Aiash, Mahdi, 2012. An integrated approach to QoS and security in future mobile networks using the Y-Comm framework. Available from Middlesex University's Research Repository.

An Integrated Approach to QoS and Security in Future Mobile Networks using the Y-Comm Framework

Mahdi Aiash

School of Engineering and Information Sciences

Middlesex University

A thesis submitted to Middlesex University in fulfilment of the requirements for degree of Doctor of Philosophy

March 2012

Abstract

Future networks will comprise a wide variety of wireless networks. Users will expect to be always connected from anywhere and at any time as connections will be switched to available networks using vertical handover techniques. However, different networks have different Qualities-of-Service (QoS) so a QoS framework is needed to help applications and services deal with this new environment. In addition, since these networks must work together, future mobile systems will have an open, instead of the currently closed, architecture. Therefore new mechanisms will be needed to protect users, servers and network infrastructure. This means that future mobile networks will have to integrate communications, mobility, quality-of service and security.

However, in order to achieve this integration without affecting the flexibility of future networks, there is a need for novel methods that address QoS and security in a targeted manner within specific situations. Also, there is a need for a communication framework wherein these methods along with the communication and handover mechanisms could be integrated together. Therefore, this research uses the Y-Comm framework, which is a communication architecture to support vertical handover in Next Generations Networks, as an example of future communication frameworks that integrate QoS, security, communication and mobility mechanisms. Within the context of Y-Comm, research has been conducted to address QoS and security in heterogeneous networks.

To preserve the flexibility of future network, the research in this thesis proposes the concept of Targeted Models to address security and QoS in specific scenarios: to address the QoS issue, a new QoS framework is introduced in this thesis, which will define targeted QoS models that will provide QoS in different situations such as connection initiation and in the case of handover. Similarly, to deal with the security side, targeted security models are proposed to address security in situations like connection initiation and handover.

To define the targeted models and map them to actual network entities, research has been conducted to define a potential structure for future networks along with the main operational entities. The cooperation among these entities will define the targeted models. Furthermore, in order to specify the security protocols used by the targeted security models, an Authentication and Key Agreement framework is introduced to address security at different levels such as network and service levels. The underlying protocols of the Authentication and Key Agreement protocol are verified using Casper/FDR, which is a well-known, formal methods-based tool.

The research also investigates potential methods to implement the proposed security protocols. To enable the implementation of some of the targeted security models, the research

also proposes major enhancements to the current addressing, naming and location systems.

Acknowledgements

This research project would not have been possible without the support of many people. It is a pleasure to convey my gratitude to them all in my humble acknowledgement.

In the first place I would like to express my gratitude to Dr. Glenford Mapp for his supervision, advice, and guidance from the very early stage of this research as well as giving me extraordinary experiences throughout the work. Above all and the most needed, he provided me unflinching encouragement and support in various ways. His truly scientist intuition has made him as a constant oasis of ideas and passions in science, which exceptionally inspire and enrich my growth as a student, a researcher and a scientist want to be. I am indebted to him more than he knows.

I gratefully acknowledge Dr. Aboubaker Lasebae for his advice, supervision, and crucial contribution, which made him a backbone of this research and so to this thesis. His involvement with his originality has triggered and nourished my intellectual maturity that I will benefit from, for a long time to come. Abu, I am grateful in every possible way and hope to keep up our collaboration in the future.

I am deeply indebted to my advisor, Dr. Raphael Phan, for his constant support. Without his help, this work would not be possible. I am very grateful for the stimulating science discussion we had in Loughborough, and I am looking forward to seeing future work together.

I am deeply and forever indebted to my parents for their love, support and encouragement throughout my entire life. My parents deserve special mention for their inseparable support and prayers. My Father, in the first place is the person who put the fundament to my learning character, showing me the joy of intellectual pursuit ever since I was a child. My Mother is the one who sincerely raised me with her caring and gentle love.

I cannot express my full gratitude to my wife whose dedication, love and persistent confidence in me, have taken the load off my shoulder. I would also like to thank my children for bringing a new colour to my life.

Finally, words fail me to express my appreciation to my brother-in-law and his family for their support all the way.

List of Publications

The work presented in this thesis has given rise to the following publications.

Journal Papers

1. M. Aiash, G. Mapp, A. Lasebae, R. Phan, J. Loo (2012), ”A formally verified AKA protocol for vertical handover in heterogeneous environments using Casper/FDR”. EURASIP Journal on Wireless Communications and Networking 2012:57 doi:10.1186/1687-1499-2012-57, February 2012.

2. M. Aiash, G. Mapp, A. Lasebae (2011), ”Security and QoS Integration for Protecting Service Providers in Heterogeneous Environments”. The International Journal of Computer Science (IJCS), Volume 38 Issue 4, Pages 384-393.

3. M. Aiash, G. Mapp, A. Lasebae, R. Phan, J. Loo (2012), ”A Formally Verified Service-Level AKA Protocol for Secure Services in Heterogeneous Environments using Casper/FDR”. The Wilye International Journal of Communication Systems, (Accepted).

4. M. Aiash, G. Mapp, A. Lasebae, R. Phan, J. Loo (2012), ”A Formally Verified Initial AKA Protocol in Heterogeneous Environments Using Casper/FDR”.EURASIP Journal on Information Security (Submitted).

Conference Papers

1. G. Mapp, M. Aiash, F. Shaikh, R. Porto Vanni, M. Augusto and E. Moreira (2009), ”Exploring Efficient Imperative Handover Mechanisms for Heterogeneous Wireless Networks”. International Symposiumon Emerging Ubiquitous and Pervasive Systems (EUPS-09) August 2009.

2. M. Aiash, G. Mapp, A. Lasebae, R. Phan (2010), ”Providing Security in 4G Systems: Unveiling the Challenges”. IEEE AICT2010, Barcelona, Spain.

3. G. Mapp, M. Aiash, A. Lasebae, R. Phan (2010), ”Security Models for Heterogeneous networking”. SECRYPT2010, Athens, Greece.

4. M. Aiash, G. Mapp, A. Lasebae (2011), ”A QoS Framework for Heterogeneous Networking”.The International Conference of Wireless Networks 2011 (ICWN11). 6-8 July, 2011. London. **Best Student Paper Award Winner**.

5. M. Aiash, G. Mapp, A. Lasebae, R. Phan, M. Augusto, R. Porto Vanni, and E. Edson (2011), "Enhancing Naming and Location Services to support Multi-homed Devices in Heterogeneous Environments". The IEEE conference on Communication, Science and Information Engineering (CCSIE 2011), 25-27 July 2011. London-UK.

6. M. Aiash, G. Mapp, A. Lasebae, R. Phan (2011), "Exploring the Concept of Scope to Provide Better Security for Internet Services". The IEEE conference on Communication, Science and Information Engineering (CCSIE 2011), 25-27 July 2011. London-UK.

7. G. Mapp, M. Aiash, H.C. Guardia and J. Crowcroft (2011), "Exploring Multi-homing Issues in Heterogeneous Environments". 1st International Workshop on Protocols and Applications with Multi-Homing Support (PAMS11). Mar 22-25, 2011. Singapore.

8. M. Aiash, G. Mapp, R. Phan, A. Lasebae, J. Loo (2012), "A Formally Verified Device Authentication Protocol Using Casper/FDR". The 2012 International Symposium on Advances in Trusted and Secure Information Systems, IEEE TrustCom 2012- Liverpool, UK.

9. M. Aiash, G. Mapp, A. Lasebae, J. Loo, R. Phan (2012), "A Survey of Potential Architectures for Communication in Heterogeneous Networks". The IEEE Wireless Telecommunications Symposium (WTS 2012), April 2012. London, UK.

Book Chapter

1. M. Aiash, G. Mapp, A. Lasebae, R. Phan, J. Loo (2012) ”INTEGRATING MOBILITY QUALITY-OF-SERVICE AND SECURITY IN FUTURE MOBILE NETWORKS”. Accepted as Book Chapters of the Springer Title: Electrical Engineering and Intelligent Systems.

Contents

1 Introduction 1

1.1 Background 1

1.2 Research Question 3

1.3 Main Contributions 3

1.4 Originality of Intended Work 4

1.5 The Scope of this Thesis 5

1.6 The Structure of this Thesis 6

2 Overview of QoS and Security Research in Heterogeneous Networks 8

2.1 Introduction 8

2.2 Network Evolution to Support Heterogeneous Networks 8

2.3 The Integrated Service (IntServ)and the Differentiated Service (DiffServ)QoS mechanisms in the Internet 9

2.3.1 The IntServ 10

2.3.2 The DiffServ 11

2.4 Handover Classification 12

2.5 IEEE 802.21 Standard 13

2.5.1 Analysis 14

2.6 Ambient Networks 14

2.7 The HOKEY WG 15

2.7.1 ERP Key Hierarchy 16

2.7.2 An Overview of the ERP method 16

2.7.3 Analysis of ERP 17

2.8 The Mobile Ethernet Architecture 18

2.8.1 Analysis 20

2.9 The Y-Comm Framework 20

2.9.1 An Overview of Y-Comm 20

2.9.2 Vertical Handover in Y-Comm 23

2.9.3 Y-Comm Security Framework 24
2.9.4 Analysis 25
2.10 A Summary and Overall Evaluation 26

3 Approach and Constraints 28
3.1 Introduction 28
3.2 Proposed Work 28
3.2.1 The Concept of Targeted Models 28
3.3 Research Methodology 29
3.3.1 The Threat Model 29
3.3.2 Solution Approach 30
3.4 QoS-Related Work 30
3.5 Security-Related Work 31
3.5.1 The Targeted Security Models 31
3.5.2 A multi-layer AKA Protocols 31
3.5.3 Verifying Security Protocols 32
3.6 Summary 33

4 A QoS Framework for 4G Systems 34
4.1 Introduction 34
4.2 Network Architecture Overview 35
4.3 Network Entities 37
4.3.1 Network Entities Structure 37
4.3.2 Network Protocols 39
4.3.3 The Network Architecture in the Context of Y-Comm 40
4.4 Targeted QoS-Signalling models 41
4.4.1 The Registration Model 41
4.4.2 The Connection Initiation Model 42
4.4.3 The Handover Model 43
4.5 Summary 44

5 The Authentication and Key Agreement Framework 46
5.1 Introduction 46
5.2 An Overview of the AKA Framework 47
5.3 The Key Hierarchy 48
5.4 Analysing the Security Protocols 49
5.5 The User-Level AKA Protocol 50

5.5.1 Introduction 50
5.5.2 Related Work 50
5.5.3 The Proposed User-Level AKA Protocol 53
5.5.4 Biometric-Information Based Authentication 57
5.6 Summary 58

6 The AKA Framework, The Network-Level AKA Protocol 59
6.1 Introduction 59
6.2 Problem Definition 59
6.3 Initial NL-AKA Protocol 60
6.3.1 Related Work 60
6.4 The Initial AKA protocol for the Mobile Ethernet 62
6.4.1 The Protocol Description 63
6.4.2 The Formal Verification of the Mobile Ethernet Protocol 64
6.4.3 Protocol Analysis and Security Consideration 66
6.5 The Proposed Solution 67
6.5.1 Defining the Security System 67
6.5.2 The Key Hierarchy 68
6.5.3 The Initial Version Of the Protocol 69
6.5.4 The Second Version of the Protocol 77
6.5.5 The Third Version of the Protocol 80
6.5.6 AKA Protocol Formal Verification and Security Analysis 80
6.5.7 AKA Protocol Formal Verification 84
6.5.8 Protocol Analysis and Security Consideration 85
6.6 Secure Vertical Handover in Heterogeneous Environment 86
6.6.1 Related Work 86
6.7 Secure Vertical Handover in Mobile Ethernet 87
6.7.1 The Formal Verification of the Mobile Ethernet Protocol 88
6.7.2 Protocol Analysis and Security Considerations 90
6.8 The Proposed Solution 91
6.8.1 The Initial Version Of the Protocol 92
6.8.2 The Second Version Protocol 94
6.8.3 The Final Protocol 95
6.8.4 AKA Protocol Formal Verification 95
6.9 Summary 95

7 The AKA Framework- The Service-Level AKA Protocol 97

7.1 Introduction 97

7.2 Problem Definition 97

7.3 Related Work 98

7.3.1 Transport Layer Security (TLS) and Secure Sockets Layer (SSL) . . . 99

7.3.2 The Stream Control Transmission Protocol (SCTP) 99

7.3.3 The Service-Level AKA of the Mobile Ethernet framework 100

7.4 The Proposed Service-Level AKA Protocol 100

7.4.1 Overview of Future Networks 100

7.4.2 The Initial SL-AKA Protocol 101

7.4.3 Light Weight SL-AKA Protocol for Handover 105

7.5 Summary 106

8 The Targeted Security Models 108

8.1 Introduction 108

8.2 Recalling the Y-Comm Integrated Security Module 108

8.3 The Ring-Based Security Model 109

8.3.1 The Motivation 109

8.3.2 The Model Outline 110

8.4 The Connection Security Model 110

8.4.1 The Connection Model Outline 111

8.4.2 Defining the Connection Security Model Mechanisms 112

8.5 The Secure Vertical Handover Model 113

8.5.1 The Model Outline 114

8.5.2 Defining the Secure Vertical Handover Model Mechanisms 115

8.6 Summary 116

9 Potential Implementation Approaches for the Proposed Security Mechanisms 117

9.1 Introduction 117

9.2 Enhancing Network Services to implement the Ring-Based Model 118

9.2.1 Investigating the Multi-Homing issue in Future Networks 118

9.2.2 Specifying the Ring-Based Model 120

9.3 The Problems of Implementing Security Protocols 124

9.4 Potential Compilers for Security Protocols 125

9.4.1 COSP-J Compiler 125

9.4.2 The ACG-C# 128

9.4.3 Comparing COSP-J and ACG-C# 128
9.5 Using Ontology to Support the Integration 129
9.5.1 An Overview of Ontologies 129
9.5.2 How Would an Ontology Help! 129
9.5.3 An Ontology for Y-Comm 130
9.6 Summary 130

10 Conclusions and Future Work 131
10.1 Introduction 131
10.2 How were the key research questions addressed? 131
10.3 Main Contributions 132
10.4 Elaboration on the main contributions 133
10.4.1 Identification of crucial gaps in knowledge in the field of providing QoS and Security in heterogeneous environments 133
10.4.2 Defining a generic structure of heterogeneous networks 133
10.4.3 Introducing a QoS framework 134
10.4.4 Investigating the Multi-homing issue of Future Mobile Devices 134
10.4.5 Providing Security in Heterogeneous network 134
10.4.6 Defining the Target Security Models 135
10.5 Future improvements to solutions from which the study can benefit 135
10.6 What are the future works that can be pursued based on this study? 136
10.6.1 Implementing the new addressing scheme and the Ring-Based Model 136
10.6.2 Implementing the Connection and the Vertical handover security model 136
10.7 Concluding remarks 136

List of Figures

2.1 The Future Internet Structure 9

2.2 The Core End Point 10

2.3 Vertical Handover Classification (GMMAM09) 13

2.4 The MIH Framework 14

2.5 The ERP Key Hierarchy 16

2.6 The Mobile Ethernet Layer (Kur05) 18

2.7 The Mobile Ethernet Network Structure (Kur05) 19

2.8 The Y-Comm Framework 21

2.9 The Proactive Handover 23

2.10 The Proactive Handover Transaction 24

2.11 The Y-Comm Complete Structure 25

4.1 The ITU Recommended Network Parties 35

4.2 The Hierarchical Network Structure 36

4.3 The Structure and Relationship Between the Network Entities 38

4.4 The Registration Model 42

4.5 The Connection Initiation Model 43

4.6 The Intra-Core End Point Handover Model 44

4.7 The Inter-Core End Point Handover Model 45

5.1 The AKA Framework Stages 48

5.2 Key Hierarchy 49

6.1 The Mobile Ethernet Protocol 64

6.2 The Attack Against the Initial AKA Protocol of the Mobile Ethernet 65

6.3 The Hierarchical Network Structure 68

6.4 The Key Hierarchy 69

6.5 The Attack Against the `Secret(MT, SK, [Auth, DA3C]` 74

6.6 The Attack Against the `Agreement(DA3C, MT, [AK])` 76

6.7 The Mobile Ethernet Protocol 88
6.8 The First Attack on the Mobile Ethernet 89
6.9 The Second Attack of the Mobile Ethernet 90

8.1 The Ring-Based Security Model 111
8.2 The Connection Security Model 112
8.3 The Steps of Connection Security Model 113
8.4 The Secure Vertical Handover Model 114
8.5 The Steps for the Secure Vertical Handover Model 115

9.1 The Address Format 119
9.2 Enforcing the Reachability Based on the Scope 121
9.3 Enforcing the Scope Concept in The Ring-Based Model 122
9.4 COSP-J and Casper Abstraction (Did09) 126
9.5 The use of Casper and COSP-J together (Did09) 126
9.6 Security Protocol Design and Implementation (CJ05) 128

List of Tables

3.1 Casper input file headers 33

4.1 Mapping the Network Entities to the Y-Comm layers 40

5.1 notation 52
5.2 Notation 54

6.1 Notation 63
6.2 Notation 71
6.3 Comparison 85

7.1 Notation 102

8.1 Mapping the AKA mechanisms to the Security module 109

9.1 The eDNS Record 120
9.2 The ML Record 120
9.3 Analysis Summary 124

Chapter 1

Introduction

The world is experiencing a huge growth in the development and deployment of several wireless technologies. Such technologies vary from second and third generation cellular networks, Wireless Local Area Networks (WLANs) to personal area networks. This widespread deployment of wireless networks will have a significant impact on the evolution of the Internet.

In the early Internet, end systems were primarily composed of Ethernet and Token Ring systems which were similar in performance to the systems used in the core network. However, with the wide-scale deployment of wireless networks as end-systems, there is a significant difference in network characteristics in terms of bandwidth, latency, packet loss and error characteristics of peripheral networks when compared with the core network. These developments mean that soon it will not be possible to think of the Internet as a single unified infrastructure (GM10). It would be better to view the Internet as an entity comprising a fast core network with slower peripheral networks attached around the core. The core network will consist of a super-fast backbone using optical switches and fast access networks which uses Multi-protocol Label Switching (MPLS).

The Next Generation Networks (NGNs) need to facilitate ubiquitous connectivity to enable users to connect from anywhere and at any time. Therefore, there is a need for a general architecture to support seamless and secure mobility between heterogeneous networks. The Y-Comm framework (GM07), (GM06) discussed in the following chapter, provides one such communication framework to support vertical handover in heterogeneous environments.

1.1 Background

As proposed by different research efforts such as the IEEE802.21, HOKEY, Mobile Ethernet and the work in ITU (IEE07), (HOK07), (Kur05), (IT04), communication in future networks will be based on IP addresses, these networks will comprise a wide variety of wireless network technologies such as 3G,WiMAX and Long Term Evolution (LTE). In this environment, Mobile Terminals (MTs) will expect to be connected to several networks at the same time and ubiquitous communication will be achieved by seamless switching between available networks using vertical handover techniques. However, since these networks might be of different technologies and controlled by different operators, vertical handover therefore raises serious threats mainly related to providing the best-possible security and QoS across different technologies.

On one hand, the network operators vary in terms of their security levels and mechanisms such as admission control, authentication and authorization algorithms. Therefore, there is a need to define a generic security module to provide security at different levels such as network and service levels. On the other hand, due to the open and flexible nature of the 4G systems, there is a need to protect the network infrastructure as well as the data. Also, there is a need to address security using a dynamic rather than a static approach that targets security in specific situations such as connection initiation, server accessibility and handover. Another challenge in this heterogeneous environments is the end-to-end, cross-operator QoS provision. When the users subscribe to a specific service, this implies that the two end systems have agreed on certain conditions of using the service, this is known as the Service-Level of Agreement (SLA). The SLA indicates contracted delivery time (of the service) along with its performance. In the case of a single operator scenario, where all the resources are controlled and managed by one administrative entity, maintaining the SLA would not be a problem. However, this may not be the case with multiple operators such as 4G systems, where the MT roams among access networks, controlled by different operators. Similar to the security provision in heterogeneous networks, there is a need for a dynamic approach for providing QoS in specific scenarios such as connection initiation and in the case of handover. Additionally, a QoS framework for 4G systems should not only provide resource allocation and reservation, but also consider the security requirement and deploy performance improvement techniques by using trade-offs between security and QoS. Furthermore, as pointed out in (JM04), vertical handover can cause radical changes in QoS. Hence, it is important that as much control as possible is exercised by mobile devices to achieve optimum vertical handover. It is therefore necessary to develop new techniques which could make other layers of the protocol stack aware of impending handover decisions and thus allow them to take steps to minimize the effects. The aforementioned challenges highlight the fact that future mobile systems must encompass communications, mobility, QoS and security, an example of such future framework is Y-Comm (GM07), (GM06). Therefore, it has been used in this research as a representative of future communication frameworks. The Y-Comm architecture provides a well defined architecture, wherein different security and QoS mechanisms could be integrated with the mobility and the communication system.

1.2 Research Question

Targeting the unaddressed challenges in QoS and Security provision in future heterogeneous networks, this research aims to find out answers to a set of important research questions which are detailed below:

'How to introduce a generic architecture for heterogeneous networks and define the main operational entities for QoS and Security provision ?'

This research question required the development of a new architecture for heterogeneous networks along with the network entities, their structures and the interfaces between them. These networks entities are the actual parties which will be involved in the proposed security and QoS mechanisms.

'How the proposed network architecture could be utilized to support an End-to-End and cross-operator QoS provision?'

The answer to this question was the development of a new QoS framework which supports QoS signalling in three different scenarios, and thus three Targeted QoS-Signalling Models are proposed namely, the Initial Registration , the Connection Initiation and the Handover Models.

'Considering the proposed network architecture, how to investigate a new approach to provide security for heterogeneous networks in the context of the Y-Comm framework ?'

The proposed approach aims to protect the data and the network resources from malicious attacks, while considering QoS and the underlying network structure, this has to be achieved without affecting the flexibility and openness of 4G environment.

1.3 Main Contributions

The research introduced a set of underlying security protocols and mechanisms. Some of these were achieved by Authentication and Key Agreement (AKA) frameworks that operated at network and application levels and in different situations such as handover and connection initiation. Others were manifested as access control mechanisms in the network infrastructure. The proposed protocols were verified using formal methods based on Casper/FDR tool (LBDH09), (PR10), (Sys93).

Furthermore, in order to provide a practical security solution and support the integration between the QoS and security, there was a need to consider the actual network structure and its operational entities. This has led to an investigation into the area of networking and QoS. Consequently, three models for signalling QoS in different situation have been proposed. These models cooperate with the security models to support full integration.

The research presented in this thesis proposes a novel approach to address QoS and security integration. The approach is based on the integration between the QoS signalling models and the AKA framework, thus three Targeted Security Models (TSMs) have been introduced. The TSMs address security in different scenarios such as connection initiation, server protection and vertical handover.

1.4 Originality of Intended Work

Many research efforts such as (HOK07), (YZ05), (Kur05), (rGPPG) have been trying to address the security side of the problem, by proposing different security mechanisms. Others such as (IEE07) tried to approach the QoS issue by providing mechanisms to support a proactive and seamless vertical handover.

The main problem with these efforts was the lack of cooperation; the author believes that, in heterogeneous systems, the security and QoS should not be addressed separately. This is due to the fact that although security and QoS are managed by different groups, their implementation will have an impact on each other. Without information about the available security level and the desired requirements, a poor assignment of the QoS parameters may lead to denial of service for vital but low bandwidth data. In contrast, an ignorance of the QoS requirements with the implementation of high security level may reduce the effectiveness of the offered QoS.

The author believe that, by integrating network QoS and security, issues such as the secure delivery of QoS parameters during connection set up, data protection during transmission as well as immunity against denial of service attacks can be achieved. If security mechanisms such as authentication, authorization and access control are enforced during the QoS signalling stage, this will help in providing secure access and authorized resource reservation. Additionally, considering the QoS requirement while determining the security parameters such as the encryption algorithms and the length of the keys will provide secure transmission path as well as content protection without any contradictions with the QoS performance. Therefore, there is a need for a new architecture wherein different security and QoS mechanisms could be integrated with the communication system. The Y-Comm framework (GM07), (GM06) discussed in the following chapter, provides one such communication framework to align with the works of the above groups, and provides the whole package to support security, QoS and connectivity, as well as introducing a model to support vertical

handover.
The research presented in this thesis addresses the shortages of the afore-mentioned related work. For this to be accomplished, the proposed approach introduces some novel concepts, as follows: Firstly, the concept of the multi-layer, Integrated Security Module (ISM) to protect the data and provide security at different levels .i.e network, transport and application levels. Secondly, to provide security for heterogeneous networking without affecting its dynamics or openness, the proposed approach introduced the concept of the Targeted Security Models (TSMs) to protect both the data and the entities such as users, servers and network infrastructure. These models will address the security in specific situations such as connection initiation and the handover. Thirdly, unlike all the related security work in the literature, the proposed security protocols in this thesis were verified using formal methods approach based on the Casper/FDR tool (LBDH09), (PR10), (Sys93). Fourthly, in order to support the QoS-Security integration, a QoS framework for 4G system was proposed. This framework will provide QoS in different situations such as connection and handover and thus, proposes targeted QoS-Signalling models which will be integrated with the targeted security models. Finally, considering a well defined architecture for communication systems in heterogeneous environments such as Y-Comm which, provides a framework to align our proposed security and QoS models. This will enable the full integration between security, QoS and communication.

1.5 The Scope of this Thesis

The research in this thesis is mainly concerned with proposing new QoS and security mechanisms that consider the unique nature of heterogeneous networks, and shows how these mechanisms could be aligned using future communication architectures such as the Y-Comm framework. This has led to the design and analysis of new security and QoS mechanisms which will be integrated with the Y-Comm communication framework. The security protocols were formally verified and integrated in designated security framework.
It is worth pointing out that although Y-Comm has been used as a model framework, the mechanisms proposed in this research to address security and QoS integration for heterogeneous networks could be used with any future communication framework that shares the author's view of how the network will evolve in the future as presented in section 1.
Casper/FDR which is a formal methods-based tool was used to verify the proposed security protocols in this thesis. We chose Casper from a range of options such as AVISPA tool (AVI06), due to its wide-spread usage and wealth of resources. However, verification using Casper only considers the protocol within the system as defined in the protocol description.

It does not consider threats or attacks resulting from the implementation environment as explained in Chapter 8.

This thesis is not focused on experimental implementation and performance analysis. For the QoS framework, there is a need to analyse the performance of the proposed QoS-Signalling models. Future implementation paths for the proposed security protocols are investigated in Chapter 9. Among the potential implementation options is using the Compiler Of Security Protocols into Java (COSP-J) (Did09) or the Automatic Code Generator into C# code (ACG-C#) compilers (CJ05), which respectively produce Java and C# implementations of protocols from Casper-like descriptions. Another option is using Ontological methods to describe the security protocols and to define the integration with the communication framework as well as enhanced network services.

1.6 The Structure of this Thesis

This thesis is structured as follows:

- Chapter 2: reviews the literature of most relevance to the research questions. This includes the following areas:
  1. The work of the IEEE 802.21 WG (IEE07) to introduce a vertical handover mechanism in heterogeneous environments.
  2. The security approach introduced by the HOKEY WG (HOK07) to support secure vertical handover.
  3. The security work introduced by the Mobile Ethernet Architecture group (Kur05) to build a secure service architecture in heterogeneous environments.
  4. Describing the related work in our group ,the Y-Comm Research Group, to build a secure 4G communication framework where security and QoS are addressed in an integrated manner.
- Chapter 3: Analyses the proposed approaches and the devised constraints needed to address the research questions
- Chapter 4: Describes the conducted research to define the actual network architecture along with the Targeted QoS models to provide QoS signalling in different scenarios.
- Chapter 5: Introduces an Authentication and Key Agreement (AKA) framework, which comprises a number of AKA protocols to meet the functionality of some of the layers of the Y-Comm's Integrated Security Module.

- Chapter 6: Develops Network-Level AKA protocols for authenticating the mobile terminal and the network in case of initial connection as well as in the case of handover.

- Chapter 7: Introduces two Service-Level AKA protocols, which are responsible for achieving mutual authentication and securing the session between the mobile terminal and the service provider in two scenarios; the initial authentication and in case of a handover.

- Chapter 8: Describes the Connection, Vertical handover and Ring-Based security models and shows how they could be defined by integrating the Targeted QoS models and the AKA framework.

- Chapter 9: Investigates possible procedures towards implementing the proposed AKA protocols and proposes enhancements on the addressing, location and naming systems in the network that could aid in implementing the Ring-Based Model.

- Chapter 10: concludes the thesis with a summary of the main contributions of this research study, along with a discussion on future work.

Chapter 2

Overview of QoS and Security Research in Heterogeneous Networks

2.1 Introduction

With the development of heterogeneous networks such as fourth Generation Networks (4G), providing security and QoS has become an essential component when accessing different networks. This provision has become a key issue for network professionals and researchers especially when dealing with vertical handover issues. This chapter acknowledges that having common mechanisms for security and QoS provision in such scenarios is a difficult task because of the various wireless systems that use different technologies. Therefore, current network mobile systems are critically analysed and evaluated in this chapter. A system will be proposed to bridge the security and QoS gaps between such heterogeneous wireless systems to create a scalable, manageable and adaptive solution for vertical hangovers. The work in this chapter is also presented in (MA12a).

2.2 Network Evolution to Support Heterogeneous Networks

There are a growing number of wireless systems being developed and deployed including 3G, WLAN, WiMax and Ultrawideband. Users will expect to be kept seamlessly connected to the Internet as they move around using whatever networks are available at that moment and whoever is the provider.

This widespread use of wireless technologies has highlighted a significant evolution in Internet Architecture. In terms of performance, it is now possible to divide the Internet into two

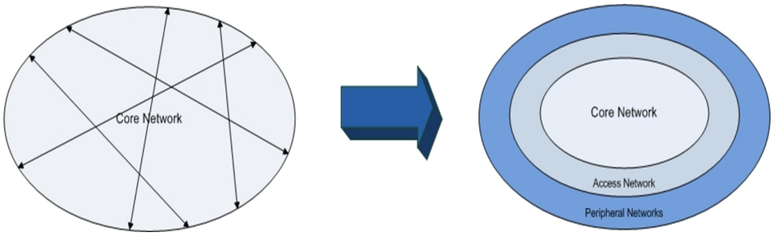

Figure 2.1: The Future Internet Structure

distinct parts: a core network and edge or peripheral networks. As shown in Fig 2.1, the core network consists of a super-fast backbone and fast access networks which are attached to the core. The backbone network is being made fast by the use of optical switches while the access networks are being upgraded using MPLS techniques. On the other hand, the peripheral network will be dominated by the deployment of wireless technology. This means that the characteristics of the core network will be very different to the peripheral wireless network on the edge.

This change needs to be reflected in new networking architecture to clearly define the functions, the order and the interlocking relationships that are necessary to support different functionalities such as handover, security and QoS in heterogeneous environment. By considering the above-described changes of the network structure, different research efforts such as the Daidalos II architecture (S.S07), the Mobile Ethernet framework (MK04) and the International Telecommunication Union (ITU-T) (IT04) have been working on defining a new architecture for heterogeneous networks. These working groups have agreed on the need for a central management entity to control and manage the resources of the different networks. This concept of a central management entity has been adopted by the Y-Comm group via introducing the concept of the Core-End Points (CEPs) as central administrative domains that control the operation of different network operators in the local area as shown in Fig 2.2. The main entities in the CEP, their structures as well as the interactions between them have been defined as part of this research in Chapter 4.

2.3 The Integrated Service (IntServ)and the Differentiated Service (DiffServ)QoS mechanisms in the Internet

These are two IETF architectures that have been proposed for providing QoS in the Internet.

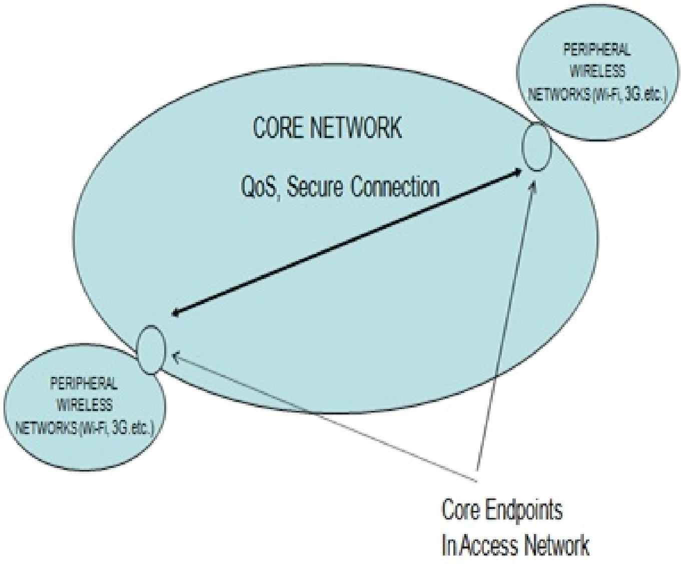

Figure 2.2: The Core End Point

2.3.1 The IntServ

The IntServ architecture provides an explicit, End-to-End resource reservation. It is based on two main features:

1. Reserved Resource: As each router needs to know how much of its own resources are allocated for each of the ongoing sessions.
2. Call Admission: before setting up a session with a guaranteed QoS, there is a need to reserve sufficient resources at each router between the source and the destination to ensure an end-to-end QoS.

   The call admission process requires each router in the session path to determine the required resources by the session consider the amount of resources allocated for other sessions and decide whether it has sufficient resources to accommodate the new session.

The IntServ architecture defines two service classes:

1. Guaranteed QoS: provides mathematically provable, firm bounds on the queuing delays that a packet will experience in a router (SS97).
2. Controlled-Load Network Service: a session in this class will receive a quality of service approximating the QoS that the same flow would receive from an unloaded network element (RB94). In this sense, the session will expect that a high percentage of its packets will pass through the router with a close to zero queuing delay in the router.

Analysis

The ability to request and reserve per-flow resources makes it possible for the IntServ to provide QoS guarantees to individual flows. However, this advantage comes with major difficulties associated the per-flow reservation as follows:

1. Scalability: The per-flow resource reservation requires each router to deal with resource reservations and to maintain per-flow state for each session. This incurs significant overhead in large networks.

2. Flexible service models: The IntServ framework provides a small number of pre-specified service classes. This set of service classes does not allow for more qualitative or relative service distinctions. Furthermore, new service classes may arise and old ones may become obsolete.

These considerations led to the so-called DiffServ activity within the IETF which will be explained in section.

2.3.2 The DiffServ

The DiffServ architecture (SB98) is flexible and more scalable as it aggregates flows into classes that receive treatment per class. This architecture is composed of a number of functional elements implemented in network nodes, including a small set of per-hop forwarding behaviours, packet classification functions, and traffic conditioning functions including metering, marking, shaping, and policing. This architecture achieves scalability by implementing complex classification and conditioning functions only at network boundary nodes, and by applying per-hop behaviours to aggregates traffic which have been appropriately marked using the Differentiated Services Field (DS) field in the IPv4 or IPv6 headers. Per-hop behaviours are defined to permit a reasonably granular means of allocating buffer and bandwidth resources at each node among competing traffic streams. Per- application flow or per-customer forwarding state need not be maintained within the core of the network.

Analysis

Unlike IntServ which requires advance setup such as call admission, DiffServ does not need such effort and thus less time is needed to setup the connection with the required QoS. Also, by removing the complexity of classification to the edge routers, this will reduce the load on the core routers in the Internet.

However, DiffServ only marks the packets with the hope of being treated differently. This does not necessarily ensure a QoS as it is up to the service provider to appreciate these marks and apply the policies to accommodate the packets as needed.

2.4 Handover Classification

As stated in (GMMAM09), handover could be classified as hard and soft handovers. Hard handovers occur when the current attachment is broken before the new connection is established while in soft handovers, the current connection is broken after the new connection is established.

Another potential classification is based on the entity that makes the decision to do a vertical handover. The options basically are network-controlled handover in which the decision to implement handover is taken by the network(s) to which the mobile node is currently attached. The second is called client-based handover in which the client is the deciding entity. Client-based handover is favoured as a more elegant solution (LP03). This is because client-based handover is more scalable as the mobile node can easily monitor the necessary parameters from its wireless interfaces.

A more advanced classification is shown in Fig 2.3, imperative handovers occur due to technological reasons only. Hence the mobile node changes its network attachment because it has determined by technical analysis that it is good to do so. This could be based on parameters such as signal strength, coverage, the quality-of-service offered by the new network. In contrast, alternative handovers occur due to reasons other than technical issues (SBC04). Hence there is no severe loss of performance or loss of connection if an alternative handover does not occur. The factors for performing an alternative handover include a preference for a given network based on price or incentives.

Imperative handovers are in turn divided into two types. The first is called reactive handover. This responds to changes in the low-level wireless interfaces as to the availability or non-availability of certain networks. Reactive handovers can be further divided into anticipated and unanticipated handovers. Anticipated handovers are soft handovers which describe the situation where there are alternative base-stations to which the mobile node may handover. With unanticipated handover, the mobile is heading out of range of the current attachment and there is no other base-station to which to handover. These handovers are therefore examples of hard handovers. The other type of imperative handover is called proactive handover. These handovers use soft handover techniques. Proactive handover policies attempt to know the condition of the various networks at a specific location before the mobile node reaches that location.

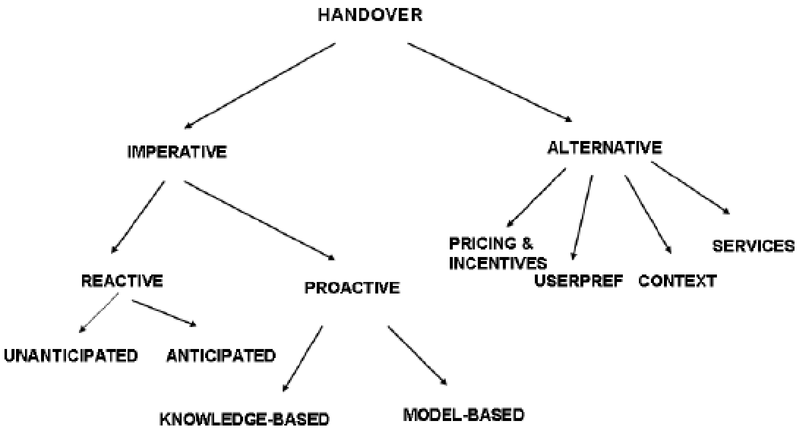

Figure 2.3: Vertical Handover Classification (GMMAM09)

2.5 IEEE 802.21 Standard

The IEEE 802.21 working group has developed standards to enable handover and interoperability between heterogeneous network types including both 802 and non-802 networks. As stated in (KT09), the purpose of IEEE 802.21 is to improve users' experience by providing Media Independent Handover (MIH) functionality that facilitates both mobile-initiated and network-initiated handover.

To optimise handover in heterogeneous environment, the IEEE 802.21 proposed an intelligent and generic interface that operated between the data link (L2) and Network layers (L3) of the protocol stack. This interface holds all the required functions to support MIHF and thus is referred to as Media Independent Handover Function (MIHF).

According to the IEEE 802.21 (IEE07), the MIHF should be available in the Mobile Terminal (MT) and the network entities. The MIHF encompasses three types of services:

- *MIH Event Services (MIES)* detect changes in link layer, report them to the upper layers. These events might be used as indicators for a potential handover.
- *Media Independent Command Service (MICS)* provides a set of commands that enables the upper layers (policy or mobility management layers) to control the status of the link such as on/off switching. Furthermore, some of the MICS commands enable the upper layer to poll the link layer about its status before making the handover decisions which is crucial to support proactive, mobile-initiated handover.
- *Media Independent Information Service (MIIS)* provides information such as topology, location and link layer parameters (data rate, throughput, delay) about different net-

works in the vicinity. This information, if provided, would aid the mobility management protocol making handover decisions.

Fig 2.4 shows the Media Independent Handover Framework and the role of the MIHF as an interface between the upper and lower layers.

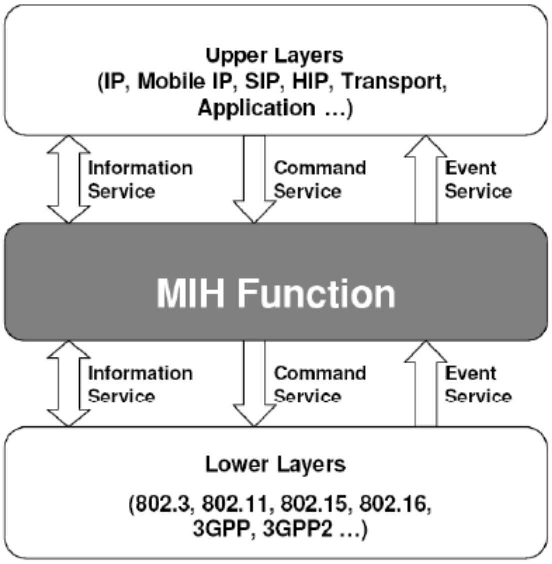

Figure 2.4: The MIH Framework

2.5.1 Analysis

The IEEE 802.21 standard provides functions and libraries to support vertical handover in heterogeneous networks. Also, its proposed vertical handover system could be considered as a reference model for future communication frameworks.

However, the IEEE 802.21 model initially came with no security features in mind and only recently, some security-related features were introduced i.e. solutions of the HOKEY WG. Recently, some other enhancements were proposed to add a QoS negotiation stage to the handover model. Furthermore, the IEEE 802.21 does not introduce a communication architecture to support the integration between QoS, security and mobility within a well-defined communication framework.

2.6 Ambient Networks

Ambient Networks (NN04) is an architecture designed to support heterogeneous networking. It is specially focused on providing seamless connectivity using a common control interface around different networks, thus converting them into Ambient Networks which are characterised by three interfaces: the Ambient Service Interface, the Ambient Network Interface and the Ambient Resource Interface. There are 4 layers in the Ambient Network design. The Connectivity Layer describes the links and infrastructure used to connect two Ambient Networks together. The Flow abstraction layer is used to define the connectivity provided by different networking technologies and to control and manage the connectivity layer. A flow is an abstract view of the connectivity provided by the underlying network technology. Flows are also defined by flow endpoints and may also pass through intermediaries called flow transits. The Bearer abstraction is a higher level abstraction which is not as location specific as flows. Bearer endpoints therefore use a unique naming space which allows mobility, address translation and media manipulation to be supported. Finally, the Application layer allows applications on Ambient Networks to use the architecture. The system therefore supports a much more flexible approach to internetworking in general which means that a global networking address can be replaced by high-level entity-names.

Ambient networks allow us to make better use of communication environment but does not address the integration of communication, mobility , QoS and security.

2.7 The HOKEY WG

The Internet Engineering Task Force (IETF) handover keying working group (HOKEY WG) (HOK07) is currently developing solutions to provide a secure, media-independent handover, also called inter-technology handover. The solutions are applicable to wireless access technologies based on the Extensible Authentication Protocol (EAP) (BA04) which is an authentication framework that supports multiple authentication protocols, these are referred to as EAP methods. Regardless of the method, the EAP key hierarchy derives two keys: the Master Session Key (MSK) and the Extended MSK (EMSK) which are used by different methods to derive further keys.

Based on EAP's terminology, three entities are defined: The EAP peer which is the client asking for authentication using an EAP method, the EAP Server which is an entity that terminates the EAP authentication method with the peer; the EAP servers are often, but not necessarily, co-located with AAA servers. And finally, the EAP authenticator which is the network Access Point that supports the authentication functionality and enforces access control based on the authentication result.

When a mobile terminal (MT) moves between different authenticators, it is desirable to avoid a full EAP authentication to support fast handover. Therefore, the HOKEY group proposed a new method for the EAP known as EAP Re-Authentication Protocol (ERP) (VN08) which will be discussed in the following sections.

2.7.1 ERP Key Hierarchy

To provide security without disturbing the handover procedure, the ERP achieves low latency handover by launching the keying materials in the target network before the actual handover takes place. Furthermore, the ERP introduces additional keys shown in Fig 2.5 , these are defined in (VN08) as follows:

- The rRK - re-authentication Root Key, derived from the EMSK.
- The rIK - re-authentication Integrity Key, derived from the rRK.
- The rMSK - re-authentication MSK. This is a per-authenticator key, derived from the rRK and is delivered to the authenticator.

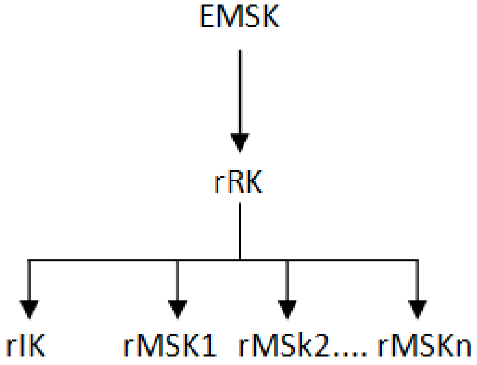

Figure 2.5: The ERP Key Hierarchy

2.7.2 An Overview of the ERP method

The ERP is a new extension to EAP to support an EAP method-independent protocol for efficient re-authentication between the peer and an EAP re-authentication (ER) server (VN08). It is assumed that, the ER server is collocated with an Authentication, Authorization, Accounting and Cost (A3C) server (GG09).

Initially, the MT performs a full normal EAP authentication with the A3C server in its home network. As a result of this authentication, the EAP's keys namely, MSK and EMSK are derived. However, when the MT roams, the ERP extension is used to achieve authentication between the MT and the ERP-Server in the target network instead of performing a full EAP authentication. This process is referred to as pre-authentication because the keying materials will be launched in the target network before the MT actually joins and it comprises:

The Pre-Authentication Process For the MT to use the ERP protocol with the access point in the target network, it needs to derive a new re-authentication root key, this key is derived using the EMSK and the domain name of the target network and hence, is called the Domain Specific Root Key (DSRK). Using this key, further domain specific keys such as the DsIK and DSrMSKs are derived, these will be used to secure the connection between the MT and the network. Additionally, proving the possession of derived keys helps in achieving authentication between the MT and the network.

2.7.3 Analysis of ERP

The HOKEY's work seemed fairly stable particularly in terms of keys hierarchy and it has influenced the direction of research when developing a Network-Level Authentication and Key Agreement (NL-AKA) protocol. However, the solutions for keys distribution are still being discussed by the HOKEY. Additionally, the ERP extension suffers from some drawbacks which are summarised as follows:

1. Although the ERP is based on the EAP platform, it introduces new messages such as EAP-Finish/Re-auth that includes a DSRK and the new domain name. This implies that, all the network entities such as the Access points have to be updated or replaced to support this extra message.

2. The ERP presumes that the MT will get the domain name of the target network either implicitly when receiving the announcement or explicitly by soliciting for it. This step should be part of the handover procedure rather than a part of the security mechanism and thus removes some of the complexity from the security mechanisms. Additionally, it is not clear how the MT would communicate with the ER server in the target network and therefore, this raises the question whether the EAP-Finish/Re-auth message include the address of the server?.

3. The EAP-Finish/Re-auth message is sent directly between the MT and the Authenticator in the new network. This message includes the domain name of the target network and is sent in an unprotected manner since there is no security agreement yet between the MT and the target network.

4. In the ERP protocol, the MT's home ERP server generates the keys and passes them to the ERP server in the target network . However, in case of heterogeneous environments,

this might not be feasible since these ERP severs might belong to different operators. There is a need for a communication framework which provides a wider context to implement these protocols.

5. The ERP protocol operates at the network level; it is only concerned with achieving secure handover at the lower layers without considering the upper (transport and application) layers.

6. Although the security consideration section of the proposal (VN08) provides some analysis of the protocol features, the protocol lacks formal analysis such as using a formal methods approach.

2.8 The Mobile Ethernet Architecture

Mobile Ethernet Architecture is a Beyond 3G network system for the all IP integrated network using MAC layer technologies (MK04), (Kur05). The architecture is based on the Wide Area Ethernet (WAE) which is a virtual private network aimed at providing connectivity based on the Ethernet (MAC) addressing and thus achieves interoperability among different IP-based operators as shown in Fig 2.6.

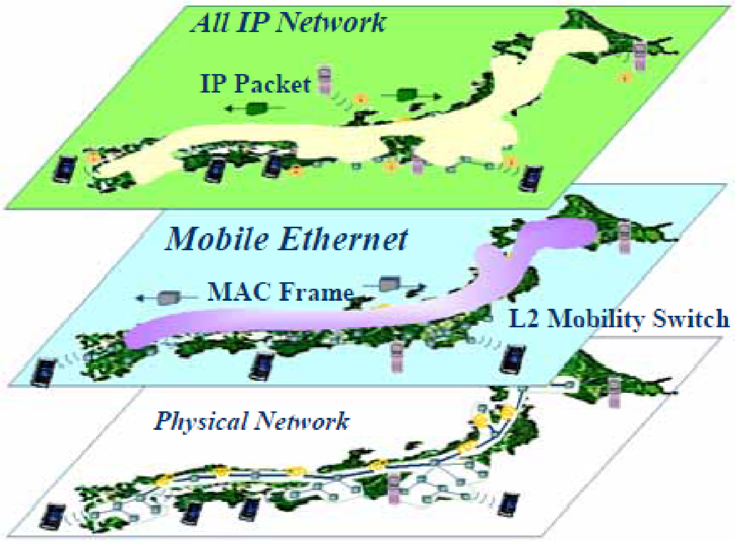

Figure 2.6: The Mobile Ethernet Layer (Kur05)

As shown in Fig 2.7, in order to achieve scalability and interoperability among different operators, the Mobile Ethernet proposes a network partitioning scheme. In this scheme the network will comprise a fast ring core network which aggregates a number of segments, each segment is attached to a one or more of peripheral networks via a number of edge switches.

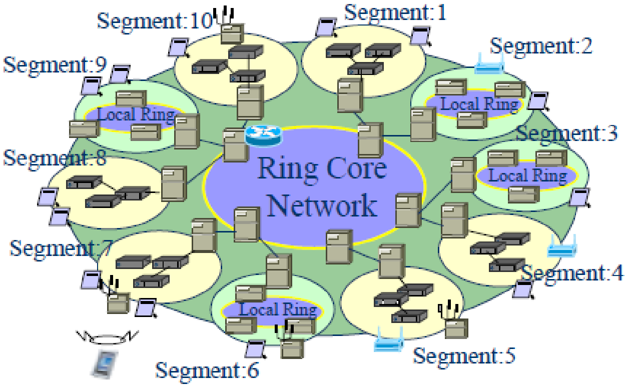

Figure 2.7: The Mobile Ethernet Network Structure (Kur05)

Handover in Mobile Ethernet Two types of handover are introduced in Mobile Ethernet's work:

- *Intra-Segment handover*: Mobility is managed in a distributed manner because each switch in the segment tracks the location of the terminal as long as it is still within the same segment. More details about Inter-Segment Mobility Management including the procedures for locations' regeneration and update are found in (Kur05).

- *Inter-Segment handover*: An Inter-Segment Mobility management provides the required functionalities to manage the handover among different segments in the core network, this includes setting up the path between segments and sending location update information to track the terminal's movement. Further details are found in (Kur05).

Security in Mobile Ethernet In (MK04), (I.D06), a secure service framework for Mobile Ethernet was proposed. The framework comprises the following elements: the mobile terminal (MT), a Contact-less smart card to hold user credentials, and a self-delegation unit between the smart card and the terminal.
As explained in (MK04), (I.D06), the framework aims to provide security at different levels, namely: Network, Service and personal levels. To achieve this, a self-delegation protocol was introduced. This protocol achieves authentication between the mobile terminal and the smart card and then delegates the terminal to perform authentication with the service provider and the network operator.

2.8.1 Analysis

This section discusses the security-related work of the Mobile Ethernet. Unlike the ERP protocol of the HOKEY group, this framework proposes addressing security at different levels through a self-delegation protocol. This concept of multi-level security can be utilised to implement security in Y-Comm. However, as stated in (I.D06), deploying the self-delegation protocol requires designated hardware and software which makes it a complicated process. Moreover, in the case of handover, the framework does not consider if this will affect the security at the service level and whether the service provider needs to know about the handover and thus the terminal might need to re-authenticate itself again. Also, the framework does not consider the QoS while providing security. Most importantly, the Mobile Ethernet's proposed protocols did not have clear goals and lacked formal verification to reveal any vulnerabilities.

Additionally, the Mobile Ethernet work deals only with security without considering other issues like QoS and vertical handover.

2.9 The Y-Comm Framework

The research aims to provide security for 4G systems and has adopted Y-Comm as a representative of these systems. Therefore, it is very necessary to fully understand the Y-Comm structure and its proposed approaches for handover and connectivity as well as its security module.

2.9.1 An Overview of Y-Comm

As proposed in (GM07), (GM06), the Y-Comm framework is a communication architecture to support vertical handover. Y-Comm uses two frameworks: the first is the Peripheral framework which deals with operation on the mobile terminal. The second is the Core framework and deals with functions in the core network to support different peripheral networks. These frameworks are brought together to represent a future telecommunications environment which supports heterogeneous devices, disparate networking technologies, network operators and service providers. The two frameworks, shown in Fig 2.8, share a common base subsystem consisting of the hardware platform and network abstraction layers. Both frameworks diverge in terms of functionality, but the corresponding layers interact to provide support for heterogeneous environments.

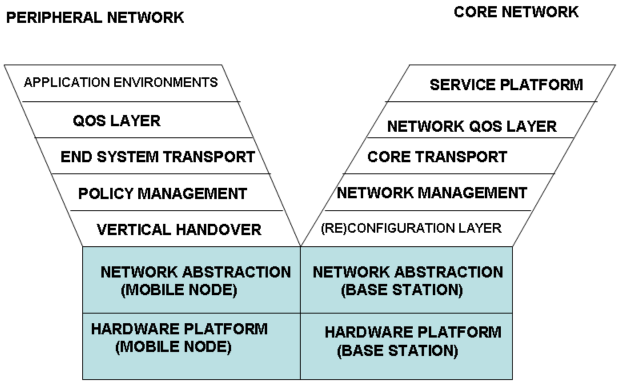

Figure 2.8: The Y-Comm Framework

The Peripheral Framework

The Peripheral Framework is concerned with activities on mobile nodes and in the wireless networks to which they are connected. The peripheral framework has seven layers:

- *The Hardware Platform Layer (HPL)*: It is used to classify all relevant wireless technologies. Hence different wireless technologies which are characterised by the electromagnetic spectrum, MAC and modulation techniques make up this layer.

- *The Network Abstraction Layer (NAL)*: It provides a common interface to manage and control different wireless technologies. The first two layers for both frameworks are similar in functionality. In the Peripheral Framework, they run on the mobile terminal to support the various wireless network technologies while in the Core Framework these two layers are used to control the functions of base stations of different networks.

- *The Vertical Handover Layer (VHL)*: This layer executes vertical handover. Therefore, this layer acquires the resources for handover, does the initial handover signalling, context transfer and packet reception after vertical handover.

- *The Policy Management Layer (PML)*: The PML decides whether, when and why handover should occur. This is done by looking at various parameters related to handover such as signal strength and using policy rules to decide both the time and place for doing the handover.

- *The End Transport Layer(ETL)*: It allows the mobile node to make end-to-end connections across the core network. This layer provides the functionalities of the Network and Transport layers of the TCP/IP module.

- *The QoS Layer (QL)*: In the Peripheral Framework, it supports two mechanisms for handling QoS. The first is defined as Downward QoS. This is when an application specifies its required quality-of-service to the system and the system attempts to maintain this QoS over varying network channels. The other definition is Upward QoS, where the application itself tries to adapt to the changing QoS. This layer also monitors the QoS used by the wireless network as a whole to ensure stable operation.

- *The Applications Environments Layer (AEL)*: It specifies a set of objects, functions and routines to build applications which make use of the framework.

The Core Framework

This framework deals with functions in the core network. The first two layers of the Core Framework are shared with the Peripheral framework. The remainders layers are:

1. *The Reconfiguration Layer (REL)*: It is responsible for managing key infrastructure such as routers, switches, and other mobile network infrastructure using programmable networking techniques.

2. *The Network Management Layer (NML)*: The NML is a management plane that is used to control networking operations in the core. This layer divides the core into a number of networks which are managed into an integrated fashion. It also gathers information on peripheral networks such that it can inform the policy management layer on mobile nodes about wireless networks at their various locations.

3. *The Core Transport System (CTS)*: It is concerned with moving data through the core network.

4. *The Network QoS Layer (NQL)*: It is concerned with QoS issues within the core network especially the interface between the core network and the peripheral networks.

5. *The Service Platform Layer (SPL)*: This layer allows services to be installed on various networks at the same time.

2.9.2 Vertical Handover in Y-Comm

Y-Comm supports a number of different handover types (GMMAM09), the most complicated of which is called proactive handover in which a mobile node decides on when and where to handover. The parameter called the Time Before Vertical Handover denoted by TBVH, is calculated (FS07). TBVH allows the higher layers of Y-Comm to take evasive action to minimize the effect of performance degradation due to handover. The interaction is shown in Fig 2.9

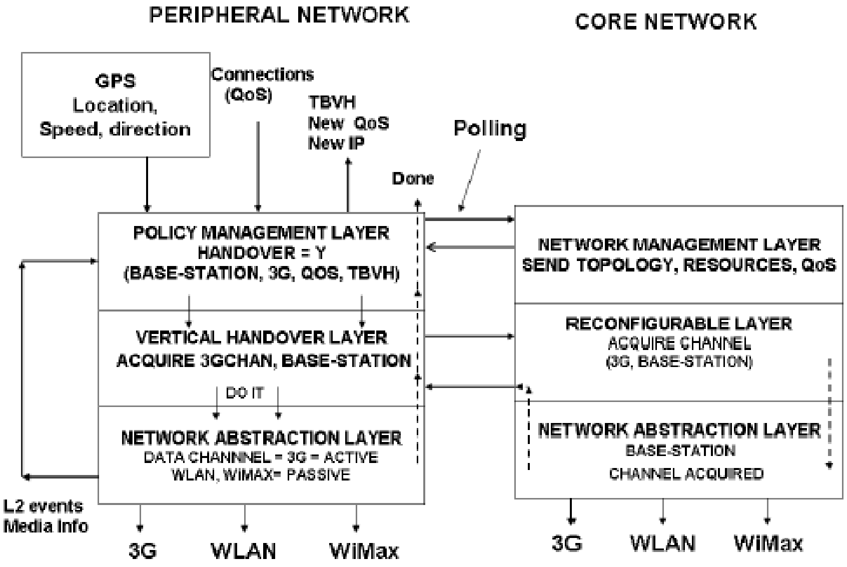

Figure 2.9: The Proactive Handover

Since proactive handover attempts to determine when and where handover should occur, it is necessary to have a knowledge of networks in the local area where the mobile is located. The mobile node therefore polls the NML to obtain information with regard to all local wireless networks, their topologies and QoS characteristics. This information along with the direction and speed of the mobile as well as the QoS of on-going connections are used by the Policy Management Layer to determine where and when handover should occur. The PML calculates TBVH - the period after which handover will occur. This information is communicated to the Vertical Handover Layer which immediately requests resources to do a handover. Even though the resources are acquired early, handover actually takes place when TBVH expires. The message sequence is given in Fig 2.10

In addition, once the PML decides to handover, the new IP address, the new QoS as well as TBVH, are communicated to the upper layers. Given TBVH, the upper layers are expected to take the necessary steps to avoid any packet loss, latency or slow adaptation. For example,

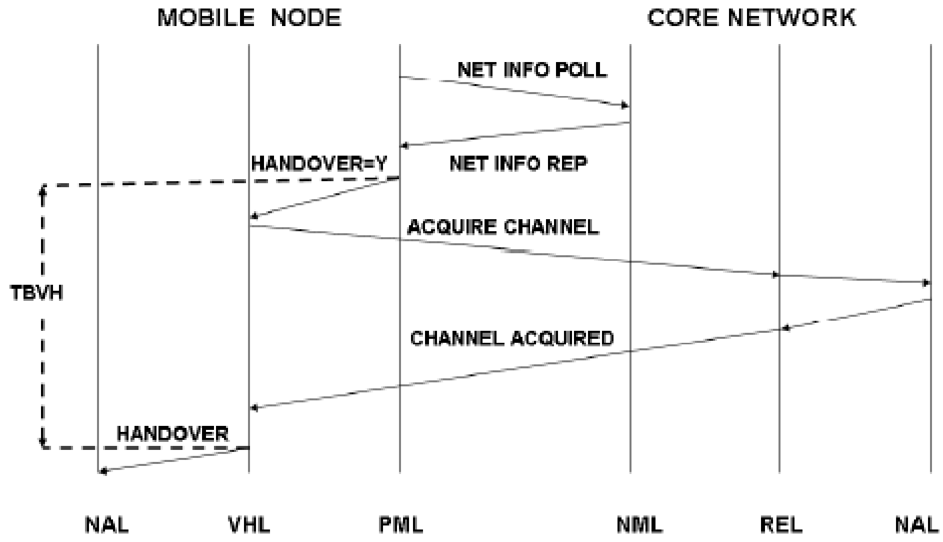

Figure 2.10: The Proactive Handover Transaction

it may be possible for the End-Transport Layer to signal an impending change in the QoS on current transport connections and to begin to buffer packets ahead of the handover. After handover, the previous channel used by the mobile node is released.

2.9.3 Y-Comm Security Framework

As shown in Fig 2.11, Y-Comm deploys a multi-layer security model which must be applied to both the Peripheral and Core Framework simultaneously to provide security in different situations. The security layers must work together across both frameworks in order to be fully integrated with the new architecture. The important point to note is that the need to support heterogeneous networking with open architectures means that security should not only protect data using cryptographic techniques but network entities as well by monitoring the utilization of these entities and making sure that they are not abused or overloaded by even legitimate users. The highest layer of security is at layer seven and is called Service and Application Security or SAS. In the Peripheral Framework, SAS defines the AAAC functions at the end-device and is used to authenticate users and applications. SAS in the Core network provides AAAC functions for services on the Service Platform in the core network.
The next security layer is called QoS-Based Security or QBS and is concerned with QoS issues and the changing QoS demands of the mobile environment as users move around. In addition, in order to meet their service-level agreements, servers may choose to replicate services closer to the current position of the mobile. So it is necessary to ensure that core

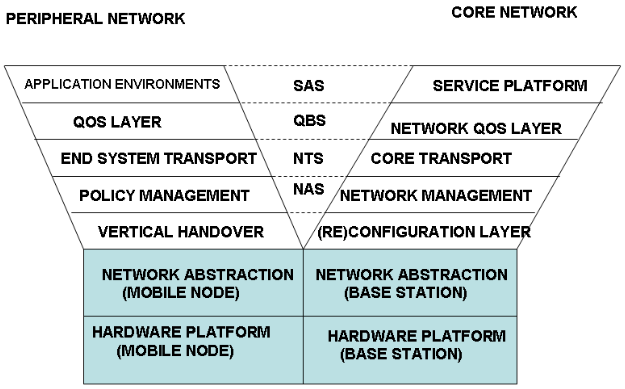

Figure 2.11: The Y-Comm Complete Structure

endpoints and peripheral networks are not overloaded. The QBS layer also attempts to block QoS related attacks, such as Denial-of-Service (DoS) attacks on networks and servers.

The next security layer is at layer five, called Network Transport Security or NTS. In the Peripheral network, NTS is concerned with access to and from end-devices and the visibility of these devices and services on the Internet. In the core network, NTS is used to set up secure connections through the core network. So NTS in the Core Framework involves setting up secure tunnels between core endpoints using mechanisms such as IPsec to ensure that moving data across the core network is done in a secure manner.

Finally, the fourth and last level of security is defined at layer four but can also encompass layers three and two. It is called Network Architecture Security or NAS. In the Peripheral Framework, it attempts to address security issues involved in using particular networking technologies and the security threats that occur from using such a technology. So when a mobile device wishes to use any given network, NAS is invoked to ensure that the user is authorized to do so. NAS also ensures in the core network, NAS is used to secure access to the programmable infrastructure. NAS in this context determines which switchlets, routelets or base-station resources may be used by the network management system.

2.9.4 Analysis

The Y-Comm provides a platform to integrate different mechanisms for QoS and Security as well to support vertical handover. In addition, because Y-Comm has proposed a detailed model of proactive handover including the concept of TBVH, it facilitates an easier integration of the QoS and security mechanisms with the communication framework. Furthermore, since the security framework is integrated with the Core and Peripheral frameworks within Y-Comm, its security functions will be part of the communications architecture and thus, could be used to much greater effect than previous methods.

However, there is a need to define the layers' underlying security protocols and some access control mechanisms. Additionally, the interface between these protocols has to be specified in order to provide an integrated security approach which addresses threats at different levels. Furthermore, because Y-Comm is an open architecture, it is also necessary to protect entities such as users, servers and network infrastructure as well as the data. This protection has to be provided without restricting the openness of the 4G systems. Therefore, we need define Targeted Security Models (TSMs) which are based around protecting a specific entity from being abused or attacked by other entities such as users in an open architecture. This is a new concept which the author believes is necessary to provide a secure environment. Therefore, defining and formally verifying these models comprise a major part of this thesis.

2.10 A Summary and Overall Evaluation

Addressing security in Next Generation Networks (NGNs) have attracted many research groups. Each attempted to tackle the problem from a different perspective and hence, many issues have been addressed while many more are waiting to be resolved.

Our research to provide QoS and security for the Y-Comm framework benefited from the work of these groups. For example, the vertical handover models of the IEEE 802.21 and the Mobile Ethernet helped in defining models for reactive and proactive vertical handover in Y-Comm (GMMAM09). Also, we benefited from the pre-authentication concept and the key hierarchy of the ERP protocol to define our AKA protocol for a secure vertical handover. The concept of a multi-level security framework of the Mobile Ethernet can co-exist with the concept of the Integrated Security Module of Y-Comm. Additionally, our research benefited from the self-delegation protocol to provide security for the SAS layer of the Y-Comm security module.

However, our research differs from the previous works in many ways: Firstly, unlike all the previous protocols, our research considered an integration between the security and QoS through the functionality of the QBS layer of the Y-Comm security module. Secondly, all the proposed mechanisms and protocols have been verified using formal methods approach based on Casper/FDR tool (LBDH09), (PR10), (Sys93). Thirdly, in contrast to all the aforementioned mechanisms which considered protecting the data being sent and received, the approach taken aims to protect the data, as well as the network's entities whether they were servers or clients and thus, mitigates major threats such as SPAM and DOS attacks. Most significantly, our security approach considers the dynamics and open nature of future networks and thus aims at providing security without affecting these features. In order to achieve this, there is a need to deal with security and QoS in a dynamic way, rather than a static one, in order to target security and QoS in different scenarios. Therefore, our approach proposes Targeted Security Models (TSMs) as well as Targeted QoS Models. More details about these models will be given in the following chapter.

Chapter 3

Approach and Constraints

3.1 Introduction

This chapter analyses the proposed approaches as well as the constraints needed to address the research questions. The first section of this chapter recalls the research questions, while the second section presents the environment and the methodology which will be followed to answer the research question. The third section discusses our research to define network architecture and propose a QoS framework. The fourth section discusses the security-related part of the research, it will describe the proposed AKA protocols and validate our choice of the verification method. Lastly, a chapter overview is given in the summary section

3.2 Proposed Work

We propose to investigate the QoS and security issues in heterogeneous systems and to develop novel approaches to be applied in different scenarios without restricting the flexibility of the systems. The solution adopts Y-Comm as a potential communication framework for heterogeneous networks to align the proposed security and QoS mechanisms in this research.

3.2.1 The Concept of Targeted Models

Due to the dynamic and open nature of future networks, QoS and security have to be provided very carefully so as not to limit network flexibility. Additionally, 4G is an open architecture where the end user might be either a mobile or stationary client starting a connection or a server receiving access requests. It is imperative that a dynamic approach to address the QoS and security are considered in these situations. In addition, it is important to observe that in this environment it is not just about delivering and protecting data, but also the entities in the system, i.e. users, servers and network infrastructure, need to be protected from each other for example SPAM, DDOS and Impersonation attacks. Therefore, new methods need to be developed to target QoS and security in specific scenarios.

In multi-technological and multi-operator environments such as 4G networks, there is a need for defining signalling mechanisms that guarantee the provision of an End-to-End QoS. Therefore, this research proposes a QoS signalling framework. The proposed framework requires certain level of cooperation among network elements. Therefore, it proposes some functional modules/ interfaces to be run on different network entities in order to signal QoS in different scenarios, thus three Targeted QoS-Signalling Models: the initial registration, the connection and the vertical handover models have been introduced.

In future heterogeneous environments, security has to be applied at different levels in an integrated manner. Therefore, there is a need for a multi-pronged security architecture where security protocols address the threats of each level. Additionally, these protocols have to operate very closely together as well as with the communication procedure to provide full integration of security and communication mechanisms.

In order to meet such requirements, the proposed approach in this research benefited from the Y-Comm's integrated security module to introduce the concept of the Targeted Security Models (TSMs) which addressed security in different scenarios, thus it yielded three security models: the connection security model which controls the connection between users and thus prevents a user from arbitrarily sending an unsolicited message to another user. The Ring-Based model helps in protecting the servers from unwanted traffic, e.g., Spam by giving the server the ability to decide on the nodes they can communicate with. The Vertical handover security model facilitates secure vertical handover and attempts to prevent network resources from being abused and overloaded.

3.3 Research Methodology

This section describes our approach to define the three security models, the approach started by investigating the security threats in 4G systems.

3.3.1 The Threat Model

As explained in (MA10), the key security threats in heterogeneous networks include access control, communication security, data confidentiality,availability and privacy. These threats are not seen in 3G networks because the network infrastructure is wholly owned by the network operators and access is denied to other network entities. However, such assumptions are no longer valid in heterogeneous systems and therefore must be addressed in any proposed security architecture. Moreover, since 4G is an IP-Based environment, it will suffer from most of the IP-specific security vulnerabilities found in the Internet. Our experience of the Internet as the best example of a successful open architecture has taught us that it is not sufficient to only protect data but it is also necessary to protect entities from each other (DoS, Spam) and also to protect the network infrastructure.

3.3.2 Solution Approach

The next step of the research was to introduce abstract security models to address security threats in 4G systems. However, because our approach proposed integration between the security and QoS, the next step of the approach was to define QoS architecture in heterogeneous environments.

After presenting the environment and defining the operational entities in the network structure, a set of Authentication and Key Agreement (AKA) protocols along with access control mechanisms were proposed and verified using formal methods approach. These protocols were integrated in AKA frameworks which will cooperate with the QoS framework to define the targeted security models

3.4 QoS-Related Work

4G networks will support global roaming across multiple wireless and mobile networks for example, from a cellular network to a satellite-based network to a high-bandwidth wireless LAN. This diversity of services and access technologies will be a universal characteristic in future communications and poses major challenges such as secure cross-domains mobility support, network resource management and QoS provision.

Initially, the client subscribes to a service with a range of desired QoS and security levels, these are defined as the Service Level of Agreement (SLA). Maintaining the client's SLA is quite feasible as the client roams within single-operator environments. However, in a multi-operator environment such as the 4G, the client might move outside its home network to a new network with entirely different QoS and security measures. In such a case, the client might not get the original SLA and a trade-off between the security and QoS has to be achieved. This concept of compromising between the required QoS and recommended security is referred to as the Quality of Security Service (QoSS) (TEL00).

As a representative of 4G systems, Y-Comm, by deploying the concept of QoSS, proposes integration between the QoS and the security through the function of the QoS Based Security

(QBS) layer. Additionally, since the security is also integrated with the communication framework, we need to define the interface that enables this integration.

The author believes that network architectures will play a key role in implementing the features required to address these issues. Therefore, part of our research considered defining the network structure and proposing models for QoS signalling in different scenarios such as the connection initiations and handover. These models will integrate with the proposed AKA frameworks to define the security models.

3.5 Security-Related Work

This section describes our security-related research in some detail. We started by describing the targeted security models then explain the approach followed to define these models.

3.5.1 The Targeted Security Models

The Targeted Security Models (TSM) are security models based around protecting a specific entity from being abused or attacked by other entities such as users in an open architecture. We have identified three security models that need to be developed, the first is called the connection security model which controls the connection between users and thus prevents a user from arbitrarily sending an unsolicited message to another user.

The second security model is concerned with restricting access to servers by introducing the concept of a scope. This is an enhancement of the 'Off By Default' (HB05) proposal. Therefore, users can only access the server when they are in the same scope as the server.

The final security model is for facilitating secure vertical handover and attempts to prevent network resources from being abused and overloaded. This can be achieved by monitoring resource requests and ensuring access to vulnerable components that does not exceed the available QoS.

The next step is to define the mechanisms and the protocols that will be used by the aforementioned models. This novel approach is explained in the following subsection.

3.5.2 A multi-layer AKA Protocols

The Y-Comm's security module comprises four layers, each of which addresses the security at a specific level. When defining the underlying security protocols of these layers three types of AKA protocols were proposed:

- *The User-Level AKA (UL-AKA) Protocol*: Operates between the Mobile device, the SIM card and the user. After running this protocol, the mobile terminal will represent the user in the remainder stages. This protocol resides at the Service and Application Security (SAS) layer of the Y-Comm's security module.

- *The Network-Level AKA (NL-AKA) Protocols*: Achieves mutual authentication between the mobile terminal and the access network and thus, addresses some functions of the Network Architecture Security (NAS) layer. For network level security, two protocols have been defined: the AKA protocol for the initial authentication process and the Pre-AKA protocol for the authentication in case of handover.

- *The Service-Level AKA (SL-AKA) Protocol*: Provides authentication between the mobile terminal, the service provider (SP) and it is related mainly to the Network Transport Security (NTS) layer.

The next step was to verify the proposed protocols and check their vulnerability to security threats.

3.5.3 Verifying Security Protocols

Protocol verification and validation can be achieved using different approaches. Discrete event simulators such as NS2 (TI09) and OPNET (Abo07) provide good capabilities to analyse the protocol performance. Other approaches based on the Unified Modelling Language (UML) or the Specification and Description Language (SDL) provide validation methods to check the protocol against its specification in order to prevent undesired states and behaviour. This includes preventing deadlocks and livelocks. However, the verification of security protocols against their claimed properties requires special toolsets such as a mathematical logic or model checks.

Generally speaking, verifying security protocols is based on theorem proofs and verification logic such as the BAN logic (MB90), (SB04) which determines the trust relationship among the protocols' parties. However, the BAN logic considered the authentication properties only so it could have not be used in confidentiality analysis. Also, the BAN logic assumed all parties to be honest and trustworthy, thus, this assumption has to be considered when interpreting the BAN results.

Another approach to simulate the security systems is by using formal specifications such as the "Z" notation (Spi92) or the Communicating Sequential Process (CSP) (Hoa85),(PR10) which uses mathematical notations to describe the properties which an information system must have. Verifying these properties is achieved using model checkers such as Failures-Divergence Refinement (FDR) (Sys93).

FDR is able to check whether a security model satisfies certain security properties by proving that an implementation is a refinement of a specification. FDR does trace refinements, failure refinements, and failure divergence refinements. However, for security analysis only the trace refinement is used to check whether a security model does satisfy security properties such as secrecy and authentication.

However, describing a system or a protocol using formal specifications is quite a difficult and error-prone task. Therefore, Gavin Lowe (LBDH09) has developed CASPER/FDR tool to model security protocols which accepts a simple and human-friendly input file that describes the system and compiles it into CSP code which is then checked using the FDR model checker. CASPER's input file consists of eight headers as explained in Table 3.1

The Header	**Meaning**
# Free Variables	Defines all the variables, functions and agents
# Processes	Where each agent in the system is represented by a process
# Protocol Description	Describes system messages
# Specification	Specifies the system's security features to be tested
# Actual Variables	Defines the real variables, in the actual system to be checked
# Functions	Has all the functions used by the agents
# System	Lists all the agents participating in the actual system with their parameters instantiated
# Intruder Information	Identifies the intruder and its initial knowledge

Table 3.1: Casper input file headers

3.6 Summary

Due to the unique nature of 4G systems and the aim to support full integration within the Y-Comm architecture, the research work attempted to define the Targeted Models in the context of the Y-Comm required research to be carried out in both security and QoS sides. Furthermore, the research needed to propose the underlying security protocols of the Y-Comm's security module, verify them using formal methods approach and then integrate them via security frameworks. It also needed to propose potential QoS-signalling models along with the network architecture which was used to define the interface that enabled the integration between security, QoS and communication framework. This issue will be investigated in the following chapter.

Chapter 4

A QoS Framework for 4G Systems

4.1 Introduction

In heterogeneous environments, mobile nodes move between different types of networks, some of which will not meet the nodes' requirements in terms of QoS or security. In such cases, the nodes need to choose the best network that meets their desired requirement, this implies that the nodes might need to trade-off between the desired requirements. For instance, nodes might choose a highly secure connection which reduces the QoS or vice-versa.

This situation becomes more serious when considering multi-homed mobile devices; that is due to the fact that, current network addresses (IPv4 and IPv6 (Alm92), (SD98)) identify the network interface cards rather than the devices holding them. Therefore, multi-homed devices could start multiple connections without any indication that these belong to the same device. Such scenarios will impact QoS and security as it will aid denial of service (DoS) attacks since a single multi-homed device can participate in multiple connections at the same time over different networks. This situation highlights the fact that in future networks, QoS and security are inter-connected and thus, should not be separately considered by any future communication framework.

Also from a security perspective, in order to provide a practical security solution for 4G systems, any proposed security mechanism has to consider the network structure and its operational entities. This will define the interface that enables the integration between the security mechanisms and the communication framework. Additionally, there is a need to deal with effects of the multi-homing issue in an integrated manner, this could be achieved by enhancing network services such as naming and locating services as will be described in Chapter 9.

Since Y-Comm provides a framework to integrate security, the QoS and communication mechanism, this chapter describes part of the conducted research that attempts to define

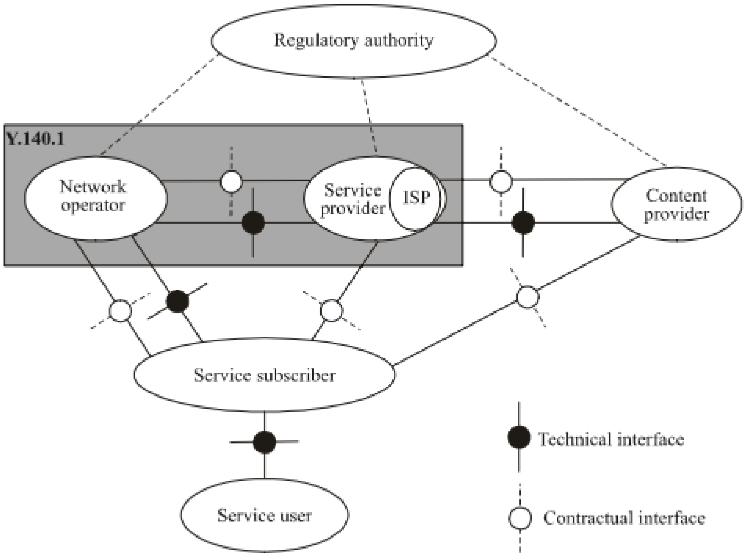

Figure 4.1: The ITU Recommended Network Parties

the actual network architecture along with QoS models to provide QoS signalling in different scenarios such as the connection initiation and handover. The first section describes a proposed network architecture and explains its major entities in terms of structure and function. The second section discusses potential QoS signalling models for different scenarios.

4.2 Network Architecture Overview

In 4G networks, multiple network operators have to cooperate in order to provide the user with ubiquitous connectivity. Since each network operator uses a different network architecture, interoperability might be one of the most challenging problems facing the deployment of 4G technology. One proposed solution for this problem is having a central management entity to control the resource of the all different technologies and coordinate the multiple operators. As explained in (MA12a), the concept of a central management entity was recommended by different architectures for NGNs such as Daidalos and Mobile Ethernet (Agu06), (Kur05). The ITU-T recommendation for NGN as in (IT04), (ITU06) proposes the concept of Regulatory Authority, shown in Fig 4.1 which controls different network operators and service providers. The Regulatory Authorities are regulatory bodies with the power to influence policies in telecommunication services, they are responsible for creating national policies to encourage the development of telecommunications, also they provide essential powers to regulate license agreements, interconnection arrangements, and monitoring unlawful telecommunication activities.

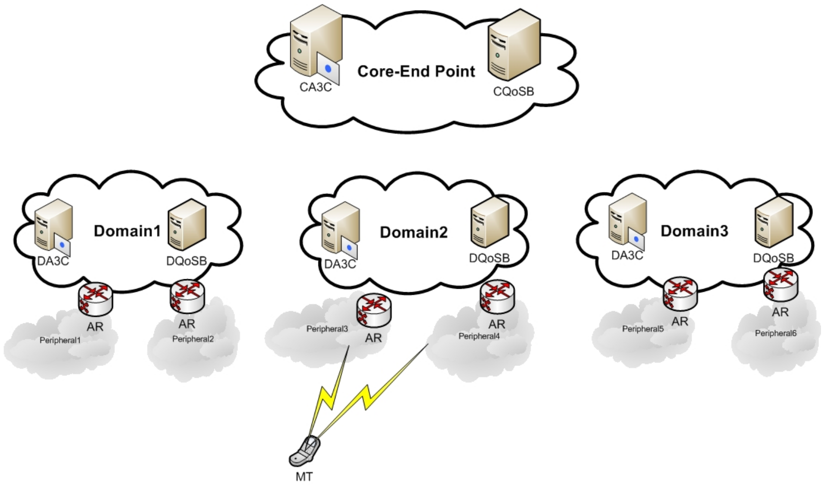

Figure 4.2: The Hierarchical Network Structure

Chapter 2 presented the Y-Comm's view of the Internet structure. This section presents a more detailed view of the network along with its components. As shown in Fig 4.2, we proposed a hierarchical structure of the network composed of three levels. The top level is the Core End-Point (CEP) which acts as a gateway to the Internet and is responsible for managing multiple, mid-level domains. Each domain is technology-specific and is controlled by a single operator. For instance the CEP might be connected to two domains, each is controlled by different technology operator such as WiMAX and GSM. The bottom level is the peripheral wireless networks, controlled by Access Routers or Base-stations through which the mobile terminal has access to the wider network.

As shown in Fig 4.2, for scalable support of Security, QoS and handover in heterogeneous networks, different operating entities exist in the network such as Domain QoS Broker (DQoSB), Core QoS Broker (CQoSB) and A3C servers (CA3C). These entities collaborate and function on both network and service management to provide QoS and security-related tasks.

- *Core A3C (CA3C)*: The top level A3C server resides in the Core End-Point and is responsible for service level management. It holds users' Service Level Agreements (SLAs) that contain the subscribed services along with the associated QoS and networks or the Operators, the user can access with the corresponding QoS. This information is passed to the CQoSB for network level management.

- *Core QoS Broker (CQoSB)*: plays a major role in managing inter-CEPs functions as well as negotiating QoS parameters with other CQoSBs in the case of cross Core End-Points connection. The CQoSB initially extracts users' Level of Agreement from the CA3C.

- *Domain A3C (DA3C)*: The DA3C is responsible for handling users' service requirements. Initially, it extracts users' profile information from the CA3C and uses this information for authorizing the users' requests to access services.

- *Domain QoS Broker (DQoSB)*: gets user's profile information from the CQoSB and manages the resources of the attached peripheral networks with respect to the user preferences and network availability, it also makes a per-flow admission control decision. In order to support handover, DQoSB uses a Network Intelligent Interface Selection (NIIS) module (IEE07), (S.S07) for load balancing and handover initiation between peripheral networks.

- *Access Router (AR)*: This is the link between the domain and peripheral networks; it enforces the DQoSB's admission control decision. The AR resides between the Mobile Terminal and the A3C server in the domain. Therefore, using security terminology, the AR acts as an Authenticator (Auth) with the DA3C server.

- *Mobile Terminal (MT)*: The MT is the user's device, used to access the network and to request a service. To comply with the heterogeneity of 4G systems, the MT should be able to get the subscribed service using the best available access network. Therefore, for the integration of Handover and QoS, the MT contains a mobility decision module called Intelligent Interface Selection (IIS) (S.S07) and a QoS module called QoS Client (QoSC).

Optionally, some service providers, not shown in Fig 4.2, such as video on-demand providers might reside in the Core end-point; these providers have agreements with the network providers to guarantee the required QoS.

4.3 Network Entities

In order for the afore-mentioned network entities to cooperate and accomplish their security and QoS tasks, a set of modules and interface should reside in each entity. This section starts by explaining the network elements structure; it then defines possible protocols for the connection between the elements.

4.3.1 Network Entities Structure

In our design, we separate the Service and Network management entities. However, for these entities to interact using the above protocols, they should contain certain interfaces as shown in figure 4.3.

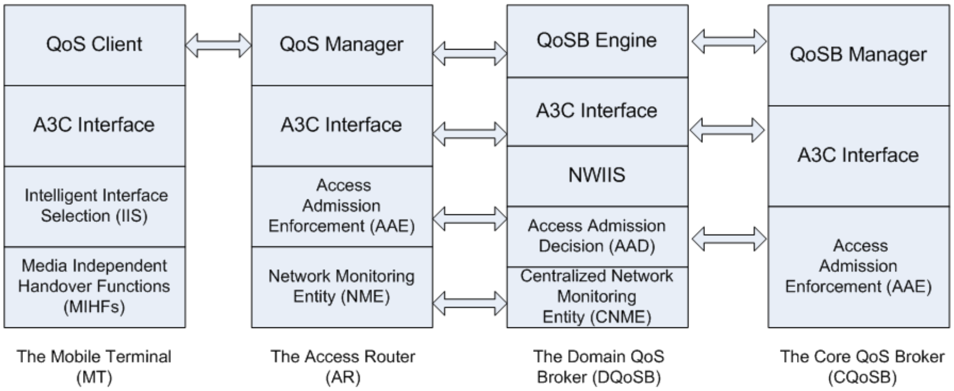
QoS Client
A3C Interface
Intelligent Interface Selection (IIS)
Media Independent Handover Functions (MIHFs)
The Mobile Terminal (MT)
QoS Manager
A3C Interface
Access Admission Enforcement (AAE)
Network Monitoring Entity (NME)
The Access Router (AR)
QoSB Engine
A3C Interface
NWIIS
Access Admission Decision (AAD)
Centralized Network Monitoring Entity (CNME)
The Domain QoS Broker (DQoSB)
QoSB Manager
A3C Interface
Access Admission Enforcement (AAE)
The Core QoS Broker (CQoSB)

Figure 4.3: The Structure and Relationship Between the Network Entities

- **The Mobile Terminal (MT)**: the MT has four interfaces. The QoS Client (QoSC) talks to the QoS Manager (QoSM) of the Access Router, A3C interface enables the client to send A3C Registration/de-registration requests to the A3C server; the Intelligent Interface Selection (IIS)is used to choose the best network for a handover based on user preference and network availability, and Media Independent Handover Functions (MIHF) which is used to control the network interfaces of the Mobile terminal and perform handover based on the IIS module decision.

- **The Access Router (AR)**: The AR comprises five modules: QoSM which has two interfaces, one with the QoSC on the mobile terminal and the other with the QoSB engine of the DQoSB, the A3C interface used to talk to the DA3C. The Access Admission Enforcement (AAE) module enforces the decision of the Access Admission Decision module (AAD) in the DQoSB; Network Monitoring Entity (NME) module monitors the utilization of network resources and reports this to the Centralized NME (CNME) module of the DQoSB, the MIHF module enables the (AR) to manage different types of peripheral networks

- **The Domain QoS Broker (DQoSB)**: the DQoSB has five modules: the QoSB Engine which makes management decisions and has two interfaces: one with the QoSM of the (AR) and the other with the CQoSB in the Core End-Point, the A3C interface is used to talk to the DA3C server in the domain; the NWIIS module manages the ARs and support load balancing, Access Admission Decision (AAD) module acts as a proxy for the high level AAD (HAAD) in the Core endpoint, and provides the AAE with policy- related decisions; the CNME module, as proposed in (Y.S08) comprises two main sub-modules: a Merger sub-module which aggregates the traces from NMEs and provides a coherent view of the traffic status. Analysis engine does a screening for network resource utilization and informs other modules of any abnormalities.

- **The Core QoS Broker (CQoSB)**: the CQoSB comprises three modules: the QoS Engine manages inter-domain connection and provides end-to-end QoS across core-end points, the A3C interface is used for the interaction with the CA3C server.

4.3.2 Network Protocols

As stated with the proposed QoS framework in (MA11a), to convey QoS-related information, network entities have to interact using a common language. Three different types of protocols are needed for the network entities interactions.

For the connection between the AAE and AAD, there is a need for policy information and configuration exchange protocol such Common Open Policy Service (COPS) (DD00). In our architecture, the access router (AR) acts as (AAE), the DQoSB acts as AAD and the CQoSB acts as a top level AAD. We used the concept of policy for a network level access control. However, for authorizing the service level request, we propose using an A3C such as DIAMETER (MA07), (PC03) or RADIUS (CR00) protocols.

The A3C protocol with its basic structure (PC03) has no QoS- related functions. Therefore, an enhanced version of the protocol (DS10) introduces three QoS-context aware entities: Resource Requesting Entity (RRE) which triggers the authorization process, Authorizing Entity (AE), an A3C server processes the access request and generates an allow/ deny decision to the Network Element (NE). The (NE) is an intermediate router between the AE and the RRE and acts as a client to the AE. Additionally, the extension proposes four new messages which are used to request QoS-related resource authorization for a given flow and then to activate the reserved resources to accommodate the connection. In the proposed architecture, the authorization process is triggered by the MT, acting as a (RRE) entity. The access router (AR) corresponds to an (NE) and the DA3C acts as (AE). For the initial request, DA3C contacts the CA3C and gets the required information for authorizing the request; this information might be cached for later requests. Since the Mobile terminal (MT) deals with different types of access networks, it needs a common interface to hide these differences. The IEEE 802.21 protocol introduces the Media Independent Handover Functions (MIHF) module (KT09) to manage the resources in the peripheral networks regardless of their technologies.

Table 4.1: Mapping the Network Entities to the Y-Comm layers

The Module	The Network Entity	The Y-Comm layer
CQoS	MT	QoS layer
QoSM, QoSB engines	the DQoSB and the CQoSB	Network QoS Layer
AAE, AAD, HAAD	AR, DQoSB, CQoSB	The Security Module
IIS, NWIIS	MT, DQoSB	Policy Management layer, the Network Management Layer
NME, CNME	AR, DQoSB	QoS , Network QoS,the security module
MIHF	MT	the Network Abstraction Layers

4.3.3 The Network Architecture in the Context of Y-Comm

Since the presented research in this thesis is in the context of Y-Comm, there is a need to show how the proposed entities could be represented in the Y-Comm architecture. In terms of functionality, this section shows a possible mapping between the afore-explained modules and the Y-Comm layers as in 8.1.

While the CQoS module of the MT corresponds to the QoS layer in the Peripheral Framework, the QoSM, QoSB engines in the DQoSB and the CQoSB are mapped to the Network QoS Layer (NQL) of the core framework. On the other hand, the Access Admission- related modules: the AAE, AAD and the HAAD provide access control in two different scenarios: controlling the access of the MT to a specific network based on the its SLA. Also, they might be used by the end point servers to specify the server's accessibility, since server's NAL defines its visibility i.e. locally, in the local network (LAN) or globally over the Internet. Such access control mechanisms will be provided as a part of the Y-Comm security module.

The IIS and NWIIS modules correspond to the Policy Management layer (PML) on the peripheral framework and the Network Management Layer (NML) of the core framework respectively. The functionality of the monitoring modules (NME, CNME) is provided through the QoS (QL) and Network QoS (NQL) layers as well as the security module. The MIHF module is used in the Network Abstraction Layers (NAL) to deal with different access networks. The A3C interfaces mainly manage the interactions with the A3C severs and thus, is considered as a part of the security module.

4.4 Targeted QoS-Signalling models

The previous section defined the network's main operational components along with their structure. These entities cooperate to provide security and QoS-related tasks. However, since there is a need for QoS provision in different situations such as when starting a connection or in the case of handover, our research adopted a similar approach to the targeted security models and proposed targeted models for signalling QoS in different scenarios.

This section explains how the network's entities cooperate to provide QoS and thus define the following three Targeted QoS-Signalling Models known as: the Registration Model describes the procedure followed when the mobile terminal (MT) first attaches to the peripheral network. The Connection Initiation Model deals with the case when the MT starts a connection to a server (SP). Finally, The Handover Model explains the QoS provision in the case of Inter and Intra Core End-Point handover. In order to provide QoS in each of these situations, the network elements interact with each other using the COPS, DIAMETER and IEEE 802.21 protocols.

4.4.1 The Registration Model

Initially, the user subscribes to the Service-Level of Agreement (SLA) which define the peripheral networks and the services that could be used by the subscriber along with the associated QoS and security parameters. For instance, based on the SLA, the user can only use two types of technologies in the peripheral networks, GSM and 3G. In either network, the SLA will define the associated security and QoS at the network level, while the SLA will specify the services the user has subscribed to, the associated security and QoS at the service level.

The SLA is shared between the mobile terminal (MT) and the CA3C server in the Core End-Point, The QoSB engine of the CQoSB gets a copy of the SLA which is related to the accessing the network. As shown in Fig 4.4, once the (MT) gets an IP address, it should be authenticated by the A3C server in order to access the network. After a successful authentication, the Access Admission Enforcement (AAE) of the AR asks the AAD of the DQoSB for a user- specific Access Decision (AD Req). Since it is the first interaction with this user, the DQoSB approaches the CQoSB- the HAAD module- for this information, the HAAD extracts user's profile from the QoSB Engine and passes the decision - via (AD Res) message- all the way back to the (AR) which configures the access policy according to the received profile and sends an acknowledgement message (Ack).

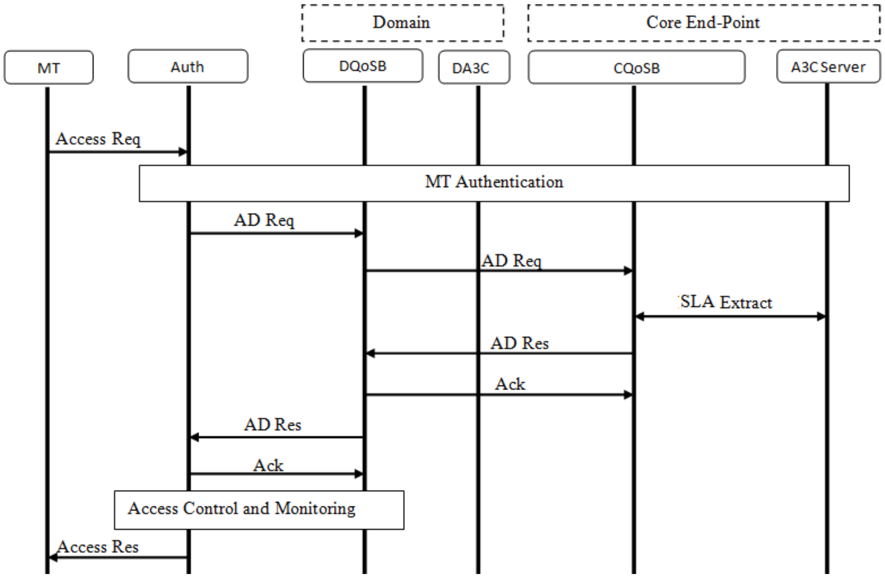

Figure 4.4: The Registration Model

4.4.2 The Connection Initiation Model

This model enables a client and the server to negotiate the QoS specifications before setting the connection. The model shown in Fig 4.5 discusses the case when the MT and the Server (S) reside in the same Core End-Point but in different domains.

The MT initiates a connection request -with a required QoS denoted in the QoS Specification (QoS-Spec) field - to the server (S). If the request complies with the network access policy configured on the AR of the source domain, an Authorization Request (Auth-Req) to access the service with the QoS stated in the QoS-Spec is initiated towards the DA3C server. If the DA3C holds a copy of the user's profile, it responds with Authorization Response (Auth-Res) message; otherwise, it passes the request to the CA3C server which holds user's contract details. In the case of a successful authorization, the QoSM of the AR in the source domain forwards the access request to the QoSM of the AR in the destination domain. This triggers the same request authorization process as in the first domain. As shown in Fig 4.5, in the case of a successful authorization, resources in the destination domain are activated using Resource- Activation request/ response messages (Resc-Act. Req / Res). The L2 resources are allocated by using IEEE 802.21 messages, and then an access response is sent back to the AR in the source network. Upon the receipt of a positive access response, resources in the source network are activated using (Resc-Act. Req/Res) messages, these activities in the source network are not shown in Fig 4.5.

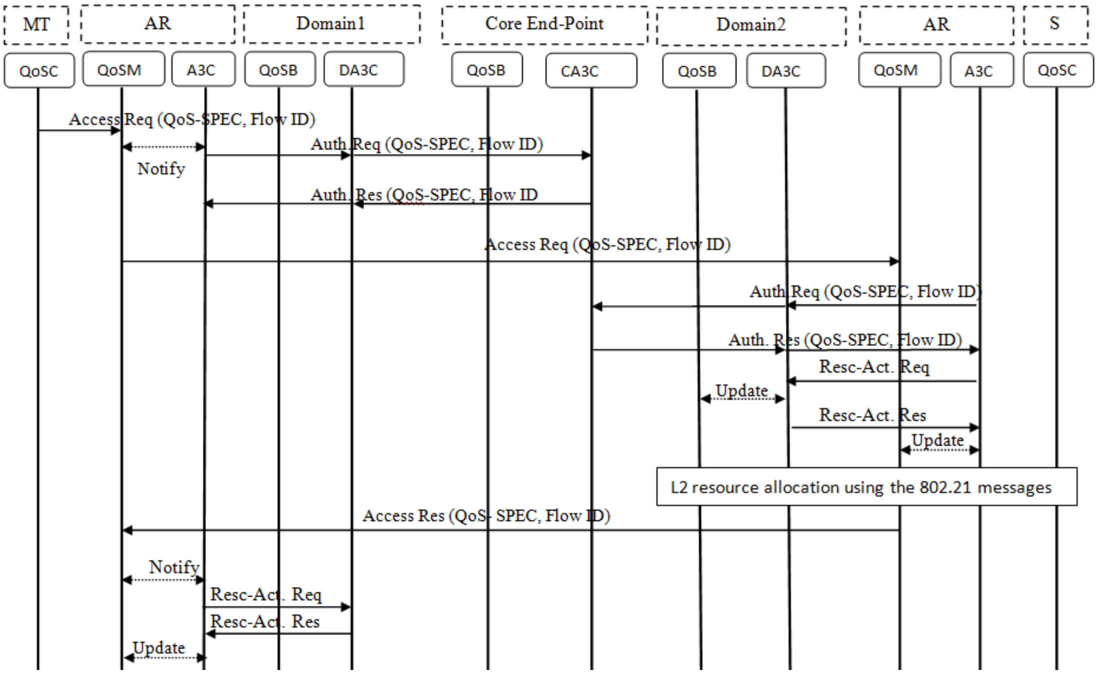
MT
AR
Domain1
Core End-Point
Domain2
AR
S
QoSC
QoSM
A3C
QoSB
DA3C
QoSB
CA3C
QoSB
DA3C
QoSM
A3C
QoSC
Access Req (QoS-SPEC, Flow ID)
Notify
Auth Req (QoS-SPEC, Flow ID)
Auth. Res (QoS-SPEC, Flow ID
Access Req (QoS-SPEC, Flow ID)
Auth Req (QoS-SPEC, Flow ID)
Auth. Res (QoS-SPEC, Flow ID)
Resc-Act. Req
Update
Resc-Act. Res
Update
L2 resource allocation using the 802.21 messages
Access Res (QoS- SPEC, Flow ID)
Notify
Resc-Act. Req
Resc-Act. Res
Update

Figure 4.5: The Connection Initiation Model

4.4.3 The Handover Model

This section explains the QoS provision in the case of intra and inter Core End-Point handover. As shown in Figs 4.6 and 4.7, the MT gets QoS -related information about available networks, the IIS module of the MT decides on the target network and a Handover request containing the desired associated QoS is sent to the QoSM module of the AR which passes it all the way to the DQoSB2 via the Core End-Point. The MT has to be authenticated; also the security keys should be launched in the target network before the handover really happens. In order to apply the right access control in the new network, the AAD module of the DQoSB2 approaches the HAAD of the Core End-Point to get the Admission Decision related to the user. After configuring the access policy in the target Access Router, it starts L2 resources reservation using IEEE802.21 messages. A successful handover response message is sent back to MT to trigger the actual handover.

In the case of an Inter-Core-End Points handover, the old Core-End Point (CEP) provides the target CEP with the user's SLA; thus, the MT's related information becomes available in the target network. The remaining steps are very similar to the intra-Core-End Point handover as shown in Fig 4.7. The communication between the CEPs takes place over the backbone of the Internet. Architectures like the Intermon (MB04), which is a research framework to facilitate Inter-domain QoS monitoring and analysis for validation, planning

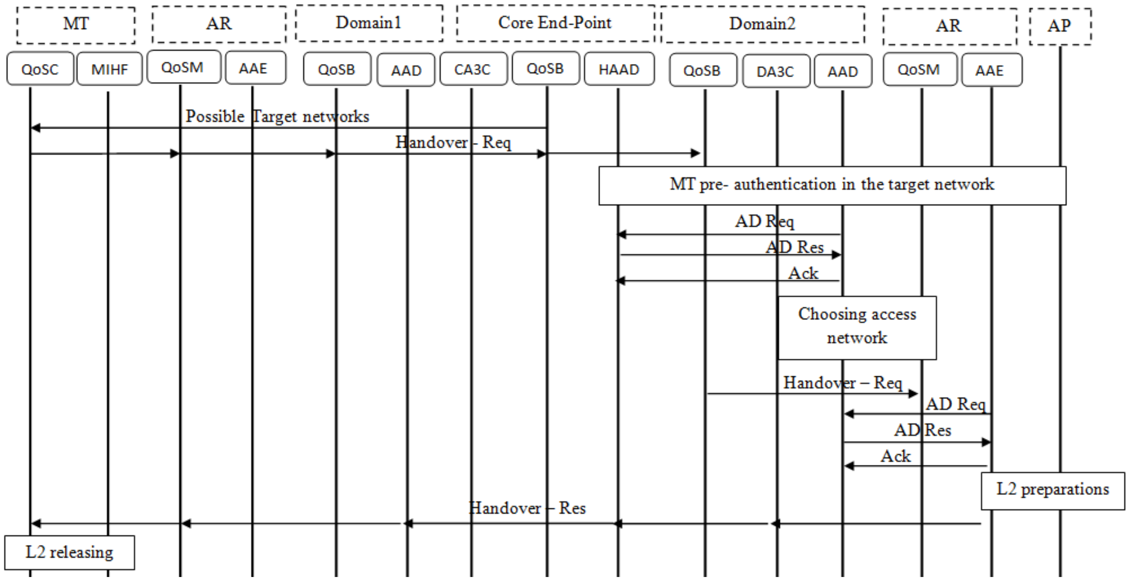

Figure 4.6: The Intra-Core End Point Handover Model

and optimisation of inter-domain QoS, could be used to manage the communication among CEPs. However, the research in this thesis is not concerned with proposing inter-CEPs communication framework

4.5 Summary

This chapter introduced a QoS framework for 4G systems, the framework proposed targeted models for signalling QoS in different scenarios such as the initial registration and connection as well as handover phases. In order to define these models, there was a need to specify the operational entities in the network. Furthermore, these entities represented the actual parties of the security protocols defined in the coming chapters and thus, facilitated the security-QoS integration.

However, for the QoS framework to be effective, it needs to consider multi-homed mobile devices which are supported with several network cards and could compromise the QoS by initiating simultaneous connections over different networks. The multi-homing issue will be investigated in Chapter 9, where further enhancements on current Internet servers such as the naming and locating systems will be introduced. However, the next chapter will describe the research, conducted to address the security issue in heterogeneous environment by introducing the Authentication and Key Agreement (AKA) framework.

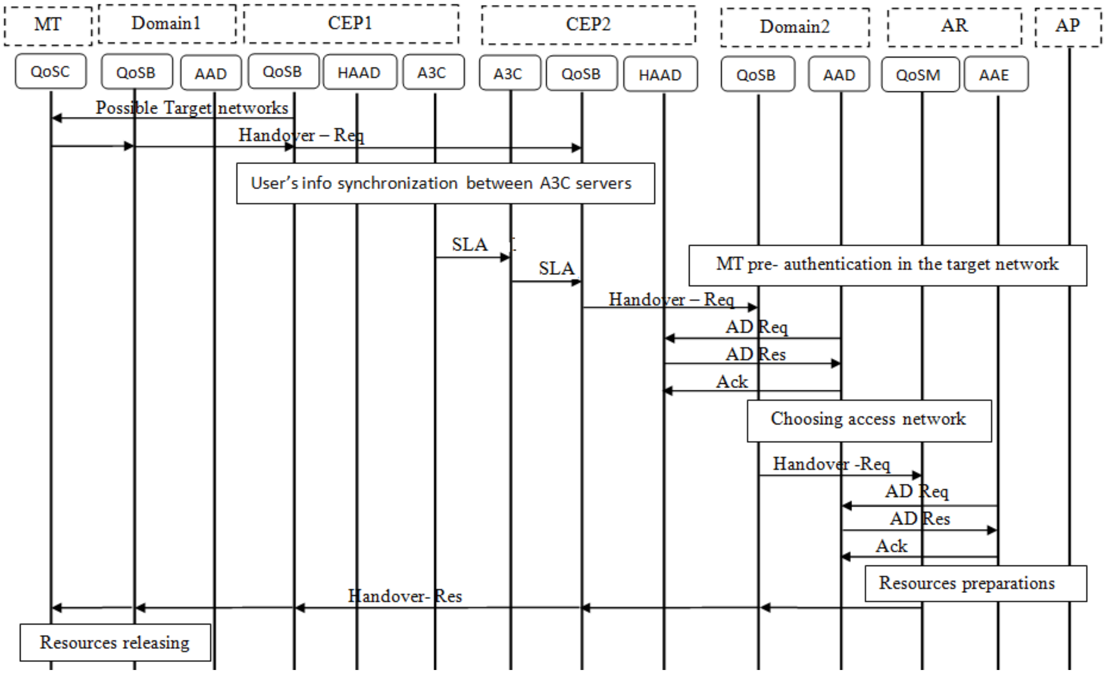

Figure 4.7: The Inter-Core End Point Handover Model

Chapter 5

The Authentication and Key Agreement Framework

5.1 Introduction

Next Generation Networks (NGNs) represent an open architecture in which two different domains need to cooperate in order to provide ubiquitous connectivity. The first is network operators domain, where multiple network operators share the core network to provide network accessibility over a wide variety of wireless technologies such as WiFi and mobile network technologies. The other is the Service Providers (SPs) domain, which launches various services ranging from the normal video-streaming to the most confidential E-Commerce services. This highlights the fact that any efficient security solution for heterogeneous networks has to consider the security in these different domains.

Therefore, as explained in chapter 2, the Y-Comm framework has proposed a four-layer Integrated Security Module (ISM) which considers the security of different levels, namely User, Network and Service levels. To define the underlying protocols of the security module, this chapter introduces an Authentication and Key Agreement (AKA)framework, which comprises a number of AKA protocols to address the functionality of some of the layers of the ISM. The framework introduces three types of AKA protocols: The Network-Level AKA (NL-AKA) protocol which provides security at the network level. The Service-Level AKA (SL-AKA) protocol sets up a secure connection between the mobile device and the service provider. The User-Level AKA (UL-AKA) protocol achieves mutual authentication between the user, the SIM/Personal ID card (PIC) and the mobile device.

This chapter is organized as follows: Section 2 defines the main parties of the AKA framework. Section 3 introduces a proposed key hierarchy for the framework. Section 4 explains a set of desired security parameters for AKA protocols. Section 5 introduces the proposed User-Level AKA protocol. The chapter is summarized in section 6.

5.2 An Overview of the AKA Framework

To address the security issue, we propose a security architecture to provide authentication and key agreement at the User, Service and Network levels. The framework considers the network structure in Chapter 4 and comprises four entities:

The Personal Identification Card (PIC):

Similarly to the SIM card in 2, 2.5 and 3G technologies (Cha05), the PIC holds user's credentials such as the subscribed services' IDs and security keys.

The Mobile Terminal (MT):

It is the user's device such as cellular phones, PDAs, and laptops, identified by a unique manufacture ID (mID). Similar to the Equipment Identity Register (EIR) used in GSM (Cha05), the mID is used to lock the MT to a specific PIC.

Application/Service Providers (SPs):

These are different types of services such as e-Commerce, on-line banking and electronic public services in addition to access to vedio on-demand/news, Grid and Cloud resources/services. which could be accessed by the subscribed clients using applications installed in the MTs. Each SP is identified by a unique Service ID (SrvID).

Network Operators:

Multiple operators (telecommunication carriers) cooperate to enable the MT to access the SPs. Referring to the network structure in Chapter 4, each domain is mapped to a single network operator such as GSM or WiMAX.

Upon signing the initial contract, the user's profile information including the Service Level of Agreement (SLA) is shared between the Centralized A3C (CA3C) in the Core End-Point and the subscriber. As shown in the Figure 5.1, the operation of the security framework goes through three main stages:

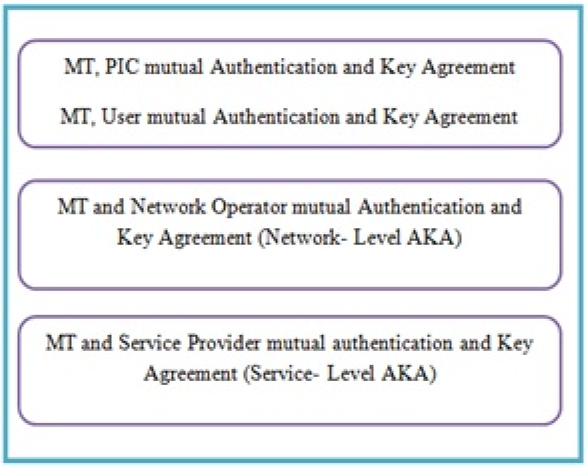
MT, PIC mutual Authentication and Key Agreement
MT, User mutual Authentication and Key Agreement
MT and Network Operator mutual Authentication and Key Agreement (Network- Level AKA)
MT and Service Provider mutual authentication and Key Agreement (Service- Level AKA)

Figure 5.1: The AKA Framework Stages

- The User-Level AKA protocol comprises two sub-stages: the First achieves mutual authentication between the PIC and the MT. In the Second, the user is authenticated, based on his biometric information, to use the mobile terminal.
- In the Second stage, the MT is authenticated to access the peripheral network using new proposed Network-Level AKA (NL-AKA) protocols. Two NL-AKA protocols are needed; the first achieves mutual authentication between the network and the mobile terminal when the MT joins the network for the first time, and thus is called Initial AKA protocol, the latter considers the case of vertical handover and authenticates the MT when it moves to a new network.
- The Third stage authenticates the MT to access the subscribed service over specific access networks. Similar to the NL-AKA protocols, there is a need for new Service-Level AKA (SL-AKA) protocols to achieve mutual authentication between the mobile terminal and the service provider in the initial and in handover scenarios.

These protocols have been verified using formal methods approach based on Casper/FDR tool

5.3 The Key Hierarchy

Upon signing the initial contract, the user's profile information including the SLA is shared between the Centralized A3C CA3C in the Core-End Point and the subscriber. The user's

PIC holds a Secret Key (SK) shared with the MT and a hashed value of the user's biometric-information as well as a Unique Secret Key (UK) shared with the CA3C.
As shown in Fig 5.2, for the Service-Level AKA, a service-specific secret key (Srvkey) are derived using the UK, service ID (SrvID); the user's subscription ID (SubID) and a lifetime value as follows: Srvkey= F(UK, SrvID, SubID, lifetime). Using the Srvkey, an Association Key (ASKey) is derived to protect the session between the client and the service provider.
For the Network-Level AKA, a Domain-Specific Master Key (DSMK) is derived for each access domain as follows: DSMK= F(UK, Seq, AAA-domain) where Seq is a random number, AAA-domain is the domain name of the corresponding DA3C server. From the DSMK key one Authentication Key (AK) and one or more Secret Keys (SKs) is derived to encrypt the connection between the authenticated entities.

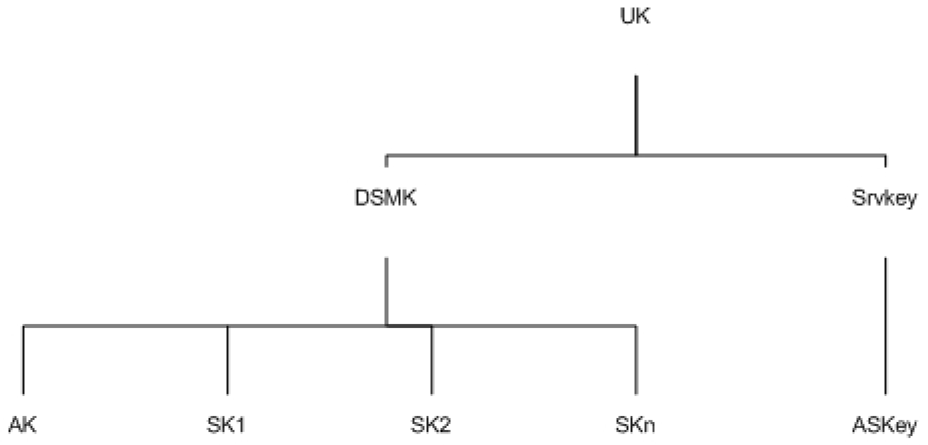

Figure 5.2: Key Hierarchy

5.4 Analysing the Security Protocols

To verify the UL-NL and SL-AKA protocols, we use a form of formal methods approach based on Casper/FDR tool (LBDH09). The Casper tool accepts an abstract, human-friendly description of the system and compiles it into Communication Sequential Processes (CSP) code, suitable for the Failures-Divergence Refinement (FDR) (Sys93) checker.
Furthermore, as stated in (AM96), it is desirable that AKA protocols meet certain security properties. Therefore, a list of these properties will be used to analyse the security features of all the proposed AKA protocols. The properties are as follows:

1. *Mutual Entity Authentication*: This is achieved when each party is assured of the identity of the other party.

2. *Mutual Key Authentication*: This is achieved when each party is assured that no other party aside from a specifically identified second party gains access to a particular secret key.

3. *Mutual Key Confirmation*: This requirement means that each party is assured that the other has possession of a particular secret key.

4. *Key Freshness*: a key is considered fresh if it can be guaranteed to be new and not reused through actions of either an adversary or authorized party.

5. *Unknown-Key Share*: In this attack the two parties compute the same session key but have different views of their peers in the key exchange. In other words, in this attack an entity A believes that it shares a key with another entity B while B mistakenly believes the key is shared with an entity $E \neq A$ instead.

6. *Key Compromise Impersonation Resilience*: This property implies that if the Intruder compromised the long-term key of one party, he should not be able to masquerade to the party as a different party.

5.5 The User-Level AKA Protocol

5.5.1 Introduction

For communication in Next Generation Networks, highly-developed mobile devices will enable users to store and manage a lot of credentials on their terminals. Furthermore,these terminal devices will represent and act on behalf of users when accessing different networks and connecting to a wide variety of services. This situation highlights the need for securing transactions between the end users and their mobile devices as well as the need for maintaining the integrity of the mobile devices. Creating such a secure environment will emphasise on the trust worthiness of mobile devices and encourage end users to delegate their devices the communication with sensitive services.

5.5.2 Related Work

Future mobile devices are expected to access different networks (such as 3rd generation network, WLAN, Bluetooth, Internet, and etc). Hence sensitivity data are stored in them.

Unfortunately, the password-based identification is not secure enough to control user's access to the mobile terminal and vulnerable to birthday and brute-force attack. This sections describes some of research efforts to address the authentication between the MT, the PIC and the User.

AKA and Authorization Scheme

In (YZ05), the authors propose an AKA and Authorization framework for 4G networks. At the initial stage, the framework combines password, biometric-information as well as public key infrastructure (PKI) to achieve mutual authentication between the user, the SIM card and the device. Based on the result of the authentication in the initial stage, the framework achieves authentication between the mobile device and the network.

Although it is stated in (YZ05) that the framework was proven to be scalable and provides some desired security features such as multi-pronged mutual authentication, the framework suffers from two major drawbacks: firstly, in order to provide a considerably robust platform for user's access to sensitive services and data and achieve the authentication process in the initial stage, the framework associates the Trusted Computing (TC) with the PKI by implementing Trusted Mobile Platform (TMP) (TMP04). These represent major modifications to the architecture of mobile devices. Secondly, some of the required functions to deal with the PKI-complexity and checking the integrity of the mobile terminal do not consider the limitations of battery and processing power in small devices such as Mobile terminals and Personal Digital Assistant (PDAs). These two reasons make the framework inapplicable with current architecture and capabilities of mobile devices.

The Device Authentication Protocol of the Mobile Ethernet Security Framework

The Mobile Ethernet group has in (MK04) proposed an AKA framework that deals with security at the network and service levels as well as achieving mutual authentication between the user, SIM card and the mobile terminal. The Mobile Ethernet's solution proposes two-stage authentication protocol; the first stage is used in the initial authentication; when the PIC is plugged into the MT for the first time. This stage is based on PKI and it aims at achieving a mutual authentication between the MT and PIC and agreeing on a secret key (K) which will be stored in the PIC and the MT. After the initial authentication, a simplified protocol, based on the derived secret key (K), is used for any subsequent authentication process.

Similar to the case of the previous AKA in subsection 5.5.2, due to the fact that setting-up a PKI is a complex and costly process that consists of several steps: registration of

Table 5.1: notation

Abbreviation	Full name and description
PIC	The Personal Identification (PIC), initially shares the key K with the MT. Key management protocols such as HMAC-Authenticated Diffie-Hellman (Euc06) are used to share the key between the PIC and the MT.
MT	Mobile Terminal
r1, r2	Random numbers
K	A pre-shared secret key between the MT and the PIC
Req	An authentication request message
$MAC\{m\}_K$	Message authentication code of message (m) using the key (K)

users, generation of keys, issuance and distribution of certificates. Additionally, PKI involves other complex processes such as certificate retrieval and certification path construction and validation. Because of these reasons, the author believes that PKI is not suitable for mobile devices in general. Furthermore, as will be explained in the following subsection, the formal verification results show attack against the authentication protocol in the second stage.

Analysing the Device Authentication Protocol of the Mobile Ethernet

After running the initial authentication protocol, the PIC and the MT will agree on a secret key (K), which will facilitate the subsequent authentications. By considering the notation in Table 5.1, the authentication protocol runs as follows:
The mobile terminal sends an authentication request (Req) to the PIC, which responds by sending a random value (R1) as a challenge towards the MT. The MT returns the hash of the R1 as well as a challenge (R2). The PIC responds to this challenge by sending the hashed R2.
$Msg1.MT \rightarrow PIC : Req$
$Msg2.PIC \rightarrow MT : R1$
$Msg3.MT \rightarrow PIC : MAC\{R1\}\{K\}, R2$
$Msg4.PIC \rightarrow MT : MAC\{R2\}\{K\}$

The full Casper's description of the protocol is mentioned in Appendix A. After modelling this protocol, Casper/FDR discovered the following attack. The notation I_PIC, I_MT represents the case where the Intruder impersonates the PIC, MT respectively. In this attack, the intruder intercepts and passively relays the messages between the MT and the PIC and thus, the mobile terminal will complete running the protocol believing that it was with the PIC, while it was with the Intruder instead. Similarly, the PIC will believe it has been running the protocol with the MT, while in reality it was with the Intruder.

```
1a.  MT -> I_PIC : req
1b.  I_PIC -> PIC : req
2a.  PIC -> I_MT : R1
2b.  I_PIC -> MT : R1
3a.  MT -> I_PIC : MAC{R1}{K}, R2
3b.  I_PIC -> PIC : MAC{R1}{K}, R2
4a.  PIC -> I_MT : MAC{R2}{K}
4b.  I_MT -> MT : MAC{R2}{K}
```

5.5.3 The Proposed User-Level AKA Protocol

As shown in Fig 5.1, the first stage of the proposed AKA framework comprises two sub-stages: the first achieves mutual authentication between the PIC and the MT, while in the second, the user is authenticated based on his biometric-information. As explained in (MA12c), the proposed AKA protocol for the MT, PIC authentication is based in the Challenge-Response paradigm.

By considering the notations in Table 5.2, the protocol runs as follows:

$Msg1.PIC \rightarrow MT : \{r1, Pseq\}\{SK(MT)\}$

Upon plugging the Personal Identification Card PIC into the Mobile terminal MT, the AKA process starts by sending a random number r1 in Msg1.

$Msg2.MT \rightarrow PIC : \{MiD, r1, r2\}\{SK(MT)\}$

The MT constructs a challenge message Msg2 containing a Mobile ID, a fresh challenge random r2 and the received random r1, this message is encrypted by the pre-shared key SK(MT). Using the information included in Msg2, both ends generate a secret key K= F (SK(MT), r1, r2, miD, PSeq) to secure the connection between the ends, the uniqueness of the derived key is based on the freshness of nonce r1, r2 and the secrecy of the pre-shared key SK.

$Msg3.PIC \rightarrow MT : \{r3, r2\}\{K\}$

The PIC responds to the challenge in Msg2 by constructing Msg3 which contains the received challenge random r2 and another challenge random number r3, this message is encrypted

Table 5.2: Notation

Abbreviation	Full name and description
PIC	The Personal Identification (PIC), initially shares SK(MT) with the MT and holds the (UK).
MT	Mobile Terminal
r1, r2, r3	Random numbers
miD	Mobile device unique ID
K	A secret key, derived to secure the connection between the MT and the PIC
SK(MT)	A pre-shared key between the PIC and the MT
PSeq	PIC unique sequence number
F	An irreversible key derivation function
Ackm	An Authentication Token: Ackm= F(MiD, PSeq, r1, r2)
$MAC\{m\}_K$	Message authentication code of message (m) using the key (K)
$Enc\{m\}_K$	Encrypting the message (m) using the key (K)

using the derived secret key K.

$Msg4.MT \rightarrow PIC : \{r3, Ackm\}\{K\}$

The MT responds by sending the received challenge r3 along with the pre-shared acknowledgement string Ackm via Msg4. As shown in Table 5.2, the Ackm is derived in a way to include the identities of the two parties (the MT and the PIC), also it includes fresh random values (r1, r2) to guarantee the freshness, this way possessing the Ackm will help in achieving entity authentication as will be described in section 5.5.3.

$Msg5.PIC \rightarrow MT : \{Ackm\}\{K\}$

The SP verifies the included Ackm in Msg4 and composes Msg5. In the case of a successful authentication among the PIC, the MT and the user, the MT represents the PIC and the user in the following stages of the AKA framework.

The proposed protocol is of the challenge-response type and is based on one secret key SK(MT) which is pre-shared between the MT and PIC. However, this type of protocols is vulnerable to replay attacks such as the one described in section 5.5.2. To avoid such attacks the proposed protocol uses three random numbers r1,r2 and r3. Furthermore, in order to meet some of the desired security requirements such as the mutual entity authentication, the authentication token Ackm is used in the protocol as will be explained in section 5.5.3.

Another possible way to stop replay attacks is by using different keys to encrypt the messages in the two directions. Based on this, we presume that PIC has two keys: the SK(MT) which is shared with the MT and the secret key (K) and the protocol goes as follows
$Msg1.PIC \rightarrow MT : \{r1, K\}\{SK(MT)\}$
$Msg2.MT \rightarrow PIC : \{r1, r2\}\{K\}$
$Msg3.PIC \rightarrow MT : \{r2\}\{K\}$
However, this solution requires one of the parties to hold two keys, one is pre-shared with the other party and will be used to encrypt the first message. The second key is sent in the first message to the second party and will be used it to encrypt the rest of the messages. Furthermore, some security requirements such as key freshness, mutual entity authentication and the resilience to key compromise impersonation attack are not achieved.

Formal Verification

We modelled our protocol by preparing a Casper input file describing the UL-AKA protocol. For conciseness, we only show here the #Specification and #Intruder headings, while the #Free Variables, #Protocol Descriptions and #System headings are included in Appendix B.
The #Free variables heading defines the participating parties, the variables and the used functions. It is worth noting that Casper does not specify a built-in method to simulate key derivation functions; therefore, we specifically defined therein the function F which is used to derive the session key (K) specific. The Protocol Description heading specifies how the intended parties will use the functions to generate the corresponding keys.
The security requirements of the system are defined under the #Specification heading. The lines starting with the keyword Secret is used to define the secrecy properties of the protocol. For example, the first line specifies SK(MT) as a secret between the PIC and MT. The lines starting with Agreement define the protocol's authenticity properties; thus, the first authenticity of the figure above specifies that the MT is correctly authenticated to the PIC and agreed on the nonce value (r3). The WeakAgreement(X,Y) specification means that if Y thinks he has successfully completed a run of the protocol with X, then X has previously been running the protocol with Y.
# Specification

```
Secret(PIC,SK(MT),[MT])
Secret(MT,SK(MT),[PIC])
Secret(PIC,miD,[MT])
Secret(PIC,K,[MT])
Secret(MT,K,[PIC])
```

```
Agreement(MT,PIC,[r3])
Agreement(PIC,MT,[r2])
WeakAgreement(MT, PIC)
WeakAgreement(PIC, MT)
```

The #Intruder Information heading shows that the intruder identity is Mallory, the identities of all agents, the nonce R1 and the function F are included in the intruder initial knowledge.

#Intruder Information

```
Intruder = Mallory
IntruderKnowledge = PICard, Mobile, R1, F
```

Running Casper/FDR tool verifies that none of the checked assertions defined in the #Specification heading was vulnerable to an attack. This is mainly due to the assumption that the SK(MT) key is secret between the PIC and the MT. Exposing this key, will lead to the entire protocol being compromised.

Protocol Analysis and Security Considerations

Although Casper/FDR has shown no attack against the proposed protocol, we need to carefully consider the result, Casper/FDR proves the protocol in the system specified in the System heading Appendix B; however, the protocol might be vulnerable in another system. Further analysis of the protocol based on the security requirement list is given in this section.

1. Mutual Entity Authentication:

   There is no direct specification within Casper to check this property, yet in order to show how our protocol could meet this requirement, we explicitly considered the Ackm value is generated as follows $Ackm = F(MiD, PSeq, random)$. This value is pre-stored in the PIC and Mobile terminal. In Msg 4, 5 each entity ensures the other party to have the right Ackm, which includes the parties' identities as parameters, thus, enforcing entity authentication. If the MiD and Pseq were exposed, it is not feasible for the Intruder to generate the Ackm, because it does not know the right random value. Even if the Intruder recorded Msg5, it could not be used in next sessions because a fresh key K is used for each session.

2. Mutual Key Authentication:

   The mutual authentication between the MT and the PIC is based on the secrecy of the derived session key (K). We got Casper to check this using the Secret (PIC, K, [MT])assertion check.

3. Mutual Key Confirmation:

   This requirement is achieved by performing the checks after Msg3 and 4 in the Protocol Description heading Appendix B. By using the Decryptable function each party makes sure that the valid secret key K is possessed by the other part. If the any of the check failed, the protocol aborts.

4. Key Freshness:

   Casper does not have any function to check this requirement, so we included freshly generated values r1, r2 in the derivation function of the the session key K: K= F(SK(MT),r1,r2,miD, PSeq) ; thus the fact that Casper does not detect any attack on the secrecy of the session key (K) implies that key freshness is not violated.

5. Unknown-Key Share:

   The Aliveness assertion is used to check this attack. Additionally, making a binding between the Keys and the parties' identity deals with this attack. This has been achieved in this protocol by including the identities of the MT and the PIC in the KDF of the K.

6. Key Compromise Impersonation Resilience: This property is not be achieved with the proposed protocol as the compromise of the long-term key Sk(MT) will lead to the session key (K) being compromised.

5.5.4 Biometric-Information Based Authentication

For this stage, we assume that the Mobile terminal is equipped with a trusted biometric-information reader such as fingerprint reader. When the user makes the initial contract, a brief hashed value- of the user's biometric-information is stored in the PIC. This hashed value could be generated using algorithms like (CV02) and (YS05) which have been designed to provide similarity preserving and eliminate any noise in the biometric sample.

After running the previous AKA protocol and setting up a secure channel between the MT and the PIC, the user is prompted to enter his biometric-information, the MT processes the data and generates a hashed value of the submitted info. This hashed value is passed to the PIC which compares it with the previously stored value. In case of match, the user is authenticated as the PIC owner and consequently to use the MT. From this point onwards, MT will represent the user in both network and service Â– level connections.

5.6 Summary

To define the underlying security protocol of the Integrated Security Module of the Y-Comm framework, this chapter introduces an AKA framework that comprises three types of AKA protocol to provide security at the network, service and user levels. The chapter explains the proposed AKA for the User-Level AKA (UL-AKA). The protocol was verified using Casper/FDR and proven to meet all the desired security requirements. The underlying protocols for the Network and Service Levels will be explained in the following two chapters.

Chapter 6

The AKA Framework, The Network-Level AKA Protocol

6.1 Introduction

As explained in chapter 5, the AKA framework comprises three stages; User-Level AKA (UL-AKA), the Network-Level AKA (NL-AKA) and the Service-Level AKA (SL-AKA). The first phase of the AKA framework achieves mutual authentication between the Personal Identification Card (PIC), the mobile terminal and the user. In the case of a successful authentication, the mobile terminal (MT) will represent the user in the following authentication stages, namely the NL-AKA and the SL-AKA.
This chapter introduces NL-AKA protocols for authenticating the MT and the network in case of initial connection; when the MT joins the network for the first time as well as in the case of handover; when the MT switches between different access networks. The NL-AKA protocols consider the open architecture introduced in Chapter 4.

6.2 Problem Definition

Due to the fact that the connectivity in the peripheral networks will be based on a wide variety of wireless technologies, provided by different operators, various network operators need to cooperate and coexist in the core network. Furthermore, new providers might choose to join the network and share the spectrum. Unlike current communication systems such as 2G and 3G, which introduce closed environments where the core network is controlled and owned by sole network operators and thus its security is mainly based on the assumption that, the core network is physically secure.

The above discussion highlights the fact that we are moving towards an open, heterogeneous environment where the core network is not controlled by a single operator, so multiple operators will have to cooperate. This tendency will bring about radical changes to handover mechanisms. Current mechanisms mainly support the network-controlled handover in which the decision to implement handover is taken by the network(s) to which the mobile device is currently attached. While this type of handover works fine in current homogeneous systems, where the core network is controlled by a sole operator and thus information about the topology of different networks is available, this type of handover is not suitable for heterogeneous environments, since multiple operators coexist in the core network. This highlights the need for the client-based handover in which the client is the deciding entity rather than the network. In this type of handover, the mobile device will be responsible for initiating the handover, acquiring and releasing the resources in the new and old network respectively. However, this situation brings about new security threats in term of authenticating the mobile device to the new access network in case of handover and maintaining data confidentiality as well as controlling the allocation of network resources and making sure that this process is accomplished by authorized parties. While the latter issue was addressed by the QoS framework in chapter 4, the first will be investigated in this chapter by analysing different research efforts such as (HOK07), (rGPPG), (KT09), (MK04), (3GP06), (Ali10), which have been trying to deal with the security issue, a detailed analysis of these approaches could be found in (MA11b) and (MA12b) .

Due to the fact that the Mobile Ethernet group (MK04), (I.D06) assumes a generic network structure, which is very close to the one described in Chapter 4, this research will consider the Mobile Ethernet's vertical handover AKA protocol as a model to investigate the security threats in the open architecture. The protocol will be analysed and verified using formal methods approach. The results discovered some security breaches in the deployment of the Mobile Ethernet's AKA protocol, which highlight the need for a new protocol.

6.3 Initial NL-AKA Protocol

The Initial NL-AKA is needed to identify and authorize mobile nodes when initially join the access network.

6.3.1 Related Work

This section describes some of the related work towards introducing AKA protocols for the initial registration in heterogeneous environments.

AKA and Authorization Scheme Based on Trusted Mobile Platform

The work in (YZ05) has introduced an AKA and authorization scheme to achieve mutual authentication between the user, the Mobile Terminal and the SIM card. This scheme deploys passwords in combination with biometric information and Public key Infrastructure, the scheme also benefits from the Trusted Mobile Platform (TMP) (TMP04) to guarantee the internal integrity of the mobile device.

As explained in (YZ05), the proposed scheme achieves many security features such as mutual authentication, protection on wired links as well as resistant to replay and man-in-middle attacks. However, the main drawbacks of the scheme are as follows:

1. The scheme proposes using the PKI approach. However, this comes at the cost of a higher overhead especially in terms of key management and cryptographic operations (MJS02).

2. Many security features of the scheme are based mainly on the hardware architecture of the trusted mobile platform, this implies that the proposed scheme is not generic and might not be compatible with none TMP-supported devices.

Handover Key Working Group (HOKEY WG)

This group is concerned with providing a set of protocols and mechanisms to secure handover. It introduced an abstract mechanism for delivering root keys from an Extensible Authentication Protocol EAP (BA04) server to another network server that requires the keys for offering security protected services, such as re- authenticating the EAP-supporting peer using the EAP Re- authentication Protocol (ERP) (VN08). The ERP protocol mainly considers the case of handover, and recommends full EAP for initial authentication. However, in either case, the solution is based on the assumption that all access networks support the EAP framework, this assumption might not be feasible in heterogeneous networks since the EAP severs might belong to different operators.

The Third Generation Partnership Project (3GPP)

The 3GPP group (rGPPG) has proposed the integration of 3GPP-WLAN and 3GPP-WiMAX as examples of heterogeneous networks. In both cases, the 3GPP recommends invoking EAP-AKA (JA06) for the initial authentication. By integrating the 3GPP-AKA (MZ05) protocol and the EAP platform, the EAP-AKA achieves many desired security features such as mutual authentication between the device and the network.

One issue with this approach is that it is fully dependent on a specific core wireless technology, in this case, the 3GPP core network. Whoever wants to add a new wireless access to an existing network will always need to develop a method that integrates wireless access with the 3GPP core infrastructure. Additionally, the solution is based on implementing the EAP platform globally which requires all AAAC severs to support the EAP.

Security in the Mobile Ethernet Architecture

Mobile Ethernet Architecture is a Beyond 3G network system for the all IP integrated network using MAC layer technologies (I.D06). The architecture is based on the Wide Area Ethernet (WAE) which is a virtual private network aimed at providing connectivity based on the Ethernet (MAC) addressing and thus achieves interoperability among different IP-based operators. More details about the Mobile Ethernet framework could be found in Chapter 2. For the Network-Level security, Mobile Ethernet has proposed AKA protocols for the initial and handover cases. The proposed protocols consider a generic structure for heterogeneous networks similar to the one in Chapter 4, and since it operates at Layer 2 (L2), it does not require underlying platforms such as EAP and thus, could be used with any operator. Furthermore, as stated in (MK04), the initial AKA protocol achieves mutual authentication between the mobile terminal and the network and meets many desired security features. Due to these factors, this initial AKA protocol in (MK04) is very relevant work and we will consider this as a model to investigate the security threats in the environment. This protocol will be analysed in Section 6.4 using the formal methods approach.

6.4 The Initial AKA protocol for the Mobile Ethernet

This section presents a formal analysis of the Initial AKA protocol for Mobile Ethernet proposed by Masahiro et al (MK04). For this protocol, the security architecture consists of the following network components:

- **The Authentication Information Server (AIS):** manages the subscriber's information, the AIS corresponds to the Core A3C (CA3C) server in the Core End-Point.

- **The Authentication Server (AS):** authenticates the subscribers based on information retrieved from the AIS. The AS corresponds to the Domain A3C (DA3C) server.

- **The Entry Points (EPs):** represent one end point for wireless communication and represent Access Points (APs) or Access Routers (ARs).

- **The Mobile Device (M):** is the mobile terminal accessing the network.

Table 6.1: Notation

The Notation	Description
M	The Mobile Node
AIS	The Authentication Information Server
AS	The Authentication Server
R1, R2	Random values
E(K, Msg)	Encrypted Msg by key K
D(K, Msg)	Decrypted Msg by key K
PRF, PRF2	Pseudo-random function
MS	Master Secret key $MS = PRF(UUK, R1\|R2)$
AK	Authentication Key $AK = PRF(MS, R1\|R2)$
SK	Secret Key used for encryption $SK = PRF2(MS, R1\|R2)$

6.4.1 The Protocol Description

The initial AKA protocol of (MK04) is based on the challenge-response paradigm. By considering the notation in Table 6.1, the protocol goes as follows: initially the mobile device (M) and the AIS pre-share User ID (UID) and user unique key (UUK). When the MD attaches to the access network, it sends its UID and a random number (R1) as a challenge all the way to the AS. The AS appends a freshly created random (R2) to the message and passes it to the AIS. Using the received UID, the AIS looks up in its database and finds the corresponding UUK, then it derives the Master Key (MS) and passes it along with the UID to the AS. The received MS is used by the AS to derive the Authentication Key (AK) and the Secret key (SK), then the AS returns the challenge (R1) encrypted using the AK and a challenge R2 to the mobile device. If the Mobile device managed to derive the required keys, it should be able to verify the received message and compose the response. The AS checks whether the Mobile device possessed the right keys and indicates the end of the authentication process by sending an acknowledgement message.

As demonstrated in Fig 6.1, this version of the protocol might be vulnerable to security threats, which are mainly due to the fact that the derived keys are insecurely distributed to the participating entities. Therefore, the authors in (MK04), have assumed that the devices of the architecture are securely installed using mutual authentication and data integrity is maintained in the core network, i.e. between the AIS and the AS. By keeping these assumptions in mind, Casper/FDR tool was used to verify the protocol and find out whether it is still vulnerable to any attacks. A detailed analysis of the protocol is in the following sections.

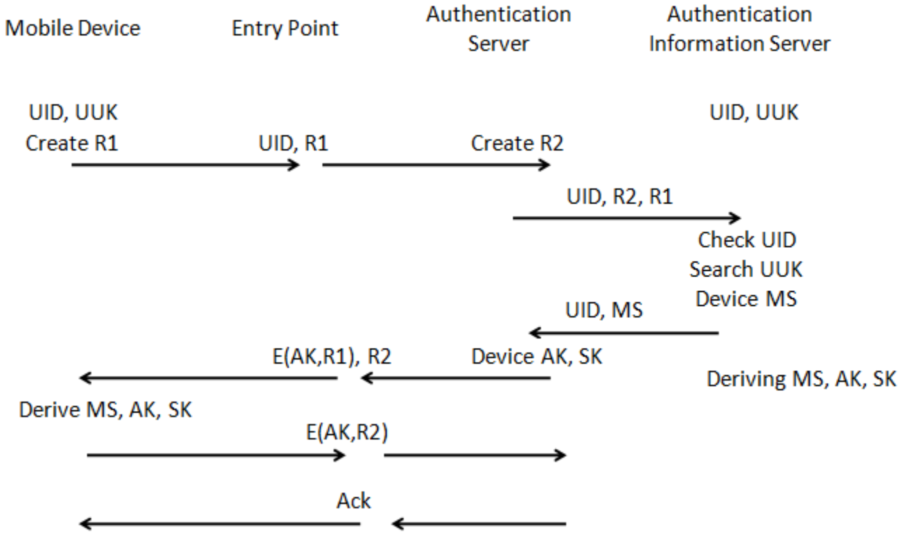

Figure 6.1: The Mobile Ethernet Protocol

6.4.2 The Formal Verification of the Mobile Ethernet Protocol

As shown in Fig 6.1, it is not clear how the Mobile device knows about the Entry Point. This knowledge could not be pre-configured as there is no way to predict which EP the mobile device will use. Similarly, there is a need to justify why the mobile device starts the protocol by sending the UID, R1 as the mobile device's first message. In order to simulate this interaction in Casper, we introduce the following preliminary messages: the Entry Points advertisement messages (Adv), The Access Request (AccReq) message, which is used by the Mobile device to indicate its intention to access the network. The Authentication Request (AuthReq) message is sent by the Entry point to trigger the authentication process. None of these messages play a security role; they are only used at the pre-authentication stage, where the entry points advertise their presence.

To formally verify the protocol, a Casper/FDR's input file was prepared. The full input file is given in the Appendix C. After compiling the Casper model and feeding the CSP output to FDR, no attacks against the secrecy of the AK and SK keys were found, this is due to the assumption that the Intruder does not know the key derivation functions of these keys despite the fact that the Master Secret Key (MS) is sent unprotected. However, attacks were found against the `Agreement( M, AS, [R2])` and `Aliveness (EP, M)`. We could find the traces for those attacks, which could be translated to the following attack sequence, where the notation I_M for instance represents the intruder taking the Mobile device's identity, either to fake a message (as in the second message 1) or to intercept a message intended for M (as in message 2).

```
0.  -> M : EP, AIS, AS
1a.  M -> I_EP : accReq
1b.  I_M -> EP : accReq
2a.  EP -> I_M : authReq
2b.  I_EP -> M : authReq
3.  M -> I_EP : M, R1
4.  I_EP -> AS : M, R1, h(M, R1)
5a.  AS -> I_AIS : M, R1, R2, h(M, R1, R2)
5b.  I_AS -> AIS : M, R1, R2, h(M, R1, R2)
6a.  AIS -> I_AS : MS, M, h(MS, M)
6b.  I_AIS-> AS : MS, M, h(MS, M)
7.  AS -> I_EP : R2, {R1}{AK}, h(R2, {R1}{AK})
8.  I_EP -> M : {R1}{AK}, R2
9.  M -> I_EP : {R2}{AK}
10.  I_EP -> AS : {R2}{AK}, h({R2}{AK})
11.  AS -> I_AIS : hoackm, h(hoackm)
12.  I_EP -> M : hoackm
```

In other words, this attack could be interpreted as follows: The Mobile device (M) thinks it has successfully completed a run of the protocol apparently with EP, while in reality it is with the Intruder, and EP has not previously been running the protocol. The steps of the attack are shown in 6.2.

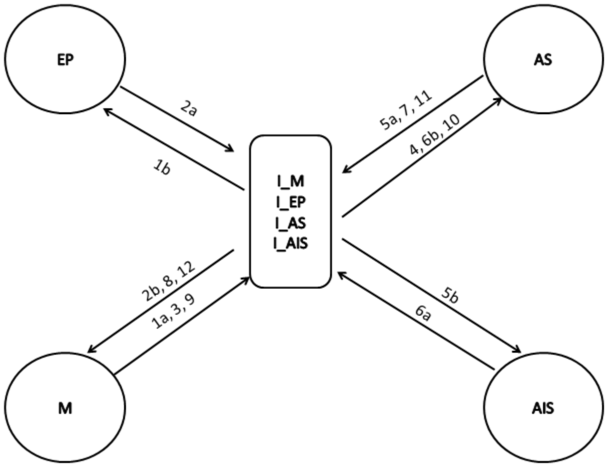

Figure 6.2: The Attack Against the Initial AKA Protocol of the Mobile Ethernet

6.4.3 Protocol Analysis and Security Consideration

In this section, we discuss how our formal modelling with Casper allows checking the security requirements described in 5.4.

- **Mutual Entity Authentication:** As stated in (AM96), entity authentication involves corroboration of a claimant's identity through actual communications with an associated verifier during execution of the protocol itself. Since the protocol does not consider verifying the identity of the participants and based on the discovered attack, we could claim that this protocol could not meet this feature.

- **Mutual Key Authentication:** the mutual authentication between the M and the AS is based on the secrecy of the AK. We got Casper to check this using the Secret (M, AK, [AS]) and Secret (AS, AK, [M]) assertion checks. Since no attack was found against the key secrecy, this property is met.

- **Mutual Key Confirmation:** Casper verifies this requirement by using the DECRYPTABLE (m, K) which checks if the message (m) is decryptable by the key (K). We performed a similar check after messages 8 and 10 as shown in the Protocol Description heading to verify that the valid Authentication key (AK) is possessed by the other party. If any of the checks fails the protocol aborts.

- **Key Freshness:** This property is guaranteed by including a fresh random value R1, R2 in the key derivation functions of the keys MS, AK and SK.

- **Unknown Key Share:** The afore-explained attack implies that the UKS requirement was not met. Despite of the fact that, the mobile device (M) and the AS share the Authentication Key (AK), the M mistakenly believes that the intruder holds this key as well. Casper/FDR indicates this fact by highlighting an attack against the Agreement and Aliveness assertions in the # Specifications header.

- **Key Compromise Impersonation Resilience:** This property could be modelled by specifying the long-term keys as crackable and then checking the Authenticity assertions. Casper verifies no breach against the authenticity feature

It is clear from the discussion above that, the initial AKA protocol failed to meet some security requirements, which are mainly related to the discovered authentication attack.
Although the protocol presumed the core network entities to be securely installed and the integrity of the exchanged messages to remain intact between the AIS and the AS, the fact that an attack could still be discovered could be due to the Intruder managing to intercept

the connections in the core network. This raises the issue of the need for providing a better security in the core network. Initially, the core network has been assumed to be physically secure, this assumption was valid in the closed, homogeneous environments, where the core network was controlled by a sole operator. However, this assumption does not hold in the case of future networks, where the core network represents an open, multi-operators environment. Additionally, there is a need to deal with identification-related attacks to meet the Mutual Entity Authentication property.

Furthermore, the process of deriving the keying materials in the Initial AKA protocol of (MK04) does not define the keys' usability scope. Therefore, there is a need to propose a more stable key hierarchy that specifies the scope of each derived keys.

6.5 The Proposed Solution

In order to address the previous security threats and to provide a better security in the core network, a novel initial AKA protocol is introduced in this section. However, instead of making assumptions of a secure core network, we need to define the part of the core network to be protected and the type of security mechanism. Therefore, as explained in (MA11b), in order to design the proposed protocol, a progressive design approach has been followed; in the initial version of the protocol, no security was considered in the core network, modelling the proposal found secrecy and authenticity attacks, analysing the discovered attacks will highlight the main source of threats. The second version simulated the case of a secure channel only between the CA3C and the DA3Cs, the discovered attacks in this draft highlight the need to secure different parts of the core network. In the third version, secure channels have been presumed between the DA3C and the Auth. After simulating this case using CSP, Casper failed to find any attacks. This could be ascribed due to the assumption that the Intruder does not know the KDFs to generate the Secret the Authentication keys (SK, AK). However, by adding this knowledge to the intruder capabilities, two attacks were discovered. These attacks highlight the need to secure all the communication between the entities (Auth, DA3C, CA3C) in the core network. Therefore, the last version of the protocol considers pre-established secure channels among these entities and simulates the situation.

6.5.1 Defining the Security System

For the proposed protocol to be practical, we consider the open network structure in Fig 6.3 which has been presented in Chapter 4.

It is crucial to show the actual parties participating in the protocol and thus, how the

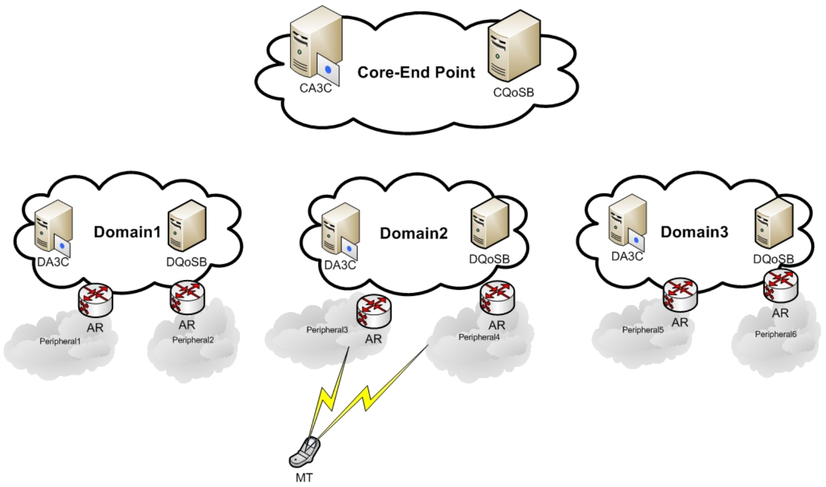

Figure 6.3: The Hierarchical Network Structure

proposed protocol could be mapped to actual entities in the network. The system comprises four entities: the Mobile Terminal (MT), the Authenticator (Auth) which runs on the Access Router (AR); the Domain A3C server (DA3C) which is responsible for authenticating and authorizing the MT to use the network and the Core-End Point which hosts the central A3C server (CA3C).

6.5.2 The Key Hierarchy

As shown in Fig 6.4, the security materials comprise a top level Unique Key uk(MT), which is pre-shared between the MT and the CA3C server. Similar to the Ki key in GSM (Sch03) , the uk(MT) is stored into the MT's SIM card and is never used for encryption purposes, rather it is only used for deriving further security keys. The second level key is the Domain Specific Master Key (DSMK), as the name implies, this key is unique at the domain level and is derived using an irreversible function F1 as follows: DSMK= F1(uk(MT), seq1, Auth_Domain_Name), where seq1 is a fresh sequence number, the Auth_Domain _Name is the corresponding domain name. Since each domain might have more than one Authenticator, the MT could join the domain via any of its Auths, thus, a different Secret Key (SK) has to be used for each Authenticator. One Authentication Key (AK) is used for mutual authentication between the MT and the network. Similar to F1, two irreversible function F2 and F3 are used to derive AK and SK as follows: AK = F2 (seq1, DSMK), SK = F3(seq1, AuthID, DSMK), where AuthID is the ID of the Auth and is broadcasted by the Auth in the form of AuthID@DomainName.

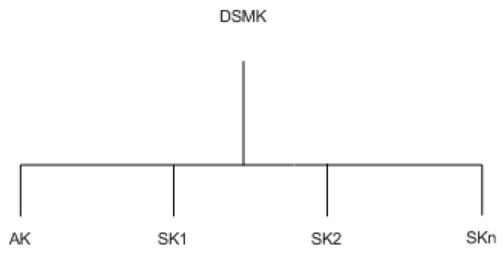

Figure 6.4: The Key Hierarchy

6.5.3 The Initial Version Of the Protocol

In the initial version, the protocol is proposed without considering security in the core network, this will help to highlight the potential threats.

The protocol Description

By considering the notation in Table 6.2, the AKA protocol is explained as follows:
After starting the mobile device, the MT picks the access routers' advertisements (Adv) which contain information about the access network such as the AuthID and the domain name. The MT uses this information to generate a Domain-Specific Master Key (DSMK).

Phase 1
$Msg1 : Auth \rightarrow MT : Adv$
Generate the DSMK= F1(uk(MT), seq1, AuthID)

The protocol starts when the MT sends a joining message Msg 2 to the Auth. The Auth responds by sending authentication request AuthReq as Msg 3.

Phase 2
$Msg2.MT \rightarrow Auth : AccReq$
$Msg3.Auth \rightarrow MT : AuthReq$

By using the DSMK, the MT derives the Authentication Key (AK) and composes Msg 4, this message consists of a fresh sequence number seq1 used as a challenge, Authentication ID (AuthID), the Mobile Terminal identity (MT) , and a set Initauth flag (InitAuth=1). The Auth passes this message to the DA3C and from there to the CA3C as Msg 5 and Msg 6. Using the included mobile ID, the CA3C looks up the corresponding uk(MT) and uses it to generate a fresh Domain Specific Master key DSMK.

Phase 3

Generate the AK = F2(seq1, DSMK)
$Msg4.MT \rightarrow Auth : MT, seq1, AuthID, Initauth$
$Msg5.Auth \rightarrow DA3C : MT, seq1, AuthID, Initauth$
$Msg6.DA3C \rightarrow CA3C : MT, seq1, AuthID,$
$Initauth$
Generate the DSMK= F1(uk(MT), seq1, AuthID)

The DSMK key is included in Msg 7. Using the information in this message, the DA3C generates the Authentication Key (AK) and returns the previously sent sequence Seq1 and a new sequence Seq2 all the way to the MT as Msg 8 and Msg 9. These messages are encrypted using the derived AK. Since the MT has the required information to derive all the keys (DSMK, SK, AK), the MT verifies the contents of Msg 9 and derives the Secret Key SK.

Phase 4

$Msg7.CA3C \rightarrow DA3C : DSMK, seq1, AuthID, MT,$
$Initauth$
Generate the AK = F2(seq1, DSMK)
$Msg8.DA3C \rightarrow Auth : \{seq1, seq2\}_{AK}$
$Msg9.Auth \rightarrow MT : \{seq1, seq2\}_{AK}$
Verify the message contents, then derive the
SK:= F3(seq1, DSMK, AuthID)

The MT returns Seq2 all the way to the DA3C as Msg 10 and Msg 11. The DA3C verifies the contents of Msg 11 and derives the Secret Key SK.

Phase 5

$Msg10.MT \rightarrow Auth : \{seq2\}_{AK}$
$Msg11.Auth \rightarrow DA3C : \{seq2\}_{AK}$
Verify the message contents, then derive the SK:= F3(seq1, DSMK, AuthID)

Upon verifying the Msg 11, the DA3C authenticates the MT and acknowledges this to the CA3C, and then generates the Secret Key (SK) and passes it to the Auth in Msgs12, 13. Using the SK, the Auth sends an encrypted access response message to the MT as Msg 14.

Table 6.2: Notation

Notation	The Description
MT	The Mobile Terminal
Auth	Is the Access Router in the peripheral network
AuthID	The Authenticator unique ID has the format AuthID@domainname
CA3C	Core-endpoint entity, which has QoS and Security related responsibilities
se1(DA3C)	Pre-shared secret key between the CA3C and the DA3C
se2(Auth)	Pre-shared secret key between the DA3C and the Authenticator (Auth)
uk(MT)	Unique secret key shared between the CA3C and the MT
DSMK	Domain specific- Master Key DSMK= F1 (uk(MT), seq1, Auth-domain name)
AK	Authentication key AK= F2 (seq1, DSMK)
SK	Secret Key SK = F3 (Seq1, AuthID, DSMK), used to encrypt all the messages between the MT and the network
F1, F2, F3	Irreversible Key Derivation Functions
InitAuth flag	A flag set only in the initial authentication. In case of handover, this flag will not be set
HoAckm	Joining/Handover Acknowledgement message used by the DA3C server to inform the CA3C in the CEP about a successful authentication
seq1, seq2	Sequence numbers
$\{m\}_K$	Encrypting the message (m) using the key (K)

Phase 6

$Msg12.DA3C \rightarrow CA3C : HoAckm$

$Msg13.DA3C \rightarrow Auth : SK$

$Msg14.Auth \rightarrow MT : \{AccRes\}_{SK}$

Formal Verification

A Casper description of the protocol was prepared. However, since this is an initial version of the protocol, only the #Specfications heading is mentioned here. A complete description of the final and completely refined version of the protocol are included in Appendix D.

# Specification

```
Secret(MT,AK,[DA3C])
Secret(DA3C,AK,[MT])
Secret(MT,SK,[DA3C, Auth])
Agreement( MT, DA3C, [seq2])
Agreement(DA3C, MT, [AK])
WeakAgreement(MT, Auth)
WeakAgreement(Auth,MT)
WeakAgreement(Auth, DA3C)
WeakAgreement(DA3C, Auth)
Aliveness (Auth, MT)
Aliveness (MT, Auth)
```

After modelling the protocol using Casper and checking the corresponding CSP code using FDR checker, the following attacks were discovered. The first attack is against the Secret(MT, SK, [Auth, DA3C] assertion, where the Intruder launches a replay attack and eventually manages to get the secret key (SK). The second attack is against authenticity specification Agreement(DA3C, MT, [AK]), in which the intruder replays messages between the different parties and manages to impersonates the DA3C to the MT, thus, the MT mistakenly believes it has completed a run of the protocol, with the DA3C, using data items AK. The third attack is against the WeakAgreement(Auth, DA3C) assertion. In this attack, the DA3C mistakenly believes it has successfully completed a run of the protocol with the Auth. However, in reality it was running the protocol with Intruder. These attacks were described in (MA11b).

The first attack is against the Secret(MT, SK, [Auth, DA3C] assertion, where the Intruder launches a replay attack and eventually manages to get the secret key (SK). The message sequence involved in the attack is given below.

```
0.  -> mt :  auth, ca3c
1a.  auth -> I_mt :  adv, da3c
1b.  I_auth -> mt :  adv, da3c
2a.  mt -> I_auth :  accReq
2b.  I_mt -> auth :  accReq
3a.  auth -> I_mt :  authReq
3b.  I_auth -> mt :  authReq
4a.  mt -> I_auth :  SEQ1, Authid, mt, InitAuth
5a.  I_auth -> da3c :  SEQ1, Authid,Mallory, InitAuth
4b.  I_mt -> auth :  SEQ1, Authid, mt, InitAuth
6a.  I_da3c -> ca3c :  SEQ1, Authid, mt, InitAuth
```

```
7a.  ca3c -> I_da3c :  DSMK,SEQ1, Authid, mt, InitAuth
5b.  auth -> I_da3c :  SEQ1, Authid, mt, InitAuth
6b.  da3c -> I_ca3c :  SEQ1, Authid, Mallory, InitAuth
7b.  I_ca3c -> da3c :  DSMK, SEQ1, Authid, Mallory, InitAuth
8a.  da3c -> I_auth :  {SEQ2, SEQ1}{AK}
9a.  I_auth -> mt :  {SEQ2, SEQ1}{AK}
10a.  mt -> I_auth :  {SEQ2}{AK}
11a.  I_auth -> da3c :  {SEQ2}{AK}
12.  da3c -> I_ca3c :  hoackm
8b.  I_da3c -> auth :  {SEQ2, SEQ1}{AK}
9b.  auth -> I_mt :  {SEQ2, SEQ1}{AK}
10b.  I_mt -> auth :  Garbage
13a.  da3c -> I_auth :  SK
11b.  auth -> I_da3c :  Garbage
13b.  I_da3c -> auth :  SK
14a.  auth -> I_mt :  {accRes}{SK}
14b.  I_auth -> mt :  {accRes}{SK}
The intruder knows SK
```

As shown in Fig 6.5, the attack could be explained as follows:

1. Initially the intruder intercepts and passively replays the messages between the MT and the Auth as in messages 1a, 1b, 2a, 2b, 3a and 3b. Once the intruder intercepts message 4a from the MT, it starts three sessions as follows:

   (a) The Intruder acts as the Auth and actively composes message 5a, which is a replay of a fake message 4a by replacing the MT with the intruder's identity (Mallory). The Intruder will then intercept and block the response from the DA3C towards the CA3C as in message 6b. Thus, the CA3C will not be able to discover the fake message.

   (b) The Intruder pretends to be the MT and passively replays message 4a towards the Auth as message 4b. Since the Auth is expecting a response to message 3a, the goal of message 4b is to make the Auth believe that he is still running the protocol.

   (c) Acting as the DA3C, the intruder replays message 4a towards the CA3C. As a result of this message, the intruder will get the DSMK from the CA3C as in message 7a

2. The intruder uses the information from messages 7a and 6b to compose a fake message 7b towards the DA3C. Since this message holds the intruder identity and the DSMK, the DA3C will believe the included identity was verified by the CA3C, and thus will generate the other keys; (AK),(SK).

3. After intercepting message 8a, the intruder starts two sessions:

   (a) The intruder pretends to be the Auth and intercepts messages 9a, 10a, 11a. By verifying message 11a, the DA3C will authenticate the intruder believing it is the MT. The intruder knows of the successful attack by intercepting the Acknowledgement message from the DA3C as in message 12.

   (b) Since the Auth is expecting a response to message 5a, the intruder replays message 8a (it was originally intended to the Auth) towards the Auth, so the Auth believes he is still running the protocol as in messages 9b, 10b and 11b.

4. Once the intruder intercepts the SK in message 13a, this means the secrecy attack was successfully completed and it is time to finish the protocol. Therefore, the intruder passes the SK to the Auth, so the Auth could send the AccRes as in messages 14a and 15a.

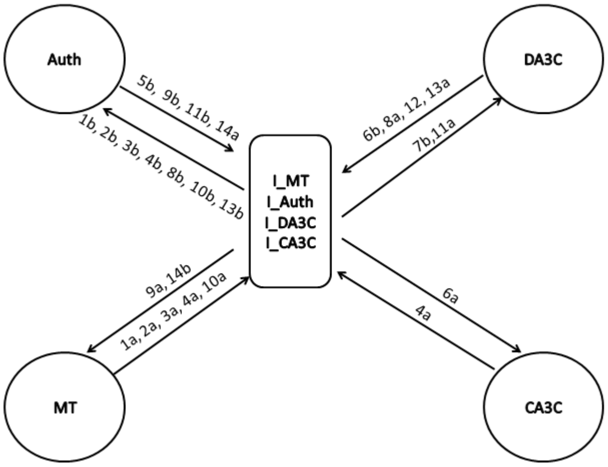

Figure 6.5: The Attack Against the `Secret(MT, SK, [Auth, DA3C]`

The second attack is against authenticity specification `Agreement(DA3C, MT, [AK])`. In which, the intruder replays messages between the different parties and manages to impersonates the DA3C to the MT, thus, the MT mistakenly believes it has completed a run of the protocol, with the DA3C, using data items AK. The message sequence involved in the attack is shown below.

```
0.  -> mt :  auth, ca3c
1a.  auth -> I_mt :  adv, da3c
1b.  I_auth -> mt :  adv, da3c
2a.  mt -> I_auth :  accReq
2b.  I_mt -> auth :  accReq
3a.  auth -> I_mt :  authReq
3b.  I_auth -> mt :  authReq
4a.  mt -> I_auth :  SEQ1, Authid, mt, InitAuth
5a.  I_auth -> da3c :  SEQ1, Authid, Mallory, InitAuth
4b.  I_mt -> auth :  SEQ1, Authid, mt, InitAuth
6a.  I_da3c -> ca3c :  SEQ1, Authid, mt, InitAuth
7a.  ca3c -> I_da3c :  DSMK, SEQ1, Authid, mt, InitAuth
5b.  auth -> I_da3c :  SEQ1, Authid, mt, InitAuth
6b.  da3c -> I_ca3c :  SEQ1, Authid, Mallory, InitAuth
8a.  I_da3c -> auth :  Garbage
7b.  I_ca3c -> da3c :  DSMK, SEQ1, Authid, Mallory, InitAuth
8b.  da3c -> I_auth :  {SEQ2, SEQ1}{AK}
9a.  I_auth -> mt :  {SEQ2, SEQ1}{AK}
10a.  mt -> I_auth :  {SEQ2}{AK}
9b.  auth -> I_mt :  Garbage
10b.  I_mt -> auth :  {SEQ2}{AK}
11a.  I_auth -> da3c :  {SEQ2}{AK}
11b.  auth -> I_da3c :  {SEQ2}{AK}
12.  da3c -> I_ca3c :  hoackm
da3c believes (s)he is running the protocol,
taking role DomainSERVER, with Mallory, using
data items AK
13.  I_da3c -> auth :  SK
14.  auth -> I_mt :  {accRes}{SK}
mt believes (s)he has completed a run of the
protocol, taking role INITIATOR, with da3c,
using data items AK
```

The first four steps of the attack are same as the previous one, the remainder steps go as follows:

1. The Intruder acts as DA3C and send a 'Garbage' message to the Auth as message 8a, the Auth considers this as a response to 5b and thus he thinks that, he is still running the protocol. Pretending to be the CA3C, the intruder uses the information in messages 7a and 6b to replace the MT identity in message 7a with it's identity (Mallory) and sends message 7b to the DA3C. Upon receiving this message, the DA3C generates the AK, SK and sends message 8a toward the Auth. This message will be intercepted by the intruder, which will pretend to be Auth and exchange messages 8a,9a,10a with the DA3C and the MT.

2. The Auth passes the 'Garbage' message towards the MT. However, the intruder blocks this message and responds by replaying 10a towards the the Auth so the Auth believes that he is still part of the protocol. The intruder pretends to be the Auth and replays message 10a towards the DA3C as in message 11a. In order not to receive a duplicated message, the intruder blocks the message sent by the Auth towards the DA3C as message 11b.

3. Upon verifying message 11a and mistakenly believing it is running the protocol with the MT, the DA3C acknowledges the successful authentication to the CA3C by sending message 12, which will be blocked by the intruder. Similarly, to the previous attack, when the intruder has managed to get the SK, the intruder prompts the Auth to send the AccRes message by sending the SK in message 13 towards the Auth. Thus, the MT mistakenly believes it has been running the protocol with DA3C, while in reality it has been running it with the intruder (Mallory)

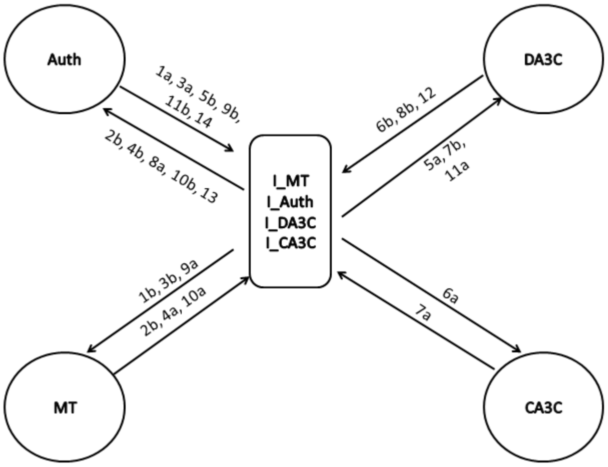

Figure 6.6: The Attack Against the `Agreement(DA3C, MT, [AK])`

The third attack, described below, is against the `WeakAgreement(Auth, DA3C)` assertion. In this attack, The DA3C mistakenly believes it has successfully completed a run of the protocol with the Auth. However, in reality it was running the protocol with Intruder. For completeness, the message sequence involved in the attack is given below.

```
0.  -> mt :  auth, ca3c
1a.  auth -> I_mt :  adv, da3c
1b.  I_auth -> mt :  adv, da3c
2a.  mt -> I_auth :  accReq
2b.  I_mt -> auth :  accReq
3a.  auth -> I_mt :  authReq
3b.  I_auth -> mt :  authReq
4.  mt -> I_auth :  SEQ1, Authid, mt, InitAuth
5.  I_auth -> da3c :  SEQ1, Authid, mt, InitAuth
6a.  I_da3c -> ca3c :  SEQ1, Authid, mt, InitAuth
6b.  da3c -> I_ca3c :  SEQ1, Authid, mt, InitAuth
7a.  ca3c -> I_da3c :  DSMK, SEQ1, Authid, mt, InitAuth
7b.  I_ca3c -> da3c :  DSMK, SEQ1, Authid, mt, InitAuth
8.  da3c -> I_auth :  {SEQ2, SEQ1}{AK}
9.  I_auth -> mt :  {SEQ2, SEQ1}{AK}
10.  mt -> I_auth :  {SEQ2}{AK}
11.  I_auth -> da3c :  {SEQ2}{AK}
12.  da3c -> I_ca3c :  hoackm
13.  da3c -> I_auth :  SK
```

Once again, these attacks could be ascribed to the fact that the Intruder managed to intercept and replay the connections between the parties in the core network.

6.5.4 The Second Version of the Protocol

As an attempt to secure the core network, we propose the presence of a certain trust relationship between the network's entities and thus secure channels have already been established between the CA3C and the DA3Cs. Such secure channels could be guaranteed by using different mechanisms such as IP security (IPSec) (SK98) or any other Virtual Private Network (VPN) protocols. Alternatively, this could be achieved using out-of-band approach such as agreeing on security materials among the multiple operators.

Formal Verification

In order to simulate this secure connection between the CA3C and the DA3C using Casper/FDR, a secret key se1(DA3C) is presumed to be pre-shared between these entities. Casper/FDR found attacks against the secrecy of the SK defined by the `Secret(MT, SK, [Auth, DA3C]` assertion, as the intruder replays the messages between the parties and impersonates the Auth to the DA3C to intercept the (Auth-DA3C) connection in Msg 13. Also, an authentication attack against the `WeakAgreement(Auth, DA3C)` assertion was discovered. It explains how the intruder intercepts and replays the messages between different participants and once it gets message 8, it impersonates the Auth so the DA3C mistakenly believes that, he has been running the protocol with the Auth while in reality it was with the intruder. These attacks were analysed in (MA11b).

The first attack below is against the secrecy of the SK defined by the `Secret(MT, SK, [Auth, DA3C]` Assertion, as the intruder replays the messages between the parties and impersonate the Auth to the DA3C to intercept the (Auth-DA3C) connection in Msg 13.

```
1a.  auth -> I_mt :  adv, da3c
1b.  I_auth -> mt :  adv, da3c
2a.  mt -> I_auth :  accReq
2b.  I_mt -> auth :  accReq
3a.  auth -> I_mt :  authReq
3b.  I_auth -> mt :  authReq
4a.  mt -> I_auth :  SEQ1, Authid, mt, InitAuth
5a.  I_auth -> da3c :  SEQ1, Authid, mt, InitAuth
6a.  da3c -> I_ca3c :  {SEQ1, Authid, mt, InitAuth}{se1(da3c)}
6b.  I_da3c -> ca3c :  {SEQ1, Authid, mt, InitAuth}{se1(da3c)}
7a.  ca3c -> I_da3c :  {DSMK, SEQ1, Authid, mt, InitAuth}{se1(da3c)}
7b.  I_ca3c -> da3c :  {DSMK, SEQ1, Authid, mt, InitAuth}{se1(da3c)}
4b.  I_mt -> auth :  SEQ1, Authid, mt, InitAuth
5b.  auth -> I_da3c :  SEQ1, Authid, mt, InitAuth
8a.  da3c -> I_auth :  {SEQ2, SEQ1}{AK}
9a.  I_auth -> mt :  {SEQ2, SEQ1}{AK}
8b.  I_da3c -> auth :  Garbage
10a.  mt -> I_auth :  {SEQ2}{AK}
9b.  auth -> I_mt :  Garbage
11a.  I_auth -> da3c :  {SEQ2}{AK}
10b.  I_mt -> auth :  {SEQ2}{AK}
12.  da3c -> I_ca3c :  {hoackm}{se1(da3c)}
```

```
13a.  da3c -> I_auth :  SK
11b.  auth -> I_da3c :  {SEQ2}{AK}
13b.  I_da3c -> auth :  SK
14a.  auth -> I_mt :  {accRes}{SK}
14b.  I_auth -> mt :  {accRes}{SK}
The intruder knows SK
```

The discovered attack highlights that, there is a need to protect other parts of the core network.

The second attack, below is an authentication attack against the `WeakAgreement(Auth, DA3C)` assertion. It explains how the intruder intercepts and replay the messages between different participants and once it gets message 8, it impersonates the Auth so the DA3C mistakenly believes that, he has been running the protocol with the Auth while in reality it was with the intruder. These attacks are similar to the previous three attacks against the initial version of the protocol in Section 6.5.3.

```
1a.  auth -> I_mt :  adv, da3c
1b.  I_auth -> mt :  adv, da3c
2a.  mt -> I_auth :  accReq
2b.  I_mt -> auth :  accReq
3a.  auth -> I_mt :  authReq
3b.  I_auth -> mt :  authReq
4.  mt -> I_auth :  SEQ1, Authid, mt, InitAuth
5.  I_auth -> da3c :  SEQ1, Authid, mt, InitAuth
6a.  da3c -> I_ca3c :  {SEQ1, Authid, mt, InitAuth}{se1(da3c)}
6b.  I_da3c -> ca3c :  {SEQ1, Authid, mt, InitAuth}{se1(da3c)}
7a.  ca3c -> I_da3c :  {DSMK, SEQ1, Authid, mt, InitAuth}{se1(da3c)}
7b.  I_ca3c -> da3c :  {DSMK, SEQ1, Authid, mt, InitAuth}{se1(da3c)}
8.  da3c -> I_auth :  {SEQ2, SEQ1}{AK}
9.  I_auth -> mt :  {SEQ2, SEQ1}{AK}
10.  mt -> I_auth :  {SEQ2}{AK}
11.  I_auth -> da3c :  {SEQ2}{AK}
12.  da3c -> I_ca3c :{hoackm}{se1(da3c)}
13.  da3c -> I_auth :  SK
```

6.5.5 The Third Version of the Protocol

In the final scenario, secure channels exist only between the DA3C and the Auth. To simulate these channels a secret key se2(Auth) is pre-shared between the Auth and the DA3C. After preparing the Casper's input file and asking Casper/FDR to verify the protocol, Casper found no attacks against the assertions in the # Specifications heading.

This result could be mainly ascribed to the fact that disclosing the connection between the CA3C and the DA3Cs will lead to exposing the DSMK key. However, this key does not have any security rule rather than acting as a primitive information, used to derive the AK and the SKs. In that sense, even if the attacker managed to get the DSMK, he will not be able to derive further keys, and thus could not launch secrecy or authentication attacks.

6.5.6 AKA Protocol Formal Verification and Security Analysis

Similar to the assumption of the AKA protocol of Mobile Ethernet in Section 6.4 that the Key Derivation Function (KDFs) are not known to the Intruder, the three versions of the proposed protocol adopted this assumption. Therefore, the key derivation functions of the AK and SK keys, F2 and F3 respectively were not included in the Intruder initial knowledge. By adding the (F2, F3) to the attacker knowledge, as stated in the #Interuder Information heading below, new attacks against the secrecy of the SK, AK are discovered.

# Intruder Information

```
Intruder = Mallory
IntruderKnowledge = {mt, da3c, Mallory, ca3c,
Authid, auth, uk(Mallory), F2, F3 }
Crackable = PresharedKeys
Crackable = Domainspecifickey
```

The first discovered attack is due to the exposure of the SK key and it goes as follows:

```
1a.  auth -> I_mt :  adv, da3c
1b.  I_auth -> mt :  adv, da3c
2a.  mt -> I_auth :  accReq
2b.  I_mt -> auth :  accReq
3a.  auth -> I_mt :  authReq
3b.  I_auth -> mt :  authReq
4a.  mt -> I_auth :  SEQ1, Authid, mt, InitAuth
4b.  I_mt -> auth :  SEQ1, Authid, mt, InitAuth
5a.  auth -> I_da3c :  {SEQ1, Authid, mt, InitAuth}{se2(auth)}
```

```
5b.  I_auth -> da3c :  {SEQ1, Authid, mt, InitAuth}{se2(auth)}
6a.  da3c -> I_ca3c :  SEQ1, Authid, mt, InitAuth
6b.  I_da3c -> ca3c :  SEQ1, Authid, mt, InitAuth
7a.  ca3c -> I_da3c :  DSMK, SEQ1, Authid, mt,
InitAuth
7b.  I_ca3c -> da3c :  DSMK, SEQ1, Authid, mt,
InitAuth
8a.  da3c -> I_auth :  {{SEQ2, SEQ1}{AK}}
{se2(auth)}
8b.  I_da3c -> auth :  {{SEQ2, SEQ1}{AK}}
{se2(auth)}
9a.  auth -> I_mt :  {SEQ2, SEQ1}{AK}
9b.  I_auth -> mt :  {SEQ2, SEQ1}{AK}
10a.  mt -> I_auth :  {SEQ2}{AK}
10b.  I_mt -> auth :  {SEQ2}{AK}
11a.  auth -> I_da3c :  {{SEQ2}{AK}
11b.  I_auth -> da3c :  {{SEQ2}{AK}
12.  da3c -> I_ca3c :  hoackm
13a.  da3c -> I_auth :  {SK}{se2(auth)}
13b.  I_da3c -> auth :  {SK}{se2(auth)}
14a.  auth -> I_mt :  {accRes}{SK, Authid)}
14b.  I_auth -> mt :  {accRes}{SK, Authid)}
The intruder knows SK
```

The second discovered attack is due to the disclosure of the Authentication Key (AK) and it goes as follows:

```
0.  -> mt :  auth, ca3c
1a.  auth -> I_mt :  adv, da3c
1b.  I_auth -> mt :  adv, da3c
2a.  mt -> I_auth :  accReq
2b.  I_mt -> auth :  accReq
3a.  auth -> I_mt :  authReq
3b.  I_auth -> mt :  authReq
4a.  mt -> I_auth :  SEQ1, Authid, mt,
InitAuth
4b.  I_mt -> auth :  SEQ1, Authid, mt,
InitAuth
5a.  auth -> I_da3c :  {SEQ1, Authid, mt,
InitAuth}{se2(auth)}
5b.  I_auth -> da3c :  {SEQ1, Authid, mt,
InitAuth}{se2(auth)}
6a.  da3c -> I_ca3c :  SEQ1, Authid, mt,
InitAuth
6b.  I_da3c -> ca3c :  SEQ1, Authid, mt,
InitAuth
7a.  ca3c -> I_da3c :  DSMK, SEQ1, Authid, mt,
InitAuth
7b.  I_ca3c -> da3c :  DSMK, SEQ1, Authid, mt,
InitAuth
8a.  da3c -> I_auth :  {{SEQ2, SEQ1}{AK}}
{se2(auth)}
8b.  I_da3c -> auth :  {{SEQ2, SEQ1}{AK}}
{se2(auth)}
9.  auth -> I_mt :  {SEQ2, SEQ1}{AK}
10.  I_mt -> auth :  {SEQ2}{AK}
11a.  auth -> I_da3c :  {{SEQ2}{AK}}{se2(auth)}
11.  I_auth -> da3c :  {{SEQ2}{AK}}{se2(auth)}
12.  da3c -> I_ca3c :  hoackm
13.  da3c -> I_auth :  {SK}{se2(auth)}
The intruder knows AK
```

The discovered attacks are ascribed to the fact that the main parameters to derive the Authentication and Secret keys, such as the DSMK, seq1 and AuthID, were sent unprotected between the CA3C and the DA3Cs and thus the intruder manages to derive the keys. These attacks highlight the need to have secure connections between the CA3C and DA3Cs. To simulate these connections a secret key se1(DA3C) is presumed to be shared between the CA3C and the DA3C, and thus the final version of the protocol is as follows:

Phase 1

$Msg1.DesAuth \rightarrow MT : Adv$

Generate the DSMK= F1(uk(MT), seq1, AuthID)

Phase 2

$Msg2.MT \rightarrow Auth : AccReq$

$Msg3.Auth \rightarrow MT : AuthReq$

Generate the AK = F2(seq1, DSMK)

Phase 3

$Msg4.MT \rightarrow Auth : MT, seq1, AuthID, Initauth$

$Msg5.Auth \rightarrow DA3C : \{MT, seq1, AuthID,$

$Initauth\}_{se2(Auth)}$

$Msg6.DA3C \rightarrow CA3C : \{MT, seq1, AuthID,$

$Initauth\}_{se1(DA3C)}$

Generate the DSMK= F1(uk(MT), seq1, AuthID)

Phase 4

$Msg7.CA3C \rightarrow DA3C : \{DSMK, seq1, AuthID, MT,$

$Initauth\}_{se1(DA3C)}$

Generate the AK = F2(seq1, DSMK)

$Msg8.DA3C \rightarrow Auth : \{\{seq1, seq2\}_{AK}\}_{se2(Auth)}$

$Msg9.Auth \rightarrow MT : \{seq1, seq2\}_{AK}$

Verify the message contents, then derive the

SK:= F3(seq1, DSMK, AuthID)

Phase 5

$Msg10.MT \rightarrow Auth : \{seq2\}_{AK}$

$Msg11.Auth \rightarrow DA3C : \{\{seq2\}_{AK}\}_{se2(Auth)}$

Verify the message contents, then derive the

SK:= F3(seq1, DSMK, AuthID)

Phase 6

$Msg12.DA3C \rightarrow CA3C : \{HoAckm\}_{se1(DA3C)}$

$Msg13.DA3C \rightarrow Auth : \{SK\}_{se2(Auth)}$

$Msg14.Auth \rightarrow MT : \{AccRes\}_{SK}$

The complete Casper description of the final version of the protocol is included in Appendix D.

6.5.7 AKA Protocol Formal Verification

The main goal of the proposed protocol is to achieve mutual authentication between the MT and the core network, thus authenticating the MT to use the peripheral network. To model the AKA protocol using Casper/FDR tool, we prepared a Casper input file that represents the system. The complete description of the protocol is found in Appendix D.
In this section, we discuss how our formal modelling with Casper allows the checking of the typical security requirements for AKA security protocols.

- **Mutual Entity Authentication:** Casper provides no direct specification to model this property. In order to show how our protocol could meet this requirement, we explicitly, and by considering the protocol transactions, could argue that this requirement could be met to a certain extent in our protocol. When making the initial contract, the MT and the CA3C share a unique key uk(MT), which acts as the root in the key hierarchy and is never used for encryption. We assume this key has been derived by running a key derivation function over identity-related information of the MT and the CA3C, and since it is never exposed and is stored in the MT's SIM card, it is unlikely for an intruder to get that key; thus, possessing this key verifies the identity of the party.

- **Mutual Key Authentication:** the mutual authentication between the MT and the DA3C is based on the secrecy of the AK. We got Casper to check this using the Secret (MT, AK, [DA3C]) and Secret (DA3C, AK, [MT]) assertion checks.

- **Mutual Key Confirmation:** Casper verifies this requirement by using the DECRYPTABLE (m, K) which checks if the message (m) is decryptable by the key (K). We performed a similar check after messages 9 and 11 as shown in the Protocol Description heading to verify that the valid Authentication key (AK) is possessed by the other party. If any of the checks fail, the protocol aborts.

- **Key Freshness:** since there is no direct function with Casper to simulate this feature, we included a freshly generated sequence seq1 in the key derivation function as explained in the key derivation subsection; thus the fact that Casper does not detect any attack on the secrecy of the secret and authentication keys (SK) and (AK) respectively implies that key freshness is not violated.

- **Unknown-Key Share:** we check this property using the Aliveness assertions. Additionally, we could address this attack by making a binding between the keys and the identity of the parties. The proposed AKA protocol has achieved this by the identity

Table 6.3: Comparison

The Security Property	The AKA of Mobile Ethernet	Initial Proposal	Refined Proposal
Mutual Entity Authentication	No	No	Yes
Mutual Key Authentication/ Keys' Secrecy	Yes	No	Yes
Mutual Key Confirmation	Yes	No	Yes
Key Freshness	Yes	Yes	Yes
Unknown-Key Share	No	No	Yes
Key Compromise Impersonation Resilience	Yes	Yes	Yes
Defining Key Scope	No	Yes	Yes

of the MT and the CA3C in the derivation of the uk(MT). Also, the authenticator's ID and the domain name are included in the key derivation functions of the SK and AK keys.

- **Key Compromise Impersonation Resilience:** this property could be modelled by specifying the long-term keys as crackable and then checking the Authenticity assertions. Casper verifies no breach against the authenticity feature.

The parameters, used as an input to derive the keys help in determining the scope of the key. For instance, the DSMK = F1(uk(MT), seq1, Auth-domain name) is unique at the domain level (since it includes the domain name) and fresh per user (it has uk(MT) and a fresh seq1). A similar discussion goes for the Authentication Key AK= F2(seq1, DSMK), which is unique and fresh per-domain. The SK= F3(seq1, AuthID, DSMK) is slightly different; as the key derivation function includes the Authenticator' ID as an input, therefore, this key is unique and fresh per authenticator. This implies that, in case of inter-domain or Vertical Handover new AK and SKs are derived while, only new SKs are derived in case of intra-domain or Horizontal Handover.

6.5.8 Protocol Analysis and Security Consideration

Table 6.3 shows a summary of the results, it compares the results between the Mobile Ethernet's AKA protocol, the first proposed solution (no security in the core network) and the final proposal (with the security in the core network).

6.6 Secure Vertical Handover in Heterogeneous Environment

The open architecture of Next Generation Networks (NGNs) implies that, to maintain ubiquity, future mobile devices need to roam between different networks using vertical handover techniques. When a mobile user moves into a new foreign network, data confidentiality and mutual authentication between the user and the network are vital issues in this heterogeneous environment.

6.6.1 Related Work

This section discusses some related work, that have been trying to provide AKA protocols to secure vertical handover mechanisms in heterogeneous networks.

The Handover Key Working Group (HOKEY WG)

In an attempt to secure vertical handover, the HOKEY WG proposed the EAP Re-authentication Protocol (ERP) (VN08). However, the ERP protocol has many drawbacks; firstly, ERP is based on the EAP platform, this implies that, all the network entities such as the Access Routers have to be updated or replaced to support this platform. Secondly, the poor confidentiality of the discovery messages at the beginning of the protocol, and thirdly, the lack of formal verification of the protocol. More detailed analysis of the ERP protocol could be found in chapter 2.

The 3rd Generation Partnership Project (3GPP)

The 3GPP project has introduced two scenarios; the 3GPP-WLAN interworking, which is introduced in Release 6 of 3GPP specifications (3GP06) and 3GPP-WiMAX interworking architectures as examples of heterogeneous environments. Both scenarios presume the presence of 3GPP technology in the core network, while WLAN or WiMax technologies are in the peripheral networks.

In the case of WiMAX to WLAN Vertical Handover, the mobile terminal (MT) invokes EAP-AKA if the WLAN domain is visited for the first time. Otherwise, fast EAP-AKA re-authentication is executed. In the case of WLAN to WiMAX handover, the MT performs the Initial Network Entry Authentication protocol (INEA) which is performed as a part of the Privacy and Key Management protocol version 2 (PKMv2) (AA09), when visiting the domain for the first time. Otherwise, WiMAX Re-authentication protocol is executed (WiM10).

One issue with this approach is that it is fully dependent on a specific wireless technology; in this case, the 3GPP core network . Whoever wants to add a new wireless access to an existing network will always need to develop a method that integrates wireless access with the 3GPP core infrastructure.

The Handover AKA protocol of the Mobile Ethernet

The Mobile Ethernet has proposed two AKA protocols: the first, which has been discussed in section 6.4 is used for the initial authentication; when the Mobile device joins the network for the fist time. The second AKA protocol is responsible for AKA functions in case of handover. The AKA protocols of the Mobile Ethernet are not technology-specific and do not require platforms such as the EAP and thus could be deployed by any operator. Also, the network architecture, proposed by the Mobile Ethernet is very similar to the open architecture in chapter 4. Due to these factors, the handover AKA protocol of the Mobile Ethernet will be act as model to investigate the potential security threats, it will be analysed in section 6.7.1 using formal methods approach.

6.7 Secure Vertical Handover in Mobile Ethernet

This section describes and formally analyses the Vertical Handover AKA protocol proposed by Masahiro et al (I.D06). By considering the notations in Table 6.1, the protocol runs as follows:

After running the initial AKA protocol in the source network, the mobile device and the Authentication Server (AS) would have shared the security context that consists of the UID, MS, AK, and SK. In case of a handover, the security context is transferred, over a presumably secure channel from the old AS to the new AS in the destination network. This means that the security context is always shared between the mobile device and the network. It also implies that only the SK is re-established on handover, while the re-establishment of the AK and the authentication process happen after the handover. As stated in (I.D06), the SK transferred during the context transfer continues to be used until the new SK is established. As shown in Fig 6.7, since both the mobile device and the Authentication server retain the security context, in the case of handover, mobile device's authentication is based on the the previous mutual authentication between the device and the old AS.

At the end of the authentication phase, the M and the AS derive a new Handover Authentication ID (HOAID), which is used to speed up the handover response. So instead of sending the UID, the mobile device will initiate the authentication protocol by sending the HOIAD and the R1 as the first message in Fig 6.7.

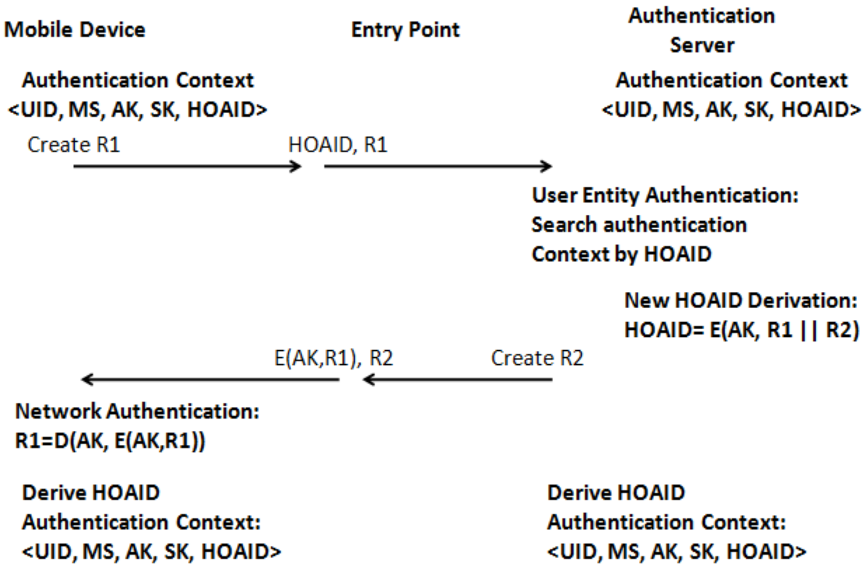

Figure 6.7: The Mobile Ethernet Protocol

6.7.1 The Formal Verification of the Mobile Ethernet Protocol

This section will formally verify the Mobile Ethernet's AKA protocol for Vertical Handover using the Casper /FDR tool, then a detailed analysis of the security properties will be introduced. As stated in (I.D06), it is assumed that, the network can trace the movement of the device and determine when handover occurs. However, in order to simulate this using Casper, we introduce the following preliminary messages: the Entry Points's advertisement messages (Adv), The Access Request (AccReq) message, which is used by the mobile device to indicate its intention to access the network. The Authentication Request (AuthReq) message, sent by the Entry point to trigger the authentication process. None of these messages play a security role; they are only used at the pre-authentication stage, where the entry points advertise their presence.

A Casper input file describing the system in Fig 6.7 was prepared. The full description is mentioned in Appendix E.

After generating the CSP description of the systems using Casper and asking FDR to check the security assertions, two attacks were found. The first discovered attack below is against the `WeakAgreement(M, EP)` and `Aliveness (M, EP)` assertions.

```
0.  -> m :  ep, as
1a.  m -> I_ep :  accReq
1b.  I_m -> ep :  accReq
```

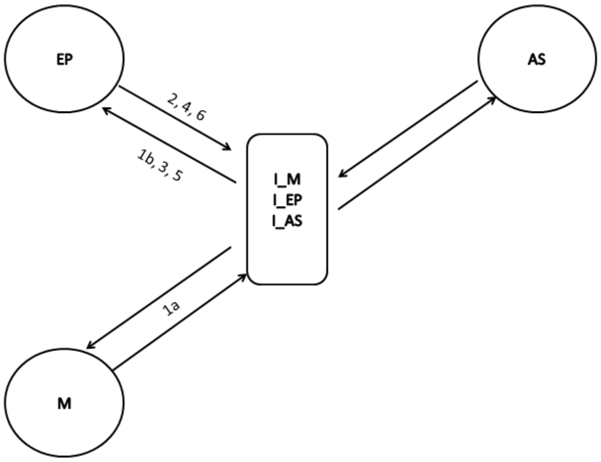

Figure 6.8: The First Attack on the Mobile Ethernet

```
2.  ep -> I_m :  authReq
3.  I_m -> ep :  Garbage
4.  ep -> I_as :  Garbage, h(Garbage)
5.  I_as -> ep :  Garbage
6.  ep -> I_m :  Garbage
```

Fig 6.8 shows the first discovered attack, which could be described as follows: Initially, the intruder intercepts the connection and replays the messages between the EP and the M as in messages 1a, 1b and 2. Pretending to be the mobile device, the intruder composes and fake message with a 'Garbage' contents as in message 3. Using this fake message, the protocol continues following the normal sequence and thus, the EP completes the run believing it has completed the run with the M, while it was with the intruder instead.

The second attack, shown in Fig 6.9 is against the `WeakAgreement(EP, M)` and `Aliveness (EP, M)` assertions. In this attack, the intruder intercepts and replays the messages between the M and the EP as in messages 1a, 1b, 2a, 2b, 3. Once the intruder intercepts message 3, it impersonates the EP and completes running the protocol as in messages 4, 5 and 6. Thus, the mobile device will complete running the protocol believing that, it was with the EP, while it was with the intruder instead.

```
0.  -> m :  ep, as
1a.  m -> I_ep :  accReq
1b.  I_m -> ep :  accReq
2a.  ep -> I_m :  authReq
```

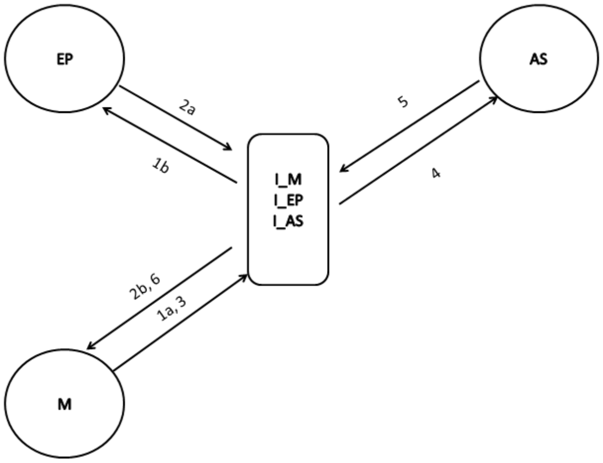

Figure 6.9: The Second Attack of the Mobile Ethernet

```
2b.  I_ep -> m :  authReq
3.  m -> I_ep :  {m, r1, hoaid1}{sk}
4.  I_ep -> as :  {m, r1, hoaid1}{sk},
h({m, r1, hoaid1}{sk})
5.  as -> I_ep :  {r2, {r1}{ak}}{sk}
6.  I_ep -> m :  {r2, {r1}{ak}}{sk}
```

6.7.2 Protocol Analysis and Security Considerations

In this section, we discuss how our formal modelling with Casper allows checking the security requirements described in 5.4.

- **Mutual Entity Authentication:** In the first discovered attack, the intruder manages to impersonate the M to run the protocol with the EP. Also, in the second attack, the intruder impersonates the EP to run the protocol with the mobile device. These attacks imply that the protocol does not fulfil this security requirement. These attacks could be ascribed due to the fact that the protocol does not consider verifying the identity of the participants.

- **Mutual Key Authentication:** The AS is authenticated to the M by proving the possession of the random value R1 and the Authentication Key AK. We got Casper to check this using the Secret (M, AK, [AS]) and Secret (AS, AK, [M]) assertion checks. Since no attack was found against the key secrecy, this property is met.

- **Mutual Key Confirmation:** Casper verifies one direction of this requirement by using the DECRYPTABLE (m, K) which checks if the message (m) is decryptable by the key (K). We performed a similar check after message 6 as shown in the Protocol Description heading to verify that the valid Authentication Key (AK) is possessed by the AS. If the check fails, the protocol aborts. For the mutual authentication, it was presumed in (I.D06) that the AK along with the security context were transferred from the old AS before the protocol starts, thus there is no need to check this by Casper.

- **Key Freshness:** Since the keying materials are transferred from the old AS, this property could be verified by considering the key derivation functions for the $MS = PRF(UUK, R1|R2)$, $AK = PRF(MS, R1|R2)$ and $SK = PRF2(MS, R1|R2)$ in the initial AKA protocol. We could claim that this property is guaranteed since fresh random values R1, R2 are included in the key derivation functions of the MS, AK and SK keys.

- **Unknown Key Share:** The second, discovered attack implies that the UKS was not met. Despite of the fact that the mobile device (M) and the AS share the Authentication Key (AK), the M mistakenly believes that the intruder holds this key as well. Casper/FDR indicates this fact by highlighting an attack against the `WeakAgreement(EP,M)` and `Aliveness(EP,M)` assertions in the # Specifications header.

- **Key Compromise Impersonation Resilience:** this property could be modelled by specifying the long-term keys as crackable and then checking the Authenticity assertions. By specifying the MS key to crackable and checking the `Agreement(AS, M, [AK, R1])` assertion, Casper verifies no breach against this authenticity feature.

It is obvious that, the Mobile Ethernet's AKA protocol for vertical handover fulfilled the Mutual Key Authentication, Key Freshness and the Key Compromise Impersonation Resilience requirements. While it failed in meeting the Mutual Entity Authentication and the Unknown Key Share, other requirements such as Mutual Key Confirmation could only be achieved if we consider the protocol pre-assumptions of a secure transfer of the security context from the previous AS. This analysis accords with the verification results of Casper/FDR, where two authenticity attacks were discovered.

6.8 The Proposed Solution

In order to address the previous security threats, this section introduces a new AKA protocol for Vertical Handover in open, heterogeneous environments. The new protocol considers the security in the core network at the design stage. However, instead of making assumptions of a secure core network, we need to define the part of the core network to be protected and the type of security mechanism. Therefore, in order to design the proposed protocol, a progressive design approach has been followed; in the initial draft, security was considered in the core network between the CA3C and the DA3Cs, modelling the proposal found secrecy and authenticity attacks, which highlight the main source of threats. The second version simulated the case of a secure channel only between the DA3C and the Auth, the discovered attacks highlight the need to secure different part of the core network. In the final version, secure channels have been presumed between the DA3C and the Auth as well as between the DA3C and the CA3C. After simulating this case using CSP, Casper failed to find any attacks. This implies that to address the afore-discovered security threats, the connections between all the entities in the core network have to be protected.

6.8.1 The Initial Version Of the Protocol

The initial version of the protocol considers the presence of a certain trust relationship between the network's entities and thus secure channels have already been established between the CA3C and the DA3Cs.

To simulate this secure connection between the CA3C and the DA3C using Casper/FDR, a secret key se1(DA3C) is presumed to be pre-shared between these entities. Thus, the connections between the CA3C and the DA3C in the source network (SrcDA3C) and the DA3C in the destination network (DesDA3C) are protected using the Srcse1(SrcDA3C) and Desse1(DesDA3C) respectively.

The Mobile Terminal (MT), residing in the source network, picks the access routers' advertisements (Adv) which contain information about the destination access network such the AuthID and the domain name. The MT uses this information to generate a Domain-Specific Master Key (DSMK).

Phase 1

$Msg1 : DesAuth \rightarrow MT : Adv$

Generate the DSMK= F1(uk(MT), seq1, AuthID)

The protocol starts when the MT sends a joining message Msg 2 to the Authenticator in the destination network (DesAuth). The DesAuth responds by sending authentication request AuthReq as Msg 3.

Phase 2

$Msg2.MT \rightarrow DesAuth : AccReq$

$Msg3.DesAuth \rightarrow MT : AuthReq$

By using the DSMK, the MT derives a new Authentication Key in the destination network (DesAK) and composes Msg 4, which consists of a fresh sequence number seq1 used as a challenge, Authentication ID (AuthID); the Mobile terminal identity (MT) , and an unset Initauth flag (InitAuth=0). Since the MT has already been authenticated in the source network, the connection with the SrcAuth will be encrypted using the Source Secret Key (SrcSK). The SrcAuth passes this message to the SrcDA3C and from there to the CA3C as Msg 5 and Msg 6. Using the included mobile ID, the CA3C looks up the corresponding uk(MT) and uses it to generate a fresh Domain Specific Master key DSMK.

Phase 3

Generate the DesAK = F2(seq1, DSMK)

$Msg4.MT \rightarrow SrcAuth : \{MT, seq1, AuthID, Initauth\}_{SrcSK}$

$Msg5.SrcAuth \rightarrow SrcDA3C : MT, seq1, AuthID, Initauth$

$Msg6.SrcDA3C \rightarrow CA3C : \{MT, seq1, AuthID, Initauth\}_{Srcse1(SrcDA3C)}$

Generate the DSMK= F1(uk(MT), seq1, AuthID)

The DSMK key is included in Msg 7, which is sent over the secure channel using the pre-shared Desse1(DesDA3C) key. Using the information in this message, the DesDA3C generates the Authentication Key (DesAK) and returns the previously sent sequence Seq1 and a new sequence Seq2 all the way to the MT as Msg 8 and Msg 9. These messages are encrypted using the derived DesAK. Since the MT has the required information to derive all the keys (DSMK, DesSK, DesAK), the MT verifies the contents of Msg 9 and derives the Secret Key DesSK.

Phase 4

$Msg7.CA3C \rightarrow DesDA3C : \{DSMK, seq1, AuthID, MT, Initauth\}_{Desse1(DesDA3C)}$

Generate the DesAK = F2(seq1, DSMK)

$Msg8.DesDA3C \rightarrow DesAuth : \{seq1, seq2\}_{DesAK}$

$Msg9.DesAuth \rightarrow MT : \{seq1, seq2\}_{DesAK}$

Verify the message contents, then derive the
DesSK:= F3(seq1, DSMK, AuthID)

The MT returns Seq2 all the way to the DesDA3C as Msg 10 and Msg 11. The DesDA3C verifies the contents of Msg 11 and derives the Secret Key DesSK.

Phase 5
$Msg10.MT \rightarrow DesAuth : \{seq2\}_{DesAK}$
$Msg11.DesAuth \rightarrow DesDA3C : \{seq2\}_{DesAK}$
Verify the message contents, then derive the DesSK:= F3(seq1, DSMK, AuthID)

Upon verifying the Msg 11, the DesDA3C authenticates the MT and acknowledges this to the CA3C, and then generates the Secret Key (DesSK) and passes it to the DesAuth in Msgs 12, 13. Using the DesSK, the DesAuth sends an encrypted access response message to the MT as Msg 14.

Phase 6
$Msg12.DesDA3C \rightarrow CA3C : \{HoAckm\}_{Desse1(DesDA3C)}$
$Msg13.DesDA3C \rightarrow DesAuth : DesSK$
$Msg14.DesAuth \rightarrow MT : \{AccRes\}_{DesSK}$

Formal Verification

A Casper description of the protocol was prepared, a complete description of the final and completely refined version of the protocol is included in the Appendix F. After modelling the protocol using Casper and checking the corresponding CSP code using FDR checker, the following attacks were discovered:
The first attack is against the `Secret(DesAuth, DesSK,[MT, DesDA3C])` assertion, where the Intruder launches a replay attack and eventually manages to get the secret key (SK).
The second attack is against the `WeakAgreement(DesAuth, DesDA3C)` assertion, where the Intruder launches a replay attack and successfully impersonates the DesAuth. Detailed analysis of the attack is in (MA12b).

6.8.2 The Second Version Protocol

In this version of the protocol, secure channels exist only between the DA3Cs and the Auths. To simulate these channels, secret keys Srcse2(SrcAuth) and Desse2(DesAuth) are pre-shared between the Auth and the DA3C in the source and destination domains respectively. After preparing the Casper's input file and asking Casper/FDR to verify the protocol, Casper found attack against the `Agreement(DesDA3C, MT, [seq1, DesAK])` assertion as described in (MA12b).

6.8.3 The Final Protocol

The first and second versions of the protocol in the 6.8.1 and 6.8.2 sections, highlight the fact that, there is a need to protect all the parts and connections in the core network. Therefore, in the final version of the proposed protocol, secure channels between the Auths and the DA3Cs, as well as the between the DA3Cs and the CA3C, have been considered. We simulated these security considerations with Casper and asked FDR to check for attacks. Despite the fact that we included the key derivation functions F2, F3 in the Intruder knowledge as stated in Appendix F, Casper/FDR failed to find attacks against any of the assertions in the #Specifications heading.

This result implies that the assumption in current systems such as 3G and 2G of a physically secure core network could not be valid any more. Therefore, in order to provide security in future heterogeneous environments, there is a need to protect each part and connection in the core network.

6.8.4 AKA Protocol Formal Verification

The main goal of the proposed protocol is to achieve mutual authentication between the MT and the core network in case of handover, thus authenticating the MT to use the destination peripheral network. To model the AKA protocol using Casper/FDR tool, we prepared a Casper input file that represents the system. The complete Casper description is detailed in Appendix F.

Furthermore, the AKA protocol for handover is related to the initial AKA protocol and uses the same keying materials. Therefore, similar to the initial AKA, the handover AKA protocol meets the desired security requirements as explained in (MA12b).

6.9 Summary

For Network-Level security, this chapter introduces two AKA protocols to provide mutual authentication between the mobile terminal and the access network in the initial connection as well as in the case of a handover. These protocols consider the open architecture of the heterogeneous networks as presented in Chapter 4, and therefore, they are considered amongst the key underlying security protocols of the AKA framework described in Chapter 5.

The protocols were verified using Casper/FDR tool and proven to meet the desired security requirements. As pointed out in our discussion on Y-Comm in Chapter 2, this work shows that as the core of the network is opened, more attacks will be possible on network entities that previously were protected in closed environments.

The next chapter will introduce the underlying protocols for the Service-Level security of the AKA framework.

Chapter 7

The AKA Framework- The Service-Level AKA Protocol

7.1 Introduction

After authenticating the Mobile Terminal (MT) in the access network using the NL-AKA protocols for the Initial and handover authentication, the MT will attempt to access the services, to which it has already subscribed. In order to provide an authoritative and secure access to these services, there is a need for a Service-Level AKA (SL-AKA) protocol that secures the session between the client and the Service Provider (SP).

This chapter introduces two SL-AKA protocols, which are responsible for achieving mutual authentication and securing the session between the MT and SP in two scenarios; the initial authentication and in case of a handover.

This chapter is organized as follows: Section 2 defines the security challenge in terms of achieving mutual authentication between the mobile terminal and the service provider in heterogeneous environments. Section 3 describes related research to address the security issue at the service-level. Two proposed SL-AKA protocols, for the initial AKA and the handover, are described in Section 4. The Chapter is summarized in Section 5.

7.2 Problem Definition

The aim of ubiquitous computing in heterogeneous environments similar to the one described in Chapter 4 is to provide mobile users anytime, anywhere and any platform access to a wide variety of computing servers. While much research has been performed to provide the infrastructure and mechanisms to support this goal at the network level such as Mobile IPv6, Fast Mobile IPv6 and IEEE802.21 (DJ04), (McC05), (IEE07), few research efforts such as (Ste07) have considered the need for application support for connectivity.

Furthermore, the issue of Application/Service-Level security has, in general, been difficult to address in future networks. This is due to many reasons. Firstly, any proposed security protocol has to consider the structure of current mobile devices as well as their limitations in terms of battery and processing power. These conditions put extra restrictions when deciding on security measurements such as encryption algorithms (Symmetric or Asymmetric) as well as keys management.

Secondly, as described in Chapters 2 and 6, current security mechanisms consider the closed nature of current communication systems. However, these differences highlight the need for enhancing current security mechanism if not introducing new ones that consider the open architecture of the future networks.

Thirdly, when a client subscribes to a service, parameters such as the desired QoS and security parameters will be defined as part of the Service Level of Agreement (SLA). However, since the service provider might have different preferences in terms of the security and QoS, the two end-points might need to provide a range of preferences where they could negotiate and trade off between security and QoS. This highlights the need for a negotiation stage to specify the connection parameters before setting up the connection.

Fourthly, in heterogeneous networks, future mobile devices are expected to switch between various access networks whilst remaining connected to the service provider. Based on their security and QoS characteristics, a server provider might choose to trust some networks more than others and hence apply different security measurements. This highlights the need for the server provider to know about the access network of the mobile terminal in order to re-assess the connection security and to decide on the required security parameters.

However, in case of handover when the mobile terminal moves into a new network with different characteristics, the mobile terminal and the service provider will need to re-negotiate the connection parameters to comply with the characteristics of the new access network. This highlights the need for a lightweight Authentication and Key Agreement (AKA) protocol for handover so the functionality of the this protocol will not disrupt the connection with the server.

7.3 Related Work

This section describes some potential mechanisms to address the Service-Level security in future networks.

7.3.1 Transport Layer Security (TLS) and Secure Sockets Layer (SSL)

Transport Layer Security (TLS) and its predecessor, Secure Sockets Layer (SSL), are cryptographic protocols that provide secure communication between two end points over the Internet. SSL is divided into two layers, with each layer using services provided by a lower layer and providing functionality to higher layers. The SSL record layer provides confidentiality, authenticity, and re-play protection over a connection-oriented reliable transport protocol such as TCP. Layered above the record layer is the SSL handshake protocol, a key-exchange protocol which initializes and synchronizes cryptographic state at the two endpoints. After the key-exchange protocol completes, sensitive application data can be sent via the SSL record layer (TD99). In this sense, SSL/TLS enables the end-points to negotiate and agree on security parameters such as the encryption and hashing algorithms.

Using public-key encryption techniques, SSL-enabled client and server will authenticate each other and establish an encrypted connection. Although SSL/TLS achieve many desired security properties, and as a result, have been widely implemented, there are many drawbacks when it comes to implementing them in future networks; firstly, they are PKI-based protocols, which involves undesired complexity as explained in 5.5.2. Secondly, the SSL/TLS run above the Transport layer which make them unaware of the characteristics of the underlying access networks, and thus cannot reflect these characteristics in the negotiation stage of the protocol. Thirdly, these protocols do not introduce a lightweight extension for re-authentication in case of handover.

7.3.2 The Stream Control Transmission Protocol (SCTP)

The Stream Control Transmission Protocol (SCTP) (Ste07) is a connection-oriented transport protocol that operates on the top of the IP protocol. The SCTP has several advantages over the traditional transport protocol such as Transmission Control Protocol (TCP) (VC74) and User Datagram Protocol (UDP) (Pos80), examples of these advantages are multi-streaming and multi-homing support. Additionally, the Secure SCTP (SSCTP) (CH12), (MT07) was designed with security features to set a secure association between the two end-points and thus addresses attacks such replay and SYN flooding. The SSCTP protocol enables the two end-points to negotiate the security parameters and thus agree on the desired algorithms.

However, the security approach proposed of the Secure SCTP is highly dependent on the SCTP protocol as the underlying transport protocol and consequently, it cannot be used with other transport protocols such as the widely implemented TCP and the UDP. Although the

SCTP protocol supports client mobility (MR05), there is neither a clear impact of this mobility on the security mechanisms or a lightweight re-authentication protocol in case of handover.

7.3.3 The Service-Level AKA of the Mobile Ethernet framework

The Mobile Ethernet framework (MK04), (I.D06), (Kur05) is an architecture for IP-based, future networks. In order to address the security between the mobile terminal and the service provider,an SL-AKA protocol was introduced in (MK04).

Although the Mobile Ethernet framework, and thus its security protocols, adopts a network structure that is very similar to our view of future networks in Chapter 4, and despite the fact that, the SL-AKA protocol achieves a set of desired security features such as mutual authentication and connection confidentiality, it suffers from some major drawbacks. These are as follows: firstly, the SL-AKA protocol does not have a negotiation stage; thus, it neither considers variations of QoS and security requirements of the access networks and the service provider nor the preference of clients. Secondly, it does not consider the case of handover and thus no SL-AKA protocol for handover has been proposed.

7.4 The Proposed Service-Level AKA Protocol

The SL-AKA protocol is needed to achieve mutual authentication between the Mobile Terminal (MT) and the Service Provider (SP). The proposed protocol is invoked when the MT sends a request to use a specific service running on the service provider (SP) side to achieve a mutual authentication and key agreement between the two ends. However, before proceeding with explaining the protocol, it might be beneficial to recall the structure of future heterogeneous networks as described in Chapter 4.

7.4.1 Overview of Future Networks

We adopt a hierarchical network structure composed of three levels. At the top level, the Core End-Point (CEP) acts as a gateway to the Internet and is responsible for managing multiple, mid-level domains. Each domain is technology-specific and is controlled by a single operator. For instance, two domains might be connected to the same CEP, each controlled by a different technology operator such as WiMAX and GSM. The bottom level comprises individual peripheral wireless networks, controlled by Access Routers or Base Stations through which the mobile terminal has access to the wider network. Additionally, some Service Providers (SPs) such as video on-demand or E-Commerce providers might reside in the core network and could be accessed over the peripheral network. Each service provided is identified by its service ID (SrvID), also, these providers have agreements with the network operators to guarantee the required QoS

7.4.2 The Initial SL-AKA Protocol

This protocol runs when the Mobile Terminal (MT) initially expresses its intention to contact the Service Provider (SP) to achieve mutual authentication and to set up a secure channel between the MT and the SP.

Considering the network structure in Chapter 4, the information about the subscribed services and the client's security preferences along with the characteristics of its access network such as the network domain name are kept by the Central A3C (CA3C) in the Core-End Point (CEP). Also, as described in the key hierarchy section of Chapter 5, for each subscribed service, the CA3C will derive a service key Srvkey= F(UK, SrvID, SubID, lifetime) and passes it to the MT and the SP. However, sharing the SrvKey between the MT and the SP is not part of the SL-AKA protocol and could be achieved as part of the QoS models in Chapter 4. Therefore, the SL-AKA protocol considers the SrvKey to be pre-shared between the SP and the MT. This key will be used to derive the Association Key (ASKey) to secure the connection between the SP and the MT.

By considering the notations in Table 7.1, the SL-AKA runs as follows:

The SL-AKA is initiated when the Mobile Terminal (MT) indicates to the CA3C its intention to access the service provider (SP). The CA3C server knows the services' subscription IDs as well as the corresponding MT's preferences in terms of security and QoS which are part of the Service Level of Agreement (SLA) stored in the CA3C server. The CA3C server passes the MT's preference as a vector of information (Vector1), which contains lists of MT's preferred encryption and hashing algorithms (Enclist) and (HMACList) respectively. Additionally, it contains a fresh random value (r1) to maintain the vector freshness.

This vector along with the domain name of the MT's access network and the MT's SubID are passed to the Service Provider (SP) as messages 1,2 and 3. Once, the SP receives message 3, it derives the Association Key ASKey = F (Srvkey(SP), Vector1, Vector2).

$Msg1 : CA3C \rightarrow DesDA3C : Vector1, SubID, ADname$
$Msg2 : DesDA3C \rightarrow DesAuth : Vector1, SubID, ADname$
$Msg3 : DesAuth \rightarrow SP : Vector1, SubID, ADname$
Generate the ASKey= F (Srvkey(SP), Vector1, Vector2)

Table 7.1: Notation

Abbreviation	Full name and description
MT	Mobile Terminal
SP	Service providers residing in the core end point
DesDA3C	The Domain AAAC server of the destination domain of the Application/service provider
CA3C	Central AAAC server stores the MT's SLA, which contains the MT's preferred QoS and Security parameters as well as the a list of all the SPs.
Srvkey(SP)	Service key: a pre-shared key between the MT and the SP: Srvkey= F(UK, SrvID, SubID, lifetime).
r1,r2	Random nonce
HMACList1, HMACList2	Lists of supported hashing algorithms.
EncList1, EncList2	Lists of encryption algorithms.
SrvID	Service ID, which uniquely identifies the service.
SubID	User subscription ID, uniquely identify the subscriber to the SP.
ADname	Access Domain name, defining the domain name of the access network.
SrvCookies	The Cookies, sent by the sever to the MT, these cookies limit replay and DoS attacks.
Vector1	r1,HMACList1,EncList1.
Vector2	r2,HMACList2,EncList2.
ASKey	Association key ASKey = F (Srvkey, Vector1, Vector2).
Ackm	Authentication Token Ackm=F(SubID, SrvID, timestamp) used as an acknowledgement messages to indicate the completion of the AKA process.

Based on the MT's preference in Vector1, the SP constructs Vector2 which represents the SP preferences in terms of encryption and hashing algorithms (EncList2), (HMACList2) -this negotiation stage will be discussed later in section 7.4.2. In message 4, the SP sends Vector2 and server cookies (SrvCookies) to the MT. These cookies will be used as a challenge and to stop re-play attacks.

$Msg4 : SP \rightarrow MT : \{Vector2, SrvCookies\}_{Srvkey(SP)}$
Generate the ASKey= F (Srvkey(SP), Vector1, Vector2)

Message 4 is encrypted using the pre-shared service key SrvKey(SP) between the MT and the SP. Therefore, the MT will decrypt the message to get Vector2 and derive the ASKey. The MT retrieves the nonce number (r2) from the received Vector2, uses the derived ASKey to encrypt message 5, which includes the server's cookies and r2. Upon receiving message 5, the SP verifies the message's contents to ensure that it contains the valid values for the SrvCookies and r2. In case of a successful verification, the SP acknowledges the successful authentication by composing the acknowledgement message (Msg6)
$Msg5 : MT \rightarrow SP : \{r2, SrvCookies\}_{ASKey}$
Verify the message contents
$Msg6 : SP \rightarrow MT : \{Ackm\}_{ASKey}$

The Negotiation Process

At this stage, we presume that each time the user subscribes to a new service, an identity-based authentication token is generated and securely stored in the MT and the SP. This token is used in the protocol as an acknowledgement Ackm=F(SubID, SrvID, timestamp) to indicate authentication completion and for achieving identity authentication as explained in the SL-AKA Analysis section 7.4.2. For the initial subscription, an out-of-band token distribution is achieved prior to the SL-AKA. In later stages, if the MT wants to subscribe to a new service, the new token is sent via secure control and management channels. However, setting these channels up is beyond the scope of the SL-AKA protocol.
In Msg1, the CA3C provides the SP with a list of the supported hashing and encryption algorithms by the MT; it also contains the domain name of the MT's access network. The reason for including the network domain name is to allow the SP to specify its security level

with regards to the credibility of the MT's access network.

Two major factors define network's credibility: the network security level in terms of the efficiency of the authentication and encryption mechanisms; geographical location of the MT's access network, some services might choose not to accept access requests from certain countries or domains, which are considered as insecure. Taking these factors into account, the SP specifies three modes of access networks: low, normal and high secure networks and as a result the SP re-orders its own hashing and encryption lists (HMACList2, EncList2) and sends them to the MT as part of the Vector2.

This way, in addition to its own lists, each end has the other end's lists. In the case of HMAC lists for instance, each end takes the first suggested algorithms in the SP's list (HMACList2) and looks it up in the MT's list (HMACList1), if no match is found, it takes the second suggested algorithm in list2 and looks it up in list1, then the third and so on. The first match is considered as the adopted hashing algorithm. The same procedure is followed for choosing the session encryption algorithm.

The SL-AKA Formal Verification and Security Considerations

Similar to the proposed UL-AKA and NL-AKA protocols, Casper/FDR tool is used to verify the SL-AKA protocol. We prepared an input file describing the protocol and its system. Refining the CSP file using Casper/FDR shows no attacks against the proposed protocol. The full Casper file, representing the protocol is in the Appendix G.

The MT and the SP could use the same SrvKey as long as it is still valid; LifeTime is less than a pre-defined threshold. This threshold is defined by the CA3C and is proportional to the MT truthfulness.

By considering the contents of Vector 1 and Vector 2, we guarantee the freshness of the Association Key (ASKey), by including two fresh nonces (r1, r2) in the derivation function: ASKey = F (Srvkey, Vector1, Vector2). Furthermore, as will be explained later in the lightweight SL-AKA protocol for a handover, a new ASKey is derived whenever the MT changes the domain of the access network, thus, in the case of multi-homed devices (GM11), different associations keys are used to protect the sessions with the SP. The side effects of this situation is the possibility that, a legitimate user might start many sessions and transfer a large volume of information; therefore, the SP needs to set a policy on the number of the simultaneous sessions for each mobile terminal identified by the subscription ID SubID. Another potential solution is by enforcing admission control at the SP's access network, such mechanisms are not parts of the SL-AKA protocol.

Security Analysis Based on the Security Requirements List

1. Mutual Entity Authentication: Similar to the UL-AKA protocol, this security property is achieved, using the Authentication Token Ackm=F(SubID, SrvID, timestamp) which has been generated based on the parties' IDs.

2. Mutual Key Authentication: The mutual authentication between the MT and the SP is based on the secrecy of the derived session key Srvkey(SP). We got Casper to check this using the Secret (SP, Srvkey(SP),[MT]) assertion check.

3. Mutual Key Confirmation: This property is met by performing the check, using the Decryptable function after Msg4 and Msg5 in the Protocol Description heading as detailed in Appendix G. By using the Decryptable function each party makes sure that the valid key is possessed by the other part. If any of the check failed the protocol aborts.

4. Key Freshness: Since Casper does not have any function to check this property, the freshness of the Association key ASKey is guaranteed by including Vector 1 & 2 in its Key Generation Function (KGF) ASKey= F(Srvkey(SP), Vector1, Vector2). These vectors comprise two fresh random values r1 & r2; thus, a new ASKey is derived for each session. Since Casper does not detect any attack on the secrecy of the ASKey, this implies that key freshness is not violated.

5. Unknown-Key Share: This requirement could be met by making a bind between the derived key and the parties' identities. This is considered by including the SrvKey in the deriving function of the ASKey; the SrvKey involves the SubID and SrvID in its derivation function: Srvkey= F(UK, SrvID, SubID, lifetime). Casper verifies this property by using the WeakAgreement assertion in the Specification heading Appendix G.

6. Key Compromise Impersonation Resilience: This check has been done by specifying the long-term key Srvkey(SP) as crackable and using the Agreement assertion to check any breach of the authenticity feature.

7.4.3 Light Weight SL-AKA Protocol for Handover

When the MT performs handover and changes its point of attachment, the new access network might offer different QoS and be of a different credibility level. There is a need to consider these changes by re-negotiating the security parameters and deriving a new Association Key (NewASKey) to secure the connection between the MT and the SP.

However, there is also a requirement not to interrupt the ongoing service; therefore, the renegotiation process in the proposed SL-AKA protocol starts before the MT actually moves to the new network, and hence, the NewASKy is derived by the MT and SP prior to the handover. Furthermore, since the MT and SP have already authenticated each other, the new fast re-authentication will be based on the previous authentication.
The light weight SL-AKA protocol goes as follows: When the MT sends a handover request to a new domain, the CA3C will send the domain name of the new network towards the SP as in messages 1,2 and 3. When the SP receives this information, it re-orders the HMACList2 and the EncList2 to suit the new characteristics of the network, and thus the SP will have a different value of the Vector2. The SP will also use the old Association Key (OldASKey) to derive the new one: NewASKey= F(OldASKey, Vector1, Vector2).

$Msg1 : CA3C \rightarrow DesDA3C : ADname$
$Msg2 : DesDA3C \rightarrow DesAuth : ADname$
$Msg3 : DesAuth \rightarrow SP : ADname$
Generate the NewASKey= F (OldASKey , Vector1, Vector2)

The SP sends the new vector (Vector2) to the MT as message 4, which is encrypted using the OldASKy. Only the MT can decrypt this message to retrieve Vector2, which will be used by the MT to generate the NewASKey. The MT acknowledges the successful derivation by sending an encrypted acknowledgement using the NewASKey.

$Msg4 : SP \rightarrow MT : \{Vector2\}_{OldASKey}$
Generate the NewASKey= F (OldASKey , Vector1, Vector2)
$Msg5 : MT \rightarrow SP : \{Ackm\}_{NewASKey}$

Formal Verification

Similar to the initial SL-AKA protocol, Casper/FDR proves no attacks against the light weight SL-AKA. The full Casper/FDR description of the protocol is in Appendix H.

7.5 Summary

As a part of the AKA framework described in chapter 5 and in order to achieve mutual authentication and set a secure connection between the Mobile terminal (MT) and the Service

Provider (SP), this chapter introduces a novel SL-AKA protocol that considers the network structure proposed in Chapter 4. Furthermore, to deal with the handover issue at the service level, a lightweight SL-AKA protocol is introduced. Both protocols are verified using Casper/FDR tool and proven to meet many desired security requirements.

This chapter along with Chapters 5 and 6 have defined the underlying security protocols of the AKA framework. The next chapter will show how to integrate the AKA framework with the QoS framework described in Chapter 4 to define the Ring-based, Connection and Vertical handover security models.

Chapter 8

The Targeted Security Models

8.1 Introduction

As highlighted in Chapter 2, one of the key concepts of the proposed security approach in this research is the notion of the Targeted Security Models (TSMs), which aim at protecting data as well as network entities in different situations. In this research, three security models have been proposed: the Connection, vertical handover and the Ring Based security models. This chapter will introduce the security models and describe how they could be defined by integrating the QoS models introduced in Chapter 4 and the AKA framework described in Chapter 5.
This chapter is organized as follows: Section 2 recalls the layers of the Y-Comm's Integrated Security Module (ISM). Section 3 describes the Ring-Based Security model. Section 4 defines the Connection security model. It starts by introducing an abstract model then explains how the model could be defined by integrating the AKA framework and the Registration and connection- QoS signalling models. Section 5 proposes an abstract Secure Vertical Handover Model, then shows how the model is defined by integrating the AKA framework and the handover-QoS signalling model. The chapter is summarized in Section 6.

8.2 Recalling the Y-Comm Integrated Security Module

As previously mentioned in Chapter 2, the Y-Comm framework proposes a four-layer Integrated Security Module (ISM) to address security at different levels of the communication framework. The security module comprises the following layers:

1. Service And Application Security (SAS): In the Peripheral Framework, SAS defines the

Table 8.1: Mapping the AKA mechanisms to the Security module

The Security Layer	The AKA Protocol
Service and Application Security (SAS)	UL-AKA
Network Transport Security (NTS)	SL-AKA
Network Architecture Security (NAS)	NL-AKA

AAAC functions at the end-device and is used to authenticate users and applications. SAS in the Core network provides AAAC functions for services on the Service Platform in the core network.

2. QoS-Based Security (QBS): is concerned with QoS issues and the changing QoS demands of the mobile environment as users move around.

3. Network Transport Security (NTS): is used to set up secure connections through the core network.

4. Network Architecture Security (NAS): is invoked to ensure that the user is authorized to use any given network.

As explained in the Table 8.1, the underlying security protocols of the AKA frameworks namely UL-AKA, NL-AKA and SL-AKA address the functionalities of certain layers of the ISM. In order to address the functionality of the QoS-Based Security (QBS) layer as well as to define the Targeted Security Models, we need to integrate the AKA framework and the QoS-Signalling models, this will be explained in the following sections.

8.3 The Ring-Based Security Model

This section describes the Ring-Based Security Model which uses the scope concept to protect servers by limiting their accessibility based on their functionality. The model will also protect network and servers resources from being abused, thus will greatly reduce Denial of Service attacks on offered services without limiting access to key parts of the infrastructure.

8.3.1 The Motivation

The open and multi-homed nature of the Next Generation Networks (NGNs) makes them vulnerable to security threats which manifest themselves mainly in form of Denial of Service (DoS) attacks. Due to many factors, these attacks will become more common in future heterogeneous networks: firstly, the concept of global reachability, which has been adopted in the design of most communication protocols, allows any host to communicate with any other host over the globe. Secondly, due to the absence of QoS-provision over the different networks, an attacker can send a huge volume of traffic towards the victim without any indication of a QoS breach.

It is obvious that the compromise and resource exhaustion attacks lead to a breach of the agreed SLA. This highlights the need for an integrated solution that considers the security and QoS sides of these attacks. Moreover, in order to deal with the above situation, there is a need for a novel approach that addresses each of the afore-mentioned factors. Therefore, to deal with the first factor, this Chapter introduces the Ring-Based security model which enhances the concept of the ”Off By Default” (HB05) which enables the end-hosts or servers to define the nodes to communicate with and thus, limits servers' accessibility based on their functionality.

8.3.2 The Model Outline

The Ring-Based concept does not allow servers to be directly accessible over a Wide Area Network such as the Internet without initially interacting with the network infrastructure. This is done by using the concept of scope where a server acts only within a given scope. Therefore, by considering the network structure introduced in Chapter 4, four scopes are presented as shown in Fig 8.1:

- **Local:** indicates that the server could only be accessed locally via mechanism such as the loopback interface.
- **Local Area Network:** defines a LAN scope which means that the server could be accessed by the nodes in the same peripheral network.
- **Core End-Point/Site:** denotes that only devices residing within the same Core End-Point could access the server; these devices might be in the same or different domains.
- **Global:** means that the server is globally accessible over the Internet; thus, it accepts connections from other Core-End Points.

Details on how to enforce these scopes are given in Chapter 9.

8.4 The Connection Security Model

This section will describe the Connection security model. It starts with introducing the model outline which defines the transactions between different network entities. Then it shows how

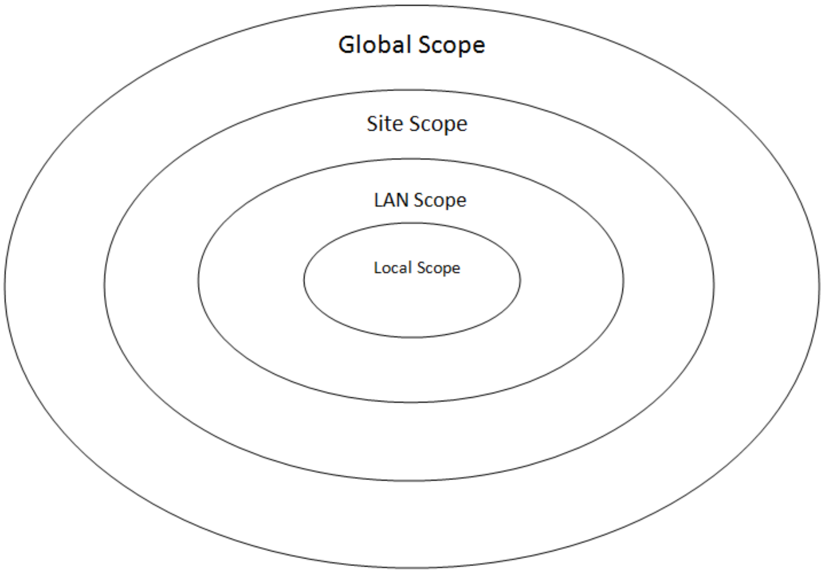

Figure 8.1: The Ring-Based Security Model

this model could be achieved by integrating the QoS-Registration and Connection Initiation models, described in Chapter 4 and the AKA framework, introduced in Chapter 5.

8.4.1 The Connection Model Outline

As stated in (MA10), (GM10), this model aims at establishing a secure connection between a mobile terminal and a service being hosted at another site, it defines the required steps to set up such secure session as shown in Fig 8.2.

- Step 1: The server is started. The NAS module in the server talks to the NAS module on the Local LAN to get access to its wireless infrastructure.

- Step 2: The QBS security module on the server informs the QBS module in the core network about its overall Service Level of Agreement which contains the QoS associated with a connection to this service.

- Step 3: The mobile node is started. The NAS module in the mobile node contacts the NAS module in the peripheral networks to gain access to the wireless infrastructure.

- Step 4: When the mobile node wants to use the service, the QBS Module in the mobile node contacts the QBS module in the core network and asks for a connection with a given quality of service to be made to the Server. The QBS module returns two core endpoints which must be used to set up the connection.

- Step 5: The NTS module on the mobile node contacts the NTS module in the core network and says that it would like a connection to the server, using the core endpoints, the QoS and security parameters.

- Step 6: The NTS module in the core network contacts the NTS module on the server to signal an incoming call. At this point, the server can also check the security of the client as well as the security of the connection.

- Step 7: If the server accepts the request, then the NTS module in the core network joins the two core endpoints.

- Step 8: It then signals to both the client and server that a connection has been established.

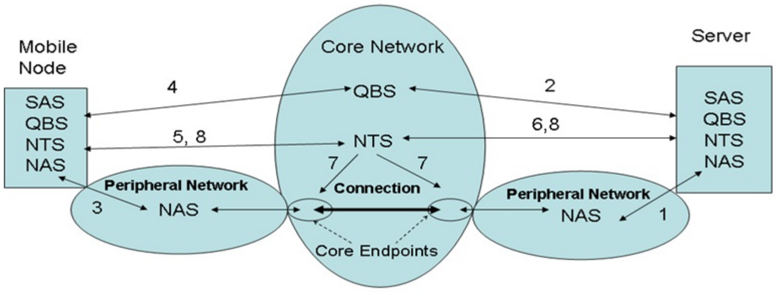

Figure 8.2: The Connection Security Model

8.4.2 Defining the Connection Security Model Mechanisms

Once the mobile terminal (MT) starts, the User-Level AKA protocol (Chapter 5) is invoked to achieve mutual authentication between the terminal, the SIM/PIC and the user. Starting from this stage, the MT will deal with all subsequent security and connectivity procedures on behalf of the user.

The second step deals with registering the MT in its network, which mainly involves:

- Authenticating and authorizing the MT to join the network, this could be achieved by the Initial Network-Level AKA protocol described in Chapter 6.

- Checking the MT's Service Level of Agreement (SLA) with the server and applying the corresponding access control policy on the network Access Router (AR), this will be achieved by the QoS-Registration model.

The third step deals with connection initiation, when the MT wants to start a connection with a remote service provider (SP). This step involves the following:

- Authorizing the connection request in both the source -the MT- and the destination -the service Provider- networks.
- Making sure that the connection specifications comply with the available QoS. The first two steps are part of the QoS-Connection model.
- Achieving mutual authentication and setting up a secure session between the MT and the SP, this will be achieved using the initial Service-Level AKA protocol described in Chapter 7.
- The final steps will be allocating the network resource to accommodate the connection, this is part of the QoS-Connection model.

Fig 8.3 shows the steps of the Connection security model.

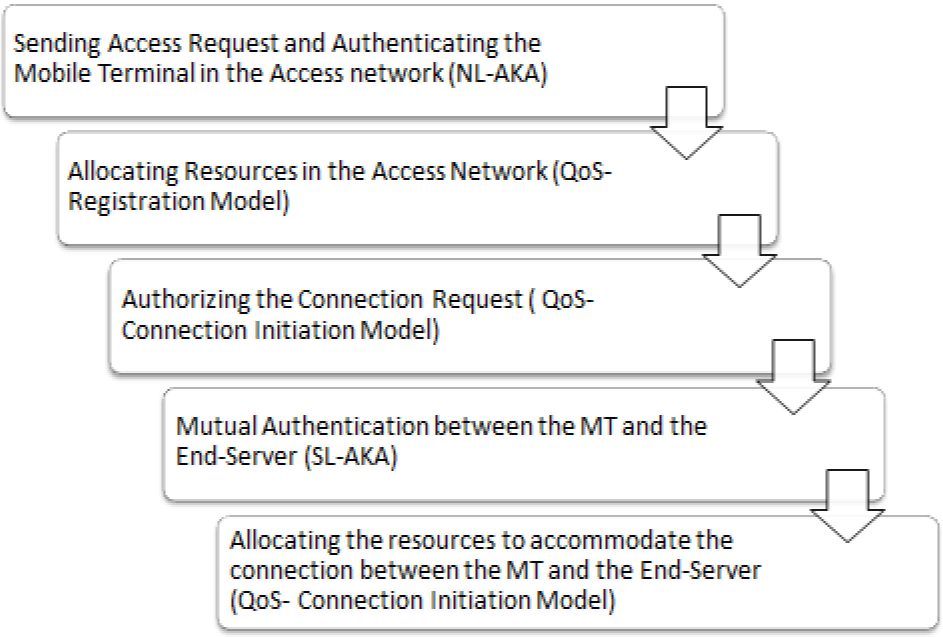
Sending Access Request and Authenticating the Mobile Terminal in the Access network (NL-AKA)
Allocating Resources in the Access Network (QoS-Registration Model)
Authorizing the Connection Request (QoS-Connection Initiation Model)
Mutual Authentication between the MT and the End-Server (SL-AKA)
Allocating the resources to accommodate the connection between the MT and the End-Server (QoS- Connection Initiation Model)

Figure 8.3: The Steps of Connection Security Model

8.5 The Secure Vertical Handover Model

This section starts with proposing a model for the secure vertical handover and then defines how this model could be achieved by integrating the AKA framework and the QoS-Handover signalling model, described in Chapter 4.

8.5.1 The Model Outline

As stated in (MA10), (GM10), the vertical handover security model is for facilitating secure vertical handover and attempts to prevent network resources from being abused and overloaded. This is done by monitoring resource requests and ensuring access to vulnerable components does not exceed the available QoS. As shown in Fig 8.4, the model involves the following steps:

- Step 1: The QBS layer of the Mobile Terminal asks the QBS of the Core Network about potential target networks for handover with required QoS and security level. If this information has not been already in the Core-End point, all the available networks are probed by the core endpoint. At the end of this stage, the MT has a clear idea of the QoS and security available at all potential networks in the vicinity and could decide on the target network for future handover.
- Step 2: The Mobile terminal sends a handover request to the target network.
- Step 3: After authenticating and authorizing the MT in the target network, network resources are allocated to accommodate the handover and a Handover Response is sent towards the MT.

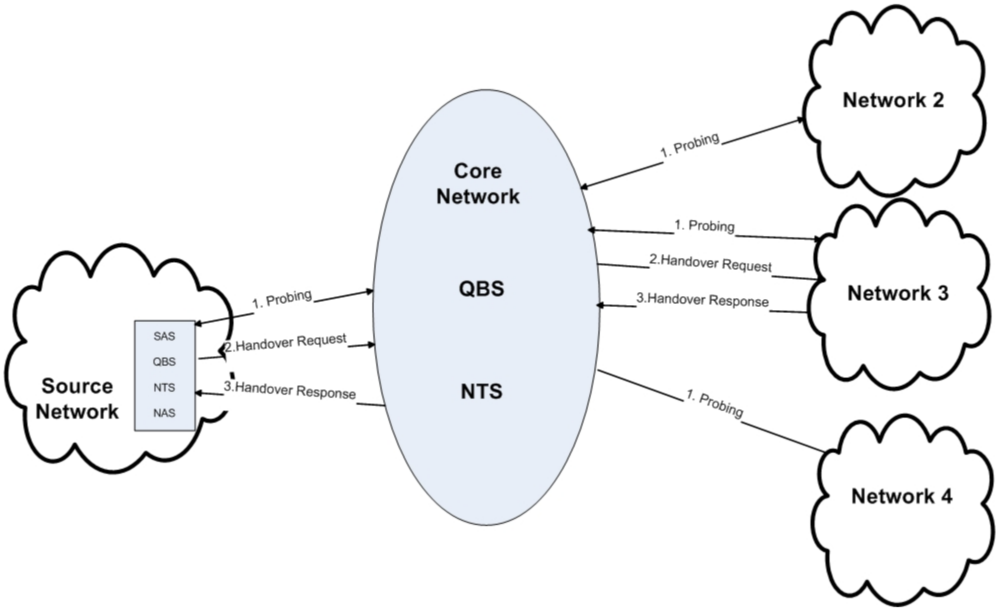

Figure 8.4: The Secure Vertical Handover Model

8.5.2 Defining the Secure Vertical Handover Model Mechanisms

After deciding on the target network for handover, the MT sends a Handover Request towards the new network. Security should be provided with minimum disturbance to the handover; therefore, before the actual handover happens, the MT should be pre-authenticated and the security materials should be launched in the target network in advance. This could be achieved using the handover Network-Level AKA, introduced in Chapter 6.

Once mutual authentication is achieved, the new target network will derive the Admission policy related to the MT from the Core network, it then passes this policy all the way to the peripheral network's Access Router. This task will be achieved as a part of the QoS-Handover Signalling model.

If the MT has a connection with a remote service provider, the handover Service-Level AKA protocol will be invoked to provide a fast re-authentication and set a secure session.

Network resources have to be allocated in the target network, then a Handover response is sent back to the MT. The old/source network will release the resources which have been reserved for the MT. This is part of the QoS-Handover Signalling model. Fig 8.5 details the steps of the Secure Vertical Handover Model.

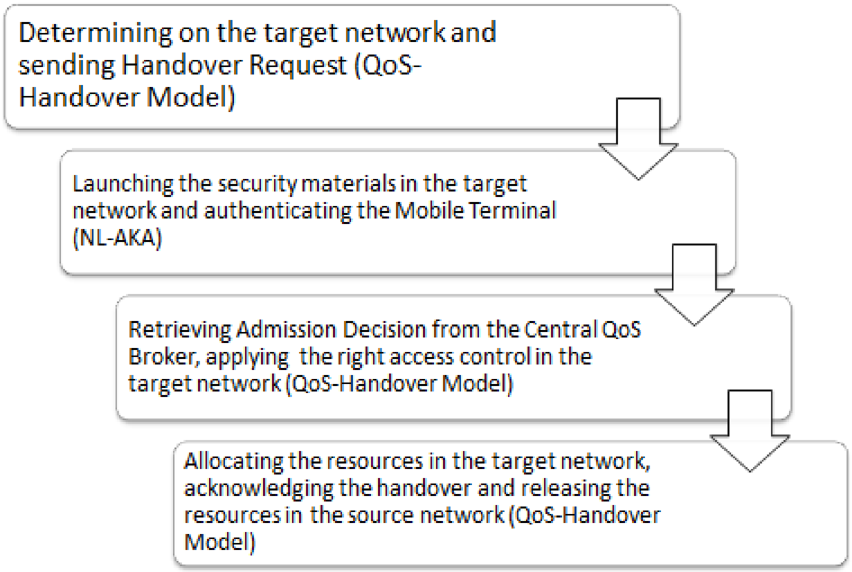
Determining on the target network and sending Handover Request (QoS-Handover Model)
Launching the security materials in the target network and authenticating the Mobile Terminal (NL-AKA)
Retrieving Admission Decision from the Central QoS Broker, applying the right access control in the target network (QoS-Handover Model)
Allocating the resources in the target network, acknowledging the handover and releasing the resources in the source network (QoS-Handover Model)

Figure 8.5: The Steps for the Secure Vertical Handover Model

8.6 Summary

After defining models for QoS-signalling in Chapter 4 and an AKA framework in Chapter 5, this chapter described how these could cooperate to define the Connection and the Vertical handover security models. The chapter also defines the Ring-Based model to protect servers in heterogeneous environment by limiting the accessibility based on their scope of functionality. The next Chapter will explore possible options to implement the proposed AKA protocols in this research as well as introduce new enhancements to the network structure to help in implementing the Ring-Based model.

Chapter 9

Potential Implementation Approaches for the Proposed Security Mechanisms

9.1 Introduction

After defining the Targeted Security Models (TSMs) as well as their underlying security mechanisms, this Chapter investigates possible procedures towards implementing the proposed NL-AKA, SL-AKA and UL-AKA protocols. Furthermore, it investigates possible approaches to support integration between the security, QoS and communication procedures using an ontology (Gru93), (Lac05). The chapter also proposes changes to the addressing, naming and location systems in the network, which could aid at implementing the Ring-Based Model.

The rest of this chapter is organized as follows: Section 2 deals with the multi-homing issue and introduces major enhancements on some network services to implement the Ring-Based Model. Section 3 defines the problem of security attacks against the implementation of the security protocols and the need for integrating the proposed QoS and security mechanism with the communication framework. Section 4 describes the Compiler Of Security Protocols into Java (COSP-J) compiler to generate a Java code of the security protocol and the Automatic Code Generator into C# code (ACG-C#) compiler which generates a C# code of the protocol. Section 5 describes how an ontology could support the integration between security, QoS and communication framework. The chapter is summarized in Section 6.

9.2 Enhancing Network Services to implement the Ring-Based Model

As stated in Chapter 8, the Ring-Based Model aims at dealing with compromise and resource exhaustion attacks as forms of Denial of Service (DoS) attacks by using the scope concept to restrict access to servers so users can only access the server when they are in the same scope as the server. Additionally, there is a QoS-side of the DoS attacks since they are mainly availability attacks and result in exhausting the network and server resources.

The author believes that the multi-homing issue in future networks will increase the severity of these attacks, as malicious, multi-homed devices use several interfaces, identified by different network addresses to launch the attack, without having anything to indicate that these addresses are collocated on the same node. In this sense, the multi-homing issue will enable attackers to start simultaneous connections with end-servers over different networks which might result in exceeding the agreed QoS and overloading the resources.

Therefore, any proposed solution should consider the multi-homed nature of future devices and limit servers accessibility to only authorized clients. Also, it has to continuously monitor the resource utilization over different networks and make sure that there is no breach of the agreed QoS.

9.2.1 Investigating the Multi-Homing issue in Future Networks

In recent work of the Y-Comm group (GM11) and (MA11c), the impact of multi-homed devices on current network addresses and structure has been investigated. The outcome highlighted the need for a new approach to map multiple network interfaces to the hosting device, thus a novel addressing scheme has been introduced in (GM11). Additionally, major changes to location and naming systems such as the Home Location Register (HLR) and the Domain Name System (DNS) (Sch03), (Moc87) have been introduced in (MA11c). It is worth pointing out that the idea of using different scopes fits very nicely with the addressing scheme that was formally associated with IPv6, in that IPv6 specified various address types closely associated with the scope concept for example, link and site addresses. Y-Comm, however, takes the view that location or network address and scope should be treated in a more orthogonal or independent fashion. This is because a scope of server is usually defined in an administrative context by a security administrator. Hence we need to decouple the concept of scope from network addresses.

Fig 9.1 shows the novel addressing scheme, the 128-bit long address has three portions: the Location_ID defines the domain of the mobile node (MN), the Node_ID is a 64-bit used to

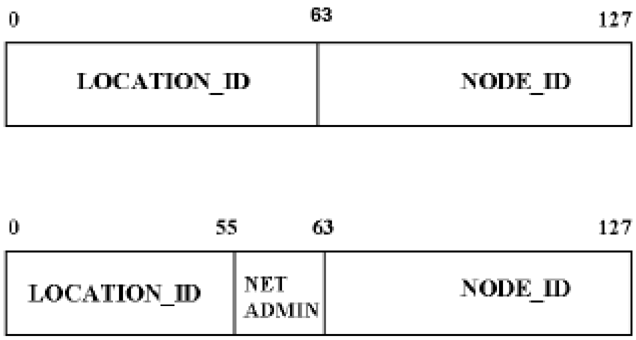

Figure 9.1: The Address Format

identify the node and is assigned by the the manufacturer. Among the fields of the NetAdmin part is the 2-bit Scope Field (SF) which is responsible for defining the node accessibility as follows:

- **SF=00:** indicates that the node could only be accessed by processes on the same machine.
- **SF=01:** defines a LAN scope which means that the node could be accessed from only the devices belonging to its LAN network.
- **SF=10:** denotes that only devices residing in the same site as the node could get access.
- **SF=11:** means that the device is globally accessible.

To support the new addresses and deals with the multi-homing issue, there is a need for major changes to the DNS and the location systems. Therefore, in (MA11c) the group has proposed the concept of the Enhanced DNS (eDNS) and the Master Locator (ML), respectively. Similarly to the current DNS, the eDNS is still responsible for resolving Addresses to names and visa-versa. However, as shown in Table 9.1, to support the new addressing scheme, there is a need to have more information about the node launched in the naming system. Examples of this information is the scope field (SF), the M and S fields, which are taken from the NetField portion of the address and indicate whether the destination is static and represent a multicast address. The ML is an evolved version of the Home Location Register HLR and is responsible for tracking the mobile terminal over different networks, Table 9.2 shows the structure of the ML.

Internet Name	Node_ID	S	M	SF	Location_ID	ML's Address
Name1	Node_ID1	1	0	01	Location_ID1	ML- Add
Name2	Node_ID2	1	0	10	Location_ID2.1 Location_ID2.2	ML- Add
Name3	Node_ID3	0	0	11	Location_ID3	ML- Add
Name4	Node_ID4	0	0	10	Location_ID4.1 Location_ID4.2 Location_ID4.3 Location_ID4.4	ML- Add

Table 9.1: The eDNS Record

Node_ID	Location_ID	INF	Mobility Vector	QoS Specifications
Node_ID1	Location_ID1.1 Location_ID1.2	INF1.1 INF1.2	Value1 Value2	QoS-Spec1 QoS-Spec2
Node_ID2	Location_ID2.1 Location_ID2.2	INF2.1 INF2.2	Value3 Value4	QoS-Spec3 QoS-Spec4

Table 9.2: The ML Record

9.2.2 Specifying the Ring-Based Model

The Ring-Based Model presented in Chapter 8 used the scope concept, derived from the (HB05), to limit servers' visibility over the network based on their functionality. This section describes how the scope concept and hence the Ring-Based Model could be implemented using the new addressing scheme and the SF field to define the server scope.

However, for the proposed security model to work in future, heterogeneous environment such as the one in Chapter 4, the SF field in the new address must be redefined and mapped to the network structure. In this structure of the network, a server could be local, accessed by clients in the same peripheral network or by clients residing within the Core-End Point (CEP)or located at the same site. The server might also be global and thus, accessible globally over the Internet. By considering the value of the (SF) field in the addressing scheme, the four scopes could be redefined as follows:

- SF=00: indicates that the server could only access locally and thus it's in a Local scope.
- SF=01: defines a LAN scope which means that the server could be accessed by the nodes in the same peripheral network. So the Location _ID should be the local LAN.

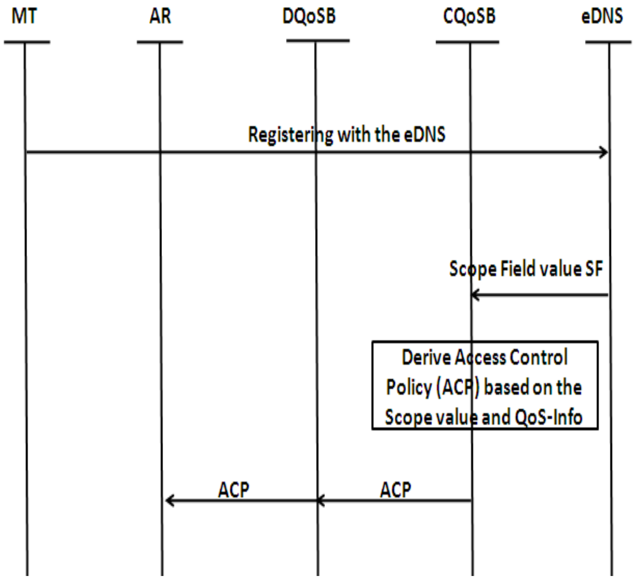
MT
AR
DQoSB
CQoSB
eDNS
Registering with the eDNS
Scope Field value SF
Derive Access Control Policy (ACP) based on the Scope value and QoS-Info
ACP
ACP

Figure 9.2: Enforcing the Reachability Based on the Scope

- SF=10: denotes that the server could only be accessed by devices residing within the same Core End-Point. So Location_ID must be a site Address.
- SF=11: means that the server is globally accessible over the Internet.

The following sections explain how the servers' accessibility could be implemented and enforced using the new addressing scheme and the hierarchical network structure. More details could be found in (MA11d).

The host registration and devising the Access Policy

As shown Fig 9.2, in this stage, the hosts register themselves in a global naming system such as the proposed eDNS along with the corresponding Scope Field's value that reflects their scope, this information is passed to the High-level Access Admission (HAAD) module in the Central QoS Broker (CQoSB). The HAAD uses the defined scope along with the Service Level of Agreement (SLA) information, retrieved from the QoS Engine to devise an Access Control Policy (ACP). The ACP is passed from the HAAD all the way to the Access Admission Enforcement (AAE) module in the Access Router (AR) using policy-conveying protocols such as the COPS protocol.

Enforcing the Access Policy

At the end of the previous stage, the end-hosts should have been registered with the eDNS along with their desired accessibility. Additionally, based on the accessible scope and other

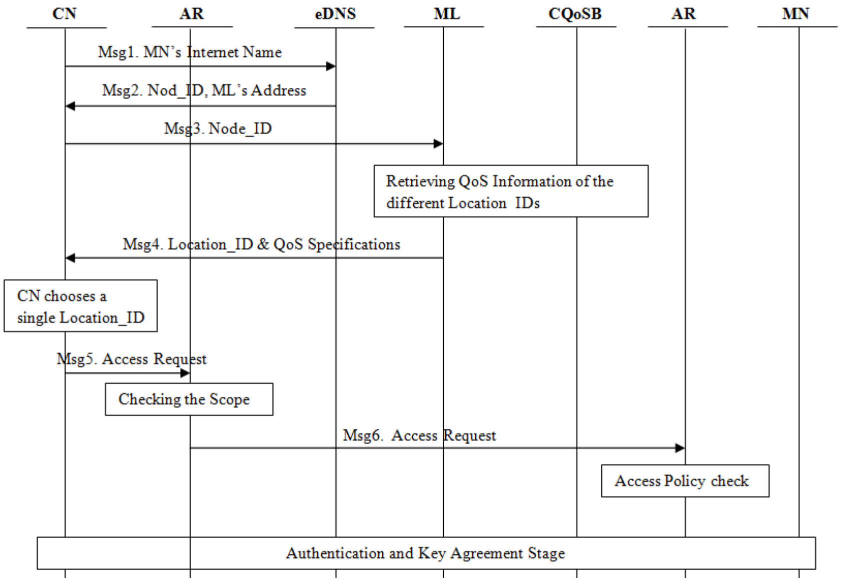
CN
AR
eDNS
ML
CQoSB
AR
MN
Msg1. MN's Internet Name
Msg2. Nod_ID, ML's Address
Msg3. Node_ID
Retrieving QoS Information of the different Location IDs
Msg4. Location_ID & QoS Specifications
CN chooses a single Location_ID
Msg5. Access Request
Checking the Scope
Msg6. Access Request
Access Policy check
Authentication and Key Agreement Stage

Figure 9.3: Enforcing the Scope Concept in The Ring-Based Model

QoS-related information, an access policy will be devised by the HAAD and transferred to the enforcement module in the AR which will, based on the policy, accept or drop access requests.

Fig 9.3 shows the transaction in the case of the Corresponding Node (CN) trying to connect to a mobile node (MN). For this scenario, we presume that, MN's scope might be any of the LAN, Domain or Global.

- **Msg1**: The CN asks the eDNS server for MN's address.

- **Msg2**: The eDNS uses the MN's name to look up its database and since the MN is a multi-homed device, its name will be resolved to different addresses with the same Node_ID, this implies that the MN is accessible over different routes/ networks. However, the eDNS cannot define the best route for the connection. Therefore, the eDNS returns the MN's Node_ID and the address of the Master Locator(ML) that manages the mobility of MN.

- **Msg3**: The CN polls the ML to find out the different networks to which the MN is currently attached. The ML approaches the CQoSB to get QoS-related information about the MN's different networks. Upon receiving this information, the ML sets the INF bits and thus maps the Location_ID to the interface address.

- **Msg4**: A list of MN's Location_IDs along with their QoS specifications is passed to the CN, which chooses the route to the MN and thus defines the corresponding

Location_ID.

- **Msg5**: Since the CN has the MN's full address, the CN can start the connection by sending Access Request to the MN. This request will be intercepted by the AR in the source network which checks the Scope Field in the destination address. Based on the SF value, if MN was accessible for CN the access request packet is forwarded, otherwise, it is dropped.

- **Msg6** When the access request gets to the destination network, the AR will check whether the request complies with the access policy or not. If the request passes the check, the AR passes it to the MN.

However, before the CN could use the service on MN, there is a need to achieve mutual authentication and set a secure session between the CN and the MN. This could be achieved using Authentication and Key Agreement (AKA) protocols similar to the one in (MA11b).

Model Analysis and Attacks Modelling

This section will describe different attack scenarios and show how our security model reacts to them.

- The first scenario is the case of a Denial Of Service (DOS) Attack where a single Corresponding Node (CN) is trying to access a server with a LAN/Site scope. Obviously, if the CN was not in the server's scope, its connection request will be dropped by the Access Routers. Otherwise, it could communicate with the server. However, if the CN initially claims more QoS than it is allowed to, this will be detected by the Access Admission Enforcement (AAE) as violation to the access policy. Furthermore, after making the connection, if the CN tries to abuse the network and exceeds the reserved QoS, this will be detected by the Network Monitoring Entity (NME) module in the Access Router, consequently, the CN will be blacklisted.

- A similar discussion applies if the server was global.

- The third scenario considers the case of a Distributed Denial of Service (DDOS) attack where multiple corresponding nodes attempt to access a server with LAN or Site scope. Only corresponding nodes in the scope of the server could communicate with the server. However, in the case where a large number of legitimate nodes managed to access the server, they could overload the server and launch a DDOS attack despite the fact that none of the nodes has individually exceeded the agreed QoS.

Table 9.3: Analysis Summary

Scope	**DOS**	**DDOS**
LAN/ Site	Fully Mitigated	Still Possible
Global	Fully Mitigated	Partially Mitigated

- In case of a server with a global scope, the previous DDOS attack could still be achievable.

Although the proposed model is based on the 'Off By Default' concept, it has avoided most of its drawbacks as explained in (MA11d) by being fully integrated with the network infrastructure and considering the multi-homing nature of future mobile devices. Furthermore, by deploying the new address scheme, which uses the Node_ID to identify the device, if a multi-homed CN attempts to start multiple sessions with the server using different network interfaces, the network and thus the server will be able to co-locate these sessions to the same CN and thus monitor the resources utilization over the different sessions. Table 9.3 summarizes the analysis result.

9.3 The Problems of Implementing Security Protocols

With the increasing popularity of wireless network technologies such as 3G, Wimax and LTE, mobile devices are expected to use these technologies to maintain connectivity to various types of services. From the viewpoint of the user, ease of use, service flexibility and secure communication are the main desired features. To deal with the security issue, there is a need to make sure that any implemented security protocols will achieve desired security goals such as confidentiality, authentication and integrity. Therefore, various formal methods such as BAN, CSP and FDR (MB90), (Hoa85), (Sys93) have been developed to verify security protocols.

In spite of the successful verification of the protocol design, there can be some errors in its implementation that can be exploited by intruders. This could be due to the mistakes made by the programmers or bugs in the programming language itself. For instance, in 2002, a Linux worm known as Linux.Slapper.Worm exploited a flaw in the secure sockets layer (SSL), which is commonly used by e-commerce sites to secure transactions between the customer's computer and the company's server. The worm does not exploit an attack against the SSL protocol itself, but is a buffer overflow attack that works only against this specific implementation (Did09), this vulnerability did not concern the security protocol itself, but its imperfect implementation. Another example was stated in (CJ05), a flaw pertaining to a buffer overflow attack was found in the OpenSSH code of the SSH protocol. Once again, it is not an attack against the protocol itself, but against some of its implementations. Furthermore, implementing security protocols is an exhaustive and error-prone process which requires a hard-core programming experience.

Accordingly, it would be interesting to have a tool that could generate the implementation automatically to remove the risk of error in this step, and also to make implementation faster and easier. Therefore, the CPSP-J and the ACG-C# compilers have been proposed in the literature to generate a program code automatically from a high-level specification of security protocols that has been verified using Casper/FDR tool.

9.4 Potential Compilers for Security Protocols

9.4.1 COSP-J Compiler

As proposed in (Did09), the COSP-J is meant to to be used in addition to Casper in order to first analyse a protocol and then compile it. Nevertheless, COSP-J and Casper provide two levels of abstractions; Casper considers a set of agents that communicate with each other, without actually considering what these agents physically are (i.e. devices or users); furthermore, Casper does not consider how the agents communicate and the nature of the communication protocols between the agents. COSP-J provides a more concrete level of abstraction, as it considers the nature of the agents, the communication protocols. For instance, with COSP-J there is a clear distinct between the user and the device (a user might log on from different machines), this difference could be shown in Fig 9.4

COSP-J Input File

As shown in Fig 9.5, the COSP-J is very related to Casper; its input file is adapted from the general structure of the input files of Casper. The input file of the COSP-J tool comprises the following headings:

- The # Protocol Description heading: This heading lists the messages to be exchanged between the participating agents. It differs from the # Protocol Description heading of Casper in the two points:

  1. We do not use the environmental message 0 anymore in COSP-J, this is due to the fact that the difference between an environmental message and an initial starting

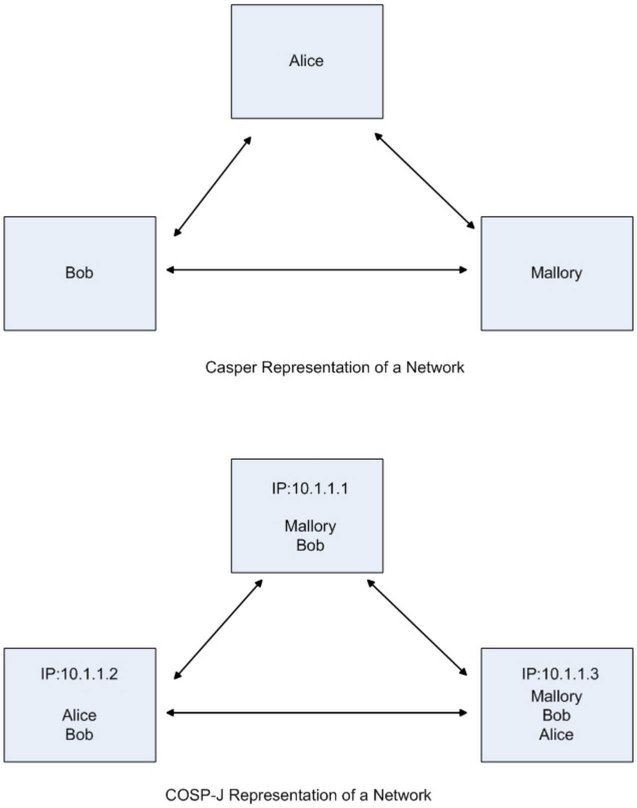

Figure 9.4: COSP-J and Casper Abstraction (Did09)

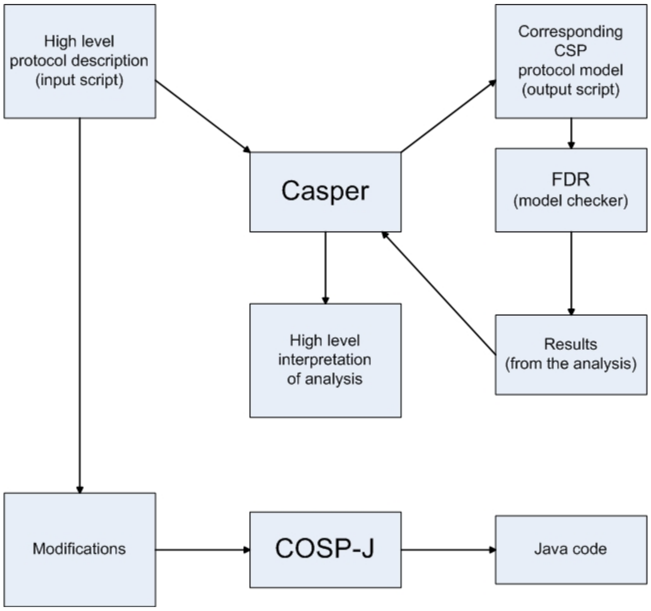

Figure 9.5: The use of Casper and COSP-J together (Did09)

assumptions is important for the analysis of a protocol with Casper, but does not change anything for an implementation (Did09).

2. To indicate the completion of a protocol, COSP-J proposed a way to express what the results are of the protocol run once it is finished. The method in the implementation returns these values to the calling code. These values may be useful after the end of the protocol, for example if this implementation of the protocol is just a part of a big network application (Did09).

- The # Free Variables heading: This section describes the types of the variables and functions that are used in the protocol definition. For COSP-J, we need to be more specific about the types of functions and variables being used, such as defining the type of the encryption and hashing functions.
- The # Processes Heading: This heading gives some information about the agents running the protocol and their initial starting assumptions. For COSP-J, there is a need to define which values are fed initially to the agents and which are generated during the run of the protocol.
- The # Functions heading: This heading defines the functions used in the protocol.
- The # External heading: This is a new heading; does not exist in Casper, and it describes the variables that are used in the input file and that will be provided at runtime.

COSP-J Output Code

The output code of COSP-J will be a script written in Java, the output defines a list of actions each agent should perform for each message in the protocol. Based on this, an agent could be of any of the following cases:

- Case 1: If the agent is not sending nor receiving, no action is assigned to the agent.
- Case 2: If the agent is sending, two actions are required; the agent needs to build the message out of its fields and then compose it. The message might contain an atomic (i.e. a nonce value, an agent identity) or an encrypted value. In the latter type, the agent has to build the sub-message first and then encrypt it.
- Case 3: If the agent is receiving a message, it decomposes and decrypts it, stores the values that were unknown and checks if the values that were already known are correct. (Did09).

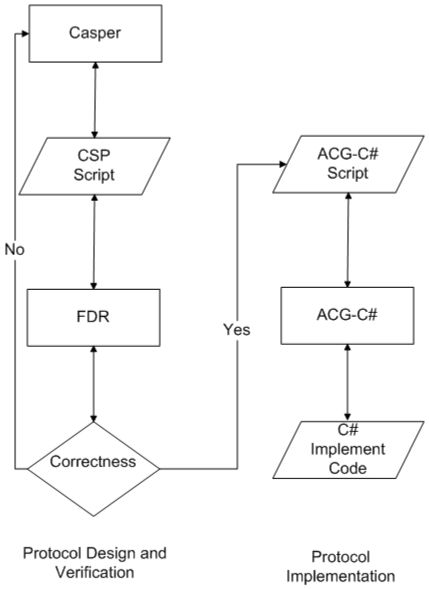

Figure 9.6: Security Protocol Design and Implementation (CJ05)

9.4.2 The ACG-C#

The ACG-C# is a tool, used to automatically generate the C# implementation code for security protocols from the high-level specification written in a variation of Casper notation. This ACG-C# tool compiles the specification to produce C# code that is a concrete implementation of the protocol.

As shown in Fig 9.6, the overall process used to verify the safety of the security protocol and to implement it. The entire process is composed of two sub processes, which are referred to as protocol design and protocol implementation. To guarantee the safety of the security protocol in the design phase, we first make an abstract model of the protocol with Casper and then generate CSP code with the compilation function of Casper. Next, we can run the FDR model checking tool to verify whether the security protocol satisfies the security properties or not. If the security properties are satisfied, then the designer inputs the slightly modified Casper script into the ACG-C# tool. In the implementation phase, the ACG-C# tool automatically generates the C# implementation code for the security protocol.

9.4.3 Comparing COSP-J and ACG-C#

These tools are automated compilers, related to Casper/FDR tool; they accept a CSP-like description of the security protocol and generate a code program for implementing the protocol, While COSP-J produces a Java code, ACG generates a C# file. In this sense, the differences between these tools will be based on the features of the corresponding language.

Furthermore, both compilers claim to offer many desired advantages such as No error-prone, Secure code and High confidence (CJ05). In this sense, deciding on the suitable compiler would be a matter of personal preference.

9.5 Using Ontology to Support the Integration

9.5.1 An Overview of Ontologies

An ontology is an explicit specification of a conceptualization (Gru93), it communicates a common understanding of a domain; declare explicit semantics, makes expressive statements, and supports sharing/reusing of knowledge (Lac05)

9.5.2 How Would an Ontology Help!

The research in this thesis dealt with three different domains; the architectural design of the communication framework, the QoS signalling as well as security provision. This implied that research efforts at different domains have to cooperate together. The efficient cooperation in this situation requires the presence of a common definition of concepts across the different domains, this in turn will aid the integration of the various research efforts.
Ontology helps in such situations because it provides the following:

- **It provides common understanding of the structured information among researchers of different domains:** An ontology deals with the issue of misunderstanding by formally describing terminological concepts and their relationships that characterise a domain. It formally corroborates one common understanding of a domain and defines semantics independent of the reader (human or computer) and context (Cho07).

- **Explicit Semantics:** By explicitly defining the main concepts in a domain and the relationships between those concepts, it can make expressive statements about the domain model. This way of documenting concepts with modelling primitives and semantic relationships assists in interpretation during information sharing in heterogeneous IT systems.

- **Sharing/reusing of knowledge:** The development of an ontology by a certain group will give the chance to other groups to reuse the ontology in their domain,and thus save time and effort.

9.5.3 An Ontology for Y-Comm

In order to use Y-Comm framework, as with any new model, novices will have to spend time understanding the approach being taken including the diversity of concepts and relationships. In addition, the use of ontologies to formalize Y-Comm and to specify semantic meanings should reduce the study time and the misunderstanding of definitions.

Therefore, work has already started on developing a new ontology for Y-Comm to facilitate the interoperability among different areas of service provision (access, content, and brokerage), ease the tasks for vertical handover management and mitigate misunderstanding of the shareable information from different providers.

The author believes that by using Y-Comm's ontology, new ontologies could easily be introduced to define the security and QoS mechanisms. This in turn will aid the integration between the different domains, as it will clearly show the potential semantic relation between objects across the domains.

9.6 Summary

This chapter describes the COSP-J and ACG-C#, two automated code-generating tools which are related to Casper/FDR; they accept a script description of the verified protocol and generate the corresponding code program in Java and C#, respectively. Since, the proposed AKA protocols in this research were verified using Casper/FDR, the author believes that any of these tools could be used to generate the code program of the protocols and thus have a chance to verify their implementations. The chapter also describes how a full integrated solution for security and QoS could be defined using ontologies. In order to implement the Ring-Based Model, the chapter describes potential enhancements on network addressing scheme, location and naming systems to implement and enforce the accessibility scopes. The analysis section shows that the proposed model succeeds in stopping DoS attacks while is also partially effective in addressing Distributed DOS attacks.

Chapter 10

Conclusions and Future Work

10.1 Introduction

This final chapter provides a succinct summary of the main ideas proposed in this thesis, the results, noteworthy achievements and the future applications of the proposed novel concepts. It captures the main theme of this research study and shows how it succeeded in answering the research questions.

10.2 How were the key research questions addressed?

The research study identified crucial gaps in addressing the issue of providing security and QoS in future heterogeneous networks. It revealed how integrating security and QoS is crucial to provide secure connectivity in heterogeneous environments. Furthermore, due to the open dynamic nature of heterogeneous networks, the research highlighted the need for addressing the issue of security and QoS provision in different scenarios. These issues were then embedded into the three important research questions

- **What is the architecture of future, heterogeneous networks and how to guarantee an end-to-end QoS provision for mobile terminals in these networks ?**

The answer to these questions is a new QoS framework proposed in Chapter 4. Furthermore, a new hierarchical architecture of the future networks has been introduced, the architecture defined the structure of the main networks entities, required for security and QoS functions. To provide an End-to-End QoS in this environment, three Targeted QoS-Signalling models have been proposed. The models will deal with QoS provision in three situations; the Initial Registration, the Connection Initiation and the case of Handover.

- **What are the underlying protocols of the Y-Comm's Integrated Security Module (ISM) and how these protocols are integrated together ?**

As stated in Chapters 5, 6, 7, to answer these questions, research has been done to define the security module's underlying protocols, part of which is represented by an Authentication and Key Agreement framework that provides AKA functions at network and application levels and in different situations such as handover and connection initiation. All the proposed security mechanisms have been verified by using formal methods approach based on the Casper/FDR. Furthermore, the integration between these protocols was defined by an Authentication and Key Agreement Framework.

- **Considering the open nature of future network, how to provide security and support the integration between security and QoS without limiting the flexibility and dynamics of these networks ?**

The answer to this question was to propose three Targeted Security Models, namely the Connection, the Secure Vertical Handover and the Ring-Based models (Chapters 8). As the names imply, the first two models address the security and QoS provision in the case of connection initiation and in the case of handover, respectively. They were defined by integrating the AKA framework with the QoS-Signalling models. The Ring-Based model aims at protecting the end-servers from unsolicited traffic such as SPAM.

10.3 Main Contributions

As stated in Chapter 1, the main contributions of this research study include:

- A critical review of existing solutions in the areas of security and QoS provision for heterogeneous networks. This review uncovered important deficiencies that hampered the successful realisation of the significance of QoS and security integration in future communication systems.
- A proposed hierarchical architecture for heterogeneous networks, the architecture defined the main operational network entities and their structure.
- Based on the proposed network architecture, a QoS framework was introduced. The framework defined three Targeted-QoS signalling models which provided QoS in different scenarios such as the initial registration, connection initiation and in the case of handover.

- A new addressing scheme for multi-homed device, which identified the device regardless of its network interfaces.

- Similar to the Targeted QoS-Signalling Models, three Targeted Security Models, namely the Connection security model, the handover model and the Ring-Based model were proposed. While the first two provided security in the case of connection initiation and handover, respectively, the latter protected the servers as well as network entities from being overloaded or abused. The underlying security protocols in these models were verified and analysed using Casper/FDR compiler as a well established model checker.

- Describing how QoS and security could be integrated as part of the proposed Targeted Security Models.

10.4 Elaboration on the main contributions

10.4.1 Identification of crucial gaps in knowledge in the field of providing QoS and Security in heterogeneous environments

The study conducted an exhaustive literature survey of related work in the areas of QoS, security and vertical handover support in heterogeneous environments. The study highlighted the following crucial drawbacks in the investigated approaches:

- Lack of cooperation; various research efforts addressed the QoS, security and vertical handover separately without actually considering integrating their effort in one unified solution.

- Uncertainty about the architecture of heterogeneous environments, which led to many abstract solutions that did not reflect into a clear network architecture or to scenario-specific solutions.

- Poor realization of the open and flexible nature of heterogeneous networks; these unique features should be considered in the design of the QoS and security mechanisms so no conflict might result of implementing them.

10.4.2 Defining a generic structure of heterogeneous networks

In order to propose practical mechanisms for addressing the security and QoS provision, our research defined a generic architecture for heterogeneous networks in Chapter 4. The proposed hierarchical architecture defined the operational network entities, their structures as well as the required interfaces between them.

10.4.3 Introducing a QoS framework

Having defined the network architecture, a novel QoS framework was proposed in Chapter 4, which realized the dynamic nature of future networks and thus introduced a dynamic approach for providing QoS. The approach introduced the concept of the Targeted QoS-Signalling models which will provide QoS in different scenarios; therefore, three models have been introduced: the Initial Registration Model, the Connection Initiation Model and the Handover Model.

10.4.4 Investigating the Multi-homing issue of Future Mobile Devices

This study proposed a new addressing scheme that identifies mobile devices regardless of their network interfaces. Furthermore, it defined the required naming and locating services to support the new address. These enhancements will aid in implementing some of the security mechanisms introduced in this thesis.

10.4.5 Providing Security in Heterogeneous network

In order to provide security in future network based on the proposed network structure, an Authentication and Key Agreement Framework was introduced in Chapter 5. The framework provided security at three levels:

- The User-Level: An Authentication and Key Agreement (AKA) protocol was introduced to achieve mutual authentication between the user, mobile terminal and the Personal ID Card (PIC).

- The Service-Level: Two AKA protocols were introduced in Chapter 7 to secure the transaction between the mobile terminal and the end-server (called Service Provider). While the first AKA protocol addressed the initial authentication when the mobile terminal contacts the service provider for the first time from its access network, the second is a lightweight pre-authentication protocol, which considers the mobility of the mobile terminal and re-authenticates the mobile terminal to the service provider when it changes its access network.

- The Network-Level: The proposed protocols for network-level security in Chapter 6 are responsible for achieving mutual authentication between the mobile terminal and the network in two cases: the initial access, when the mobile terminal joins the network for the first time and in the case of handover when the mobile terminal joins the access network coming from another network.

All the protocols of the AKA framework have been formally verified using Casper/FDR compiler. The choice of using Casper/FDR was based on the fact that this tool has been widely acknowledged and hence, used to verify many security protocols as stated in (LBDH09). Furthermore, due to the multi-homed feature of future mobile devices, the Ring-Based Security Model adopted the new addressing scheme in Chapter 9.

10.4.6 Defining the Target Security Models

The Connection and the Vertical handover security models were defined by integrating the AKA framework and the QoS-Signalling models as described in Chapter 8.

10.5 Future improvements to solutions from which the study can benefit

This section lists a set of improvements to the proposed solutions and related work upon which this study is based, which will result in a further improvement in performance:

- Analysing the performance of QoS-Signalling models. This could be achieved using analytical modelling or simulation tools such as OPNET or NS-2 (Abo07), (TI09).

- Defining the cryptographic algorithms as well as the mathematical algorithms in the key derivation functions. This involves comparing different algorithms and analysing how these might affect the performance of the whole protocol.

- Enhancing the current naming and locating services to cope with the mutli-homed nature of the mobile devices. Examples of such enhancements are represented by the enhanced DNS (eDNS) and the Master Locator (ML) in Chapter 9.

10.6 What are the future works that can be pursued based on this study?

The solutions proposed in this study have led to the discovery of a new set of key mechanisms which enhance user experience and provide a secure environment for communication in future heterogeneous networks. The successful derivation of these new mechanisms has created new opportunities for improved research activity in different areas of QoS, security management and handover optimisation. This section looks at how the proposed research ideas can develop further.

10.6.1 Implementing the new addressing scheme and the Ring-Based Model

In order to explore this further, the author considers building a testbed composed of mobile devices and Linux routers capable of supporting 3G using OpenBTs (DB08) as well as Wi-Fi networks. This would be used to implement the new addressing scheme along with concept of scope in the Ring-Based model.

10.6.2 Implementing the Connection and the Vertical handover security model

As explained in Chapter 9, two code-generating compilers namely, the COSP-J and ACG-C# have been introduced to generate Java and C# code of the protocols verified by Casper/FDR. The author considers using COSP-J to have the Java executable programs of the proposed AKA protocols, and then run these protocol on smart phones with Java platform.

10.7 Concluding remarks

This thesis has addressed the key issues of providing security and QoS in heterogeneous environment. I hope that this contribution will play a significant role in the development of future mobile heterogeneous networks.

APPENDIX A

The UL-AKA For the Mobile Ethernet

```
# Free Variables
PIC, MT : Agents
r1, r2, r3 :  Nonce
miD : DeviceID
K : SessionKeys
h :  HashFunction
Req:  message
InverseKeys = (K, K)
# Processes
INITIATOR(PIC,r1, K )
RESPONDER(MT,PIC, r2, miD, K, Req)
# Protocol Description
0.  -> PIC : MT
1.  MT -> PIC : Req
2.  PIC -> MT : r1
3.  MT -> PIC : {r1}{K}%v, h(r1), r2
[decryptable(v, K)andnth(decrypt(v, K), 1) == r1]
4.  PIC -> MT : {r2}{K}%w
[decryptable(w, K)andnth(decrypt(w, K), 1) == r2]
# Specification
Secret(PIC,K,[MT])
Secret(PIC,r2,[MT])
Secret(PIC,r1,[MT])
Agreement(MT,PIC,[r1, K])
Agreement(PIC,MT,[r2, K])
WeakAgreement(MT,PIC)
WeakAgreement(PIC,MT)
# Actual Variables
PICard, Mobile, Eve :  Agents
R1,R2, R3 :  Nonce
MID : DeviceID
k :  SessionKeys
InverseKeys = (k, k)
req:  message
# System
INITIATOR(PICard,R1, k)
```

```
RESPONDER(Mobile,PICard, R2, MID, k, req)
```

# Intruder Information

```
Intruder = Mallory
IntruderKnowledge = {PICard, Mobile}
```

APPENDIX B

The Proposed UL-AKA Protocol

# Free Variables

```
PIC, MT : Agents
r1, r2 :  Nonces
r3 :  challNonce
SK : Agents -> presharedKeys
F : presharedKeys x Nonces x Nonces x DeviceID -> SessionKeys
miD : DeviceID
K : SessionKeys
h :  HashFunction
Ackm:  Acknolwedgment
InverseKeys = (K, K), (SK, SK),(F, F)
```

# Processes

```
INITIATOR(PIC,r1,r3,Ackm) knows SK(MT)
RESPONDER(MT,PIC, r2, miD, Ackm) knows SK(MT)
```

# Protocol Description

```
0.  -> PIC : MT
1.  PIC -> MT : {r1}{SK(MT)}
```

$< K := F(SK(MT), r1, r2, miD) >$ `2.  MT -> PIC : {miD,r2,r1}{SK(MT)}`

$< K := F(SK(MT), r1, r2, miD) >$ `3.  PIC -> MT : {r2,r3}{K}%v`

$[decryptable(v, K)andnth(decrypt(v, K), 1) == r2]$

$< r3 := nth(decrypt(v, K), 2) >$

```
4.  MT -> PIC : ({Ackm, r3}{K})%w
```

$[decryptable(w, K)andnth(decrypt(w, K), 1) == r3andnth(decrypt(w, K), 2) == Ackm]$

```
5.  PIC -> MT: {Ackm}{K}%w1
```

$[decryptable(w1, K)andnth(decrypt(w1, K), 1) == Ackm]$

# Specification

```
Secret(PIC,SK(MT),[MT])
Secret(PIC,miD,[MT])
Secret(PIC,K,[MT])
Agreement(MT,PIC,[r3])
Agreement(PIC,MT,[r2])
WeakAgreement(MT, PIC)
WeakAgreement(PIC, MT)
Aliveness (PIC, MT)
Aliveness (MT, PIC)
# Actual Variables
PICard, Mobile, Mallory :  Agents
R1,R2:  Nonces
R3, R4 :  challNonce
MID : DeviceID
k :  SessionKeys
InverseKeys = (k, k)
ACKM: Acknolwedgment
# Functions
symbolic SK, F
# System
INITIATOR(PICard,R1,R3, ACKM)
RESPONDER(Mobile,PICard, R2, MID, ACKM)
# Intruder Information
Intruder = Mallory
IntruderKnowledge = PICard, Mobile,R1, F
```

APPENDIX C

The Mobile Etherent's Initial NL-AKA Protocol

```
# Free Variables
M: MobileTerminal
EP : AccessRouterAuthenticator
AS : DomainA3CServer
AIS : CentralA3CServer
AuthID : Identity
```

```
Initauth :  Flags
R1 :  initialSeq
R2 :  Sequence
UUK : MobileTerminal-> PresharedKeys
AK : AuthenticationKeys
SK : SecretKeys
MS: Domainspecifickey
RPF: PresharedKeys x initialSeq -> Domainspecifickey
rpf:  initialSeq x Domainspecifickey ->
AuthenticationKeys
F3:  initialSeq x Domainspecifickey -> SecretKeys
h :  HashFunction
AccReq, AccRes,AuthReq, Adv:  Messages
HoAckm :  AcknowledgementMessage
InverseKeys = (AK, AK), (UUK, UUK) , (SK, SK),
(MS, MS), (RPF,RPF), (rpf,rpf),(F3,F3)
# Processes
INITIATOR(M, EP, R1,AuthID,Initauth, AccReq, AuthReq) knows UUK(M)
Authenticator(EP,M,AS, AuthReq, Adv,AccRes)
DomainSERVER(AS,AIS, R2, HoAckm)
CentralSERVER(AIS) knows UUK(M)
# Protocol Description
0.  -> M : EP, AIS, AS
1.  M -> EP: AccReq
2.  EP -> M : AuthReq
< MS := RPF(UUK(M), R1);
AK:= rpf(R1, MS)>
3.  M -> EP : M,R1
4.  EP -> AS : M,R1, h(M,R1)
5.  AS -> AIS : M,R1, R2, h(M,R1,R2)
< MS := RPF(UUK(M), R1)>
6.  AIS -> AS : MS,M, h(MS, M)
< AK:= rpf(R1, MS)>
7.  AS -> EP: R2,({R1}{AK}%z)%x, h(R2,({R1}{AK}%z)%x)
8.  EP -> M : x%(R1AK%z), R2
[decryptable(z,AK)andnth(decrypt(z,AK),1) == R1]
```

```
< SK := F3(R1, MS) >
9.  M -> EP : ({R2}{AK}%y)%q
10.  EP -> AS: (q%{R2}{AK})%y, h((q%{R2}{AK})%y)
[decryptable(y, AK)andnth(decrypt(y, AK), 1) == R2]
< SK := F3(R1, MS) >
11.  AS -> EP :HoAckm, h(HoAckm)
12.  EP -> M : HoAckm
# Specification
Secret(M,AK,[AS])
Secret(AS,AK,[M])
Secret(M,SK,[AS, EP])
Agreement( M, AS, [R2])
Agreement(AS, M, [AK, R1])
WeakAgreement (EP, M)
WeakAgreement (M, EP)
Aliveness (EP, M)
Aliveness (M, EP)
# Actual Variables
m, Eve:  MobileTerminal
ep :  AccessRouterAuthenticator
as :  DomainA3CServer
ais :  CentralA3CServer
Authid :  Identity
InitAuth :  Flags
r1 :  initialSeq
r2 :  Sequence
ak :  AuthenticationKeys
sk :  SecretKeys
ms:  Domainspecifickey
accReq, accRes,authReq, adv:  Messages
hoackm :  AcknowledgementMessage
InverseKeys = (ms, ms), (ak, ak), (sk, sk)
# Functions
symbolic UUK, RPF, rpf, F3
# System
INITIATOR(m,ep, r1,Authid,InitAuth, accReq,
authReq)
Authenticator(ep,m,as, authReq,adv, accRes)
DomainSERVER(as,ais, r2,hoackm)
CentralSERVER(ais)
# Intruder Information
Intruder = Eve
IntruderKnowledge = m, as, Eve, ais, Authid,ep,
UUK(Eve)
Crackable = PresharedKeys
Crackable = Domainspecifickey
```

APPENDIX D

The Proposed Initial NL-AKA Protocol

```
# Free Variables
MT: MobileTerminal
Auth :  AccessRouterAuthenticator
DA3C : DomainA3CServer
CA3C : CentralA3CServer
AuthID : Identity
Initauth :  Flags
seq1 :  initialSeq
seq2 :  Sequence
se1:  DomainA3CServer-> PresharedKeys se2 :  AccessRouterAuthenticator-> PresharedKeys
uk :  MobileTerminal-> PresharedKeys
AK : AuthenticationKeys
SK : SecretKeys
DSMK: Domainspecifickey
F1:  PresharedKeys x initialSeq x Identity ->
Domainspecifickey
F2:  initialSeq x Domainspecifickey ->
AuthenticationKeys
F3:  initialSeq x Domainspecifickey
x Identity -> SecretKeys
```

```
AccReq:AccReqMessage
AccRes:AcceResMessage
AuthReq:AuthReqMessage
Adv:  AdvMessages
HoAckm :  AcknowledgementMessage
InverseKeys = (AK, AK), (uk, uk) , (SK, SK),
(DSMK, DSMK),(se2, se2), (F1,F1), (F2,F2),(F3,F3), (se1,se1)
```

# Processes

```
INITIATOR(MT,seq1,AuthID,Initauth, AccReq, AuthReq)
knows uk(MT)
Authenticator(Auth,MT,DA3C, AuthReq, Adv,AccRes) knows se2(Auth)
DomainSERVER(DA3C,CA3C, seq2, HoAckm) knows se2(Auth), se1(DA3C)
CentralSERVER( CA3C) knows uk(MT), se1(DA3C)
```

# Protocol Description

```
0.  -> MT : Auth, CA3C
1.  Auth -> MT :Adv,DA3C
< DSMK := F1(uk(MT), seq1, AuthID)>
2.  MT -> Auth:  AccReq
3.  Auth -> MT : AuthReq
<AK:= F2(seq1, DSMK)>
4.  MT -> Auth :  MT,seq1,AuthID, MT, Initauth
5.  Auth -> DA3C : {MT,seq1,AuthID, MT, Initauth}{se2(Auth)}
6.  DA3C -> CA3C : {MT,seq1,AuthID, MT, Initauth}{se1(DA3C)}
< DSMK := F1(uk(MT), seq1, AuthID)>
7.  CA3C -> DA3C : {DSMK, seq1,AuthID, MT, Initauth}{se1(DA3C)}
< AK:= F2(seq1, DSMK)>
8.  DA3C -> Auth:  {({seq2, seq1}{AK}%z)%x}{se2(Auth)}
9.  Auth -> MT : x%({seq2, seq1}{AK}%z)
```

$[decryptable(z, AK)andnth(decrypt(z, AK), 2) == seq1]$

$< SK := F3(seq1, DSMK, AuthID);$

$seq2 := nth(decrypt(z, AK), 1) >$

```
10.  MT -> Auth :  ({seq2}{AK}%y)%q
11.  Auth -> DA3C: {(q%{seq2}{AK})%y}{se2(Auth)}
```

$[decryptable(y, AK)andnth(decrypt(y, AK), 1) == seq2]$

$< SK := F3(seq1, DSMK, AuthID) >$

```
12.  DA3C -> CA3C : {HoAckm}{se1(DA3C)}
```

```
13.  DA3C -> Auth :{SK}{se2(Auth)}
14.  Auth -> MT : {AccRes}{SK}
# Specification
Secret(MT,AK,[DA3C])
Secret(DA3C,AK,[MT])
Secret(MT,SK,[DA3C, Auth])
Agreement( MT, DA3C, [seq2])
Agreement(DA3C, MT, [AK])
WeakAgreement(MT, Auth)
WeakAgreement(Auth,MT)
WeakAgreement(Auth, DA3C)
WeakAgreement(DA3C, Auth)
Aliveness (Auth, MT)
Aliveness (MT, Auth)
# Actual Variables
mt, Mallory:  MobileTerminal
auth :  AccessRouterAuthenticator
da3c :  DomainA3CServer
ca3c :  CentralA3CServer
Authid :  Identity
InitAuth :  Flags
SEQ1 :  initialSeq
SEQ2 :  Sequence
ak :  AuthenticationKeys
sk :  SecretKeys
DSMK: Domainspecifickey
accReq:AccReqMessage
accRes:AcceResMessage
authReq:AuthReqMessage
adv:  AdvMessages
hoackm :  AcknowledgementMessage
InverseKeys = (DSMK, DSMK), (ak, ak), (sk, sk)
# Functions
symbolic uk, se2, se1, F1, F2, F3
# System
INITIATOR(mt,SEQ1,Authid,InitAuth, accReq, authReq)
```

```
Authenticator(auth,mt,da3c, authReq,adv, accRes)
DomainSERVER(da3c,ca3c, SEQ2,hoackm)
CentralSERVER(ca3c)
```

# Intruder Information

```
Intruder = Mallory
IntruderKnowledge = {mt, da3c, Mallory, ca3c, Authid, auth, uk(Mallory), F2, F3}
Crackable = PresharedKeys
Crackable = Domainspecifickey
```

APPENDIX E

The Mobile Ethernet Secure NL-AKA for Vertical Handover

# Free Variables

```
M: MobileTerminal
EP : AccessRouterAuthenticator
AS : DomainA3CServer
AuthID : Identity
Initauth :  Flags
R1 :  initialSeq
R2 :  Sequence
HOAID1:  OldToken
HOAID2:  NewToken
AK : AuthenticationKeys
SK : SecretKeys
MS: Domainspecifickey
F: AuthenticationKeys x initialSeq x Sequence -> NewToken
h :  HashFunction
AccReq, AccRes,AuthReq, Adv:  Messages
HoAckm :  AcknowledgementMessage
InverseKeys = (AK, AK), (SK, SK), (MS, MS), (F,F)
```

# Processes

```
INITIATOR(M, EP, R1,AuthID,Initauth, AccReq, AuthReq, MS, AK, SK, HOAID1)
Authenticator(EP,AS, AuthReq, Adv,AccRes)
DomainSERVER(AS,M, R2, HoAckm, MS, AK, SK, HOAID1)
```

# Protocol Description

```
0.  -> M : EP, AS
1.  M -> EP: AccReq
2.  EP -> M : AuthReq
3.  M -> EP : {M,R1, HOAID1}{SK}%w
4.  EP -> AS : w%{M,R1,HOAID1}{SK}, h(w%{M,R1,
HOAID1}{SK})
5.  AS -> EP: {R2,{R1}{AK}%z}{SK}%v
6.  EP -> M : v%{R2,{R1}{AK}%z}{SK}
[decryptable(z, AK)andnth(decrypt(z, AK), 1) == R1]
# Specification
Secret(M,AK,[AS])
Secret(AS,AK,[M])
Secret(M,SK,[AS, EP])
Agreement(AS, M, [AK, R1])
WeakAgreement(M, EP)
WeakAgreement(EP,M)
Aliveness (EP, M)
Aliveness (M, EP)
# Actual Variables
m, Eve:  MobileTerminal
ep :  AccessRouterAuthenticator
as :  DomainA3CServer
Authid :  Identity
InitAuth :  Flags
hoaid1:  OldToken
hoaid2:  NewToken
r1 :  initialSeq
r2 :  Sequence
ak :  AuthenticationKeys
sk :  SecretKeys
ms:  Domainspecifickey
accReq, accRes,authReq, adv:  Messages
hoackm :  AcknowledgementMessage
InverseKeys = (ms, ms), (ak, ak), (sk, sk)
# Functions
symbolic F
```

```
# System
INITIATOR(m,ep, r1,Authid,InitAuth, accReq, authReq, ms, ak,sk,hoaid1)
Authenticator(ep, as, authReq,adv, accRes)
DomainSERVER(as,m, r2,hoackm, ms,ak,sk,hoaid1)
# Intruder Information
Intruder = Mallory
IntruderKnowledge = m, as, Mallory, Authid, ep
Crackable = Domainspecifickey
```

APPENDIX F

The Proposed NL-AKA Protocol for Secure Vertical Handover

```
 # Free Variables
MT: Agent
SrcAuth :  SrcAccessRouterAuthenticator
DesAuth :  DesAccessRouterAuthenticator
SrcDA3C : SrcDomainA3CServer
DesDA3C : DesDomainA3CServer
CA3C : CentralA3CServer
AuthID : Identity
Initauth :  Flags
seq1 :  initialSeq
seq2 :  Sequence
Srcse1 :  SrcDomainA3CServer-> PresharedKeys
Desse1 :  DesDomainA3CServer-> PresharedKeys
Srcse2 :  SrcAccessRouterAuthenticator-> PresharedKeys
Desse2 :  DesAccessRouterAuthenticator-> PresharedKeys
uk :  Agent-> PresharedKeys
SrcAK, DesAK : AuthenticationKeys
SrcSK, DesSK : SecretKeys
DSMK: Domainspecifickey
AccReq, AccRes,AuthReq, Adv:  Messages
HoAckm :  AcknowledgementMessage
F1:  PresharedKeys x initialSeq x Identity
-> Domainspecifickey
```

F2: initialSeq x Domainspecifickey ->

AuthenticationKeys

F3: initialSeq x Domainspecifickey x Identity ->

SecretKeys

InverseKeys = (SrcAK, SrcAK), (uk, uk) , (SrcSK, SrcSK),

(DSMK, DSMK), (Srcse1,Srcse1),(Srcse2, Srcse2),

(Desse1,Desse1),(Desse2, Desse2),(DesAK,DesAK),(DesSK, DesSK), (F1, F1), (F2,F2),(F3,F3)

# Processes

INITIATOR(MT,seq1,AuthID,Initauth, SrcAK, SrcSK, AccReq) knows uk(MT)

SrcAuthenticator(SrcAuth,MT,SrcDA3C,SrcSK, AuthReq)

knows Srcse2(SrcAuth)

DesAuthenticator(DesAuth,MT, DesDA3C, AuthReq, Adv,

AccRes) knows Desse2(DesAuth)

SrcAAASERVER(SrcDA3C,CA3C, SrcAuth, SrcAK, SrcSK) knows Srcse1(SrcDA3C), Srcse2(SrcAuth)

DesAAASERVER(DesDA3C,CA3C,DesAuth, seq2,HoAckm) knows Desse1(DesDA3C), Desse2(DesAuth)

CentralSERVER(CA3C, SrcDA3C, DesDA3C) knows

Srcse1(SrcDA3C), Desse1(DesDA3C), uk(MT)

# Protocol Description

0. -> MT : SrcAuth, DesAuth, SrcDA3C

1. DesAuth -> MT :Adv, DesDA3C

$< DSMK := F1(uk(MT), seq1, AuthID) >$

2. MT -> DesAuth: AccReq

3. DesAuth -> MT : AuthReq

$< DesAK := F2(seq1, DSMK) >$

4. MT -> SrcAuth : {seq1,AuthID, MT, Initauth}{SrcSK}

5. SrcAuth -> SrcDA3C : {seq1,AuthID, MT, Initauth}{

Srcse2(SrcAuth)}

6. SrcDA3C -> CA3C : {seq1,AuthID, MT, Initauth}{Srcse1(SrcDA3C)}

$< DSMK := F1(uk(MT), seq1, AuthID) >$

7. CA3C -> DesDA3C : {DSMK, seq1,AuthID, MT, Initauth}

{Desse1(DesDA3C)}

$< DesAK := F2(seq1, DSMK) >$

8. DesDA3C -> DesAuth: {({seq2, seq1}{DesAK}%z)%x}

{Desse2(DesAuth)}

9. DesAuth -> MT : x%({seq2, seq1}{DesAK}%z)

$[decryptable(z, DesAK) andnth(decrypt(z, DesAK), 2) == seq1]$

$< DesSK := F3(seq1, DSMK, AuthID);$

$seq2 := nth(decrypt(z, DesAK), 1) >$

```
10.  MT -> DesAuth :  ({seq2}{DesAK}%y)%q
11.  DesAuth -> DesDA3C: {(q%{seq2}{DesAK})%y}{Desse2(DesAuth)}
```

$[decryptable(y, DesAK) andnth(decrypt(y, DesAK), 1) == seq2]$

$< DesSK := F3(seq1, DSMK, AuthID) >$

```
12.  DesDA3C -> CA3C : {HoAckm}{Desse1(DesDA3C)}
13.  DesDA3C -> DesAuth :{DesSK}{Desse2(DesAuth)}
14.  DesAuth -> MT : {AccRes}{DesSK}
```

# Specification

```
Secret(MT,DesAK,[DesDA3C])
Secret(DesAuth,DesSK,[MT, DesDA3C])
Agreement( MT, DesDA3C, [seq2])
Agreement(DesDA3C, MT, [seq1, DesAK])
WeakAgreement(MT, DesAuth)
WeakAgreement(DesAuth,MT)
WeakAgreement(DesAuth, DesDA3C)
WeakAgreement(DesDA3C, DesAuth)
Aliveness(MT, DesAuth)
Aliveness(DesAuth,MT)
```

# Actual Variables

```
mt, Mallory:  Agent
srcAuth :  SrcAccessRouterAuthenticator
desAuth :  DesAccessRouterAuthenticator
srcDA3C : SrcDomainA3CServer
desDA3C : DesDomainA3CServer
ca3c :  CentralA3CServer
authID : Identity
initauth :  Flags
SEQ1 :  initialSeq
SEQ2 :  Sequence
srcAK, desAK : AuthenticationKeys
srcSK, desSK : SecretKeys
dsmk:  Domainspecifickey
accReq :  AccessReqmessages
accRes:  AccessResmessages
```

```
authReq:  Authmessage
adv:AdvMessages
hoAckm :  AcknowledgementMessage
InverseKeys = (dsmk, dsmk), (srcAK, srcAK), (srcSK, srcSK), (desAK, desAK), (desSK,
desSK)
```

# Functions

```
symbolic Srcse1,Srcse2,Desse1,Desse2,uk, F1, F2, F3
```

# System

```
INITIATOR(mt,SEQ1,authID,initauth, srcAK, srcSK,accReq)
SrcAuthenticator(srcAuth,mt,srcDA3C,srcSK, authReq)
DesAuthenticator(desAuth,mt, desDA3C, authReq,adv, accRes)
SrcAAASERVER(srcDA3C,ca3c,srcAuth, srcAK, srcSK)
DesAAASERVER(desDA3C,ca3c,desAuth, SEQ2,hoAckm)
CentralSERVER( ca3c, srcDA3C, desDA3C)
```

# Intruder Information

```
Intruder = Mallory
IntruderKnowledge = {mt, srcDA3C,desDA3C, Eve, ca3c, authID, srcAuth,desAuth,
uk(Eve), F2, F3}
Crackable = PresharedKeys
Crackable = Domainspecifickey
```

APPENDIX G

The Initial SL-AKA Protocol

# Free Variables

```
MT: Agent
SP : Service
DesAuth :  DesAccessRouterAuthenticator
DesDA3C : DesDomainA3CServer
CA3C : CentralA3CServer
r1, r2 :  Nonce
ADname :  AccessDomainname
Srvkey :Service -> ServiceSpecificKeys
F: ServiceSpecificKeys x Vectors x Vectors -> AssociationKeys
```

```
SubID : ServiceSubscribtionID
ASKey :  AssociationKeys
Vector1,Vector2:Vectors
SrvCookies:  Cookies
Ackm :  AcknowledgementMessage
InverseKeys = (Srvkey, Srvkey), (ASKey,ASKey), (F,F)
```

# Processes

```
INITIATOR(MT,Ackm,r1,Vector1, SubID,ADname) knows Srvkey(SP)
DesAuthenticator(DesAuth,SP,DesDA3C)
DesAAASERVER(DesDA3C,CA3C,DesAuth)
CentralSERVER(CA3C, DesDA3C,Vector1,SubID, ADname)
RESPONDER(SP, MT, DesAuth, DesDA3C, r2, Vector2, SrvCookies, Ackm) knows Srvkey(SP)
```

# Protocol Description

```
0.  CA3C -> DesDA3C: Vector1, SubID, ADname
1.  DesDA3C -> DesAuth:  Vector1, SubID, ADname
2.  DesAuth -> SP : Vector1, SubID, ADname
```

$< ASKey := F(Srvkey(SP), Vector1, Vector2) >$

```
3.  SP -> MT : {r2, Vector2, SrvCookies}{Srvkey(SP)}
```

$< ASKey := F(Srvkey(SP), Vector1, Vector2) >$

```
4.  MT -> SP : {r2, SrvCookies}{ASKey}%v1
```

$[decryptable(v1, ASKey) and nth(decrypt$

$(v1, ASKey), 1) == SrvCookies]$

```
5.  SP -> MT : {Ackm}{ASKey}%w3
```

$[decryptable(w3, ASKey) and nth(decrypt$

$(w3, ASKey), 1) == Ackm]$

# Specification

```
Secret(SP,ASKey,[MT])
Secret(MT,ASKey,[SP])
Secret(SP,SrvCookies,[MT])
Agreement(SP,MT,[ASKey])
Agreement(MT,SP,[ASKey, SrvCookies])
WeakAgreement (SP, MT)
WeakAgreement (MT, SP)
```

# Actual Variables

```
mt, Mallory:  Agent
```

```
desAuth :  DesAccessRouterAuthenticator
desDA3C : DesDomainA3CServer
sp :  Service
ca3c :  CentralA3CServer
R1, R2 :  Nonce
ADNAME : AccessDomainname
VECTOR1,VECTOR2:  Vectors
SRVCookies:  Cookies
SUBID : ServiceSubscribtionID
ACKM : AcknowledgementMessage
ASKEY : AssociationKeys
InverseKeys = (ASKEY,ASKEY)
# Functions
symbolic Srvkey, F
# System
INITIATOR(mt, ACKM, R1, VECTOR1, SUBID, ADNAME)
DesAuthenticator(desAuth, sp, desDA3C)
DesAAASERVER(desDA3C, ca3c, desAuth)
CentralSERVER(ca3c, desDA3C, VECTOR1, SUBID, ADNAME)
RESPONDER(sp, mt, desAuth, desDA3C, R2, VECTOR2, SRVCookies, ACKM)
# Intruder Information
Intruder = Mallory
IntruderKnowledge = {mt, desDA3C, ca3c, authID}
Crackable = ServiceSpecificKeys
```

APPENDIX H

The Ligh-Weight SL-AKA Protocol for Handover

```
# Free Variables
MT: Agent
SP : Service
DesAuth :  DesAccessRouterAuthenticator
DesDA3C : DesDomainA3CServer
CA3C : CentralA3CServer
r1, r2 :  Nonce
```

```
ADname :  AccessDomainname
Srvkey :Service -> ServiceSpecificKeys
F: ServiceSpecificKeys x OldAssociationKeys x Vectors x Vectors -> NewAssociationKeys
SubID : ServiceSubscribtionID
NewASKey :  NewAssociationKeys
OldASKey :  OldAssociationKeys
Vector1,Vector2:  Vectors
SrvCookies:  Cookies
Ackm :  AcknowledgementMessage
InverseKeys = (Srvkey, Srvkey), (NewASKey,NewASKey),(OldASKey,OldASKey) ,(F,F)
```

# Processes

```
INITIATOR(MT, Ackm,r1,Vector1, SubID,ADname, OldASKey) knows Srvkey(SP)
DesAuthenticator(DesAuth,SP,DesDA3C)
DesAAASERVER(DesDA3C,CA3C,DesAuth)
CentralSERVER( CA3C, DesDA3C,Vector1,SubID, ADname)
RESPONDER(SP, MT, DesAuth, DesDA3C, r2, Vector2, SrvCookies, Ackm, OldASKey) knows
Srvkey(SP)
```

# Protocol Description

```
0.  CA3C -> DesDA3C:Vector1, SubID,ADname
1.  DesDA3C -> DesAuth:Vector1, SubID,ADname
2.  DesAuth -> SP : Vector1, SubID,ADname
```

$< NewASKey := F(Srvkey(SP), OldASKey,$
$Vector1, Vector2) >$

```
3.  SP -> MT : {Vector2}{OldASKey}
```

$< NewASKey := F(Srvkey(SP), OldASKey,$
$Vector1, Vector2) >$

```
4.  MT -> SP : {Ackm}{NewASKey}
```

# Specification

```
Secret(SP,NewASKey,[MT])
Secret(MT,NewASKey,[SP])
Secret(SP,SrvCookies,[MT])
Agreement(SP,MT,[OldASKey])
Agreement(MT,SP,[Ackm])
```

```
WeakAgreement (SP, MT)
WeakAgreement (MT, SP)
# Actual Variables
mt, Mallory:  Agent
desAuth :  DesAccessRouterAuthenticator
desDA3C : DesDomainA3CServer
sp :  Service
ca3c :  CentralA3CServer
R1, R2 :  Nonce
ADNAME : AccessDomainname
VECTOR1,VECTOR2:  Vectors
SRVCookies:  Cookies
SUBID : ServiceSubscribtionID
ACKM : AcknowledgementMessage
InverseKeys = (newASKey,newASKey)
,(oldASKey,oldASKey)
newASKey :  NewAssociationKeys
oldASKey :  OldAssociationKeys
# Functions
symbolic Srvkey, F
# System
INITIATOR(mt, ACKM, R1, VECTOR1, SUBID, ADNAME, oldASKey)
DesAuthenticator(desAuth, sp, desDA3C)
DesAAASERVER(desDA3C, ca3c, desAuth)
CentralSERVER( ca3c, desDA3C, VECTOR1, SUBID, ADNAME)
RESPONDER(sp, mt, desAuth, desDA3C, R2, VECTOR2, SRVCookies, ACKM, oldASKey)
# Intruder Information
Intruder = Mallory
IntruderKnowledge = {mt, desDA3C, ca3c, authID}
Crackable = ServiceSpecificKeys
```

Bibliography

[3GP06] *3gpp technical specifications.: 3g security; wlan interworking security (release 7)*, Technical specification, 3rd Generation Partnership Project, 2007-2006.

[AA09] S. Naseer A. Latif A. Altaf, M. Younus Javed, *Performance analysis of secured privacy and key management protocol in ieee 802.16e-2005*, International Journal of Digital Content Technology and its Applications (2009).

[Abo07] E. Aboelela, *Network simulation experiments manual: A systems approach*, Morgan Kaufmann, 2007.

[Agu06] R. Aguiar, *Pervasive services for next generation heterogeneous networks*, WTC06, May 2006.

[Ali10] A.S. Ali, *Authentication and key management in heterogeneous wireless networks*, Ph.D. thesis, Electrical and Computer Engineering, The University of British Columbia, 2010.

[Alm92] P. Almquist, *Type of service in the internet protocol suite*, RFC 1349, University of Southern California, July 1992.

[AM96] S. Vanstone A. Menezes, P. van Oorschot, *Handbook of applied cryptography*, CRC Press, 1996.

[AVI06] AVISPA ORG, *Avispa 1.1. user manual*, 2006.

[BA04] J. Vollbrecht J. Carlson H. Levkowetz B. Aboba, L. Blunk, *Extensible authentication protocol (eap)*, Standards Track 3748, Network Working Group, June 2004.

[CH12] T. Dreibholz C. Hohendorf, E. Unurkhaan, *Secure sctp*, Internet-draft, 2012.

[Cha05] P. Chandra, *Bulletproof wireless security : Gsm, umts, 802.11 and ad-hoc security*, Newnes. Oxford, 2005.

[Cho07] K.S. Choi, *It ontology and semantic technology*, Natural Language Processing and Knowledge Engineering, 2007.

[CJ05] J.Y. Choi C.W. Jeon, I.G. Kim, *Automatic generation of the c# code for security protocols verified with casper/fdr*, 19th International Conference on Advanced Information Networking and Applications, 2005.

[CR00] A. Rubens W. Simpson C. Rigney, S. Willens, *Remote authentication dial in user service (radius)*, RFC 2865, Network Working Group, June 2000.

[CV02] A. Mayerhoefer C. Vielhauer, R. Steinmetz, *Biometric hash based on statistical features of online signatures*, IEEE International Conference on Pattern Recognition (ICPR), 2002.

[DB08] H.A. Samra D.A. Burgess, *The open bts project*, Kestrel Signal Processing, Inc., 2008.

[DD00] R. Cohen S. Herzog R. Rajan A. Sastry D. Durham, Ed. J. Boyle, *The cops (common open policy service) protocol*, RFC 2748, Network Working Group, January 2000.

[Did09] X. Didelot, *Cosp-j: A compiler for security protocols*, Master's thesis, Oxford University Computing Laboratory, 2009.

[DJ04] J. Arkko D. Johnson, C. Perkins, *Mobility support in ipv6*, RFC 3775, 2004.

[DS10] P. McCann H. Tschofenig T. Tsou A. Doria G. Zorn Ed D. Sun, Ed, *Diameter quality-of-service application*, RFC 5866, 2010.

[Euc06] M. Euchner, *Hmac-authenticated diffie-hellman for multimedia internet keying (mikey)*, RFC 4650, Network Working Group, September 2006.

[FS07] A. Lasebae F. Shaikh, G. E. Mapp, *Proactive policy management using tbvh mechanism in heterogeneous networks*, NGMAST'07, 2007.

[GG09] E. Demaria J. Bournelle R. Lopez G. Giaretta, I. Guardini, *Authentication, authorization, and accounting (aaa) goals for mobile ipv6*, RFC 5637, Network Working Group, September 2009.

[GM06] F. Shaikh P. Vidales L. Patanapongpibul J. Balioisian J. Crowcroft G. Mapp, D.N. Cottingham, *An architectural framework for heterogeneous networking*, International Conference on Wireless Information Networks and Systems (WINSYS'06), 2006.

[GM07] D. Cottingham J. Crowcroft J. Beliosian G. Mapp, F. Shaikh, *Y-comm: A global architecture for heterogeneous networking*, in *International Wireless Internet Conference (WICON 2007)* (GM07).

[GM10] A. Lasebae R. Phan G. Mapp, M. Aiash, *Security models for heterogeneous networking*, in *SECRYPT'10* (GM10).

[GM11] H. Crestana Guardia J. Crowcrof G. Mapp, M. Aiash, *Exploring multihoming issues in heterogeneous environments*, PAMS'11, 2011.

[GMMAM09] F. Shaikh G. Mapp, M. Augusto M. Aiash, R. Vanni, and E. Moreira, *Exploring efficient imperative handover mechanisms for heterogeneous wireless networks*, EUPS-09, 2009.

[Gru93] T. Gruber, *Translation approach to portable ontology specification*, Knowledge Acquisition, 1993.

[HB05] S. Ratnasamy T. Roscoe S. Shenker H. Ballani, Y. Cathwathe, *Off by default*, HotNets-II, 2005.

[Hoa85] C. A. R. Hoare, *Communicating sequential processes*, Prentice Hall International, 1985.

[HOK07] HOKEY, *Hokey wg*, 2007.

[I.D06] K.Masahiro I.Daisuke, *secure service framework on mobile ethernet*, Journal of the National Institute of Information and Communication Technology (2006).

[IEE07] IEEE802.21, *Ieee 802.21/d8.0: Draft standard for local and metropolitan area networks: Media independent handover services*, 2007.

[IT04] ITU-T, *Global information infrastructure internet protocol aspects and next generation networks, y.140.1*, International Telecommunication Union. ITU-T, 2004.

[ITU06] ITU, *Principles for the management of next generation networks, m.3060/y.2401*, International Telecommunication Union ITU-T, 2006.

[JA06] H. Haverinen J. Arkko, *Extensible authentication protocol method for 3rd generation authentication and key agreement (eap-aka)*, RFC 4187, 2006.

[JM04] F. Zhu J. McNair, *Vertical handoffs in fourth-generation multinetwork environments*, Wireless Communications, IEEE In Wireless Communications, IEEE **11** (2004), no. 3, 8–15.

[KT09] V. Fajardo S. Das M. Tauil Y. Cheng A. Dutta D. Baker M. Yajnik D. Famolari K. Taniuchi, Y. Ohba, *Ieee 802.21: Media independent handover: Features, applicability, and realization*, 2009.

[Kur05] M. Kuroda, *Apg-report: Scalable mobile ethernet and fast vertical handover*, Tech. report, Communications Research Laboratory, 2005.

[Lac05] L.W. Lacy, *Owl: Representing information using the web ontology language*, Trafford Publishing, 2005.

[LBDH09] G. Lowe, P. Broadfoot, C. Dilloway, and M. L. Hui, *Casper: A compiler for the analysis of security protocols*, 1.12 ed., September 2009.

[LP03] G. Mapp L. Patanapongpibul, *A client-based handoff mechanism for mobile ipv6 networks*, ISCC'03, 2003.

[MA07] S. Sargento V. Jesus R. Aguiar M. Almeida, D. Corujo, *An end-to-end qos framework for 4g mobile heterogeneous environments*, OpenNet Workshop, 2007.

[MA10] A. Lasebae R. Phan M. Aiash, G. Mapp, *Providing security in 4g systems: Unveiling the challenges*, AICT'10, 2010.

[MA11a] A. Lasebae M. Aiash, G. Mapp, *A qos framework for heterogeneous networking*, WCE'11, 2011.

[MA11b] A. Lasebae R. Phan J. Loo M. Aiash, G. Mapp, *A formally verified initial aka protocol in heterogeneous environments using casper/fdr*, Submitted For Publication in the International Journal of Information Security, Springer, 2011.

[MA11c] A. Lasebae R. Phan M. Augusto R. Vanni E. Moreira M. Aiash, G. Mapp, *Enhancing naming and location services to support multi-homed devices in heterogeneous environments*, CCSIE '11, 2011.

[MA11d] G. Mapp M. Aiash, *Security and qos integration for protecting service providers in heterogenoues environments*, International Journal of Computer Science **38:4** (2011), 384–393.

[MA12a] A. Lasebae J.Loo F.Sardis R. Phan M. Augusto R. Vanni E. Moreira M. Aiash, G. Mapp, *A survey of potential architectures for communication in heterogeneous networks*, IEEE Wireless Telecommunications Symposium (WTS), 2012.

[MA12b] A. Lasebae R. Phan J. Loo M. Aiash, G. Mapp, *A formally verified aka protocol for vertical handover in heterogeneous environments using casper/fdr*, EURASIP J. Wireless Comm. and Networking (2012), 57.

[MA12c] R. Phan A. Lasebae J.Loo M. Aiash, G. Mapp, *A formally verified device authentication protoc*, In Proceedings of TrustCom 2012, 2012.

[MB90] R. Needham M. Burrows, M. Abadi, *A logic of authentication*, ACM Transactions on Computer Systems **8** (1990), 18–36.

[MB04] C. Brandauer T. Braun S. Kardos F. Orl M. Scheidegger J. Seger M. Bartoli, F. Baumgartner, *The intermon simulation framework*, In Proceedings of the Second Inter-Domain Performance and Simulation Workshop, 2004.

[McC05] P. McCann, *Mobile ipv6 fast handovers for 802.11 networks*, RFC 4260, 2005.

[MJS02] P. Ebinger M. Jalali-Sohi, *Towards efficient pkis for restricted mobile devices*, CCN'02, 2002.

[MK04] A. Okubo T. Sakakura K. Shimizu F. Adachi M. Kuroda, M. Inoue, *Scalable mobile ethernet and fast vertical handover*, IEEE Wireless Communications and Networking Conference, 2004.

[Moc87] P. Mockapetris, *Domain names - implementation and specification*, RFC 1035, 1987.

[MR05] M. Tuexen M. Riegel, *Mobile sctp*, Internet-draft, 2005.

[MT07] P. Lei E. Rescorla M. Tuexen, R. Stewart, *Authenticated chunks for stream control transmission protocol (sctp)*, Internet-draft, 2007.

[MZ05] Y. Fang M. Zhang, *Security analysis and enhancements of 3gpp authentication and key agreement protocol*, Wireless Communications, IEEE Transactions **4** (2005), 734–742.

[NN04] H. Abramowicz G. Malmgren J. Sachs U. Horn C. Prehofer H. Karl N. Niebert, A. Schieder, *Ambient networks: An architecture for communication networks beyond 3g*, IEEE Wireless Communications **11** (2004).

[PC03] E. Guttman G. Zorn J. Arkko P. Calhoun, J. Loughney, *Diameter base protocol*, RFC 3588, 2003.

[Pos80] J. Postel, *User datagram protocol*, RFC 768, 1980.

[PR10] M. Goldsmith G. Lowe A.W Roscoe P. Ryan, S. Schneider, *The modelling and analysis of security protocols*, PEARSON Ltd, 2010.

[RB94] S. Shenker R. Braden, D. Clark, *Integrated services in the internet architecture: an overview*, RFC 1633, Network Working Group, June 1994.

[rGPPG] 3rd Generation Partnership Project (3GPP).

[SB98] M. Carlson E. Davies Z. Wang W. Weiss S. Blake, D. Black, *An architecture for differentiated services*, RFC 2475, Network Working Group, December 1998.

[SB04] A. Schuchart S. Small K. Watkins S. Bono, M. Brotzman, *Ban logic reading guide*, 2004.

[SBC04] A. Lasebae S. Bansal and R. Comley, *Freedom to choose along with freedom of mobility*, ICCTA'04, 2004.

[Sch03] J.H. Schiller, *Mobile communications*, Addison-Wesley, 2003.

[SD98] R. Hinden S. Deering, *Internet protocol, version 6 (ipv6) specification*, RFC 2460, 1998.

[SK98] R. Atkinson S. Kent, *Security architecture for the internet protocol*, RFC 2401, 1998.

[Spi92] J.M. Spivey, *The z notation: a reference manual*, Prentice Hall, 1992.

[SS97] R. Guerin S. Shenker, C. Partridge, *Specification of guaranteed quality of service*, RFC 2212, Network Working Group, September 1997.

[S.S07] F. Sousa F. Mitrano T. Strauf J. Gozdecki G. Lemos M. Almeida D. Corujo S.Sargento, V. Jesus, *Context-aware end-to-end qos architecture in multi-technology, multi-interface environments*, 16th IST Mobile & Wireless Communications Summit, 2007.

[Ste07] R. Stewart, *Stream control transmission protocol*, RFC 4960, 2007.

[Sys93] Formal Systems, *Failures-divergence refinement. fdr2 user manual and tutorial*, June 1993, Version 1.3.

[TD99] C. Allen T. Dierks, *The tls protocol*, RFC 2246, 1999.

[TEL00] C. E. Irvine T. E. Levin., *Quality of security service*, 2000.

[TI09] E. Hossain T. Issariyakul, *Introduction to network simulator ns2*, Springer, 2009.

[TMP04] *Trusted mobile platform protocol specification document – revision 1.00*, Online, May 2004.

[VC74] Y. Dalal V. Cerf, *Specification of internet transmission control program*, RFC 675, 1974.

[VN08] L. Dondeti V. Narayanan, *Eap extensions for eap re-authentication protocol (erp)*, Standards Track 5296, August 2008.

[WiM10] *Interworking specification*, WiMAX Forum® Approved WMF-T37-008-R016v01, WiMAX Forum® Network Architecture, 2010.

[YS05] N. Memon Y. Sutcu, T. Sencar, *A secure biometric authentication scheme based on robust hashing*, ACM MM-SEC Workshop, 2005.

[Y.S08] K. Tan U. Deshpande B. Vance H. Yin C. McDonald T. Henderson D. Kotz A. Campbell J. Wright Y.Sheng, G. Chen, *Map: A scalable monitoring system for dependable 802.11 wireless networks*, IEEE Wireless Communication Special Issue on Dependability Issues with Ubiquitous Wireless Access, 2008.

[YZ05] X. Tang H. Wang Y. Zheng, D. He, *Aka and authorization scheme for 4g mobile networks based on trusted mobile platform*, ICICIS'05, 2005.