EID-to-RLOC mapping lookup if does not have already an EID-to-RLOC mapping for the EID in its cache. After this EID-to-RLOC search, a LISP routing cache introduces a new binding entry; this cache contains the EID-to-RLOC mappings for destination EIDs which have already communicated with it. In case the LISP Cache does not have the mapping for the destination EID, it will be the LISP Mapping System who takes charge of obtaining it on behalf of the ITR.

- `Egress Tunnel Router (ETR)`: is a router that receives LISP-encapsulated IP packets from an ITR, decapsulates and sends the decapsulated IP packets to EID destination.

- `Proxy Ingress Tunnel Router (PITR)`: this is a LISP ITR that allows non-LISP sites to send packets to LISP sites without any changes to protocols or equipment at the non-LISP site. It acts as the ITR for traffic received from the public Internet (non-LISP sites).

- `Proxy Ingress Tunnel Router (PETR)`: this is a LISP ETR that allows LISP sites to send packets to non-LISP sites. It acts as the ETR for the traffic received from the LISP sites.

- `X Tunnel Router(xTR)`: router that can perform ITR or ETR functionalities; this happens when direction of data flow is unknown.
- `LISP Map Cache`: is a virtual table in an ITR that stores and is responsible for EID-to-RLOC mappings entries and their time-out. This cache is different from the database on the mapping system, it is dynamic, local to the ITRs, and relatively small while the other is distributed, relatively static, and much more global in scope.
- `LISP Site`: is a set of routers and devices in an edge network that are under a single technical administration. Furthermore, LISP architecture is separated into LISP sites in the edge network which EIDs are inserted, core network which is responsible to routing, and RLOCs and the mapping system, and finally the non-LISP sites where public internet devices are encompassed.
- `EID-to-RLOC Database`: is a global distributed database in the mapping system that contains all known EID-prefix to RLOC mappings. Each potential ETR typically contains a small part of the database: the EID-to-RLOC mappings for the EID prefixes "behind" the router.
- `Map-Server`: is a network infrastructure component which learns EID-to-RLOC mapping entries from an ETR. Further, the Map-Server publishes these mappings in the distributed mapping database.
- `Map-Resolver`: is a network infrastructure component that receive LISP Encapsulated Map-Requests, usually from an ITR, and determines whether or not the destination IP address is part of the database; if it is not the case, a Negative Map-Reply is returned. Otherwise, the Map-Resolver finds the appropriate EID-to-RLOC mapping by consulting a mapping database system.

3.3 LISP Encapsulation Messages Details

When a host in a LISP capable domain emits a packet, it inserts its EID in the packets source address, and the EID of the correspondent host in its destination address. Then, the ITR maps the destination EID to a RLOC which corresponds to an ETR which is either in the destination domain or proxy's for the destination domain. It is also possible that the MN does the encapsulation and decapsulation instead of ITR and ETR, but this mostly happen in one hop mobility, not precluding the possibility to making into more than one hop. When the packet arrives at the destination ETR, it is decapsulated and

sent to the EID destination. Figure 3.2 shows the packet format when an IPv4 packet is LISP-encapsulated in another IPv4 packet. There are some LISP packet messages to take into account, which are:

- `Map-Request`: when an EID tries to reach another one in another LISP site, the ITR may query the mapping system by sending a Map-Request message into the mapping system to request a particular EID-to-RLOC mapping. To make this happen ITR is responsible to encapsulate the Map-Request message before being sent to the Map-Server: the outer IP header contains the RLOC of the requesting ITR and of the Map-Server, in order to route the packet correctly to the destination. As soon as the Map-Request is received by the Map-Server, it is decapsulated and read. The Map-Server will look for the EID prefix requested in the database. If the Map-Server does not contain the EID requested, the Map-Request will be forwarded into the Mapping System until it is found; in case it is not found, a negative Map-Reply message is received.

- `Map-Reply`: this message is used to "answer" to the requesting ITR, sending back the EID-to-RLOC mapping requested, in case that this binding is found in the mapping system as mentioned before. This message is sent straight to the ITR and therefore to EID-prefix without encapsulation. Further, it is important to mention that, to find the EID-to-RLOC mapping in the mapping system, the EID through ETR must be registered on that previously, as soon as it starts the connection.

- `Map-Register`: this message is sent by an ETR to a Map-Server to register its associated EID-Prefixes. In addition this message brings the RLOC available to reach any EID behind the corresponding EID-prefix forming the EID-to-RLOC binding. This RLOC is needed to be used by the Map-Server in order to answer forwarding Map-Requests received through the database mapping system. An ETR may request that the Map-Server respond Map-Requests on its behalf by setting the proxy Map-Reply flag bit in the message.

- `Map-Notify`: this message is a Map-Register answer sent by a Map-Server to an ETR to confirm that a registration has been received and processed.

In the User Data Protocol (UDP) packet formats, used by the LISP control plane, inside of LISP Message field we can find LISP control message formats which are represented in figure 3.3.

The main field in the figure 3.3 is the Type field which can be:

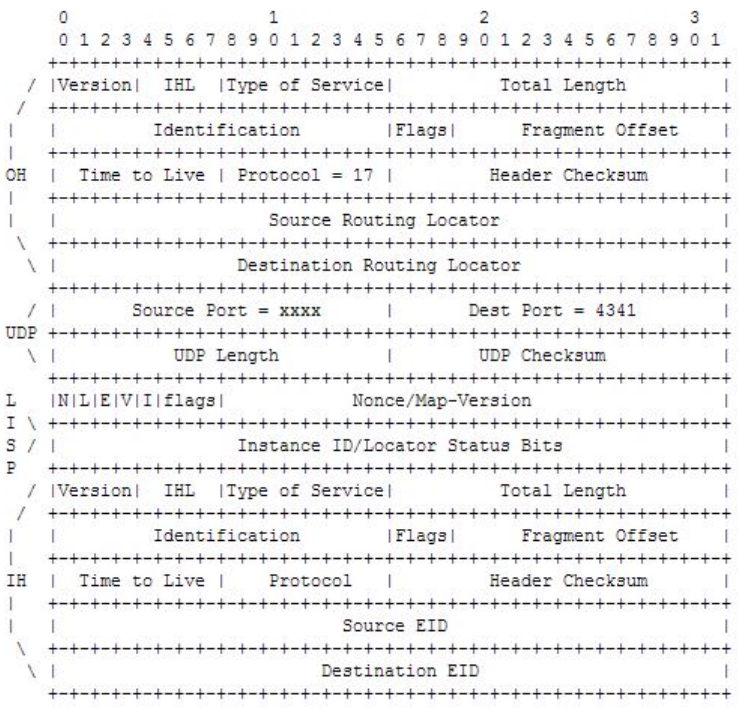


Figure 3.2: LISP IPv4-in-IPv4 Header Format [13]

- 1 - LISP Map-Request
- 2 - LISP Map-Reply
- 3 - LISP Map-Register
- 4 - LISP Map-Notify
- 8 - LISP Encapsulated Control Message

In addition, each letter also triggers a bit, for instance the type field with number 1 and letter S is the Solicit-Map-Request (SMR), which is an important bit in the handover testbed process. Further, in the LISP implementation section, it will deeply described.

3.4 LISP-MN

The basic LISP architecture does not support mobility of end hosts as it was previous mentioned. Nowadays, it is possible to support mobility because the extension LISP Mobile Node (LISP-MN) has already been done by LISPmob organization.

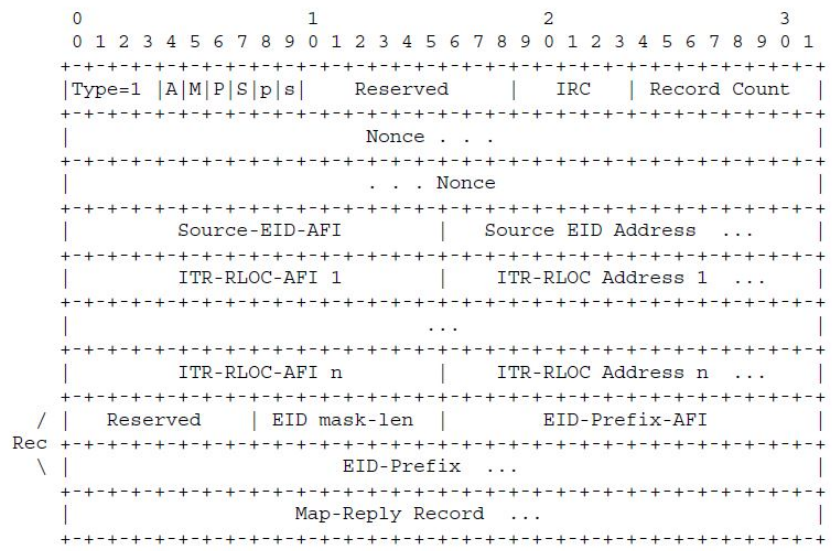


Figure 3.3: LISP Control Plane Messages Format [13]

3.4.1 Introduction

LISP-MN [14] enables MNs to have a permanent EID while roaming into LISP and non-LISP sites; this means that the MN can be always reachable, even whether it changes the network and consequently its point of attachment because the EID address remains the same independent of the network and the attachment point which is connected.

EID is used for identification but not for forwarding. Forwarding is provided by the RLOCs which represent the location of EIDs and are used for routing. Thus, LISP provides support for location/identity separation making it a suitable mobility protocol.

LISP-MN assumes that a MN forms a separate LISP domain and implements the ITR/ETR functionality for incoming and outgoing traffic. For example, to send traffic, a MN must encapsulate outgoing traffic to some ETR or PETR, and it must be configured with the RLOC of ETR or PETR. Besides, for receiving traffic, the traffic must be tunnelled to the MN from some ITR, PITR or even from another MN.

The current point of attachment to the network defines the current RLOC for the MN. The location of the host can change several times during an ongoing connection without breaking the connection. When the hosts location (RLOC) changes, the LISP-MN will encapsulate the packets towards the new RLOC. This is done through the LISP Mapping System, a distributed database that contains EID-to-RLOC bindings, which has always the latest RLOC for the MN's EID. Moreover, this also happens because the MNs register

their currently valid locator at their configured Map-Server and refresh this information by sending Map-Register messages as soon as they are connected to one or more new attachment points.

3.4.2 LISP Mapping System

The LISP Mapping System [45] is a central part of the LISP-MN architecture, and it is an accessible service that stores and gives location information associated with EIDs (EID-to-RLOC mappings). In the LISP mapping system, the included elements are Map-Servers and Map-Resolvers. The EID-to-RLOC mappings are stored in Map-Servers, and in the case of existing more than one Map-Server, each one is associated with a portion of the EID name space, and stores the location information for those EID prefixes, forming a partition EID-to-RLOC bindings. Thus, with a distributed mapping system (with more than one Map-Server), the scalability issues could be avoided. Further, each LISP MN is associated with a specific Map-Server where it registers its EID-to-RLOC mapping, and updates it according to its movement. In order to do that, Map-Servers have assigned a set of prefixes (EIDs) and delegate them to LISP tunnel routers or to MNs.

Map-Resolvers are used as an interface to the mapping system for looking up the EID location information. This function has similar functionality as DNS resolvers have in today's Internet. For instance, the LISP MN sends EID Map-Request to the mapping system through Map-Resolver; therefore, this EID lookup is going across the mapping system to the respective Map-Server which will reply with the respective RLOC for the requested EID.

3.4.3 Registering EID and obtaining an RLOC

Each time that a MN roams across providers, it remains with the same EID, but otherwise it gets a different RLOC in each location it is attached. In that context, it is required a previous registration by all MNs in the Mapping System. This registration is a LISP message called Map-Register, which includes a EID-to-RLOC binding; this means that the carried message is filled with a permanent EID and its respectively location, where it is connected (RLOC).

When the MN is moving, it is constantly changing its position, so it is mandatory to register the new location, the new EID-to-RLOC binding into the Map-Server every times this occurs. Thus, in every new RLOC a LISP Map-Register message is triggered by the MN in order to Register it and the new location to be reachable by others MN when they

require their EID for further communications. In the multihoming case, the MN is connected to several attachment points at the same time, thus it may include multiple RLOCs in the Map-Register message.

However, it is important to know that LISP-MN and the Map-Server share a preconfigured key in the previous settings, which is made to ensure the authentication. Therefore, if the key does not have a match validation, the Map-Register is not recorded in the Map-Server database.

Further, the LISP Map-Notify message is triggered by the Map-Server to answer to the Map-Register. Upon receiving this message, the MN is aware if the registration is valid or invalid.

To be familiar with the register process described above, the figure 3.4 illustrates this process.

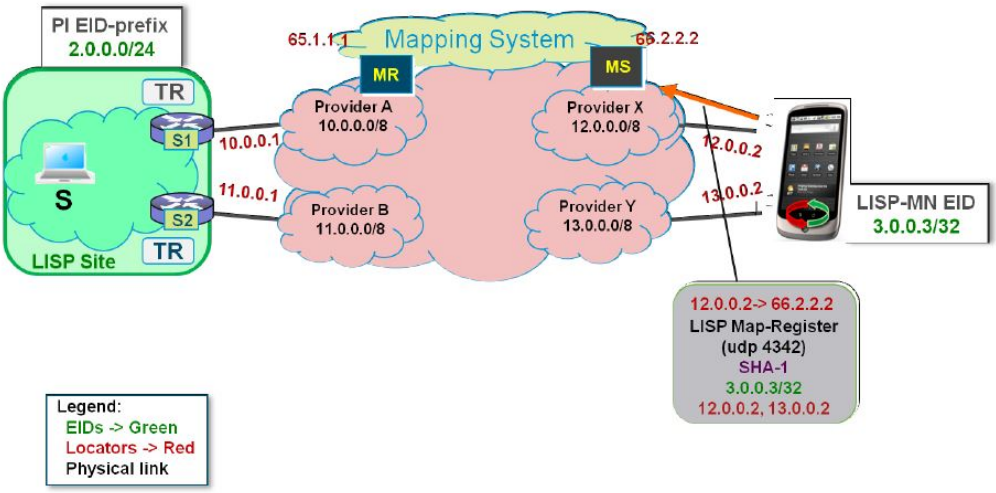


Figure 3.4: Registering an EID-to-RLOC bindings [3]

3.4.4 Signalling EID-to-RLOC bindings and transmitting data-packets

We are already familiar with LISP messages described before, but here it will be explained the LISP procedure with some examples.

First of all, as it is described in picture 3.5 after retrieved the destination EID, it is possible to transmit packets to this EID. The packet transmitted by the EID of the static node (SN) to the EID of the MN is routed to the tunnel router (TR). Upon reception of

this packet, TR checks if it has an EID-to-RLOC for this EID in its Map-Cache. If the Map-Cache does not contain the mapping, a Map-Request message is triggered in order to discover which is the location associated to the required EID of the MN. This message will query the mapping system (Map-Resolver and Map-Server typically co-located) to search and find a valid EID-to-RLOC binding for this EID. Once this binding is found, a Map-Reply message, presented in figure 3.6, is triggered from the Map-Server to the TR, which contains the actual location of the required MN; further the TR stores the received EID-to-RLOC binding in order to avoid a future Map-Request message while the MN does not change his position.

Despite RLOC being the main element in the Map-Reply message, there are others to take into account, such as Time-To-Live (TTL), the EID of the MN required and the priorities and weights of each locator, if there are more than one (multihoming). TTL is the time which the stored EID-to-RLOC binding is valid; afterwards a Map-Request message is triggered again to update that. The priorities and weights are previous assigned to each locator: usually they are equal for all of them, but they can be different, for example in multihoming case if it is required to chose a favourite point of attachment to establish communications instead of another.

Once RLOC is discovered by the TR, the SN is able to route packets until the MN through a created tunnel. Furthermore, if the SN is sending packets to the MN, first the packets are going straight to the TR which has in Map-Cache the respective RLOC of the required EID, and use that to encapsulate packets towards the MN until the TTL expires or until the MN remains attached to the same location. Therefore, when the MN moves, such as handover, another messages are exchanged to realize that. These messages are described in the handover processes section.

3.4.5 Deployment Scenarios

This section will present several distinct connectivity scenarios considered by the LISP-MN design [17].

There are many different possible scenarios regarding LISP-MN mobility, such as:

- A MN in a non-LISP domain communicates with a SN in a LISP domain.
- A MN in a non-LISP domain communicates with a MN in a LISP domain.
- A MN in a non-LISP domain communicates with a non-LISP node.
- A MN in a LISP domain communicates with a non-LISP node.

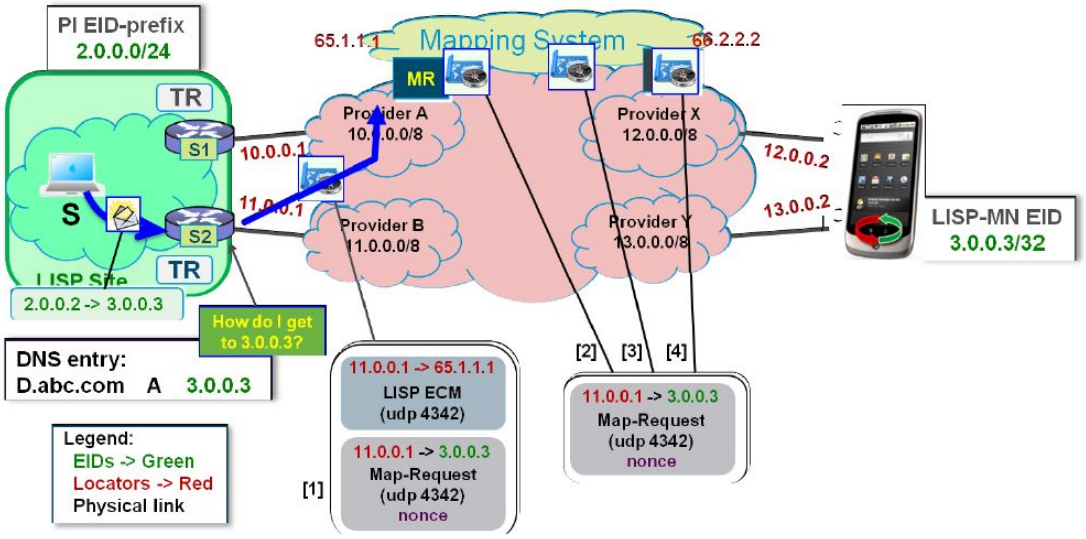


Figure 3.5: Map-Request example [3]

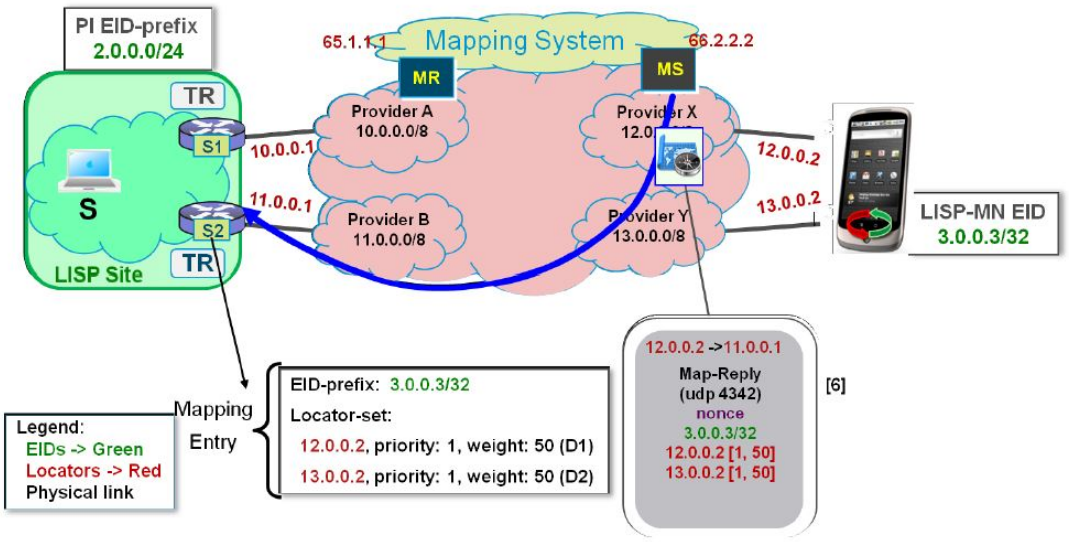


Figure 3.6: Map-Reply example [3]

- A MN in a LISP domain communicates with a SN in another LISP domain.
- A MN in a LISP domain communicates with a MN in another LISP domain.
- A MN in a LISP domain communicates with a SN in the same LISP domain.

- A MN in a LISP domain communicates with a MN in the same LISP domain.

Here it will be described two of them. These are chosen specially because they cover mostly all the LISP messages exchanged to ensure mobility for the several cases. A case that a MN in a LISP domain communicates with a SN in another LISP domain is present in figure 3.7.

It is important to note that the MN is roaming while the SN is stopped. In that case, the MN tries to communicate with the SN. If it does not have the corresponding RLOC of the SN in its Map-Cache, it must query the Map-Server with a Map-Request message requesting the RLOC of EID2. Upon receiving the Map-Reply message, the Map-Cache is updated and a tunnel between them is created in order to send traffic directly through the tunnel. Thus, through that tunnel, the MN encapsulates and sends data packets straight to the SN. Once the MN moves and changes its location, it must update the location to the Map-Server sending a new Map-Register message. After the MN updates the EID-to-RLOC binding, the SN can retrieve the new mapping data and further it is able to decapsulate data again straight from the MN.

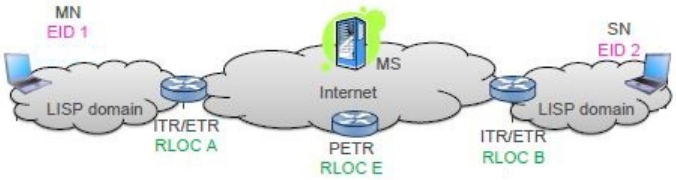


Figure 3.7: A MN in a LISP domain communicates with a SN in another LISP domain [17]

Another case, the second one, happens when a MN in a LISP domain communicates with a non-LISP node in figure 3.8. The procedure is the same way as the one mentioned above, with some differences as described below.

The MN in the LISP domain addresses a packet towards the IP address of a non-LISP node. As there is no RLOC corresponding for that, it encapsulates the packet towards the PETR which corresponds to this node; in that case RLOC F is assumed as the PETR of the corresponding node. First, when the packet is sent by the MN towards the PETR, it is received and encapsulated by ITR, and further, it is sent towards RLOC F. When it is received, the PETR decapsulats the packet and, the non encapsulated packet is carried to the non-LISP node.

In the reverse direction, the non-LISP node addresses a packet towards the EID of the MN. The packet is forwarded to a PITR (RLOC-I), which is responsible to encapsulate

the packet and then forward to an ETR. Upon receiving the packet, ETR decapsulates it and finally sends it to the MN.

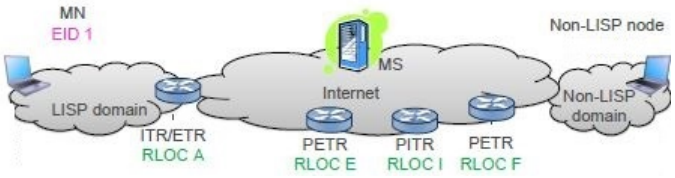


Figure 3.8: A MN in a LISP domain communicates with with a non-LISP node [17]

3.5 Chapter Considerations

This chapter focused on the LISP protocol description.

At the beginning an overview of LISP protocol has been done. In this introduction, it has been described what it is the protocol, how it works and what are the main features.

Then, a detailed description regarding LISP components was made, as well as it was depicted their functionalities. Further, as there are many LISP messages, they were deeply explained in order to understand how the protocol works.

As the central subject of this work is mobility, the existing extension mobility for LISP is presented. Thus, the extent of mobility LISP-MN was presented, divided into various topics and explained in detail in order to understand its mode of operation for future use. Last, it was described the set of possible communication scenarios using LISP mobility protocol.

After describing deeply the LISP, the next chapter focuses on the specification and implementation of the mobility protocol in vehicular OBUs and RSUs.

Chapter 4

Implementation of the LISP mobility protocol

In order to analyse the performance of LISP in vehicular environments, it has been developed a prototype capable to support mobility in vehicles. The proposed prototype, LISP-CAR, comprises a virtual management server acting as MS and MR, an extended LISP-MN and all elements required to allow cars and users to connect to several networks. During this chapter, it is described the implementations done in order to guarantee mobility to the entire network regardless of the access technology.

Section 4.1 shows the LISP architecture and the modifications or adaptations performed in order to build the LISP-CAR architecture.

Section 4.2 presents the components used to build the architecture for the future evaluation of the protocol.

Section 4.3 describes the implementation of the LISP-CAR architecture. The Mapping System implementation is detailed as well as all the configurations performed. Then, the software tools used in LISP-CAR architecture are explained.

Section 4.4 explains how the *radvd* and *rdisc6* are used, implemented and why they are so useful to the LISP mobility. It is also present the problem and the solution of the *radvd* and *rdisc6* using WAVE technology.

Section 4.5 describes the importance of DHCP and where it is used on this architecture.

Section 4.6 details the handover process and all the LISP messages exchange.

Section 4.7 describes the connection manager implementation as well as its operation.

Finally, section 4.8 summarizes the previous sections described above.

4.1 LISP Architecture

In this section, its made a brief overview according to the LISP architecture and its adaptation to a vehicular architecture.

The LISP architecture presented in figure 4.1 is used as the base to build the LISP-CAR architecture. The figure highlights three fundamental parts, which are:

- Destination Space (EIDs)
- Transit Space (RLOCs)
- Mapping System

Those mainly parts will be kept in the LISP-CAR architecture, but with the appropriate elements, which will be described in the components section.

Every subsection goes deep in the details on how to configure or reprogram the specified network component.

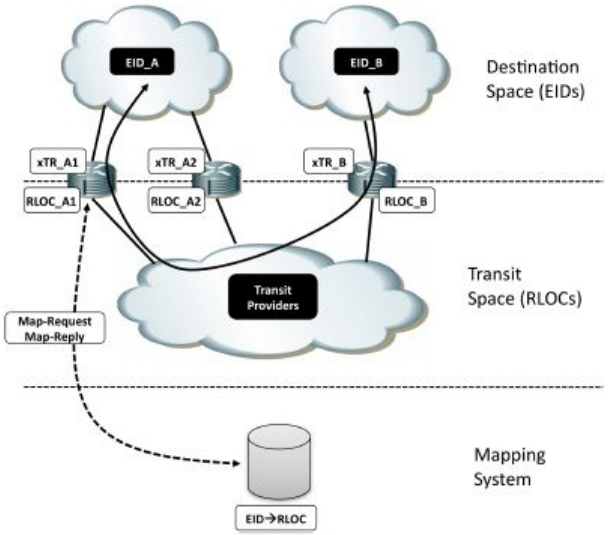


Figure 4.1: LISP Architecture [35]

4.2 Components

In this section the fundamental components to the LISP-CAR architecture are referred below.

So, the components used to build the new architecture are:

- LISP Map-Server and LISP Map-Resolver co-located.
- RSUs
- OBUs
- Laptops as MNs

In the following section these components will be described in order to detail their functionality and their implementation.

4.3 LISP adaptation to Vehicular Network

Taking into account the architecture presented in figure 4.1 and the fundamental components detailed previously, the LISP-CAR architecture presented in figure 4.2 is capable of supporting vehicular mobility to the vehicles and their passengers using LISP.

As can be seen according to the figure, there is a Map-Server co-located with the Map-Resolver, two LISP sites, RSUs and OBUs. In this context, there is the possibility to extract several scenarios from the architecture to further evaluate the vehicular mobility using LISP communications. The addresses assigned in the figure above are just shown as possible example.

4.3.1 Mapping System Implementation

A distributed database, the mapping system, is one of the most important part; it is responsible for maintaining the associations between EIDs and RLOCs and it is comprised of LISP Map-Servers and Map-Resolvers.

In the course of this thesis, it has been mentioned that the LISP Protocol is a Cisco developed protocol. Furthermore, LISPmob [23] is an organization that provides an open-source LISP and LISP Mobile Node (LISP-MN) implementation for several operating systems. In a first approach, it was decided to use Map-Server (MS) and Map-Resolver (MR) provided from LISPmob, more precisely for LISP Beta network. LISP Beta network [46] is a multi-company, multi-vendor effort to research real-world behaviour of the LISP protocol.

The first approach has become unused as soon as we discovered that it contained several problems. It was necessary to had a publicly routable, non-firewall IP address on the

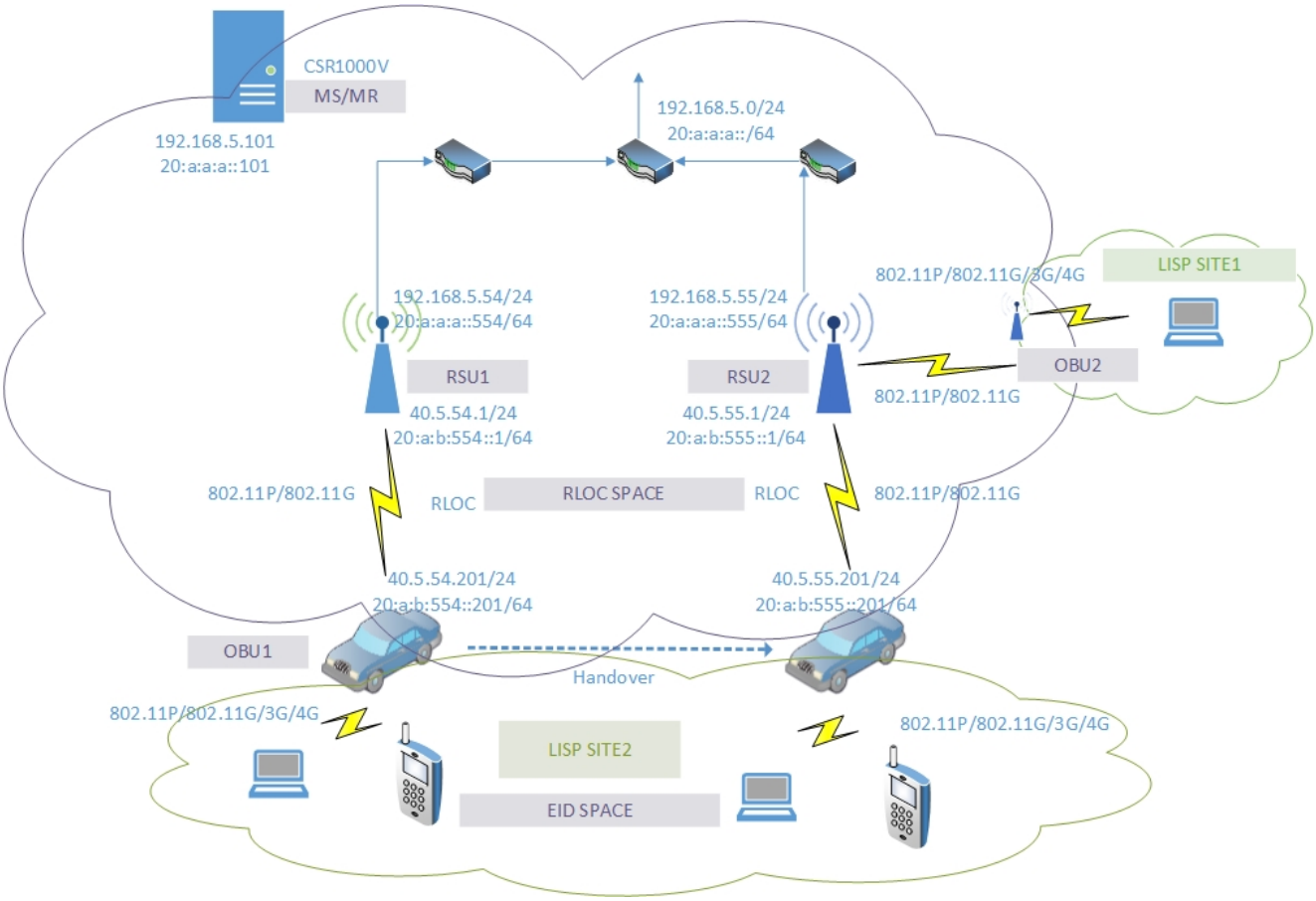


Figure 4.2: Prototype of LISP-CAR Architecture

device to connect to the LISP Beta network, because Beta network does not have support for Network Address Translation (NAT) traversal. A set of private IPs had to be made available as well as opening several ports such as control port and data port. Beyond these problems, the Map-Server provided from LISP Beta network is located in London, which would bring another delay to the handover times. The total handover time would suffer an increase: this increase corresponds to the sum of the round trip time of all LISP messages which come from or to the Map-server. Despite being a small time increase, it has a negative impact in the vehicular mobility which should be avoided.

Given those facts, and seen that it is the first work to address the LISP protocol in vehicular networks, the public idea was abandoned, in other words, the Map-Server provided from LISP Beta network was not used, and it emerged the idea of creating one in the private environment.

In order to create that it was implemented Cisco Cloud Services Router (CSR) 1000V

Series, a virtual router provided by Cisco with several benefits and uses-cases. Cisco allows the costumers to download and use for free for 60 days with full access to all features and a throughput of 50 Mbps. Thus, the CSR 1000v was emulated on a laptop, and several commands were executed in order to make CSR 1000v to become operational.

The CSR 1000v was chosen due to the fact that it supports LISP mobility and routing. With these conditions, the virtual router can be used as MS and MR, both co-located for IPv4 or IPv6 communications. Therefore, as the router fits perfectly, it was used as MS and MR in a private environment.

In this context various configurations were made to use CSR 1000v as MS/MR, and others settings were performed according to LISP-CAR architecture. So these settings are:

- MR/MS configuration.
- LISP sites configuration.
- Interfaces configuration.
- Routes configuration.

In order to allow CSR 1000v to act as MS and MR, it is necessary to activate them as presented in table 4.1. From that, the router is able to work as a database maintaining the associations between the vehicle and its position, acting as an anchor point, which provides control-plane scalability.

Regarding LISP protocol operation, all connectivity cases involve communications with LISP sites or non-LISP sites. EIDs are used within sites while RLOCs are used by the transit network. Consequently these sites have to be created in the MR/MS in order to recognize the future EID-to-RLOC bindings during the evaluation of the protocol with whole elements implemented. So, two LISP sites are created and the settings are presented in table 4.2. According to these settings, there are two LISP sites, each one with IPv4 and IPv6 EID-prefix available and one authentication key. EID-prefix represents a set of EIDs available to the nodes within the LISP site, under a single technical administration. Both EID-prefix support accept-more-specifics, which is the condition necessary to support mobility. However, the authentication key is essential to validate all EID-to-RLOC bindings from any EID behind the respective EID-prefix. The reasoning behind the importance of the key will be clarified in detail in the following section.

In order to reach the MR/MS, it is necessary configure at least one interface. The interfaces configuration is presented on table 4.3 and it shows two addresses in one interface, one for IPv4 and another for IPv6; thus MS/MR becomes reachable independent of the IP

type.

Finally, to allow communication, both input and output information, several routes must be configured presented in table 4.4. In the table it is presented mainly the destination network as well as the gateway to achieve the network. It is important to clarify that the LL shown there means Link-Local, a permanent IPv6 address, and the number inside it represents which is the RSU, so for instance the LL[554] is the Link-Local of RSU with number 554 and the corresponding interface depending on the access technology. In sum, regardless of the access technology, WI-FI or WAVE, which handover has been performed, the MS/MR is prepared according to the routes defined.

Table 4.1: MR/MS Configuration

	IPv4	IPv6
Map-Server	Enable	Enable
Map-Resolver	Enable	Enable

Table 4.2: CSR 1000V Sites Configuration

Site name	Authentication-key	Eid-prefix IPv4	Eid-prefix IPv6
Site1	mob	172.16.1.0/24	2001:db8:a::/48
Site2	mob1	172.16.2.0/24	2001:db8:b::/48

Table 4.3: CSR 1000V interfaces Configuration

Interfaces Name	IPv4 address	IPv6 address
GigabitEthernet1	192.168.5.101/24	20:a:a:a::101/64
GigabitEthernet2	-	-
GigabitEthernet3	-	-

To make sure that the MS/MR is properly working, a debug level feature is very important as well as the LISP Site registration information.

The MS/MR on a debug level can intercept all LISP messages, such as Map-Register, Map-Notify, Map-Request and Map-reply. In figure 4.3 it is presented an example of the received Map-Register message in MS/MR. It is also possible to view the EID-to-RLOC bindings stored on it: into the database it is possible to observe the location of each

Table 4.4: Routes Configuration

Routes technology	IPv4-addr	IPv4-gw	IPv6-addr	IPv6-gw
WI-FI	40.5.54.0/24	192.168.5.54	40:A:B:554::/64	LL[554]
	40.5.55.0/24	192.168.5.55	40:A:B:555::/64	LL[555]
WAVE	20.5.54.0/24	192.168.5.54	20:A:B:554::/64	LL[554]
	20.5.55.0/24	192.168.5.55	20:A:B:555::/64	LL[555]

MN, and where each permanent EID is located at that moment. The table 4.5 represents an example of the information that the MS/MR can store when LISP is running on the network. Taking a look at this example table, it is possible to verify that site1 is off while site2 is on. This means that LISP is only running in part of the network, the part of network with the site2 informations. According to site2, all MNs with their permanent EIDs belonging to these EID-prefix have been registered in MS/MR at 1 second ago through the RLOC address mentioned on the last registration. As the EIDs of all MNs remain the same, every time that each MN behind this EID-prefix change its attachment point, it has to update the MS/MR database and in this table the RLOC last registered will be replaced to the new one.

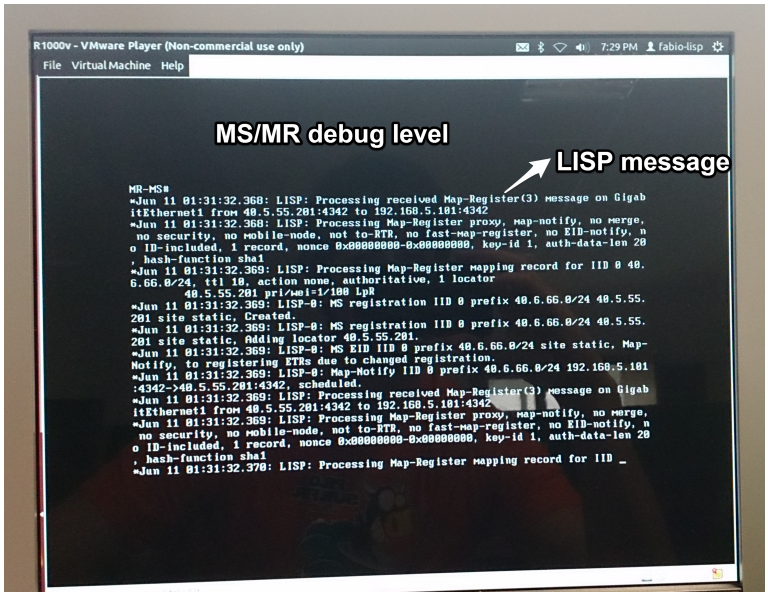


Figure 4.3: MS/MR debug level

Table 4.5: LISP Site Registration Information

Site name	Last Register	up	Last Registered(RLOC)	Eid-prefix IPv4/IPv6
Site1	never	no	–	172.16.1.0/24;2001:db8:a::/48
Site2	00:00:01	yes	20:a:b:555::201	172.16.2.0/24;2001:db8:b::/48

4.3.2 Network Implementation

Regarding network implementation it is taken into account all the necessary implementation performed in RSUs and OBUs.

The RSUs act as a simple static station, which provide several types of wireless connections, and every packets that go through them are routed normally as an ordinary router. In this case all RSUs are connected via Ethernet to the MS/MR.

Regarding OBUs, they are able to function as a node as well as a router. This means that, on one hand, it can connect via WI-FI or WAVE to RSUs and, on the other hand, they can diffuse WI-FI or WAVE to other nodes that will bind to it. Moreover, OBUs may act as MN or SN depending on the condition being tested.

Furthermore, in order to communicate with MS/MR and to ensure vehicular mobility, to keep the connection or communication alive between MN or SN during the handover procedures, it is necessary to apply LISP mobility protocol in the OBUs.

The LISP-MN open-source code for openWRT provided by LISPmob [23] was used and changed to work for vehicular networks. Through the LISP-MN mobility is guaranteed; however the time of handover is very large. As the goal is to ensure mobility between vehicles which are in constant and fast movement, changes were made in the LISP-MN accordingly. In the *lispd_iface_mgnt.c*, the function responsible for the LISP management, it has been changed in order to that and the differences are present in the software tools subsection. Thus, these functions were extended in order to work as fast as possible according to the handover technology used.

Further, once LISP-MN is compiled, the binary is running together with a configuration file in order to ensure mobility for the vehicles and their passengers. Each configuration file is different for each OBU. This file comprises important information presented and detailed in several tables below, which their information consists primarily of:

- Daemon configuration.
- RLOC-probing configuration.
- MR configuration.

- MS configuration.
- Database-Mapping Configuration.

According to the daemon configuration presented in table 4.6, it is enabled a debug level, which may be within the range [0,3]. As higher is the level, more verbose is the information essential to see if everything is working correctly. If there is no issue, the 0 debug level is the most appropriate. Router mode is off, so LISP is working on MN mode; thus the OBU is considered a MN, as well as their users connected to the OBU. The Map-Request retries represent the number of times that Map-Request message could be sent. It is presented a value of 2 to avoid any mistake, but 1 is enough.

The RLOC-probing configuration in table 4.7 exposes if there are or not RLOC probes. This means that, if enable, the MN will be probing all RLOCs of all MNs present in its cache in order to know whether they remain valid or not. The number of times that this happens and the interval between them is described by RLOC retries and RLOC retries intervals. These settings were made to be truly reliable, because without that everything works.

Regarding table 4.8, it represents both address, IPv4 and IPv6 to reach the MR.

Further, table 4.9 portrays two Map-Server configurations, each one corresponding to each OBU. The difference between them lies in a key important to know whether a Map-Register is valid or not, comparing this key with the already existent on Map-Server informations at CSR 1000V, which is presented in table 4.2. The Map-Server IPv4 and IPv6 addresses are the same as previously mentioned for the MR, due to the fact that they are co-located.

To sum up, in the table 4.12 it is demonstrated two different database-mapping for each OBU, due to the fact that each OBU represents different LISP sites: two different technical administration with different IPv4 and IPv6 EID-prefix that should correspond with the already configured at CSR 1000v MR/MS. Consequently, the configured RLOC interface depends on the technology to connect to the RSUS: if it is WAVE it corresponds to wlan1 interface, while for WI-FI it is wlan0 interface. Thus, the RLOC assigned to each OBU is the address presented at the mentioned interface, allowing communication to the OBU and to all users binding to it from other MN, or even from THE MS/MR, in the same way it will route packets to other MNs and also to reach the MS/MR.

After explaining the main LISP configurations in the OBUs as well as the LISP-MN and all network implementations, in figure 4.4 presents the flow diagram of the LISP mobility operator when LISP is starting in OBUs, together with the configurations required in order to ensure the vehicular mobility.

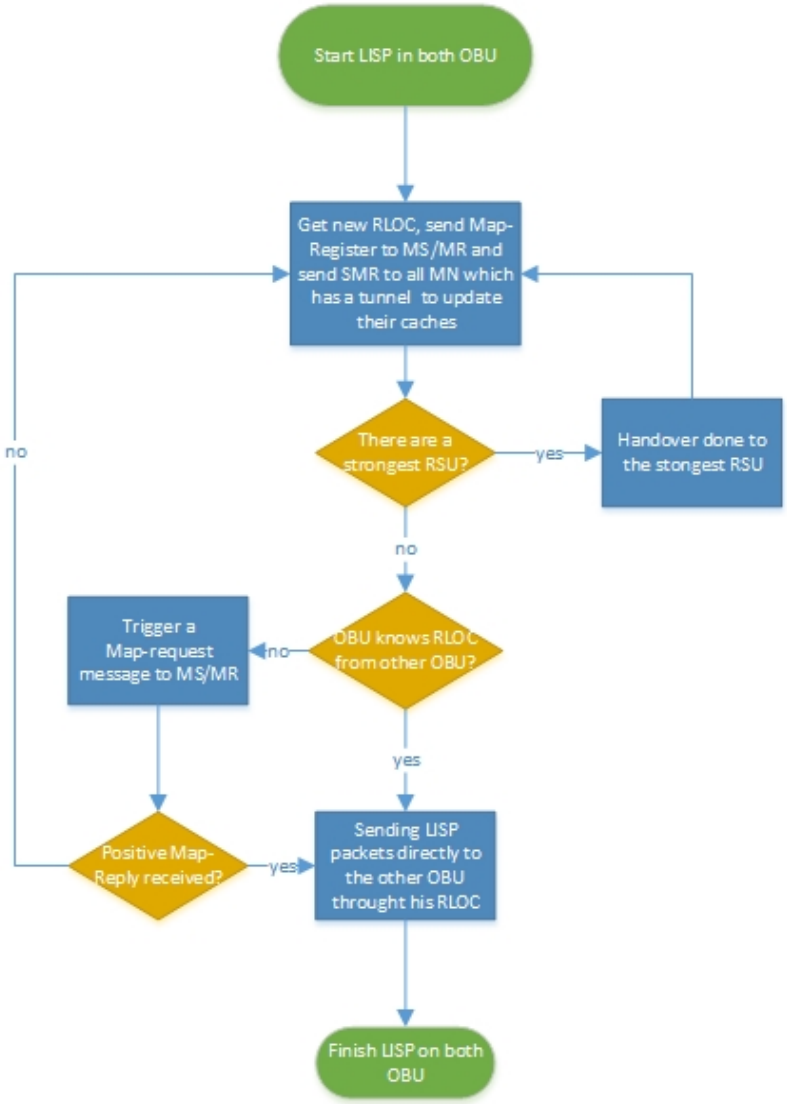


Figure 4.4: LISP Mobility operation flow diagram

According to figure 4.4, imagine that an OBU and its users are moving and receiving packets from another OBU through a tunnel already created. Thus, while the OBU is moving, it changes its point of attachment (RSU), it has to register its new RLOC to the MS/MR, and it must send the SMR for the other OBU which it has a connection established and is receiving its packets. So, the other OBU, upon receiving a SMR, is able to update its cache, querying again the MS/MR with Map-Request message in order to obtain the new RLOC. The new RLOC will be used to create the new tunnel, whereas the other is no longer valid because the OBU is no longer there. So, the packets are sent

Table 4.6: Daemon Configuration

	config Daemon
router mode	off
debug level	3
map request retries	2

Table 4.7: RLOC-Probing Configuration

	RLOC-probing
RLOC probe intervals	0
RLOC probe retries	2
RLOC probe retries interval	5

through the new tunnel to the OBU which had the established connection.

Thus, as described above, the SMR is sent by who makes the handover to all those who have an established communication, whose function is driving the LISP Map-Request message to find the new location and keeping the connection alive.

4.3.3 Software Tools

In this work, several software tools have been used in order to implement a vehicular mobility using LISP. Some of them are software platforms which have been taken as a basis for development, and other are programs which main functionality is to carry out the evaluation part of the implementation. In this section, a description of these tools is reported:

- `OpenWrt`: All the RSUs and OBUs used on the testbeds run a version of the OpenWrt operating system modified by VeniamWorks company. The OpenWrt [37] is a Linux distribution for embedded devices, with a strong integration of network components. It provides a fully writeable file system with packet management that allows the user to customize the device through the use of packages to suit any application. Given these facts it is concluded that OpenWrt is a suitable operative system for developers and it is easily modifiable operating system for router.

- `LISPmob`: All OBUs run LISP-MN code in C and C++ languages provided by LISPmob [23]. LISPmob is an open-source LISP and LISP-MN implementation for several

Table 4.8: Map-Resolver Configuration

	Map-Resolver
address	192.168.5.101
address	20:a:a:a::101

Table 4.9: Map-Server Configuration

Table 4.10: OBU1

address	192.168.5.101
address	20:a:a:a::101
key-type	1
key	mob
proxy-reply	on

Table 4.11: OBU2

address	192.168.5.101
address	20:a:a:a::101
key-type	1
key	mob1
proxy-reply	on

Table 4.12: Database-Mapping Configuration

Table 4.13: OBU1

EID-prefix	172.16.1.0/24
EID-prefix	2001:db8:a::/48
RLOC interface	wlan1
priority-v4v6	1
weight-v4v6	100

Table 4.14: OBU2

EID-prefix	172.16.2.0/24
EID-prefix	2001:db8:b::/48
RLOC interface	wlan1
priority-v4v6	1
weight-v4v6	100

operating systems. Some changes were made in LISP-MN in order to work for vehicular networks. The LISP-MN has a very slow handover which is based on SMR procedure presented in section 6.6.2 of [13]. This process could take a few seconds, roughly between 4 and 7 seconds. As LISP-MN is used for the vehicles which moves very fast performing quickly handover, then was changed that. Further, during the mobility tests, when LISP-MN is running in OBUs, it was figured that SMR is triggered as soon as a new RLOC is felt on RLOC interface bringing some issues. The problems happen when the SMR is triggered but the default route is not already defined. Due to that, SMR is not sent and consequently the destination of SMR does not update its cache resulting on a loss of connections and a wrong location sending packets. This mainly happens in case of handover via WI-FI because it takes a long

time to get the new RLOC and their default route, so as the SMR leaves before that, the destination of the SMR does not update its caches unless it is changed the process between losing the address and gateway, and the allocation of the new ones. Only after that the SMR must be triggered.

- `VMware Player`: is a virtualization software package used to emulate CSR 1000v, which is configured to work as MS/MR.
- `Builder`: is an ubuntu image running in the VirtualBox which has OpenWrt buildroot installed. OpenWrt builroot is explained in the next subsection.
- `VirtualBox`: is a full virtualizer for x86 hardware, targeted at desktop, server and embedded use.
- `Wireshark`: it is used to listen on the physical interface(s) in order to see the encapsulated packets or listen on the lisp TUN interface (lispTun0) to see the packets before or after being encapsulated. With "lisp" and "lisp-data" filters it is possible to look for LISP control or LISP data packets in order to know whether the packets are reaching the destination.

4.3.4 OpenWrt buildroot

OpenWrt Buildroot, a greatly modified version of buildroot, is a set of patches and Makefiles that allows users to easily generate both a root filesystem (filesystem in the same partition as the root) and a cross-compilation toolchain for an embedded system. The cross-compilation toolchain uses uClibc and a tiny C standard library in order to generate the binary files from a host system to the embedded device. Thus, in order to modify the LISP-MN code, the OpenWrt buildroot was used.

A makefile LISP-MN was downloaded and inserted in the OpenWrt buildroot containing the following information:

- Where to download the package.
- How to compile.
- Where to installed the compiled binaries.

Using *kconfig* (Linux Kernel menuconfig), it was possilble to enable the LISPmob feature. Further, when something is modified in the LISP-MN code, it is again compiled in the OpenWrt buildroot generating a binary file. Then, the binary file is sent to the OBUs allowing them to use the LISP protocol.

4.4 RADVD/RDISC6 Configuration

The Router Advertisement Daemon *radvd* [1] is essentially an open-source software product that implements advertisements of IPv6 router prefixes using the Neighbour Discovery Protocol (NDP) [40]. To take better advantages from that, *rdisc6* is also used to lookup the list of IPv6 prefixes: *rdisc6* is an Unix program which implements the ICMPv6 Router Discovery.

There is a *radvd* process on OBUs which is responsible to send RAs with the mainly following information:

- Advertisement interface.
- Advertisement prefix.

On the other hand, there is a script *rdis.py* to run in MNs which is in charge of automating the process of the EID allocation with the following steps:

- Router Solicitation.
- Extract the prefix and add the suffix.

During the execution of this research, *radvd* is used in OBUs in order to answer requests with router advertisement (RA) messages; *rdisc6* is used in MNs to router solicit (RS) an IPv6 prefix.

The main purpose of *radvd* running on OBUs lies in the fact that it can distribute the EID-prefix for all those users that are connected to the OBUs. Thus, according to *radvd* configurations, the RA messages mainly comprise: the interface used to send the RAs, the routing prefix as well as the address of the interface which provides those RAs. As users connect to OBUs via WI-FI, the chosen interface is wlan0, corresponding to WI-FI addresses, and the prefix to be advertise is the EID-prefix defined in each OBU.

Thus, when a passenger enters in a car represented by the OBU and wants to connect to the car, it must acquire an EID. In this context, a *rdisc6* is triggered in order to require an IPv6 prefix in which *radvd* will answer with the EID-prefix. Upon receiving the EID-prefix of the respective OBU, it extracts the prefix and adds a random number, to form an IPv6 EID. If there are multiple MNs on the same vehicle, it is necessary to certify that the random number provided for the construction of the EID is not repeated. A script is included in all MNs to provide this function.

4.4.1 RADVD/RDISC6 problem and solution using WAVE technology

The *radvd* is also used in RSUs in order to provide the IPv6 network prefix when required by the OBUs, which they will use to form their IPv6 address: it is comprised by the received prefix and its own interface LL. Thus, when an OBU wants to connect to a network or a network provider from the RSUs, it has to send a *rdisc6*, which will be answered through a RA. Further, the respective OBU has to read and extract the IPv6 prefix provided by the requested RSU, and together with its suffix, which is its own LL, it forms its IPv6 address.

In case of the connection between the RSU and the OBU be performed through the WI-FI network, the suffix will be the LL in the wlan0 interface; while in the WAVE connection case, the suffix will be the LL presented in the wlan1 interface. However, the RS is an ICMP message sent to a specific multicast address, the ff02::2 [6]. This is not a problem in the WI-FI connection established between the OBU and the RSU, but it is indeed a problem in the connection established through the WAVE technology.

When it is used the WI-FI connection, it is established a session, so it is guaranteed that the packets sent by the OBU will only be received by the RSU with which it is connected, even if there are other RSUs within range. Otherwise, with the WAVE technology a problem arises, since there is no prior session establishment on the connection by the OBU to the provider RSU. Thus, when the OBU sends the RS message to the multicast address ff02::2, all RSUs will answer containing different prefixes into each RA messages, while the RS should only be sent to one RSU which the OBU required, and then only the RSU would respond with the desired prefix.

The RS and the RA performed by the OBUs and the RSUs, respectively, are used in order to the respective OBU to connect to the RSU with the best connection available, in which this subject is handled in the implementation of the connection manager. Further, and deeply detailed in the connection manager implementation, with the command "uwme getAvailable" performed by the OBUs, among many things, it is possible to know which is the best RSU network available and its MAC address. This MAC address will be converted in the respective LL address. Thus, with this LL, it emerges a solution in order to assign the correct IPv6 prefix address between all of them presented in the RA messages when a RS from the OBU is triggered. So, upon receiving multiples RA messages, the correct prefix is obtained by filtering the RA messages with the LL of the RSU with the best quality obtained earlier as mentioned above. Thus, this eliminates the problem mentioned above and makes it possible to obtain an IPv6 prefix through the WAVE technology, even

though the RS is sent to a multicast address.

With the *radvd* and *rdisc6* described above, we can ensure automatically IPv6 EID to the MNs as well as the IPv6 prefix, in order for the OBU to connect to the required RSU.

4.5 DHCP Considerations

The *radvd*/*rdisc6* takes care of IPv6 addresses while Dynamic Host Configuration Protocol (DHCP) [38] is responsible for IPv4 addresses.

DHCP is a protocol that offers especially dynamic configuration of terminals, with the grant of IPv4 host addresses, subnet mask and default gateway.

So, DHCP is very important in the IPv4 networks being responsible for the following characteristics:

- Range of addresses
- Interface that disseminates the network

In this work DHCP has been very important in all WI-FI connections using IPv4 addresses, such as between RSU and OBU and between OBU and MN.

In this case the network interface that disseminates the WI-FI network is respectively the wlan0 and all users that connect to that network will get an address within that range of addresses becoming connected to that.

For example, when a vehicle is roaming, it constantly changes its connection between RSUs. In order to get the IPv4 addresses it is used DHCP.

On the other hand, between OBU and MN, DHCP is very important in order to establish passengers WI-FI connection inside the vehicle and get an EID for users from the EID-prefix available. Thus, DHCP ensures "automatic" IPv4 addresses to the OBU and MN when required.

4.6 Handover Process

Regarding the handover process, when the MN receives a new RLOC, it has to update its EID-to-RLOC mapping in the associated Map-Server to maintain reachability at its new location. This flow process is illustrated in figure 4.5 with an OBU1 as an example.

In order to OBU1 and its MNs maintain the connection with other MNs (vehicles and their users) who are already connected, it is necessary to send a SMR bit to those MN, allowing those MNs to do a new search to find the new RLOC, which they will establish

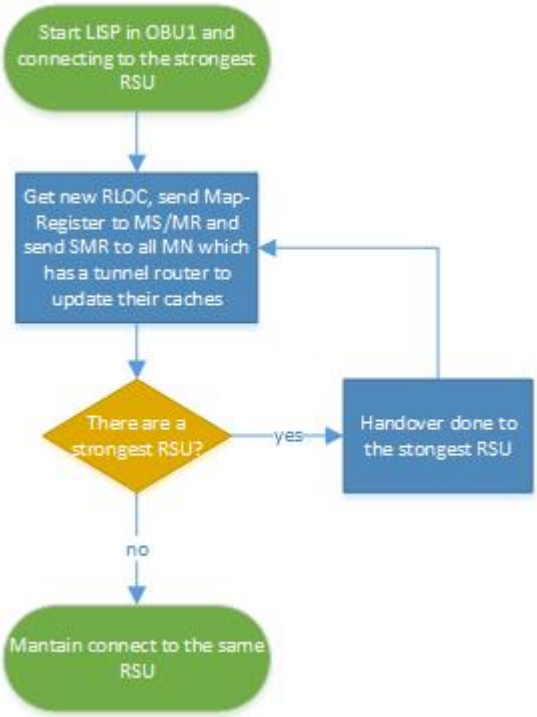


Figure 4.5: Handover operation flow diagram

---

a new tunnel to send packets. Going deeply in this subject, when the MNs receive the SMR, they automatically trigger a Map-Request towards the MS in order to know the new RLOC. Upon receiving the reply containing the updated RLOC, they establish the new tunnel which becomes available to send the packets again.

Finally, if everything described above is fast, the users are unaware that they changed the network provider, since they keep the connection alive. Thus, the handover has been made successful.

4.7 Connection Manager Implementation

In order to make the handover process of the vehicles while they are moving automatic, it is necessary a system capable of monitoring the available networks at their range and trigger the handover to the strongest network, this means to the available network with the best signal.

Regardless of the access technology, the connection manager does a search of possible connections and chooses the one that has the strongest signal. Once chosen, if it is equal

to the previous network selected, the handover is not made and all previously existing settings are maintained; otherwise, the handover is done and it shall proceed with sending a solicitation message to know which is the network with the best signal and extract its ID which identifying the RSU to attach. In case of an IPv4 handover, upon receiving the ID corresponding to the board which belongs to the network with better signal, it is added to the prefix and then with the suffix which representing itself forming the IPv4 address, or it can also be required an address with DHCP in WI-FI technology. Further, a default route to the respective interface of the RSU with the best signal is added, and finally the old address is deleted. In case of an IPv6 handover, it must trigger a RS to the LL of the strongest network as explained in 4.4 section in order to receive the IPv6 prefix and with their LL forming his own IPv6 address; so when the handover is done, it is repeated this process to obtain the new IPv6 address. Further, it must also change the route to forward the traffic accordingly, so it is added a new default route to the LL of the new RSU connected. This LL is obtained by converting the MAC of the new RSU in LL, and this MAC is in turn obtained from the response to the solicitation message done previously. Afterwards, it is necessary to eliminate both the old address as well as their old route.

When the request is made to receive all available networks, then several parameters are presented from each network and the metric that represents the strongest signal is the Received signal strength indication (RSSI); therefore the network that has the highest RSSI, it will be the chosen network.

A script *handover.py* was made in order to implement the operation method of the connection manager.

The connection manager operation flow diagram for WAVE handover can be observed in figure 4.6. The same procedure goes for WI-FI handover.

The script for WI-FI is done in the same way as for WAVE, so the following information describes the script made for the IEEE 802.11p access technology:

- `uwme getAvailable`: This command acts as a solicitation and performs a scan to all available networks. In each available service, the main information is the "Provider Service Context", who let us to know who is the available provider indicated by the ID, the "RSSI", which gave us the received signal strength indication and the "MAC address", which indicate the valid MAC address of the WAVE interface (wlan1) of the concerned provider. Then, the service with the strongest RSSI is chosen.

- `Mac to Link-local`: The MAC address of the WAVE interface of the chosen provider is analysed and has made the effort to convert for Link-local (LL).

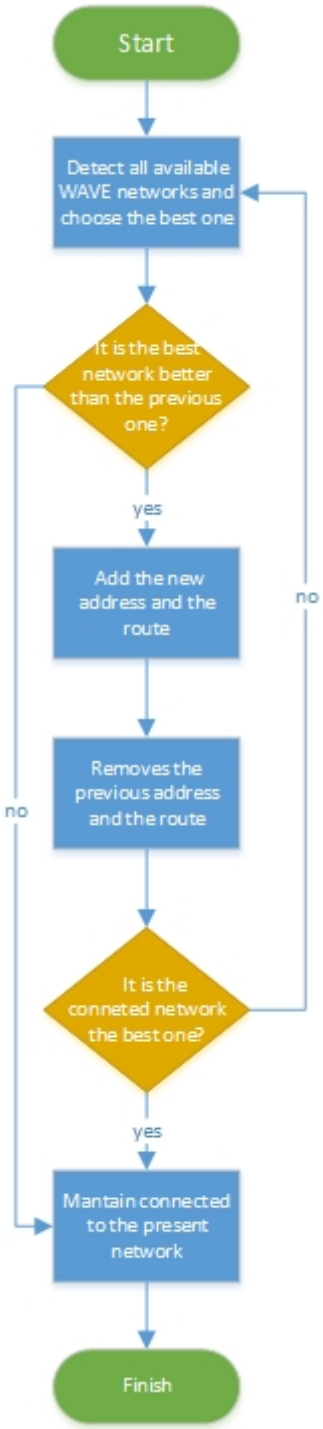


Figure 4.6: Connection manager operation flow diagram

- `Lookup IPv6 prefix through LL`: Send a RS using the command: "*rdisc6 wlan1*" and then extracts the IPv6 prefix of the chosen provider, filtering through the converted LL all RA messages received in order to find the desired IPv6 prefix.

- `Add Addresses/Routes`: Adds the new address with the new prefix received from the new strongest provider, and the suffix is the own Link-local presented at the WAVE interface, which is always the same regarding IPv6 rules. Then, it configures the default route via the new converted LL in the WAVE interface.

- `Delete Addresses/Routes`: Removes the default route through the Wave interface to the converted Link-local, and it also removes the address existent at the moment on the wlan1 interface.

To sum up, it is important to note that these actions described above ran in an infinitive loop, and every second (could be changed) it is triggered the top action: if the strongest RSSI is the same than before, the handover does not occur and the other four actions do not happen; if the handover occurs, these four actions are triggered seamlessly.

4.8 Chapter considerations

This chapter presented the LISP mobility implementation.

Once realized how this protocol works, the types of messaging exchanged, the communication between various elements and other mains features, the LISP-CAR architecture (figure 4.2) has been created in order to use the LISP protocol, but adapted to the vehicular environment. The main topics of this chapter were the following:

- Described the fundamental components to the LISP-CAR architecture.

- Explained the Mapping System implementation, the central/top part in this private architecture which comprises the Map-Server, and the Map-Resolver responsible of the association between EID and RLOC of each MN.

- Depicted the network implementation which comprises all the implementation performed on the RSUs and on OBUs highlighting the LISP-MN operation and the parallel configurations.

- Described the tools used to implement the LISP protocol into vehicular networks, as well as in the LISPmob tool it is presented the changes performed in LISP-MN in order to ensure mobility to the vehicular networks.

- Described the *radvd* and *rdisc6* fundamental configurations in order to provide and lookup the respective IPv6 prefix to the OBU or MN to form its IPv6 address comprised by the IPv6 prefix obtained by those tools and with the IPv6 suffix which is the own LL, a permanent address of each interface. Further it was reported a problem and detailed a solution in the *radvd* and *rdisc6* using IEEE 802.11p.

- Depicted the necessary *DHCP* configurations in order to provide IPv4 addresses in all WI-FI connections used. So, after establishing the session through WI-FI connection, if DHCP is ran in the AP which is disseminating WI-FI network, an automate IPv4 address is obtained in the client as soon as it connects to that AP.

- Explained what and how it is the operation of the handover process.

- Described the connection manager implementation, which is done in order to automate the handover process, and it is shown its operation to perform seamless handover.

To conclude, in order to validate this protocol in the VANETs, it is necessary to make tests, both in laboratory and real environments. This will be the subject of the next chapter.

Chapter 5

Evaluation

This chapter tests and evaluates the performance of the LISP protocol in vehicular environments.

In section 5.1 an introduction about the evaluation is made, explaining briefly what will be done in order to test the mobility in vehicular networks using the splitting between identification and location of the MNs.

Section 5.2 details the used scenarios in order to evaluate the LISP mobility protocol into vehicle environments. Further, the equipment and the access technology used in those scenarios are depicted in this section.

In section 5.3 it is described which are the tools used in order to test LISP mobility protocol, as well as the considered metrics which are considered in the handover process to further extract the results according to those metric(s).

In section 5.4 it is presented the handover results obtained in laboratory environment according to the testbed evaluated, and section 5.5 presents the handover results obtained in the road environment according to the testbed evaluated.

Lastly, section 5.6 presents an overview regarding the topics described.

5.1 Introduction

The work carried out in this Dissertation requires tests to prove its truthfulness and verify whether it is a protocol with future in vehicular networks. The tests and results will assess the feasibility of the LISP protocol to provide seamless handover in vehicular networks.

Thus, in this chapter it is evaluated the handover process between the RSUs in different networks which can transmit multi-technology, such as IEEE 802.11g or IEEE 802.11p,

different internet protocol versions, such as IPv4 or IPv6, as well as in different environments such as laboratory and road. The OBUs will be changing constantly their point of attachment (RSU) in order to establish the best connection available. This connection can be done in IPv4 or IPv6, as well as in different access technologies as mentioned above.

These handovers will be also evaluated with passengers acting as MNs inside the cars connected through WI-FI to the OBUs in the cars. Thus, OBUs must be able to connect to the RSUs and, at the same time, be able to disseminate WI-FI network allowing the passengers, represented by laptops, establish their connection. Moreover, taking into account all implementations made in chapter 4, the communication between MN to MN, this means, vehicle to vehicle, or passenger to passenger, or passenger to vehicle, will be tested while one or both MNs are roaming into different technologies, different internet protocol versions and different environments.

5.2 Testbed

5.2.1 Equipment Used

The equipment used consists on the Map-Server collocated with the Map-Resolver, the RSUs, the OBUs and the MNs. In a laboratory (lab) environment, a laptop with UBUNTU 12.04 operating system (OS), 2.9 GB memory, 1 processor, 8 GB hard disk and three virtual interfaces bridged is used as CSR 1000v router configured to behave as Map-Server and Map-Resolver. Those entities communicate with fixed RSUs using the building Ethernet network on lab tests and via WI-FI on road testes. Further, the RSUs displayed in figure 2.1 act as normal routers, enabling the OBUs to connect through WI-FI or WAVE technology.

The OBUs also illustrated in figure 2.1 represent vehicles in the lab and road, which can be moving or stationary; on the other hand, the MNs represent car's passengers and they are emulated through laptops with UBUNTU 12.04 OS.

For the real tests, in the road environment, it is also needed batteries to turn on OBUs and RSUs, tripods to hold them and a vehicle to move along the road performing the expected handover with an OBU inside.

5.2.2 Testbeds implemented

In order to test the LISP mobility protocol for vehicular networks, it is evaluated the handover in IPv4 and IPv6, in the laboratory and the road environments with all possible

combinations of intra and inter-technologies present in the table 5.1. Some scenarios were extracted based on the LISP-CAR architecture shown in figure 4.2. Thus, two possible scenarios were drawn from there.

Table 5.1: Technology Handover Cases

Name	Handover Case
P2P	IEEE 802.11p to IEEE 802.11p
P2G	IEEE 802.11p to IEEE 802.11g
G2P	IEEE 802.11g to IEEE 802.11p
G2G	IEEE 802.11g to IEEE 802.11g

The first testbed chosen is the communication between the SN and the MN into different LISP Sites. That testbed aims to test how the LISP mobility protocol behaves when a SN is sending traffic to a MN which is constantly roaming, changing his network attachment point (RLOC) in order to connect to the best connection available. This can be compared to a vehicle moving along the road performing handover between the available RSUs, while passengers are communicating and sharing information with a stopped vehicle and their passengers, or can be also compared with a connection or communication between car's passengers (MN) to the internet (SN). Further, this can be tested in a more realistic environment: imagine that in the future this SN may be a facebook server, any MN is able to establish communication with him executing the same procedure described for this testbed, but or the facebook server is covered by the LISP protocol (ideal case), or it is necessary to use a PxTR in order to encapsulate the ingress packets and decapsulate the egress packets, because this is no longer a LISP site to another LISP site communication. Then, the communication would be made between a SN in the non-LISP site to any MN in a LISP site.

The second testbed chosen is the communication between two MNs into different LISP Sites. Thus, this testbed is also done to understand how the LISP mobility protocol reacts to this type of connections when two MNs into different LISP sites are moving at the same time as they establish and maintain their connection. This can be compared to the communication between two passengers into different vehicles when they are moving along the road.

For the first testbed in the lab environment, illustrated in figure 5.1, the MS/MR is connected to the RSU1 and the RSU2 by the wired network of the building. The MS/MR configurations were described in section 4.3, and regarding both RSUs, their addresses

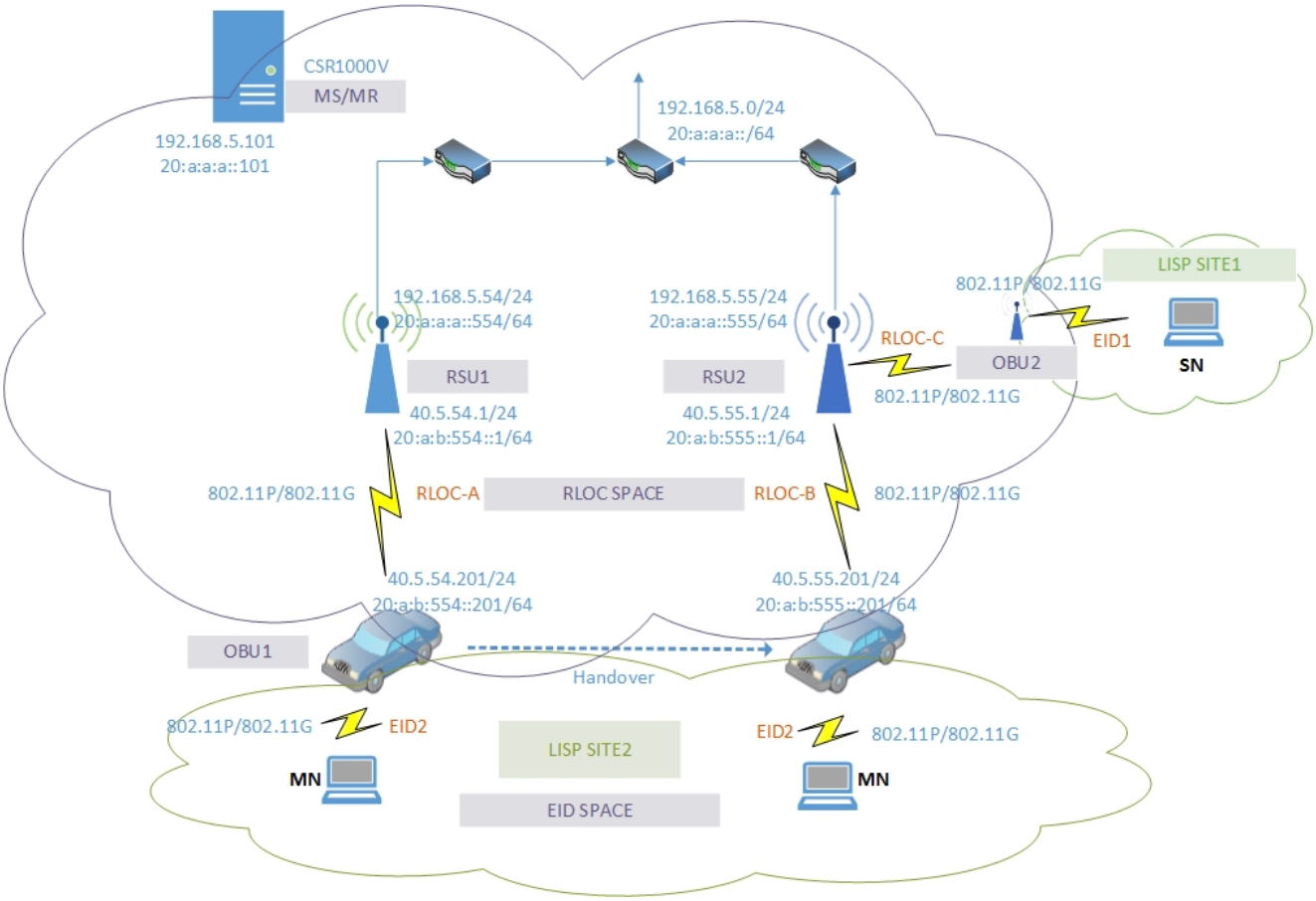


Figure 5.1: LISP testbed 1

---

were set up in order to automatically acquire addresses. For IPv6, the prefix is presented on that figure in order to create and indicate the network, and the suffix which can be its own link-local or the one described in that figure. For IPv4, the address is comprised of a portion that indicates the network, and the other by the ID of each board which represents itself. The important, so far, is that both RSUs and the MS/MR are in the same network connected by the wire.

RSUs are able to broadcast WI-FI network or to be a WAVE provider. The OBUs must run the adapted LISP-MN protocol with the correspondent configuration file with each corresponding site, in which the OBU will register its EID-to-RLOC binding into the MS/MR (this was better explained in section 4.3). OBUs are configured with the IP addresses presented in the figure, and they disseminate WI-FI network for the MNs, and at the same time they are WI-FI or WAVE clients. In this last option, the OBU creates a channel in order to connect with the RSUs, and then, it starts the connection manager

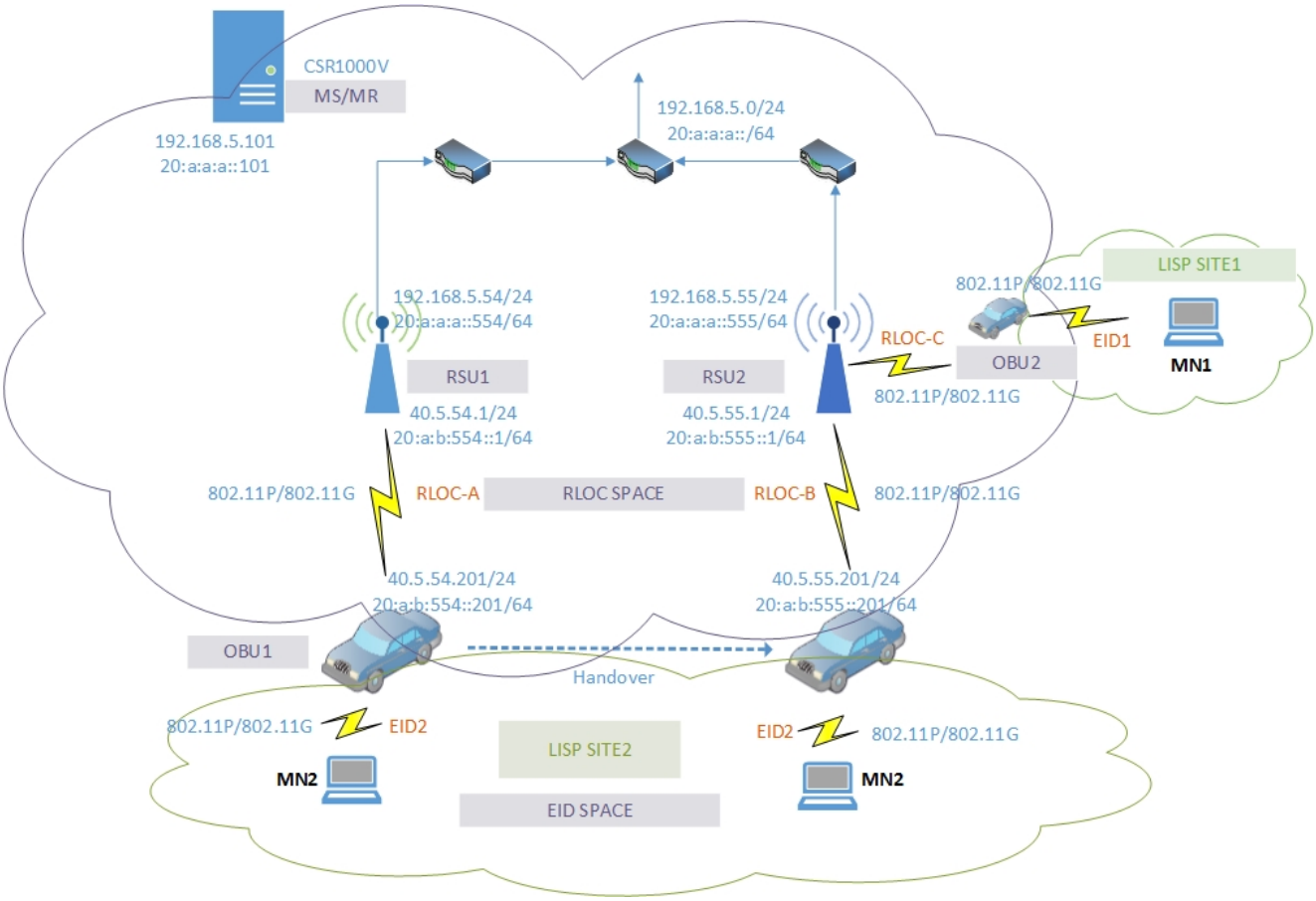


Figure 5.2: LISP testbed 2

program allowing an automatic handover. On the other hand, the MN and the SN connect to the respective OBU as in a regular WI-FI connection, and get an EID address from the EID-prefix of the corresponding OBU. Each EID-prefix of each LISP site was previously defined in the OBU configuration file. In order to obtain the EID address, which remains unique and independent of the connected RSU, it is used the *rdisc6* and *radvd* for IPv6 as well as DHCP for IPv4 as described in section 4.4 and section 4.5, respectively.

The second testbed lab environment is illustrated in figure 5.2. It is necessary to clarify that the MN1 as well as the MN2 are in constant movement while they are changing the connection between RSUs, in order to connect to the best connection available. On this testbed, the characteristics are the same as the first one, except for the SN that is now replaced by the MN1; thus, both the OBU2 and the MN1 are also in movement, and it is needed to run also a connection manager as in the OBU1 in order to connect to the best network available.

Figure 5.3: RSU 1

Figure 5.4: RSU 2

The real testbed is performed in the real vehicular environment the same way as in the laboratory. However, the main difference is the connection type used between the RSUs and the MS/MR: previously it was cable and now it is WI-FI. This test is made in the public road with the RSUs placed alongside the road, as it can be observed in figures 5.3 and 5.4. The Map-Server is placed in a middle of both RSUs, allowing them to connect through WI-FI. Furthermore, the OBU2 as well as the SN that is attached to it and is represented by a laptop, they are also placed alongside the road and connected to the RSU2. The OBU2 and the MN attached are presented in figure 5.5 and are placed inside the vehicle with the necessary antennas as depicted in figure 5.6. Then, the OBU2 performs the handover between RSUs in order to connect to the best network, while the MN is in communication with the SN.

In order to clarify certain doubts, the figure 5.7 is shown. In this figure it is presented three important parts to execute both testbeds. With the MS/MR database debug, it is possible to view all arriving messages on the virtual machine, which runs CSR 1000v router acting as MS and MR. The two red arrows point to two LISP messages previously detailed in section 3.3, which are captured in MS/MR. With this feature enabled, it is perceptible

if all the processes are carried out properly and, in the negative case, the problem is reported. Further, the debug of LISP-MN is shown on the OBU when a communication between two OBUs is performed. Pointing with red arrow is detailed the packet header, with visible RLOC source, RLOC destination as well as the EID source and destination. Finally, the connection manager is presented upon working. For example, when one OBU has a better connection through the RSU 555 instead of 554, the handover is triggered to that network, so the OBU changes its default route as well as the RLOC, and consequently, in the MS/MR database debug it will be observed the exchange of several LISP messages, such as Map-Register, Map-Notify, Map-Request and Map-Reply.

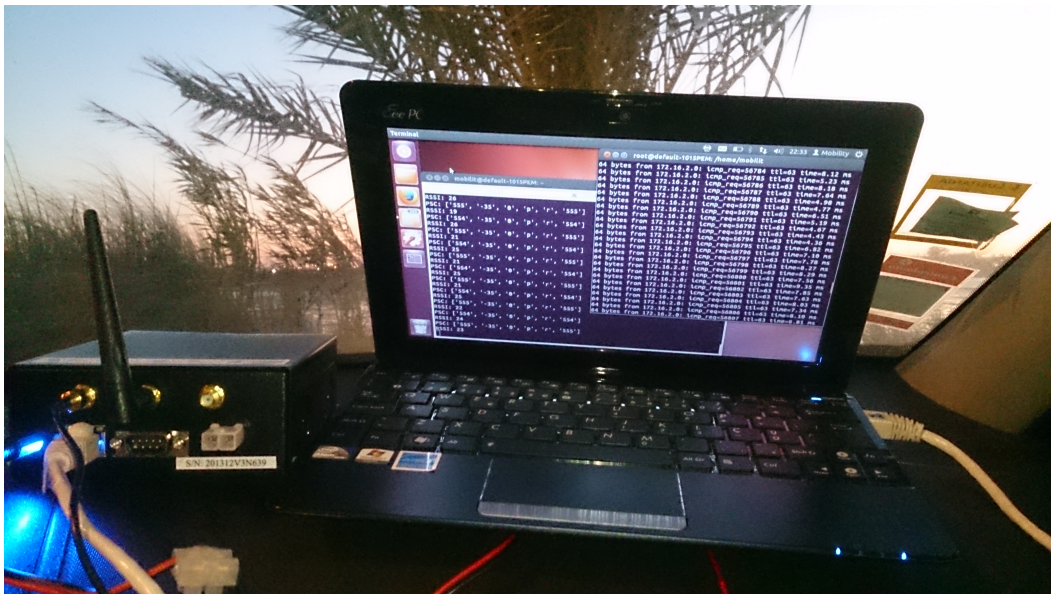


Figure 5.5: OBU1 and MN inside the vehicle

5.3 Tools and metrics

In order to get a good characterization of the handover process, the most import metric, handover latency, is taken into account.

The handover latency defines precisely the time interval when there is no connection while the OBU and the corresponding MN moves from one network to another. Thus, based on this metric, it may be possible to know which are the suitable access technologies, the WAVE or WI-FI, and if LISP mobility protocol is a suitable mobility protocol to use in VANETs.

Figure 5.6: Vehicle and 802.11p antenna

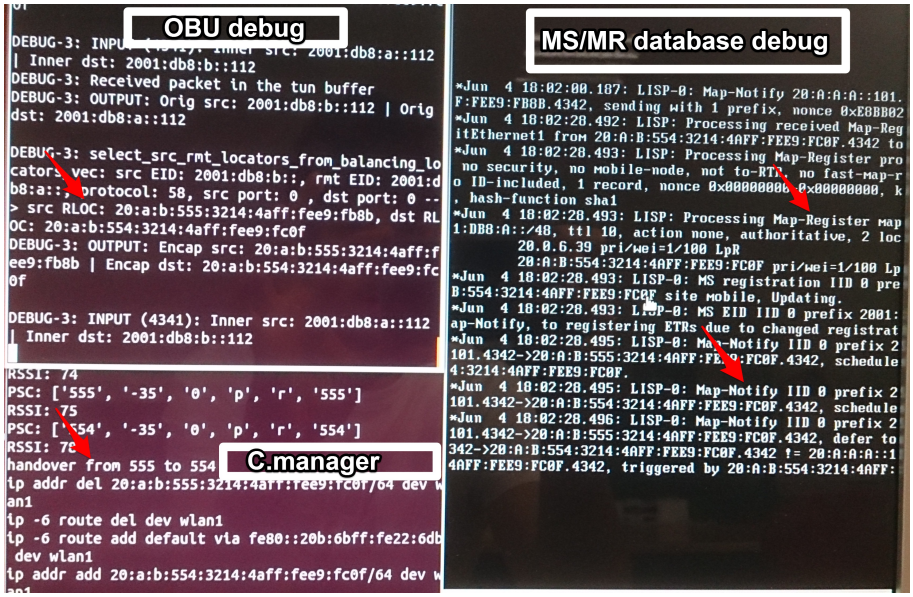


Figure 5.7: Process Debug

In this context, to obtain the handover latency results, it is necessary in both testbeds to setup tools capable to send traffic from one MN or SN to another. Thus, as it is common knowledge, a *ping* tool is used, simulating the transmission of the packets to test that metric and at the same time, the *Iperf* tool [43] is used, in order to generate traffic

using the transport protocols Transmission Control Protocol (TCP) or UDP. For these tests, only the UDP traffic is generated because with LISP, there are no retransmissions, which allows obtaining the results of this metric with greater reliability.

To analyse the output of the *ping* tool with sending traffic simultaneously through *iperf*, one *python* program is used in order to calculate the difference between the time the OBU is not receiving any packet any more, and when it is again receiving from another network. In addition, for each handover occurrence, the handover latency is always calculated this way. Then, the results are processed in a MATLAB [24] script in order to create the graphs. These results are obtained from 50 repetitions of each of test in the lab environment and 3 in the road, and the confidence intervals shown are of 95%.

Another tool was strongly useful in the evaluation of this protocol. VLC media player (VLC) is a free open-source written by VideoLan project [11], which has a streaming video server platform. With that, a video can be provided by a node and received by another, in other words, the video can be constantly received by and presented to a node while being delivered by a node provider. Those nodes can be static or in movement, and thanks to this tool, it is possible to test the handover in other way as is presented in the lab and the road experimental results section.

Besides *VLC*, in order to stream a video, it is also necessary to use an Hypertext Transfer Protocol (HTTP) session. So, according to [39], HTTP Live Streaming is a way to transmit audio and video over HTTP from a server to a client. The server is responsible for sending video and encode it digitally, while the client is responsible for determining the convenient media to request, downloading those resources, and then reassembling them so that the media can be shown to the user in a continuous stream.

5.4 Lab Experiments Results

In this section it is presented the results of the testbeds in the laboratory environment. The tests were made using the *ping* tool as well as using the *iperf* to provide 3 different traffic rates while handover occurs, 256Kbit/s, 512Kbit/s and 1Mbit/s, and all the bar graphics are in that order for each handover technology case.

5.4.1 Handover Latency

5.4.1.1 First Testbed

Regarding the first testbed, the handover is tested with intra and inter-technology as shown in figure 5.1 and different internet protocol versions, such as IPv4 and IPv6.

With LISP-MN running in both OBUs and with the OBU1 handover procedure managed by connection manager, the test is made with a communication between one MN and a SN, both connected to OBU1 and OBU2 respectively, with the help of *PING* tool.

The SN through the OBU2 is pinging MN, while this one is moving among both RSUs in order to establish the best connection available. Further, the WI-FI network is broadcasted by the OBUs to enable SN and MN to be attached to them.

Every time that the handover occurs, there is a period of time without connection, known as handover latency, which is measured enough times to achieve precise results. The handover process is performed several times, using different internet protocol versions in all elements and different access technologies.

Thus, the handover latency results using IPv6 addresses in all the elements are presented in figure 5.8 and detailed in figure 5.9, which illustrates the handover latency for each handover technology case used in the handover process.

On the other hand, using IPv4 in all elements, the handover latency results are presented in figure 5.10 and detailed in figure 5.11, which also shows the handover latency for each handover technology case used in the handover process. The reason for the handover to be slower using IPv6 than in IPv4 lies in the fact that it takes approximately one more second in a loop cycle which comprises the netLink messages used to communicate with the kernel in order to obtain the new address and the new gateway; on the other hand, this does not happen with IPv4 addresses.

In order to improve the evaluation of this testbed, a video streaming is performed. With *VLC* tool, SN begins transmitting the data, this case a video, while MN is changing the network connection to the best one provided by the RSUs, and at the same time it requests the video.

It is Important to note that VLC runs on SN acting as server and also on the MN, acting as a client. In addition, the HTTP session is applied, and with *VLC* a streaming video, illustrated in figure 5.12, is performed. Furthermore and taking into account the meaning of HTTP streaming described in section 5.3 and according to the documents presented in [11], there are three commands to take into account:

- "vlc -vvv name.mp4 –sout '#transcodevcodec=mp4v,acodec=mpga,vb=800,ab=128"

- ":standardaccess=http,mux=ogg,dst=dstaddress:8080'"
- "vlc -vvv http://dstaddress:8080"

The first command runs together with the second one by the SN with all settings filled, while the MN runs the third one in order to require the desired video.

Taking into account the IPv4 and IPv6 results, the mobility was ensured; however, the handover latency time results are significantly worse in G2G handover than P2P, P2G and G2P handovers. This is due to the fact that G2G handover is performed between WI-FI networks, while the others are done across WAVE network or inter-technology networks, such as G2P and P2G handover. In WI-FI networks, it is only possible to communicate after establishing one session at the same time with an AP; due this fact, when the handover occurs, there is a period of time without connection. This time includes the breakdown of previously established session while scanning for another network which is going to move, the time until it is finally connected, and the period of time to receive the netLink messages in the LISP-MN in order to get the new address and the new route for the new network. Further, as the MN shares, at the same time, the WI-FI interface for the connected network and for the network that it broadcasts into the vehicle, it even makes this technology more inappropriate for vehicular mobility.

Given those facts, and focusing in these results, the seamless handover happens between WAVE networks or through an IEEE 802.11g to an IEEE 802.11p network (inter-technology). Disregarding G2G handover, all other handover cases presented in the table 5.1, allow the MNs to connect to another network without releasing the first one. For example, the MN can connect to a WI-FI network and WAVE network at the same time, and then leave one which results in a not significant loss of time between breaking and establishing again the connection, which does not happen in G2G handover. In intra-technology handovers using only IEEE 802.11p, there is no notion of association to a network, which makes the process of handover much faster than with WI-FI.

Compared to the results obtained for different traffics for each handover technology case, the difference is not relevant, due to the fact that both, the WAVE channel and the WI-FI network, well support these traffic speeds not delaying the handover process.

Furthermore, in this testbed with the handovers technology cases described above, it is mandatory to run LISP-MN in the OBUs and thus, the following messages are exchanged in order to ensure vehicular mobility:

- The MN obtains an EID address of the EID-prefix defined in the OBU1, as well as the SN gets its EID address of the OBU2.

- The OBU1 and the OBU2 trigger the Map-Register message to MS/MR in order to register their EID-prefix and the corresponding RLOC.
- The MS/MR answers with the Map-Notify message to both OBUs informing about the validity of their register.
- When the ping or streams to a MN, it sends a Map-Request message to MS/MR in order to know the location of the MN EID address.
- The MS/MR sends a Map-Reply message to the SN containing the RLOC address of the MN, the same of OBU1.
- The SN achieves in its cache the RLOC of the MN and it establishes a directly tunnel to the MN, sending traffic through the tunnel.
- When the MN changes the RLOC, it sends a Map-Request message to the SN containing the SMR bit enabled.
- Upon receiving the SMR, the SN triggers a Map-Request message to the MS/MR to know the new location.
- Upon receiving the new RLOC, the SN establishes a new tunnel to the MN, communicating now through this tunnel, since the previous tunnels are no longer reachable.

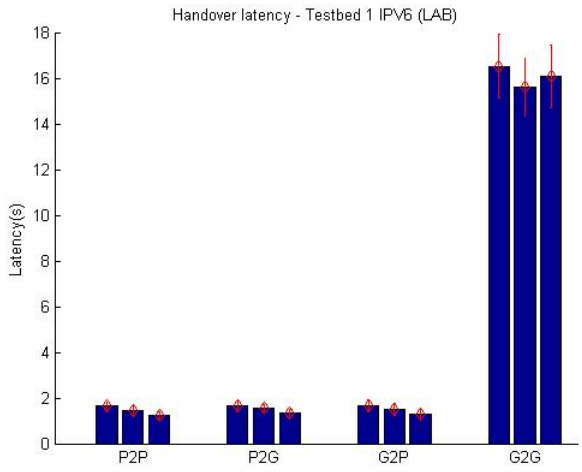


Figure 5.8: Hand-latency - T1 IPV6 (LAB)

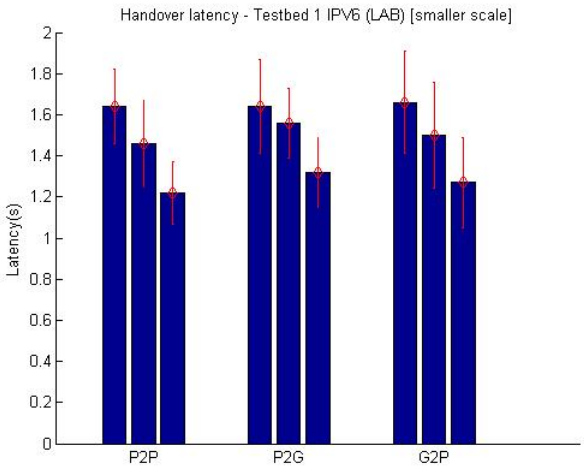


Figure 5.9: Detail of figure 5.8

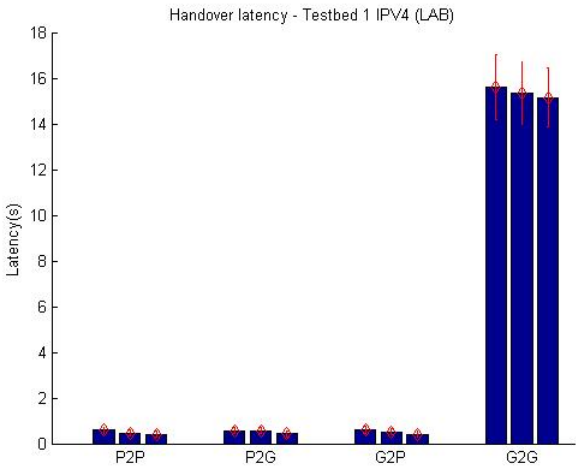


Figure 5.10: Hand-latency - T1 IPV4 (LAB)

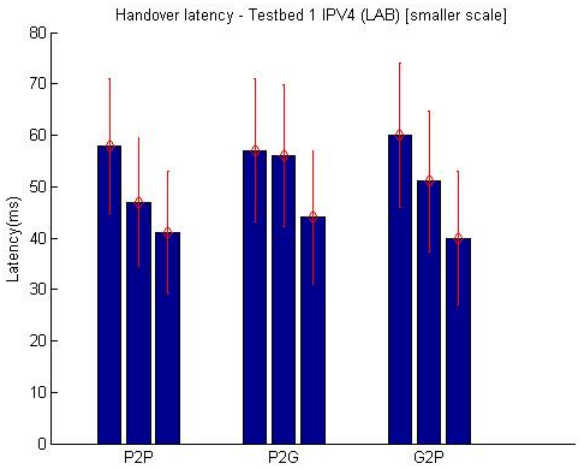


Figure 5.11: Detail of figure 5.10

5.4.1.2 Second Testbed

Regarding the second testbed, the tests are done the same way as in the first one, with the SN replaced by another MN, so it is tested the communication between two MNs into different LISP sites.

Thus, as expected, the results of this testbed presented in figure 5.13 and better detailed in figure 5.14 with all the elements in IPv6, and in figure 5.15, and detailed in figure 5.16 with all the elements in IPv4, were similar to the ones of the first testbed.

The small difference between the results of this testbed comparing to the first one can be explained by the fact that the communications were performed in a simultaneous roaming with both MN in movement. In this case, there are more LISP messages exchanged, which means that the caches are constantly updating, slightly delaying the process. Nevertheless, this small difference is not relevant.

Finally, once again, taking into account these handover latency results, it is clear that WI-FI technology is not an appropriate technology for the vehicular handovers, since it takes a longer time to perform handover.

5.5 Road Experiments Results

In this section it is presented the results obtained in the road environment.

Two types of tests were carried out with a vehicle moving at a speed of approximately 40, 50 and 60 km/h, and the connection between the RSUs towards the MS/MR is now

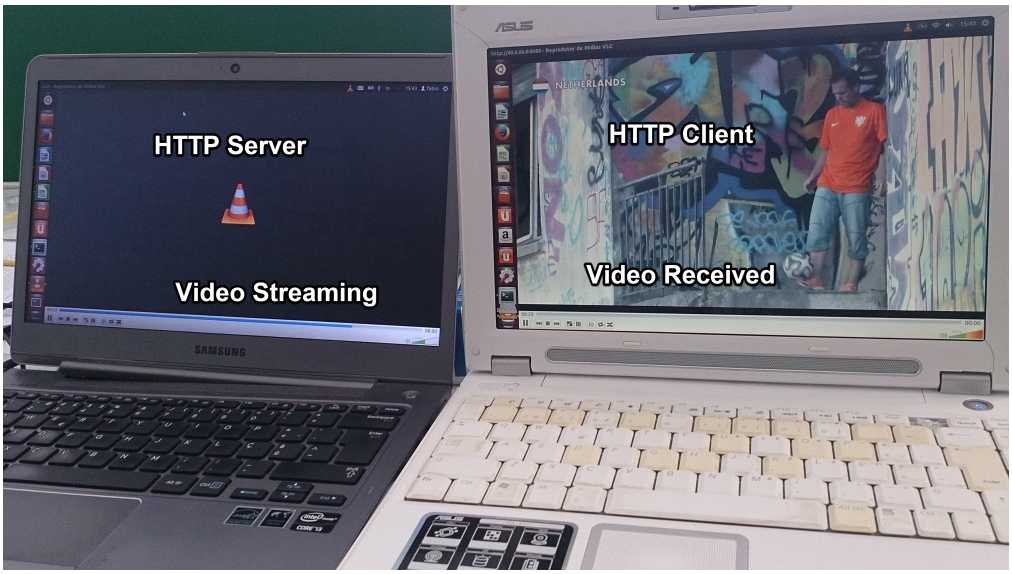


Figure 5.12: Video Streaming Process

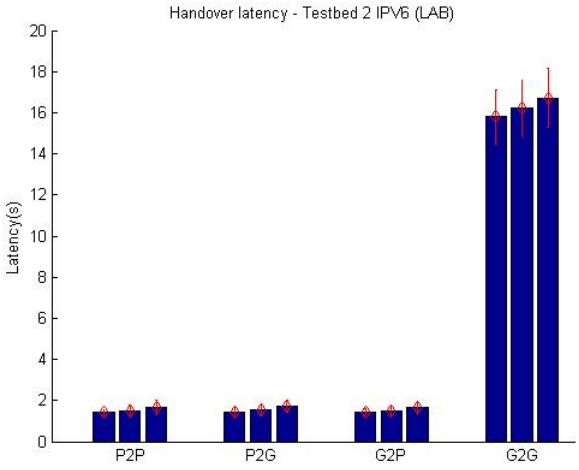


Figure 5.13: Hand-latency - T2 IPV6 (LAB)

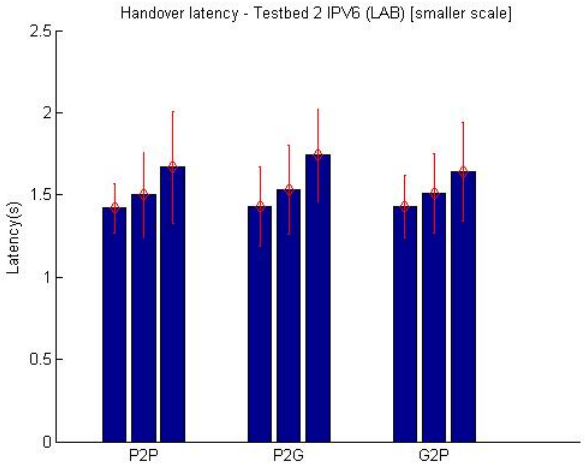


Figure 5.14: Detail of figure 5.13

performed by a WI-FI connection. In the first one, the RSUs are separated about 80 meters (m), while in the second one they are separated about 120 m. Since the WAVE is the most appropriate technology for vehicular networks, and taking into account the results obtained in the lab, it was just performed the P2P handover on the real environments using the *ping* tool.

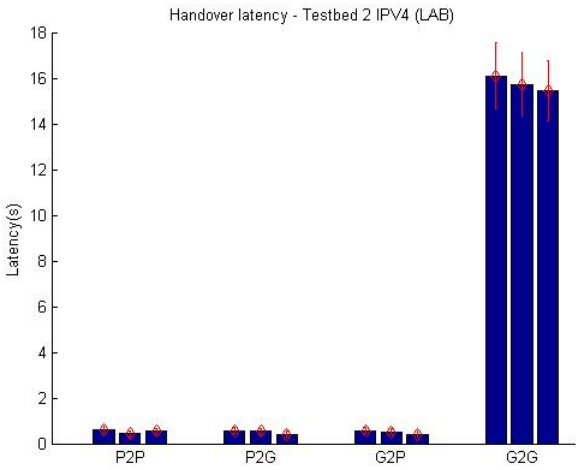


Figure 5.15: Hand-latency - T2 IPV4 (LAB)

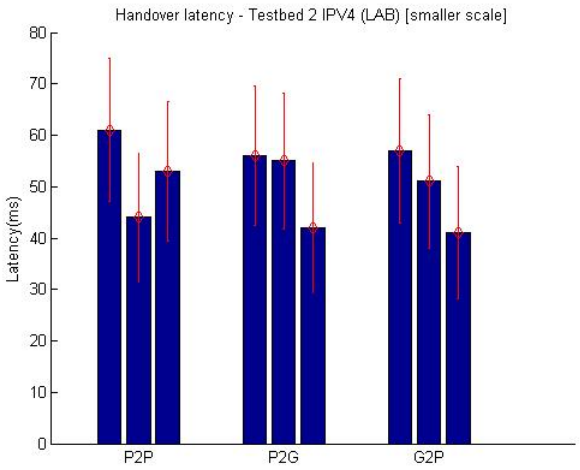


Figure 5.16: Detail of figure 5.15

5.5.1 Handover Latency

The handover latency results can be seen in figures 5.17 and 5.18.

Comparing between each bar graph mentioned above, it is possible to show that with 80 m distance between RSUs the handover times are slightly lower than when spaced between 120 m. However, when the RSUs are separated by 120 m, although not significant, the transition is less abrupt than when separated by 80 m.

Comparing the road results with the lab tests of the testbed 1, it is possible to confirm that they are very similar with a slight increase in the handover latency times of the road tests. Despite not being a significant increase of handover latency times, this happens due to the fact that the connection between the RSUs to the MS/MR is performed by WI-FI technology instead of Ethernet cable, and due to the adverse conditions encountered in the real environment.

Comparing the LISP results (handover times between 60 and 70 msec) with the ones of N-PMIPv6 in [33] in the same scenario conditions (handover times between 40 and 50 msec), we observe that LISP times are slightly increased, due to the overhead of the MS and all the signalling associated. However, the values are in the same order of magnitude, and we can state that LISP is a suitable protocol for vehicular networks.

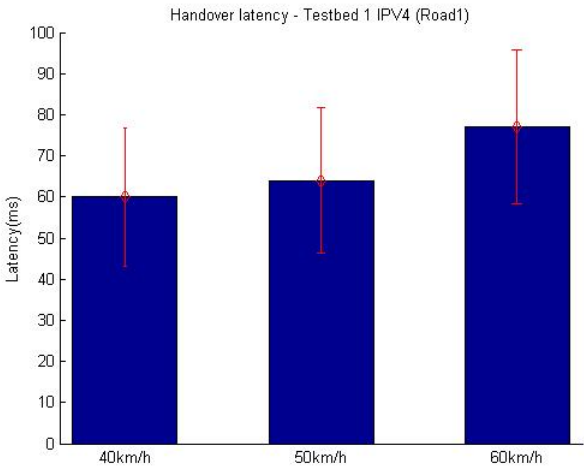


Figure 5.17: Testbed 1 ROAD (80 m)

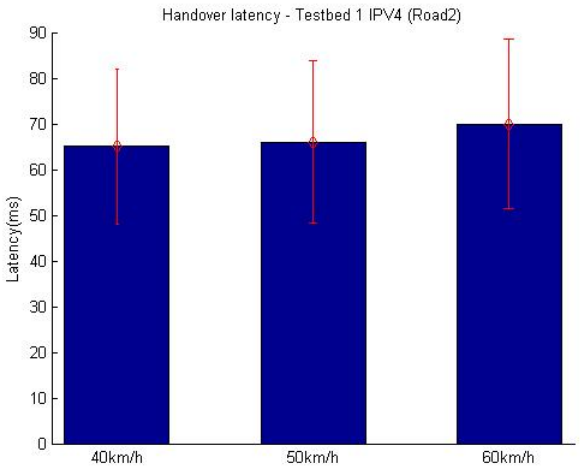


Figure 5.18: Testbed 1 ROAD (120 m)

5.6 Chapter Considerations

In this chapter it has been shown and explained the overall results in both laboratory and road testbeds applied in different environment conditions, and it is assessed the feasibility of the LISP protocol in vehicular environments.

Firstly, it was defined two possible scenarios in order to test LISP mobility protocol in vehicular networks. According to the lab tests, they have shown the correct mobility protocol operation, since the OBU can move through different attachment points, and it is still able to be reachable as well as to communicate with other MNs or SNs. It has also been shown the capability not only to support horizontal handover, but also vertical handover; in other words, the mobility is ensured between attachment points of the same (horizontal) and different (vertical) technologies. However, WI-FI technology is inappropriate for vehicular mobility, which reflects in a higher handover latency comparing to the other intra and inter-technology handovers cases performed.

Regarding the results obtained in a real vehicular environment, they have also shown the correct mobility protocol operation, since the road results are very similar to the lab ones with a slight increase of the handover times, due to the fact that the connection between RSU towards to the Map-Server is done via WI-FI, as well as due to the adverse conditions present in the real environments.

Comparing these results with those obtained with N-PMIPv6 in our group [33], we conclude that the WAVE is the most suitable access technology in the communication between vehicles providing handover latency times of around milliseconds (ms), in which it is quite

favourable for handover with fast transitions that is the case of vehicle handover.

To sum up, with the implementations done and described in chapter 4, it was ensured the mobility to the vehicles and to the to MNs connected in the vehicles, which the best results happen when the handover is performed using WAVE technology.

Chapter 6

Conclusions and Future Work

6.1 Conclusions

Along this thesis, as a goal of this research, the LISP mobility protocol, a LISP Mapping System and a connection manager have been adopted, implemented, setup and adapted to vehicular networks in a multi-technology network approach.

In this Dissertation it was explored the LISP mobility protocol in vehicular environments which, to the best of our knowledge, it is the first work developed with this protocol in VANETs.

Thus, during this work, a distributed database, known as Map-Server and Map-Resolver was set up and implemented in a virtual machine, more specifically in CSR 1000v router acting as an anchor in charge of storing the locations of the corresponding identifiers, as well as to provide the location of any looked up identifier. Further, along this work, just one Map-Server was implemented in a private environment, but it is possible to implement as much as we want and spread that over the world, and therefore work perfectly as distributed database in order to solve the scalability issues.

With LISP and LISP-MN, the IP address is split into two name spaces, the location and the identifier which ensure natively mobility and also become able a natively multihoming, allowing the MNs to connect to more than one RSU or AP at the same time, saving the network resources and becoming more appealing.

A LISP-MN implementation provided by LISPmob has suffered several changes described previously, and together with the mentioned configuration files in the OBUs, it worked in order to provide fast and seamless handovers to the vehicles and their passengers.

A connection manager was developed in order to automatically perform handover to

the best connection available; thus, when a vehicle is moving along the road within the range of one stronger network, it triggers the handover to that network without losing the connection to a SN or a MN previously established.

With *radvd*, *rdisc6* and *DHCP* configurations, it was developed a mechanism for the MNs to connect to the OBU through WI-FI technology to automatically obtain an IPv6 or IPv4 addresses respectively. This is done in order to enable every MN attached to the OBUs via WI-FI to obtain an EID address from the correspondent EID-prefix defined in each OBU on the LISP configuration file. In addition, the *radvd* and *rdisc6* were included in the connection manager in order for the OBU to obtain the requested IPv6 prefix address from the RSU and form its IPv6 address.

Moreover, those developments were tested according to two different testbeds, two different internet protocol versions and in two different environments with intra and inter-technology handovers. Therefore, taking into account the results observed in the evaluation section, we can conclude that the mobility with LISP protocol was ensured to the vehicular networks, as well as the fact that the WAVE access technology is the most appropriate to the VANETs providing seamless handovers.

Although there is a believe that vehicular networks are the future in a society always connected, it still contains a plenty of things to discover and develop to enhance the network vehicular world. The LISP mobility protocol applied in vehicular networks is not an exception; for this reason, there is still much to do, and some of the issues will be included in the next section.

6.2 Future work

As was previously referred, there are several issues that still exist and need to be overcome to make LISP a reality in vehicular networks. Some of them will be mentioned below:

- `Multihoming`: Multihoming described deeply in [15] is an important feature that must be implemented in LISP mobility protocol in order to enable all MNs to connect to more than one RSU or AP at the same time through multiple access technologies. Thus, the resources would be better used with lower network overhead.

- `Extend MS/MR`: In order to solve the scalability problem existent in internet and in several mobility protocols, the LISP provides a distributed database known as mapping system with the possibility to have more than one MS and MR. During this

dissertation, just one MS and MR were implemented in a private environment, so to the future, to solve the routing stability issues and to improve the LISP protocol into vehicular environment, it shall be implemented more than one MS and MR in different locations in a public environment.

- `Evaluation in different Scenarios`: Regarding the evaluation chapter, two testbeds were performed. According to the subsection 3.4.5 there are several scenarios that could be implemented. In addition, unless LISP is running in the Internet, it must be tested a communication between a MN and a SN with the MN in a LISP site representing a vehicle, and the SN representing an internet server in a non-LISP site. Thus, the users inside the vehicles can have an internet connection, which is nowadays fundamental. As the communication between a SN and a MN was performed successfully on this dissertation, it can be a good starting point.

- `Handover time improvements`: Although the handover latency results are very good, in IPv6 it may be possible to improve if some modifications are performed in the kernel layer related with the messages exchange between the OBUs and the LISP-MN.

Bibliography

[1] Linux IPv6 router advertisement daemon (radvd). Available: `http://www.litech.org/radvd/`, retrieved on 1 June 2014.

[2] Bachelor of Communications Engineering Ahmad Rasem. O-PMIPv6: Optimized proxy mobile IPv6. Master's thesis, Carleton University, April 2011.

[3] Loránd Jakab Vina Ermagan Preethi Natarajan Albert Cabellos, Alberto Rodríguez Natal and Fabio Maino. LISPmob: Mobile networking through lisp. 2012.

[4] Carlos Ameixieira, José Matos, Ricardo Moreira, André Cardote, Arnaldo Oliveira, and Susana Sargento. An IEEE 802.11 p/WAVE implementation with synchronous channel switching for seamless dual-channel access (poster). In *Vehicular Networking Conference (VNC), 2011 IEEE*, pages 214–221. IEEE, 2011.

[5] Varun Bhatia. IEEE 802.11p – the future of connected cars. Available: `http://technicafe.net/2013/01/ieee-80211p-future-of-connected-cars.html`, retrieved on 1 February 2014.

[6] Marc Blanchet et al. Special-use ipv6 addresses. IETF RFC 5156, April 2008.

[7] Perkins Calhoun and Charles Perkins. Mobile ip network access identifier extension for IPv4. Technical report, IETF RFC 2794, March, 2000.

[8] Jonathan Carvalho. Distributed mobility in dynamic environments. Master's thesis, University of Aveiro, 2013.

[9] Ming-Chiao Chen and Teng-Wen Chang. *"Introduction of Vehicular Network Architectures." Telematics Communication Technologies and Vehicular Networks*. IGI Global, December 31, 2009.

[10] Yuh-Shyan Chen, Ching-Hsueh Cheng, Chih-Shun Hsu, and Ge-Ming Chiu. Network mobility protocol for vehicular ad hoc networks. In *Wireless Communications and Networking Conference, 2009. WCNC 2009. IEEE*, pages 1–6. IEEE, 2009.

[11] Um projeto e uma organização sem fins lucrativos composta de voluntários. VideoLAN organization, VLC media player. Available: `http://www.videolan.org/vlc/`, retrieved on 10 April 2014.

[12] L. Delgrossi D. Jiang. IEEE 802.11p: Towards an international standard for wireless access in vehicular environments. pages 2036 – 2040. Vehicular Technology Conference, 2008.

[13] D. Meyer D Lewis D. Farinacci V. Fuller. RFC 6830: The locator/ID separation protocol (LISP). Cisco Systems, 2013.

[14] D. Farinacci D. Meyer, D. Lewis. LISP mobile node, draft-meyer-lisp-mn-06.txt. Internet Engineering Task Force, 2011.

[15] Amine Dhraief and Abdelfettah Belghith. Multihoming support in the internet: A state of the art. 2010. International Conference on Models of Information and Communication Systems (MICS 2010), Rabat, Morocco.

[16] Jorge Dias, André Cardote, Filipe Neves, Susana Sargento, and Arnaldo Oliveira. Seamless horizontal and vertical mobility in VANET. In *Vehicular Networking Conference (VNC), 2012 IEEE*, pages 226–233. IEEE, 2012.

[17] Michael Hoing Dominik Klein, Matthias Hartmann and Michael Menth. Improvements to LISP mobile node. University of Wuerzburg, Germany, 2010.

[18] Old Dominion Universit Dr. Michele Weigle, Department of Computer Science. Standards:WAVE / DSRC / 802.11p. Spring 2008.

[19] G. Tsirtsis et al. Flow bindings in mobile ipv6 and network mobility (NEMO) basic support. RFC 5648, 2011.

[20] Gundavelli et al. Proxy mobile IPv6, IETF RFC 5213, August 2008.

[21] Yan Zhang Hassnaa Moustafa. *Vehicular Networks: Techniques, Standards, and Applications.* Auerbach publications, April 2009.

[22] Universidad Politécnica de Madrid; Carlos J. Bernardos Ignacio Soto, Universidad Carlos III de Madrid; María Calderón, and Alcatel Lucent Bell Labs Telemaco Melia. PMIPv6: A network-based localized mobility management solution.

[23] Cisco Systems Inc and Barcelona Tech University. An open-source LISP implemantation for linux, android and openWRT. Available: `http://lispmob.org`, retrieved on 1 February 2014.

[24] The MathWorks Inc. Matlab version 7.10.0 r2010a. *Available: http://www.mathworks.com/products/matlab/*, 2010.

[25] Treck Inc. Treck mobile IPv6 mobile node. Available: `http://www.treck.com/treck-mobile-ipv6-datasheet`, retrieved on 1 February 2014.

[26] Seil Jeon, Rui Aguiar, and Behcet Sarikaya. Network mobility support using mobile mag in proxy mobile IPv6 domain. IETF draft-sijeon-netext-mmag-pmip-00, 2012.

[27] Daniel Jiang and Luca Delgrossi. IEEE 802.11p: Towards an international standard for wireless access in vehicular environments. In *Vehicular Technology Conference, 2008. VTC Spring 2008. IEEE*, pages 2036–2040. IEEE, 2008.

[28] David Johnson, Charles Perkins, Jari Arkko, et al. Mobility support in IPv6, 2004.

[29] James Kempf et al. Goals for network-based localized mobility management (NETLMM). IETF RFC 4831, April 2007.

[30] John B Kenney. Dedicated short-range communications (DSRC) standards in the united states. *Proceedings of the IEEE*, 99(7):1162–1182, 2011.

[31] R. Koodli. Fast handovers for mobile ipv6 (FMIPv6). RFC 4068, 2005.

[32] Clemson Vehicular Electronics Laboratory. Dedicated short range communications. Available: `http://www.cvel.clemson.edu/auto/systems/dsrc.html`, retrieved on 1 June 2014.

[33] Diogo Lopes and Susana Sargento. Network mobility for vehicular networks. In *International Symposium on Computer and Communications (ISCC), Madeira, Portugal.* IEEE, Junho 2014.

[34] Diogo Miguel Augusto Lopes. Acesso à internet com handover de veículos através de gateways móveis. Master's thesis, University of Aveiro, 2013.

[35] Albert Cabellos-Aparicio Loránd Jakab and Jordi Domingo-Pascual. Supporting mobility in LISP. Technical report 2009-55, Universitat Politècnica de Catalunya, Barcelona, Oct. 2009.

[36] T. Narten, E. Nordmark, and W. Simpson. H. soliman, neighbor discovery for ip version 6 (IPv6). Technical report, RFC 4861, September, 2007.

[37] Peteruithoven. About openwrt. Available: `http://wiki.openwrt.org/about/start`, retrieved on 1 March 2014.

[38] Bucknell University R. Droms. Dynamic host configuration protocol (DHCP). IETF-RFC 2131, March 1997.

[39] Tim Siglin. HTTP streaming: What you need to know. 2010 Streaming Media Sourcebook.

[40] W. Simpson T. Narten, E. Nordmark. Neighbor discovery for IP version 6 (IPv6), RFC 2461, December 1998.

[41] T.Condeixa and S.Sargento. Dynamic mobile IP anchoring. IEEE ICC, June 2013.

[42] Fumio Teraoka and Tetsuya Arita. PNEMO: a network-based localized mobility management protocol for mobile networks. In *Ubiquitous and Future Networks (ICUFN), 2011 Third International Conference on*, pages 168–173. IEEE, 2011.

[43] Ajay Tirumala, Feng Qin, Jon Dugan, Jim Ferguson, and Kevin Gibbs. Iperf: The TCP/UDP bandwidth measurement tool. *Available: http://dast.nlanr.net/Projects*, 2005.

[44] Asanga Udugama, Muhammad Umer Iqbal, Umar Toseef, Carmelita Goerg, Changpeng Fan, and Morten Schlaeger. Evaluation of a network based mobility management protocol: PMIPv6. In *Vehicular Technology Conference, 2009. VTC Spring 2009. IEEE 69th*, pages 1–5. IEEE, 2009.

[45] D. Meyer V. Fuller, D. Farinacci and D. Lewis. LISP alternate topology (LISP+ALT), draft-ietf-lispalt-05. Internet Engineering Task Force, 2010.

[46] Various volunteers from various organisations. LISP beta network. Available: `http://www.lisp4.net/beta-network/`, retrieved on 1 February 2014.

[47] Mohamed Watfa. *Advances in Vehicular Ad-Hoc Networks: Developments and Challenges*. Information Science Reference, April 2010.

[48] Kun Zhu, Dusit Niyato, Ping Wang, Ekram Hossain, and Dong In Kim. Mobility and handoff management in vehicular networks: a survey. *Wireless Communications and Mobile Computing*, pages 1–20, 2009.