Rashad Mahmood Saqib

Lightweight ECC Based Multifactor Authentication Protocol (LEMAP) for Device to Device Cellular Network

Rashad Mahmood Saqib

Doctor of Philosophy

Faculty of Computer Science and Information Technology
Universiti Malaysia Sarawak
2022

2022

Lightweight ECC Based Multifactor Authentication Protocol (LEMAP) for Device to Device Cellular Network

Rashad Mahmood Saqib

A thesis submitted

In fulfillment of the requirement for the degree of Doctor of Philosophy

Computer Science

Faculty of Computer Science and Information Technology
UNIVERSITI MALAYSIA SARAWAK
2022

DECLARATION

I declare that the work in this thesis was carried out in accordance with the regulations of Universiti Malaysia Sarawak. Except where due acknowledgements have been made, the work is that of the author alone. The thesis has not been accepted for any degree and is not concurrently submitted in candidature of any other degree.

……

Signature

Name: Rashad Mahmood Saqib

Matric No.: 15010175

Faculty of Computer Science and Information Technology

Universiti Malaysia Sarawak

Date :

DEDICATION

To the Almighty ALLAH (SWT) and the Holy Prophet Muhammad (P.B.U.H)

&

Also dedicated to my great and very beloved father Mr. Ali Asghar, my beloved mother Haleema Bibi, my brother Shaukat Ali, my sisters, wife and my beloved son Usman Rashad, Umer Shaukat, Zunaira Shaukat, Umaira Shaukat. Special dedication to my best friend Dr. Yasir Javed Kiani for his great support, guidance and mentor Dr. Adnan Shahid Khan who always motivated, guided and encouraged me in achieving this milestone of life.

ACKNOWLEDGEMENT

I owe my gratitude to ALLAH (SWT), the most gracious and the most merciful Whose guidance and protection sustained me throughout my entire life and makes my educational career a reality with enormous success. Special gratitude goes to my supervisor Senior Lecturer Dr. Adnan Shahid Khan for his proper supervision, motivation, fatherly advice, support and encouragement throughout the entire study period at Universiti Malaysia Sarawak. His knowledge and expertise in D2D Security will always remain with me as guidance to follow. Without his guidance, the research and achieving this milestone of life would have been impossible. I would also like to thank my co-supervisors, Associate Professor Dr Nazim Jambli for his sincere advice.

My special and profound gratitude goes to my late father Ali Asghar, mother Haleema Bibi and sisters, niece Zunaira Shaukat and my entire family members and friends for their support and prayers. Immense gratitude and special thanks to my wife and son Usman Rashad for their constant prayers, patience, tolerance and sense of understanding.

I would like to sincerely appreciate the effort and assistance of my beloved friend Dr. Yasir Javed Kiani for his constant support, guidance and sincere advice towards the successful completion of the programme. I would like to say special thanks to my colleagues and friends for their advice, encouragement and prayers. The contributions of authors whose work have being duly cited in this research are hereby acknowledged. I am grateful to all.

My sincere gratitude to the Centre for Graduate Studies, for the advice and support given during my period of study in Universiti Malaysia Sarawak.

Finally, I would like to thank the management of the Universiti Malaysia Sarawak for making it possible for me to complete my study here in Sarawak. Thank you all.

ABSTRACT

Device to Device (D2D) communication is a type of technology where two devices can communicate directly with each other without the need to contact Base Station or any central infrastructure. With emerging of Long Term Evaluation (LTE) and Fifth Generation (5G) technology, D2D has gained a lot of attention for communication between closely located mobile devices for offering high speed, energy efficiency, throughput, less delay, and efficient spectrum usage. D2D has changed recent wireless networks with new trends as D2D can play a vital role in sharing resources by load off the network in local areas by direct communication between devices and useful in natural disasters where BS is destroyed. D2D has revolutionized the direct communication as it is a basis for 5G network. D2D allows miniature devices like cell phone, tablets and radio devices to work as Non-Transparent Relays (NTR) where they can provide services as well as forward traffic, request services by direct communication without the need of Base Station (BS) or central network infrastructure. Multi-hop D2D can be used for peer-to-peer communication or even access to cellular networks. This concept of multihop D2D communication has introduced a number of issues and challenges that were not prevalent in traditional current cellular communication. One of the major issues in D2D is security that is required in D2D communication to transmit information securely over non secure channel. The major challenge when considering security is that current established security techniques cannot be modified as security-requiring devices are miniature with restricted processing and storage or are constrained by power and bandwidth issues. Another issue is that how devices can get secure mutual authentication for secure communication. To tackle these issues, a lightweight multifactor authentication scheme that allows multihop secure communication over open channel is designed called as Lightweight ECC based Multifactor Authentication

Protocol (LEMAP) in multihop D2D communication. Formal analysis of scheme is performed using well known BAN Logic method which is used to check correctness of protocol. The formal analysis of LEMAP proves that it can mitigate replay attack, Man-in-the-Middle (MITM) attack, Rogue device attack, Denial of Service (DoS) attack, timestamp exploitation attack, impersonation attack and masquerading attack. LEMAP also achieves security requirements confidentiality, integrity, privacy, non-repudiation, secure mutual authentication and anonymity. The communication cost and computational overhead of benchmark protocols and the proposed scheme LEMAP are also calculated. The results show that LEMAP is 6%-28% percent stronger than the selected benchmark algorithms such as 2PAKEP, Chaotic based authentication and TwoFactor authentication protocol. Additionally, LEMAP provides additional security by using trust validation, double hashing, and reduced authentication overhead. Discrete logarithm analysis shows that LEMAP is more secure compared to current security algorithms or current security algos are used as attacks against LEMAP. LEMAP is a lightweight and flexible scheme which can be used in 5G as well as multihop D2D communication to provide secure communication environment.

Keywords: D2D security, multihop D2D security, multi factor, light-weight security, ECC

Protokol Pengesahan Multifaktor Berbasis ECC Ringan (LEMAP) untuk Rangkaian Selular Peranti ke Peranti

ABSTRAK

Komunikasi Peranti ke Peranti (PkP) adalah sejenis teknologi di mana dua peranti dapat berkomunikasi secara langsung antara satu sama lain tanpa perlu menghubungi Stesen Pangkalan atau infrastruktur pusat. Dengan munculnya teknologi LTE dan 5G, D2D telah mendapat banyak perhatian untuk komunikasi antara peranti mudah alih yang terletak berdekatan untuk menawarkan kelajuan tinggi, kecekapan tenaga, throughput, penundaan yang lebih sedikit, dan penggunaan spektrum yang efisien. D2D telah mengubah rangkaian tanpa wayar yang terkini dengan tren baharu kerana D2D dapat memainkan peranan penting dalam berkongsi sumber dengan memuatkan rangkaian di kawasan setempat dengan komunikasi langsung antara peranti dan berguna dalam bencana alam di mana BS dimusnahkan. D2D telah merevolusikan komunikasi langsung kerana ia adalah asas untuk rangkaian 5G. D2D membenarkan peranti miniatur seperti telefon bimbit, tablet, peranti radio berfungsi sebagai Relay Tidak Transparan (NTR) di mana mereka dapat menyediakan perkhidmatan serta lalu lintas ke hadapan, meminta perkhidmatan dengan komunikasi langsung tanpa memerlukan Stesen Pangkalan (SP) atau pusat infrastruktur rangkaian. Multi-hop D2D dapat digunakan untuk komunikasi rakan sebaya atau bahkan akses ke rangkaian selular. Konsep komunikasi multihop D2D ini telah memperkenalkan sejumlah masalah dan cabaran yang tidak lazim berlaku dalam komunikasi selular semasa tradisional. Salah satu masalah utama dalam D2D adalah keselamatan yang diperlukan dalam komunikasi D2D untuk menghantar maklumat dengan selamat melalui saluran yang tidak selamat. Cabaran utama ketika mempertimbangkan keselamatan adalah bahawa teknik keselamatan yang ada saat ini tidak dapat diubahsuai kerana peranti yang

memerlukan keselamatan adalah miniatur dengan pemprosesan dan penyimpanan terhad atau dibatasi oleh masalah kuasa dan lebar jalur. Masalah lain ialah bagaimana peranti dapat mendapatkan pengesahan bersama yang selamat untuk komunikasi yang selamat. Untuk mengatasi masalah ini, skema pengesahan multifaktor ringan yang membolehkan komunikasi selamat multihop melalui saluran terbuka dirancang dipanggil sebagai Protokol Pengesahan Multifaktor Berbasis ECC Ringan (LEMAP) dalam komunikasi D2D multihop. Analisis skema secara formal dilakukan dengan menggunakan kaedah BAN Logic yang terkenal yang digunakan untuk memeriksa kebenaran protokol. Analisis formal LEMAP membuktikan bahawa ia dapat mengurangi serangan ulangan, serangan Man-in-the-Middle (MITM), serangan perangkat rouge, serangan Denial of Service (DoS), serangan eksploitasi cap waktu, serangan peniruan dan serangan penyamaran. LEMAP juga mencapai kerahsiaan, integriti, privasi, penolakan, pengesahan bersama yang selamat dan tanpa nama. Kos komunikasi dan overhead komputasi juga dikira protokol penanda aras dan skema LEMAP yang dicadangkan. Hasil kajian menunjukkan bahawa LEMAP adalah 6 hingga 28 peratus lebih kuat daripada algoritma penanda aras yang dipilih seperti 2PAKEP, pengesahan berasaskan Chaotic dan protokol pengesahan TwoFactor. Selain itu, LEMAP memberikan keselamatan tambahan dengan menggunakan pengesahan kepercayaan, hash berganda, dan overhead pengesahan yang dikurangkan. Analisis logaritma diskrit menunjukkan bahawa LEMAP selamat daripada algoritma keselamatan semasa dan juga serangan kuantum. LEMAP adalah skema ringan dan fleksibel yang dapat digunakan dalam komunikasi 5G dan multihop D2D untuk menyediakan persekitaran komunikasi yang selamat.

Kata kunci: *Keselamatan D2D, keselamatan D2D multihop, multi factor, keselamatan ringan, ECC*

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

3G	Third Generation Cellular Communication
4G	Fourth Generation Cellular Communication
4G LTE	Fourth Generation Long Term Evolution
4G LTE-A	Fourth Generation Long Term Evolution Advanced
5G	Fifth Generation Cellular Communication
AES	Advance Encryption Standard
AP	Access Point
Auth	Authentication
BAN Logic	Burrows, Abadi and Needham Logic
BFA	Brute Force Attack
BFC	Blind Fold Challenge
BS	Base Station
BsGs	Baby Step, Giant Step
CA	Certification Authority
C_{dev}	Communication Devices
CU	Cellular Users
ECC	Elliptic Curve Cryptography
Ch	Challenge Scheme
CH	Cluster Head
CN	Core Network
D2D	Device to Device Communication
DES	Digital Encryption Standard

DH	Diffie Hellman
D_1	Device with ID 1
D_2	Device with ID 2
DLP	Discrete Logarithm Problem
DoS	Denial of Service
DDOS	Distributed Denial of Service
DT	Trudy Device
ECC	Elliptic Curve Cryptography
ECDH	Elliptic curve Cryptography with Diffie Hellman
eNB	Evolved Node B
E-UTRA	Evolved Universal Terrestrial Radio Access
f_i	Share frequency
GW	GateWay
IoT	Internet of Thing
LTE	Long Term Evolution
LEMAP	Lightweight ECC based Multifactor Authentication Protocol
MANET	Mobile Adhoc Networking
MAC	Medium Access Control
MITM	Man in the Middle Attack
M-MIMO	Massive Multiple Input Multiple Output
MRBS	Master Relay Base Station
MFA	Multi Factor Authentication
NIST	National Institute of Standards and Technology
NTR	Non-Transparent Relay

OTP	One Time Password
P_B	Device B Public Key
P2P	Peer-to-Peer
PKI	Public Key Infrastructure
ProSe	Proximity Based Services
QoS	Quality of Service
RDp	Reactive Discovery Protocol
Req	Request
Resp	Response
RSA	River Shamir Adleman
SHA v3	Secure Hashing Algorithm Version 3
SP	Secret Point
SRP	Secure Routing Protocol
SK	Secret Key
ts	Timestamp
UE	User Equipment
WIFI	Wireless Fidelity

CHAPTER 1

INTRODUCTION

1.1 Research Background

Increase in cellular devices and demand for higher and fast data rate has put forward research and evolved Fourth Generation (4G) Long Term Evolution (LTE) towards Fifth Generation 5G (Alsharif & Nordin, 2017). Device-to-device (D2D) communication is a transmission in which devices directly communicate with each other without an intermediate device. It helps in expansion of cell coverage and increases frequency reuse in a 5G network. D2D communication is a core technology of 5G communication and adopts approach to facilitate work without centralized control mechanism to provide wireless network more spectrum and energy efficient with traffic offloading. 5G-D2D technology promises to improve current technologies by providing a proficient and reliable strategies for efficient resource utilization solutions to meet increased demand for future high network performance.

D2D apportions cellular traffic to offload from network-centric cellular links to peer-to-peer (P2P) wireless systems enabling low computational devices to take partial decisions besides increase in data rate, coverage and network-capacity (Li et al., 2018; Lee et al., 2016). It permits the communication to happen under the presence of infrastructure called as network assisted or without network assistance (Liu et al., 2015) but in any case, the devices must be registered first to a cellular network to start the provision of services. D2D utilizes the direct link between user equipment's (UE) to soften the load on cellular link such as base station (BS) thus empowering the user devices to act as an intelligent authorization and data dissemination devices (Ahmad et al., 2017, Gandotra & Jha, 2016). However, typical D2D

communication in 4G has various security challenges including impersonation, man in the middle (MITM), replay, eavesdropping, privacy, and denial of service (DOS) attacks etc. When devices are allowed to work as intelligent relays and authorization, this has exposed devices communication to several security loopholes and breaches for instance: quantum attacks, 51% attacks, MITM attack, masquerading attack, DoS attack, rouge relay attack and privacy issues in current cellular network settings (Cheng et al., 2017; Haus et al., 2017; Habib et al., 2017). These security challenges are more crucial and harder to mitigate because of the resource-constrained nature of cellular devices. Therefore, there is an immense need to consider security design requirement for ensuring secure and trustworthy environment for D2D cellular communication. To solve these security challenges in D2D communication, a lightweight and secure D2D communication system is required that can provide secure mutual authentication, data confidentiality/integrity and anonymity.

D2D communication is an integral part of 5G and is based on LTE (Kato, 2016) and uses proximity services to communicate between devices and share data without the need for authorization by central device. D2D even can provide services with low signal strength or even no coverage and is a big relief for area with weak infrastructure. D2D communication provides fast data rate as there is no need to request services from the central network (Gandotra & Jha, 2016). Evolution of wireless networks and Internet of Things (IoT) have convinced network operators to support D2D and provision of D2D especially in LTE (Astely et al., 2013). D2D communication using proximity services not only reduces communication cost but also provides higher data rate and reusability. D2D also permits sharing of resources such as spectrum, power application, services, data and social contents for any number of users that are near the proximity range forming a base for mobile cloud computing (Ghudayyer et al., 2017; Tehrani & Yanikomeroglu, 2014).

There are several D2D applications such as local communication where multiple devices can communicate with each other to provide services like social data services, local address services and cellular data offloading. These days, one of the common D2D applications is IoT communication where multiple devices can be connected to each other using wireless network and provide data related to home, offices, vehicles, and emergency services. With IoT provisioning, a single device can collect all sensor traffic as one unit and then forwards this to BS. One of the major applications of D2D is emergency communication services where even with weak network or no coverage, mobile devices can be connected to form a network and help the restoration and rescue operations. D2D can be similarly used for coverage extension without the need for installation of expensive BS. Also, the spectrum sharing and re-usage in one area, massive multiple input and multiple output (M-MIMO) are few of the applications (Liu et al., 2015; Wei et al., 2014). With the provision of devices to provide services, data, applications and spectrum sharing gives rise to vulnerability in terms of security and trust.

For D2D communication, each device is referred as user equipment (UE) or devices in literature but in this study, the devices are referred to as devices. Each device communicates with the evolved Node B (eNB) to get the authorization if Base Transceiver Station (BTS) or mostly referred as the base station (BS) is not in range. eNB is an authorized small base station installed by cellular companies having enough resources and a direct link to allow D2D communication and authorization. The devices that receive authorization from eNB or BS can communicate directly with each other (Lien et al., 2016; Raghothaman et al., 2013). The security and device validation must be established before communication between two devices as any compromise can result in all major security attacks possible that

were even non-existent before the D2D technology (Haus et al., 2017; Tehrani & Yanikomeroglu, 2014).

Several algorithms have been proposed for D2D communication security. These algorithms include usage of asymmetric and combination of asymmetric and symmetric cryptographic algorithms such as River Shamir Adleman (RSA) (Ometov et al., 2016; Olshannikova et al., 2016), Diffie Hellman (DH) (Sedidi & Kumar, 2016), Elliptic Curve Cryptography (ECC) (Javed et al., 2017), Elliptic Curve Cryptography with Diffie Hellman (ECDH) (Hossain & Hasan, 2017; Yao et al., 2016) or some other combinations have also been used. In most of these techniques, BS serves as a certification authority (CA) for initial registration of devices. Once the devices are registered with the CA, they hold their public and private key pair for communication through which devices send secure message for authorization to secure communication to eNB or CA. The receiving eNB then checks the nearby devices and sends the authorized request to neighboring devices as well as authorized response to requesting devices. Each device must be authorized before communication. Once the authorization is done then communication can occur through the same asymmetric or mostly symmetric algorithm such as Advanced Encryption Standard (AES) or Digital Encryption Standard (DES).

The security layer of D2D is still at infancy and doesn't address all security challenges and attacks. Authorization of devices is one of the key challenges that usually is addressed by strict security classification of behavior (Wong, 2017). Authorization of devices is performed through eNB based message verification that requires each device to authenticate before any communication and thus requires additional authorization messages before communication (Habiba & Hossain, 2018; Nasrallah et al., 2018).

Usually the security is achieved by having a larger key size but it requires higher computational power and higher space (Othmen et al., 2017). Devices in D2D communication have limited computation and limited space and thus cannot have higher operational and computational cost (Hu et al., 2003). Authors have addressed security issues as D2D communication is over open non-secure channel. Privacy issues are also addressed by many researchers that keep secrecy of user when communication occur through eNB. Thus many schemes have been proposed to handle security issues like MITM, DoS and masquerading attacks especially to focus on single attack mitigation (Javed & Khan, 2019; Javed et al., 2019; Khan et al., 2017).

Multi-hop communication requires a new class of security algorithm as it results in newer attacks such as rogue relay and 51% attack (Gupta et al., 2018; Javed et al., 2017). Some of the solution provided use pseudo identity for each user but require eNB to provide the pseudo identity at time of each communication (Zhang & Shen, 2015). Collaboration and cooperation are key in D2D network which require device validation that must be established before starting communication. Validating the trust has been addressed by number of authors (Tata & Kadoch, 2014; Imai et al., 2006) but adaptive techniques are required to handle mix behavior of the devices that is acting partially as malicious and in some cases as selfish (Melki et al., 2016; Qin et al., 2016; Kim & Han, 2012).

This research mainly focuses on identifying security challenges associated with D2D communication by designing a lightweight and trustworthy security scheme which can mitigate above mentioned attacks with low authentication overhead, reduced computation cost as well as lower communication cost. The proposed scheme is based on Elliptic Curve Cryptography (ECC) which is one of light weight asymmetric-key technique compared to

currently adapted public-key encryption algorithms such as RSA. ECC provides 3072-bit of RSA cryptographic security using a 256-bit key.

1.2 Problem Statement and Research Challenges

Multi-hop D2D communication has several challenges. One of the major constraints is the lack of secure channel that leads towards several attacks specifically DoS, replay attack, MITM attack and identity reveal attacks. Many other challenges also exist like privacy issues, frequency hopping (Adam et al., 2019), network coding (Huang et al., 2019; Vanganuru et al. 2012), interface management (Tuan et al., 2019; Droste et al., 2015), network pairing (He et al., 2019), device discovery (Kaleem et al., 2019; Doppler et al., 2011), device security and trust (Javed et al., 2017; Company, 2003).

In multi-hop D2D communication, device and information security is a key challenge which can destroy the whole communication if compromised. Although current security schemes have addressed some of challenges, yet a number of challenges require further attention. One of the challenges is that key size must be small as larger key size results in higher operational cost. Normally, larger key size is required to make security scheme safer against quantum attack but this results in higher operational cost. Researches have shown that a very large key size cannot be used in D2D security scheme due to memory, storage and computation requirements (Javed et al., 2017; Wang et al., 2017; Zhang et al., 2017).

Most of the authors focused on mitigating well-known and common MAC Access Control (MAC) layer attacks such as Replay attack, MITM, DoS attacks and impersonation attacks. This research is based on primitive assumptions such as identity can be shared publicly, communication channels are always secure, some of the credentials can be sent without encryption, timestamp cannot be modified, single hashing can resolve the integrity

problem and cryptosystem is free of quantum brute force and password guessing attacks. However, several researchers believe that the discussed assumptions are not addressed carefully and it can provide space to common MAC layer attacks. For instance, identity reveal attack can lead to theft of identity which further leads towards impersonation attack or MITM attack.

As compared to normal cellular services, D2D communication offers high authentication overhead because numerous messages are exchanged for registration and mutual authentication. In order to complete authorization, devices must be verified and validated by Cluster Head or eNB and get authorization before starting actual communication (Habib et al., 2017; Hossain & Hasan, 2017; Wang et al., 2017; Ramadan et al., 2016; Yao et al., 2016; Zhang et al., 2016). In this whole authorization process, many messages are exchanged for communication which not only increase authentication overhead but also increase communication cost.

Privacy is another big issue in D2D communication as D2D is based on proximity services. In proximity services, real identity and location of devices is not hidden so the real identity and location of devices is apparent to other devices that is not desired by most of the devices. The real identity of devices should be hidden to maintain their privacy and real identity details. If a real identity of device is compromised, a communication pattern can be established which can also result in finding identity of any devices participating in communication even if the channel is secure (Hsu et al., 2017; Wang et al., 2017; Ramadan et al., 2016).

D2D allows communication directly between two devices which is referred as single hop communication. All this communication usually occurs on non-secure channel while the devices can be semi or non-trustworthy for the communication.

In multi-hop communication, a number of attacks such as Replay attack, Denial of Service (DoS) or Distributed Denial of Service (DDOS), Man in the Middle (MITM) and Identity Reveal attacks can occur (Haus et al., 2017; Javed et al., 2017; Wang et al., 2017; Zhang et al., 2017; Sedidi & Kumar, 2016; Zhang et al., 2016; Liu et al., 2015). This can also result in impersonation attack and masquerading attack where false devices can claim as real one (Ji et al., 2016; Nomikos et al., 2014). Replay attack is performed by repetitive delivery of a message sent by an adversary (Haddad et al., 2015). Attack on confidentiality will result in leakage of all sensitive information and may also result in compromised integrity (Khan et al., 2017; Wang et al., 2017; Zhang et al., 2017; Zhang et al., 2016; Ometov et al., 2015). D2D devices can act as intelligent relays where the devices can decode and forward the whole traffic that can make it vulnerable to already discussed attacks.

Literature shows a very limited work in area of threat mitigation of multi-hop communication and usually focuses on identification of security challenges (Nardini et al., 2017). Current schemes focus on mitigation of either one or two or a few of the security attacks, mostly focusing on MITM attack; whereas a comprehensive scheme that can mitigate most of security challenges at once is missing (Khan et al., 2017; Shen et al., 2016). Few techniques that have tried to address this challenge resulted in additional computation and authentication overhead on small miniature devices which have already limited computation and storage, as these devices are mostly in the form of mobile phones, tablets

and other hand-held devices. Thus a comprehensive security scheme must address this issue (Javed & Khan, 2017; Javed et al., 2017; Khan et al., 2017).

Most of the authors achieved secure end to end communication at the expense of high computation cost and message authentication overhead in multi-hop scenario. However, it is well agreed that using lightweight security mechanism and transmitting a small number of messages over communication link are always desirable as well as reducing the challenges of security threats, authentication overhead, communication and computation cost which should be feasible for miniaturized devices.

1.3 Idea and Motivation

D2D communication is an integral part of LTE-Advance (LTE-A) and 5G (Khan et al., 2018) where data availability and data rate must be higher with the lowest latency. The usage of public devices to act as authorization devices or decode and forward devices have resulted in several security issues such as DoS attacks, Rogue Relay attacks and MITM attacks which were not possible in earlier form of cellular communication. The motivation behind designing such a security algorithm is that numerous researchers conducted research in designing such a security scheme but most of the security algorithms have either unrealistic assumptions or do not address all security requirements simultaneously. As D2D promises number of applications that are necessary for high speed and low latency communication in 5G, it can be only possible if a lightweight secure environment can be created.

1.4 Research Objectives

This research mainly focuses on contriving a secure and lightweight algorithm for a multi-hop D2D communication that can satisfy all major identified security requirements.

This research designs a lightweight security algorithm called Lightweight ECC based Multi-factor Authentication Protocol (LEMAP). Especially, the following research objectives are addressed in this research:

i. To identify and investigate security challenges in existing D2D communication network.

ii. To develop:

   a. Lightweight ECC based Multifactor Authentication Protocol (LEMAP) that has secure end-to-end communication and reduced authentication and computation overhead.

   b. A certificate-less secure lightweight multi-hop D2D communication scheme for mitigating above mentioned security challenges.

   c. To evaluate the network performance of the proposed scheme using NCTUns for packet overhead, packet delivery ratio, effects of increasing rouge cluster head and processing time.

iii. To conduct the Formal Security Analysis for evaluation of the proposed Protocol.

iv. To check correctness and validation of proposed protocol using formal methods and non-formal security conformity methods against set of given attacks.

v. Performance evaluation of proposed protocol through communication cost, authentication overhead and computation overhead and Quantum attack safety using discrete logarithm problem.

1.5 Scope of Work

To accomplish above mentioned objectives, the scope of the study covers that the research work assumes two devices are connected to eNB where Cluster Head (CH) works as a relay device. Cryptographic technique Elliptic Curve Cryptography (ECC) is used to provide lightweight security. Multi-factor authentication is achieved using onetime password (OTP), timestamp is used for message freshness, integrity of messages is ensured using Secure Hashing Algorithm version 3 (SHA v3) and simple Blind Fold Challenge (BFC) is used for mutual authentication.

This study identifies and investigates security challenges in D2D such as MITM, Replay, Identity Reveal, Rogue Relay, DoS, Masquerading and Quantum attacks. To evaluate efficiency and performance of the proposed scheme, computational overhead is computed using computation time and Capkun equation is used to estimate communication overhead. Security analysis is performed using Pollard Rho methods. The validity and correctness of the scheme is evaluated using Burrows–Abadi–Needham (BAN) logic, a formal analysis scheme.

1.6 Thesis Contributions

There are number of contributions in this research dissertation which are listed below:

The first contribution is a lightweight cryptographic multi-factor authentication scheme that helps secure D2D communication in an open insecure environment. This is a novel cryptosystem that utilizes ECC with Elgamal for achieving confidentiality, integrity, and non-repudiation. Elgamal has a smaller key size which helps not only in reduction of operational and communication cost but also makes it usable on small miniature devices.

Usage of Elgamal helps in achieving asymmetric key exchange that is one of the key requirements in open and mobile environment. It also applies digital signature to achieve authentication while double hashing using SHAv3 combined with timestamp and blind fold challenge scheme provides three-dimensional security that is integrity, freshness, and mutual authentication.

Three-dimensional security using Double Hashing based on SHAv3 combined with multi-factor authentication provides integrity, confusion, diffusion, freshness and mutual authentication. Multi-factors used for authentication include onetime password OTP (Biometrics, random number), timestamps, challenge and password which provide mitigation of all major security attacks for complete secure communication. This scheme also offers reduced authentication overhead, communication and computation cost due to merged challenge-response scheme.

The proposed scheme enhances the security performance by addressing all security requirements. It reduces the authentication overhead for single hop and multi-hop communication. It has smaller computation overhead that helps devices to consume less computational power and has also lower communication overhead resulting in reduction of network traffic significantly.

1.7 Thesis Organization

The thesis comprises five chapters. The first chapter provides introduction to the thesis. The topics covered in the first chapter include research background, research challenges, problem statement, scope and importance of this research.

Chapter 2 discusses the security challenges of D2D communication and highlights the significance that D2D provides. Security requirements that are key for assessment of any security protocols are also highlighted. Security challenges for single hop and multi-hop such as MITM, replay, impersonation, identity reveal, rouge relay, DoS and masquerading threats are also identified. Current benchmark approaches are highlighted according to their workings. Analysis and verification techniques used in various research works are highlighted in this chapter.

Chapter 3 provides a detailed flow of LEMAP where only two devices are directly communicating but the communication must pass through multiple hops. This chapter also provides a detailed system design of two sub layers proposed as D2D security layer and MFA layer. The first layer is referred as Multi Factor Authentication (MFA) layer and second layer is referred as security level layer. Complete scheme details are explained in the section after system design. Analysis methods based on performance and computation are discussed at the end of the chapter along with formalization techniques used for validation and verification.

Chapter 4 provides verification of LEMAP and benchmarking using the formal analysis with BAN logic. The verification of LEMAP against security attacks is tested and performed and computational security is also verified using the mathematical proven security methods. The cost analysis of benchmarks and LEMAP is performed while in the last section, the simulation analysis of the proposed method is also performed.

In Chapter 5, the security requirements which are fulfilled by LEMAP are explained. The overall thesis is concluded with the details on significant contribution of this research. The future directions of this research are also outlined.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

The newly evolved 5G cellular wireless network is developed to provide high speed data, broadband and multimedia services with lower latency and energy consumption for various mobile devices. To achieve higher data rate, technologies like D2D are integrated into 5G. D2D communication is one of the key technologies incorporated to enhance 5G performance (Hussein et al., 2020). D2D is a direct communication method that enables two devices to communicate directly with each other without involving infrastructures such as access points (AP) or BSs. Bluetooth and WiFi-Direct are the most popular D2D techniques which work on unlicensed bands but on the other hand, cellular networks do not support direct over-the-air communication between users and devices. However, with the emergence of 5G, high data rate consuming applications and fast growth of Machine-to-Machine (M2M) applications, D2D communication plays progressively important role. It facilitates the discovery of geographically close devices and enables direct communication between these neighboring devices which improves communication capability with high data rate. Moreover, it reduces power consumption and delay in communication.

Despite many advantages, still many challenges and threats exist related to D2D communication. This research focuses mainly on security issues of D2D, threats involved in D2D during communication and secure mutual authentication. In this research, multifactor based mutual authentication scheme is proposed to overcome security issues during communication. Correctness and validity of the proposed scheme is verified by using formal

method BAN Logic. Next section provides basics of D2D, the challenges associated with D2D and the related works that have been done in D2D security.

2.2 Device to Device Communication

Device to Device (D2D) communication is a major and integral part of LTE-A and 5G. Especially, in the next generation cellular network D2D will be a major role player as it ensures high data rate and low latency. The research on D2D has been drawing attention since last few years where academia and industry are working together to standardize the process (Gandotra et al., 2017; Wong, 2017; Gandotra & Jha, 2016; Noura & Nordin, 2016; Gupta & Jha, 2015). D2D allows communication with the neighborhood devices which can communicate directly, and both have infrastructural network eNB, Core Network (CN) and Access Point (AP). D2D communication has shown that it can reduce huge network load, manage power consumptions, enhance network coverage and boost up traffic as well as reduction in delays (Gandotra et al., 2017). In academia, the D2D is introduced and used in LTE-A network to enhance communication performance (Li et al., 2017). D2D communication can work even though there is partial or no involvement of network infrastructure which is making it more popular for usage in the disastrous areas or low coverage services area (Biswash et al., 2017).

D2D techniques such as Wi-Fi-Direct (Trifunovic et al., 2011), Bluetooth, short-range wireless are already popular and in place. However they work without involvement of cellular network for communication and operate as an independent network. Cellular network doesn't allow direct communication even in the area of the cell device that is a fundamental block of the cellular network (Grossglauser & Tse, 2002). To achieve the cellular network services in 5G and LTE-A, the complete architecture and infrastructure

need to be updated for effective communication. There are plenty of researches that have been conducted in this area and have focused in some areas such as mode selection (Jänis & Ribeiro 2010; Shirani, & Kossentini, 2000), allocation of resources (Phunchongharn et al., 2013; Zulhasnine et al., 2010), interference control (Gu, J., Bae et al., 2011; Jänis et al., 2009) and power control (Janis et al., 2009; Yu et al., 2009). Industrial researches have been in the area of D2D for development of protocols such as the introduction of FlashLin Q (Theobald et al., 2014; Baccelli et al., 2012) that is used for communication of two devices. The standardization of D2D is going on through the Third Generation Partnership Project (3GPP) (Shen et al., 2012).

Another important aspect of D2D is the proximity-based services (ProSe) that allows the devices to find its neighborhood or nearby devices (Ta et al., 2014; Yang et al., 2013). The study of ProSe is underway in LTE-A network and its system architecture and network functions have been made. Current researches are being underway to develop the security and discovery aspects as well as on radio services (Oueis et al., 2017; Ferrús & Sallent, 2014; Husain et al., 2014). D2D assumes that communication can only occur if the devices are in proximity area. Initially for D2D communication, the authorization happens at the network controller and then devices can communicate once the full trust is established. The basic function of ProSe is to develop the safety infrastructure for public and this feature was first released as LTE-R12. There are two important parts of ProSe range that are Device tier and Macro Cell tier. Device tier refers to ProSe area of device where the device can locate and communicate with nearby devices while Macro Cell Tier means that area of ProSe is of BS. The Macro Cell manages the complete list of the possible neighbor list. The Macro Cell tier is responsible for the communication of cellular-related services. On the other hand, device tier is used for D2D communication.

2.2.1 D2D Communication Modes

The communication mode selection is a very important aspect of D2D communication and is considered as a major challenge when conventional cellular network is combined with D2D communication. The mode selection is the technique that determines whether the traffic between two communicating devices is established via the central network, Base Station (BS) or D2D mode. The method of spectrum allocation is also determined by the mode selection. Depending on whether devices in close proximity communicate directly or not, and whether D2D users and devices use dedicated channels or reuse channels of cellular users (CUs), three basic communication modes for D2D systems exist which are cellular mode, dedicated mode and reuse mode (K. Doppler et al., 2010). In the cellular mode, D2D devices can communicate conventionally through Base Station and requires more resources than the D2D mode due to the long communication range. In the dedicated mode which is also known as overlay mode or device to device mode, D2D devices are assigned dedicated radio resources and a dedicated spectrum to establish a direct link between the devices. In the reuse mode, which is also referred to as underlay mode, D2D devices reuse the same radio resources with cellular devices and this mode utilizes the spectrum resources shared with other cellular devices to establish communication between the D2D devices. The mode selection has been studied by many researchers and several mode of selection methods have been proposed.

Figure **2.1:** shows two major types of D2D communication that is Inband communication and Outband communication. Inband refers to licensed cellular frequencies or working under licensed frequencies that is a major part of the cellular network. Normal

network communication and D2D communication both work under the same licensed frequencies. Inband is more widely used as it provides a higher data rate and efficient utilization of frequencies (Asadi & Jacko., 2014; Asadi et al., 2014). Outband refers to the unlicensed frequencies in which different communication band is required and it is managed by devices itself. In Outband communication, data rate is quite low as compared to Inband. But it can use both D2D communication and cellular communication at the same time which requires extra computation and overhead (Asadi & Jacko, 2014; Asadi et al., 2014).

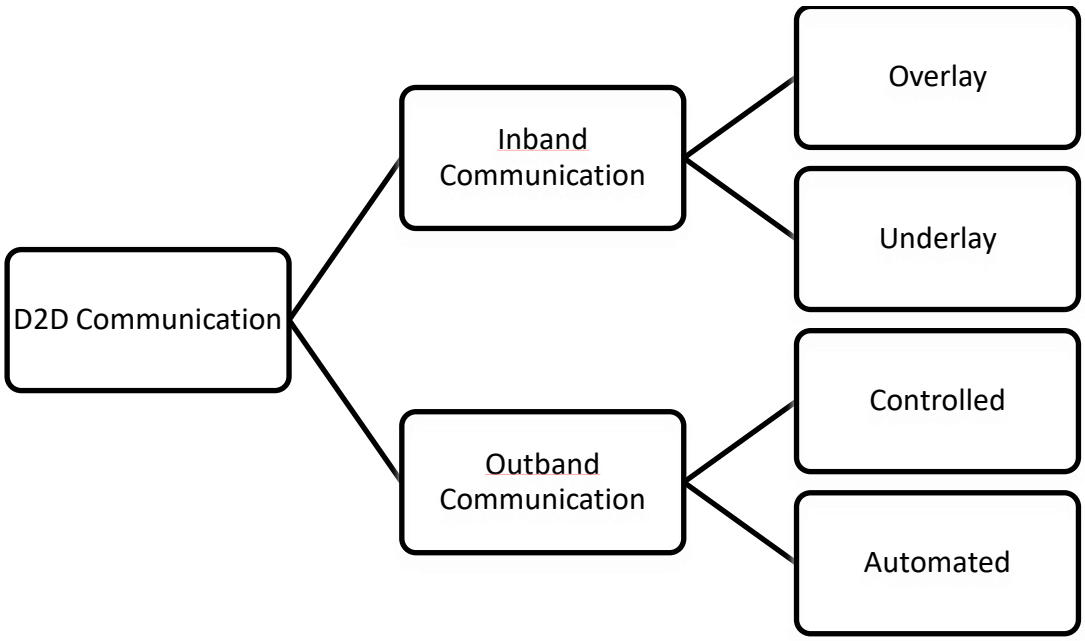

Figure 2.1: Modes of D2D Communication

Inband communication consists of Overlay and Underlay modes. In overlay mode different cellular band is used for communication while in underlay the same frequency band is used for cellular communication (Asadi & Jacko., 2014; Malandrino et al., 2014; Lin et al., 2013). Outband communication also consists of controlled and autonomous modes. In controlled transmission, the access point (AP) is serving as partial eNB and all communication take place through the trust established through the same cellular network.

In autonomous mode, communication occur without AP or eNB and gives very low coverage enabling some devices to take hold of the transmission mostly in disastrous area (Javed et al., 2019; Javed & Khan, 2019; Xu et al., 2012). Another important feature of D2D communication is soft handover which is possible even at local wireless communication level and provides an extended capacity about 5-10 Gbps between the devices. This was not possible in the current short-range free communication. The feature of handover is still in infancy and security is a big issue in D2D communication. Many researchers are focusing on this issue and proposing security algorithms to address it.

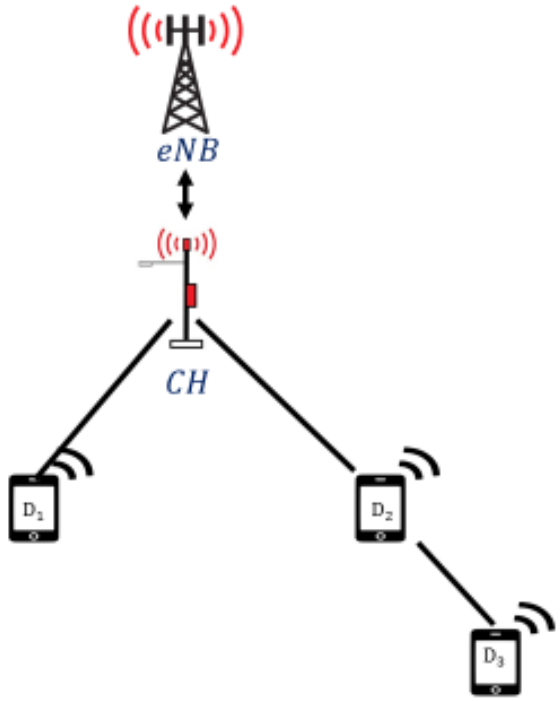

Figure 2.2: Basic Multi-hop D2D Communication

Figure 2.2 shows the multi-hop communication architecture of D2D that shows devices are controlled by eNB and devices can connect after establishing the authorization

and authentication with eNB. Cluster Head (CH) works as a relay agent for devices and provides services to devices in specific area. In recent trends, it is observed that D2D communication has achieved advanced features where multi-hop D2D communication is also possible. Multi-hop D2D communication will allow local area communication between multiple devices where one or more devices will act as relays for communication. Multi-hop D2D communication will also allow extended coverage so that device outside the coverage area can be accommodated.

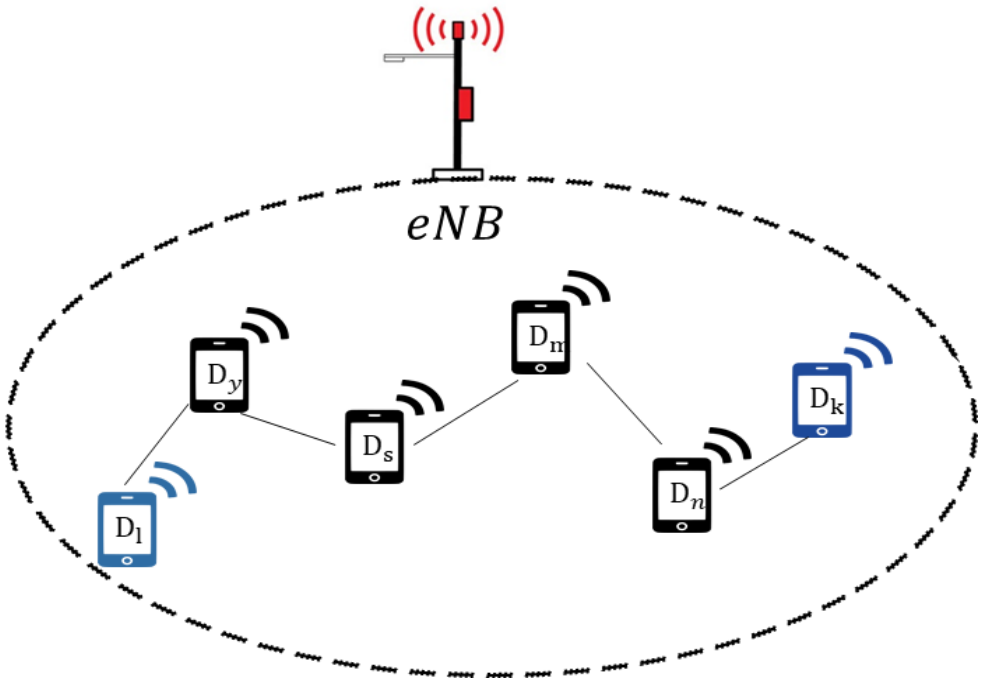

Figure 2.3: Scenario of Direct Communication

Figure 2.3 shows the scenario where devices are under the jurisdiction of eNB but would like to communicate directly after establishing successful trust. In this scenario, there are multiple devices as shown in Figure 2.3, however in terms of home setup or friend set up these devices have already established trust and would like to communicate directly by using relay devices (Tata & Kadoch, 2014; Imai et al., 2006). Figure 2.4 shows the scenario of multi-hop communication where some devices are out of the coverage range of eNB. This

scenario may arise in many cases such as disastrous area, non-cellular coverage areas or distant locations. To support the communication, the coverage to these devices can be provided if any of the registered devices is in range. This device will work as an acting eNB to those devices (Melki et al., 2016; Qin et al., 2016; Kim & Han, 2012). It is shown in (Nardini et al., 2017) that multi-hop communication must be done without boundaries and it supports a maximum coverage area without any geographical boundaries. This kind of communication will allow geo-fencing services to specific or multiple devices, in contrast to mobile-TV where everything is broadcasted in the cell. Integration of cellular network along with multi-hop network provides better Quality of Service (QoS) and flexibility. Another powerful feature is adaptability according to situations. There are several benefits associated with multi-hop D2D such as QoS, better data rate and network capacity.

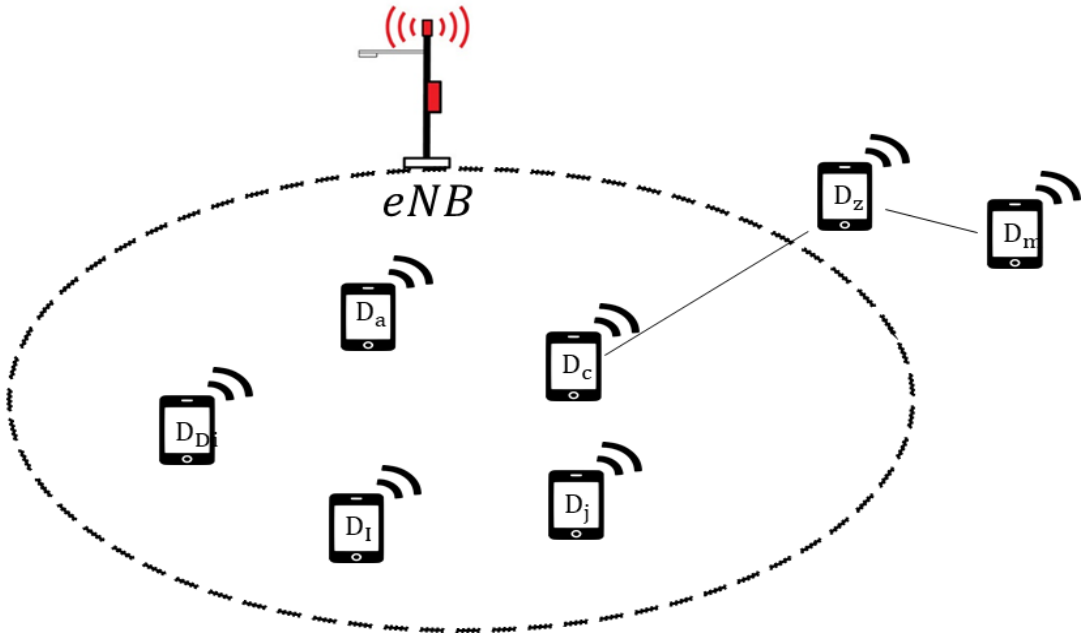

Figure 2.4: Out of Coverage Communication

QoS for devices in proximity is better since it has lower noise and signal to interference ratio as compared to traditional cellular networks. It is seen that multi-hop

results in the achievement of better QoS, but interference management must be made. In multi-hop D2D communication, higher modulation and available coding schemes result in saving bandwidth and high data rate due to small requirement of using channels to communicate to eNB. This results in two benefits as the same frequency for local communication can be used at full extent while main channel availability will result in a greater number of users, thus, increases network capacity. When communication will not be passing through the main cellular network and will be only used mostly for control message, it will result in reduction of great amount of network load. In case of partial or no network coverage, services will be available through the devices that are connected to the main network thus saving the cost of deployment of cellular systems and providing services in an area where the network cannot be deployed.

2.2.1 D2D Components

D2D components include devices, Cluster Head (CH) and eNB or Base Station. Each device must have been registered with the service provider through CH that acts as a gateway (GW) as shown in figure 2.5. Each device is assigned a unique pseudo-identity at the time of registration by eNB. Communication between devices takes place using these pseudo-identities. Each device works as decode and forward node to communicate with other devices. Initial registration is done through BS that broadcasts requests and allows verification. CH works as a forwarding device for specific area. Devices send requests to eNB through CH. eNB is a computationally powerful node that can manage all its underlying devices registration, authorization, verification, trust and all other data or service-related requests. Brief description of each component is given in Table 2.1.

Table 2.1: Description of D2D Components

Name	Description
BS	Base Station (BS) is a computationally powerful node that connects the local network to the core network. It performs routing control, internet provision, and all network related services. It manages eNB, verification and authorization control and also runs the proximity service control function that updates the neighbourhood information for devices (Tran, 2018).
eNB	eNB is an important part of Evolved Universal Terrestrial Radio Access (E-UTRA) in an LTE-A and 5G network. It performs authentication control for all devices and acts as security and interference control. It has high computational and storage resources and can manage parallel requests. It also broadcasts network membership and services to its members to manage the communication requests (Mach et al., 2018).
CH	Cluster Head (CH) is an intelligent forwarding device which serves a specific geographical area. It has low computational power and usually works as a forwarding device for all devices connected to the network. It forwards requests to eNB which manages authorization for all devices.
D_1, D_2	D_1, D_2 refer to devices that want to communicate with each other. D_1 sends communication request with D_2 to CH that forward it to eNB which can authorize both devices to communicate. It is a requirement that each device must be registered before becoming part of the area (Tran, 2018).

All the devices must trust CH and eNB, where eNB manages all the authorization, registration, re-authorization and communication control. It maintains the information table that contains all information about actual and pseudo identities of devices, its public key and neighboring information. The detailed explanation will be given in Chapter 3 in Section 3.3.3.

Figure 2.5 shows two devices D_1 and D_2 in a single cellular network which is managed by eNB. Devices are directly connected to CH in specific area which forward requests to eNB. Cluster Head works and manages device registration and handover for each area. Neighboring information is managed by Base Station (BS). The information that is shown in Figure 2.5 is based on infrastructural deployment of D2D.

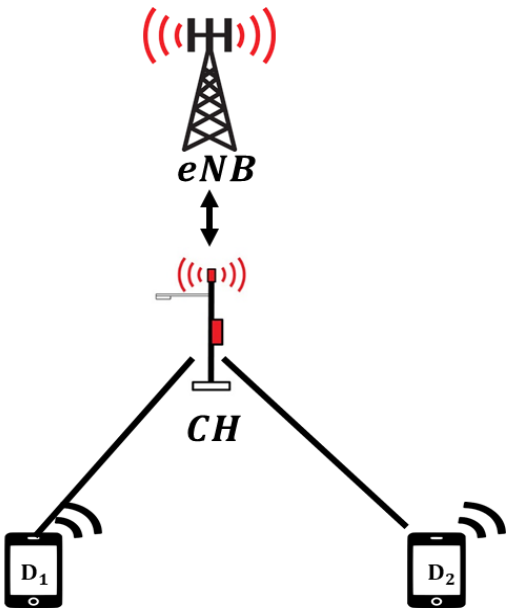

Figure 2.5: D2D Cellular Model

2.3 Security Requirements of D2D Protocols

In D2D communication, several security issues and threats still exist which have been identified in the literature. Many techniques have been proposed to overcome these threats.

These techniques cover some of the security requirements (Suo et al., 2012; Renauld et al., 2011; Fabian et al., 2010; Raya et al., 2006; Wallner et al., 1999). This section describes several security requirements for any D2D security protocols. These security requirements are generally for all types of communication either it is single hop or multi-hop network.

2.3.1 Confidentiality

The first security requirement is confidentiality which means data transmitted on open channel must be encrypted and only the intended recipient can decrypt and read it. To achieve confidentiality, a strong encryption scheme is required (Borgohain et al., 2015) to encrypt data so that it must be secure from adversaries during transmission.

2.3.2 Integrity

The second important security aspect is integrity which means when data is sent by the sender, it must be received to the recipient as it was sent. It means there is no tampering during transmission. To achieve integrity, a strong hashing technique must be applied (Li et al., 2014; Meister, 2013). After applying hashing, when recipient will receive data, its hash is checked and matched with actual message. If both are same, it means there is no tampering during transmission and hence data integrity is maintained.

2.3.3 Availability

Availability means services are available anywhere and at any time. In D2D communication, services should be accessible anytime and anywhere even under DoS or free-riding attacks(Lu et al., 2015). Free riding resistance refers that each UE must be registered as member of some eNB and have a unique identity. No free users can join the network until they register.

2.3.4 Non-repudiation

To ensure the reliability of the information transmitted, both parties must attach a private seal of a legitimate user as the basis for the information during communication. At the same time, this also makes it impossible for a sender to deny that one had not sent a message. Non-repudiation is achieved when none of the transmitting devices deny the sent message request. Digital signatures are used to achieve non- repudiation. Timestamps also help to document date and time when the message was sent which can help in achieving non-repudiation.

2.3.5 Mutual Authentication

To ensure a secure communication process between two devices, communicating parties must register with base station or central server and authenticate each other securely. In mutual authentication two devices authenticate each other before start of actual communication. Thus, the security scheme must offer secure mutual authentication.

2.3.6 Anonymity

Anonymity is a state in which somebody public identity is unknown or hidden. Anonymity must be achieved as participating communicating devices should not be recognizable by other communicating devices. To achieve anonymity, pseudo identity is used to protect the real identities (Hsu et al., 2017).

2.4 Security Challenges in D2D

There are many challenges that are associated with D2D communication like device design (Fodor et al., 2012), distance between UE, distance between BS and device (Wang & Chu, 2012), interference management (Shalmashi et al., 2013), device mobility management

(Zhang et al., 2017; Yilmaz et al., 2014), energy consumption (Jiang et al., 2015; Zhou , 2014) , spectrum usage and frequency selection (Monserrat et al., 2015; Sakr & Hossain, 2015), offloading techniques (Andreev et al., 2015). Resource allocation and resource management (Phunchongharn et al., 2016; Jiang et al., 2015) is one of major challenges due to its inherent limitations. Mode of D2D either licensed or unlicensed or another form such as Assisted or Un-Assisted makes it a challenge to support all modes (Asadi et al., 2014; Han et al., 2012). Security is a major challenge of D2D (Cheng et al., 2017; Haus et al., 2017; Habib et al., 2017; Hossain & Hasan, 2017; Javed et al., 2017; Wang et al., 2017; Zhang, 2017; Fujdiak, 2016; Li, 2016; Kato, 2016; Ramadan et al., 2016; Sedidi & Kumar, 2016; Yao et al., 2016; Zhang et al., 2016; Abd-Elrahman et al., 2015; Liu et al., 2015; Xi et al., 2014). Other aspects related to security are also identified in multiple types of research (Li et al., 2018; Ahmad et al., 2017; Lien et al., 2016; Kato, 2016; Tehrani & Yanikomeroglu, 2014; Raghothaman et al., 2013).

This research focuses on security and privacy issues that have appeared with the advent of D2D. Security solutions presented for LTE and cellular networks cannot address the issues that have arisen due to the inception of D2D. For instance, in LTE there is only mutual authentication between the users and main cellular network. Moreover, there is an agreement between users in the main cellular network. While in D2D, there is mutual authentication between the devices and similarly, key exchange after negotiation between devices also takes place. In the absence of security measures, the whole communication architecture can be compromised. There are a number of attacks that can occur either actively or passively (Zhang et al., 2017; Dholeshwar & Salapurkar, 2016; Wang et al., 2015; Wen et al., 2013; Jibi & Rakshanda, 2013). Some of the examples of most common attacks that are possible on D2D are eavesdropping attack, DoS, MITM attack, impersonation attack,

side channel attack, forge attack, free riding attack and rogue device attack. There are several privacy attacks or security issues that can be made successful in the current infrastructure of cellular network. Among those issues are privacy issues related to user identity privacy, location privacy, data privacy and unwanted messages.

Owing to the increased number of attacks on D2D communication, researchers in academia and industry have invested many efforts in designing secure communication algorithm. Research about D2D security is still at early stages. Some researchers used most widely used security algorithms for normal communication over network. A few researchers have used Mobile Adhoc Networks (MANETS) algorithms for proposing security solutions (Panaousis & Alpcan, 2014). Most of the security algorithms only focus on one or two security requirements that must be met for designing a secure technique for D2D. Another issue is that D2D designing is still under development and create an emerging security requirement issue. Some algorithms are proposed based on old requirements while a new requirement has emerged. Another issue is that devices involved in D2D have variable computation power and due to applications probability across different devices; any device can act as communicating device. Hence, the developed solution must consider a low computation device. Owing to this issue, many of the proposed solutions may be vulnerable to new era attacks (Yin et al., 2018; Aggarwal et al., 2017). Numerous techniques and approaches focused on secure data transfer, others focused on authentication and key agreements. These proposed techniques for D2D security usually focused on the security of data, sharing of data securely, authentication and agreement of keys. Most of the security techniques focus on single operator and security issues are considered individually without combining the security techniques with key management techniques. Research in D2D especially security in D2D is still at early stages and standardization has not been developed.

2.5 Security Threats in D2D

D2D inherently poses number of challenges and threats that can be categorized into single hop and multi-hop challenges.

2.5.1 Single hop Challenges

As discussed in previous section that D2D is more open to any kind of attacks that were even non-existent in traditional 4G and 4G-LTE network. But allowing the devices to act as relaying devices opens as a whole bunch of attacks that should be handled in a different way; as the devices have low computing power and are required to do number of new tasks which were not its actual part prior (Javed & Khan, 2019; Javed et al., 2019; Khan et al., 2017). The following is the list of attacks that can occurr in a single-hop D2D communication.

2.5.1.1 Denial of Service Attack

In this type of attack, an attacker targets and sends huge traffic to a device or service and makes it inaccessible for legitimate users. An attacker can both flood the server or the IoT device with many fake requests and crash it due to resource constraints. This attack can be from one device or multiple devices. In D2D DoS attack, an attacker exploits the capabilities of eNB or BS by sending repeated delivery of false messages. DoS usually performed through replay attack that may result in stoppage of access to some service or complete resource (Javed & Khan, 2019; Javed et al., 2017).

2.5.1.2 Replay Attacks

The intruder can send intercepted message again and again or delaying message. Adding timestamp with message is a solution to avoid replay attacks because it ensures message freshness. During mutual authentication process, an adversary can intercept the

traffic and takes the message from legitimate node, later use this message after a certain amount of time and send to the base node to perform replay attack against base node (Hu & Evans, 2003). It is possible that adversary may not be able to decrypt the message, but it continues to send the message to BS or eNB that can result in DoS attack. If this attack is initiated from BS to sub-nodes or devices, this results in MITM attack. An example of attack is shown in Figure 2.6.

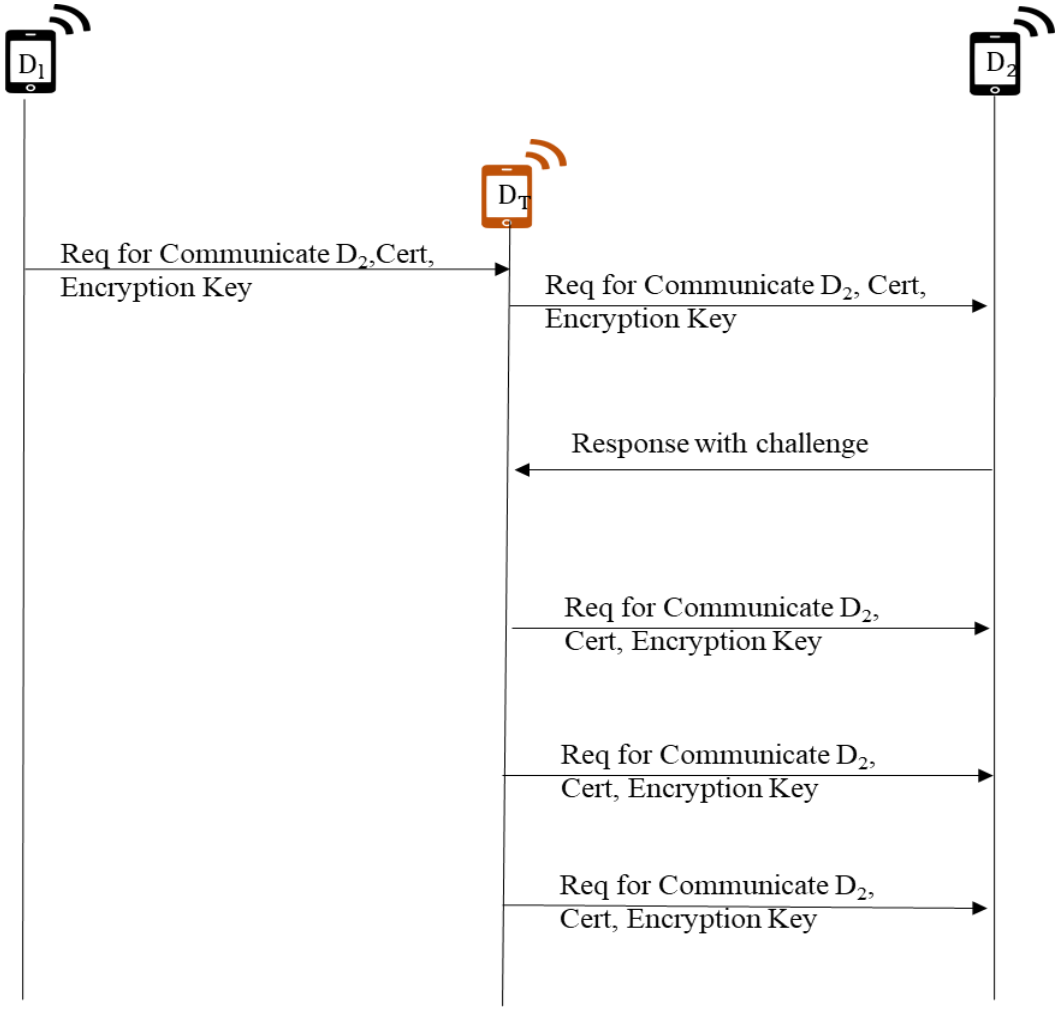

Figure 2.6: Replay Attack

2.5.1.3 Masquerading Attack

A masquerade attack uses a forged or fake identity of a legitimate device to gain unauthorized access to target device or network. Masquerade attacks are generally performed by stealing login credentials (IDs, Passwords) or by finding gaps in authentication process. If authentication process is not fully secure, it is highly vulnerable to masquerading attack. A masquerade attacker can be from inside network or from outside network as well.

In this kind of attack, one device pretends to be another device (Faria & Cheriton, 2006). This device can be a device acting as the Internet of things (IoT) or Master Relay Base Station (MRBS) or Non-transparent relay (NTR). This attack may happen by using two kinds of techniques; one is identity theft, and another is Rouge Base Station. If the identity theft is successful, then it will result in MITM attack and even if it not successful it will result in DoS attack (Khan et al., 2017; Javed et al., 2017).

2.5.1.4 MITM

MITM is a type of attack in which an adversary eavesdrop communication between two devices, intercepts the communication and starts conversation with devices. The legitimate devices cannot recognize either they are talking to legitimate device or an adversary. The main reason of MITM attack may be weak encryption, weak passwords or not using digital signatures in communication.

MITM attack is successful if encryption technique used is not too strong. In MITM, normally an attacker can decrypt the response messages successfully (Javed et al., 2019). The attack is illustrated in Figure 2.7 that shows two devices directly communicating while the attacker is listening to the whole communication. MITM attack is one of the most challenging attacks as it may result in whole communication compromise.

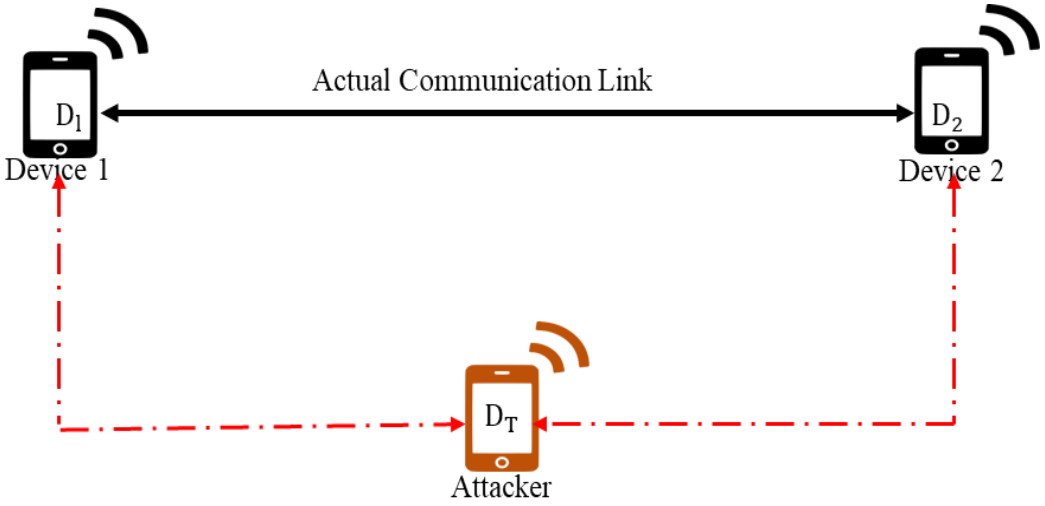

Figure 2.7: Man in the Middle Attack

2.5.2 Multi-hop Challenges

Multi-hop D2D communication introduces the following security attacks:

2.5.2.1 Rogue Device Attack

A rogue device is a device that is connected to your network without authorization. This type of device gets authorization due to lack of proper authorization rules in the network and can become a bot to launch DoS attack. Also a rogue device can be a sniffer which eavesdrop network and causes sensitive data theft, destruction of data, loss of services and can insert malicious data in the form of viruses, keylogging or pornography(Gupta et al., 2018; Javed et al., 2017).

2.5.2.2 Timestamp exploitation attack

Timestamp is used for message freshness which ensures that received message is fresh and within the time limit specified for that session. If the timestamp exceeds the time limit, the message is considered by an adversary and it is discarded. Sometimes an adversary intercepts the session within the specified time limit. This opens window for adversary to

initiate timestamp exploitation attack. In D2D communication, an adversary can intercept a message during specified timestamp to use it later.

2.5.2.3 51% Attack

The 51% attack occurs when several malicious devices supersede the number of legitimate devices. As most of the trust-based system relies on feedback from the devices, thus if the number of malicious devices is higher they can jeopardize the whole communication by giving false beliefs (Bastiaan, 2015).

2.5.2.4 Eavesdropping Attack

In an eavesdropping attack, an adversary sniffs network communication to gain unauthorized access to data, steal IDs information, modify or delete sensitive information. An adversary uses some software or application to sniff network communication and takes advantage of weak passwords, weak encryption, and weak privacy to steal information. Later adversary uses this private information to compromise nodes and degrade network performance. To avoid eavesdropping attack, strong encryption and authentication mechanism is required.

2.5.2.5 Pre-image Attack

In Pre-image attack, an attacker tries to find specific hash value of some message by using brute force attack. A cryptographic hash function should resist attacks on its preimage. To mitigate pre-image attack, double hashing technique is used. All the messages sent or received by one device to other, use double hashing digitally signed by the private key of the sender. Double hashing is a collision resolving technique in Open Addressed Hash tables. Double hashing uses the idea of applying a second hash function to key when a collision occurs.

2.5.2.6 Impersonation Attack

Impersonation attack is an attack in which an adversary adopts a fake identity of a legitimate device and starts communication with the target device. The target device cannot recognize either it is the legitimate device or the fake device.

2.6 BAN Logic

One of the famous techniques for formalization called as BAN logic is used for verification of proposed security algorithm called as LEMAP. The various inference rules are listed below; the lists above are the inference rules of the. Burrows, Abadi, and Needham (BAN) Logic. There are many rules but below mentioned are used commonly. A detailed description of the rules can be found in (Sufatrio & Yap, 2008; Cohen, 2005).

First of the formalization techniques was made by Dolev and Yao for the security algorithms but their proposed methods have a restricted use due to complexity involved and lack of system details. Later, the Dolev and Yao introduced the capabilities model for attackers, and it was based on assumption that attackers can take control of the system resulting in wide usage of the protocol.

Still the proposed system had couple of limitations such as limitation of intruder was the restriction on usage of random number. Encryption and decryption were not part of the intruder system that resulted in in-complete attack. With this method any kind of weak security would have the same result as strong encryption.

One of the well-known computation model that was proposed was Bellare and Rogaway that was based on scenarios such as each attack must be made in scenario and is

played or executed like a game. The task is to create an algorithm where user cannot win after endless possibilities.

Burrows, Abadi and Needham algorithm, often referred as BAN logic is one the decent algorithms for checking the security of the system and still widely used for verification of security protocol using formal ways. BAN logic is based on symbols and can be used to perform attack analysis on security algorithms. It is based on rule that each goal means the authentication properties of security algorithm. It is also based on condition that algorithm is converted into definition and rules based on BAN logic principles.

Another condition is that security model must be identified before conversion. Next condition is to use the inferencing rule. Based on all these assumption, model and rules it is verified that security goal can be achieved or not. If it cannot be achieving, then the security protocol is vulnerable to attack and thus modification in algorithm is required else it passes the security test.

There are number of issues that were identified in BAN logic are that it doesn't handle the certification authority rules properly and it also cannot handle the Public key infrastructure (PKI) well. It also makes some assumption that might result into security weakness. In order to handle this an extension on BAN logic has been proposed. The extension in BAN logic resulted in the betterment of certification authority rules but made it complex.

One of the extensions in BAN logic is proposed by Sufatrio and Roland that handles few weaknesses such as goodness property of base station (BS) (Sufatrio & Yap, 2008; Cohen, 2005). The first issues identified by authors was the basic assumption on certificate trust as shown in Equation 2.1

$$\prod(K_{BS}^{-1})$$ Equation 2.1

The assumption statues that BS believes in Equation 2.1 as shown in Equation 2.2.

$$P \mid \equiv \prod(K_{BS}^{-1})$$ Equation 2.2

Another issue that is identified is jurisdiction of message, it must be assumed that sender can send the message as he has jurisdiction as shown in Equation 2.3.

$$A \mid \equiv B \rightarrow BS\ (X, P)$$ Equation 2.3

where X is secret key and P is public key

The revocation of certification possibility can be introduced but it must be assumed that is missing in first that keys of CA are good and protected.

2.6.1 Extension by Sufatrio and Roland

To deal with identified issues, there were extensions proposed by authors.

2.6.1.1 For the idealization of certificates

It is assumed that certificate is fully signed by CA and it has good and secure secret key and public key pair. It removes the need of creating another message stream of trust between CA or BS and other nodes.

2.6.1.2 For the message Recipient

In order to make sure that message is sent by a correct recipient the message is signed using the private key of his own as shown in Equation 2.4

$$\prod(K_{BS}^{-1}\ ,(\ BS\ (X, P)))$$ Equation 2.4

It ensures that the certificate is valid and can only be generated by CA. The goals for Device to Device(D2D) authentication protocols should be

$$D_B| \equiv \ D_A \overset{K_{D(A,B)}}{\longleftrightarrow} D_B \qquad \text{Equation 2.5}$$

The above rule in Equation 2.5 says the device D_A believes that there is a secret key must be calculated without sharing between both devices D_A and D_B.

Similarly, the device D_{Ai} believes that there is a secret key must be calculated without sharing between both devices D_B and D_A. If the secret key $K_{D(A,B)}$ or SP is created and shared that key will be used as communication point for both devices as shown in Equation 2.6.

$$D_A| \equiv \ D_B| \equiv \ D_A \overset{K_{D(A,B)}}{\longleftrightarrow} D_B \qquad \text{Equation 2.6}$$

The message in BAN logic in extension are same as old one and thus all the old rules holds true along with trust.

2.7 Simulation Software

There are number of simulations software's that can be adapted for this research, out of which NS2, NS3, OPNET and NCTUns are one of the notable software. This research adapted a D2D communication of discrete even that why NCTUns is selected to be simulation software. It also provides implement IEEE802.16j protocol that helped this research in focusing on real implementation. NCTUns is high fidelity network simulator that can also work as emulator. Its an opensource platform that has integrated wireless network and VANET implementation that make it easier and more widely used among other

simulators. It also provides fast feedback look that helped this research in implementing sender and receiver messages among base station, devices, and head. This research also used Microsoft Excel to process the data logs received from NCTUns.

2.8 Current Approaches

Several approaches and techniques have been discussed in previous sections which can cover few security requirements and do not complete support for all the security requirements. Some of the research work that has been done used RSA, ECC, DH and other algorithms. This research selected three benchmark protocols based on their completeness, citations, standardization, and openness. Openness refers to the openness of algorithm along with its source code or easy to reproduce. The following three protocols explained are treated as a benchmark for this research.

2.8.1 Chaotic Map-Based Anonymous User Authentication Scheme

Chaotic Map-Based Authentication Scheme (Sandip Roy et al., 2018) is used for securing the data and perform D2D communication. The protocol uses an assumption that eNB is secure and trusted third party while two devices A and B want to communicate with each other. The device A will have to send a request to eNB for authentication to communicate with device B. The proposed scheme consists of registration phase and then authentication phase.

Firstly, registration phase is completed. In this phase, a user U_i gets his smart card SC from the medical Server S. Then communication between user U_i and Server S takes place on secure channel.

Secondly authentication phase starts. The user U_i selects its ID_i, password PW_i, biometrics B_i and 1024-bit random number b. Next by using fuzzy extractor generation procedure, U_i generates (αi, βi) = Gen(Bi), and computes masked password RPWi = H(H(IDi||PWi)||αi) and C = H(H(IDi||PWi||b)||αi). Then U_i submits (ID_i, C) to server S via a secure channel. S receives the registration (ID_i, C) from U_i and selects a 1024-bit number mk as its secret master key, which is known to this server only, also S selects a 128 bit random number r. S saves some parameters into smart card and sends it to user via secure channel. S also saves these parameters in its database as well. As user receives smart card SC from server S, it saves some parameters in smart Card. To access services from server S, user must login to the system. Here login phase starts.

User inputs his ID, Password and biometric at the sensor. Using reproduction procedure, smart card computes and confirms with the stored credentials. If mismatch is found, the login request is terminated immediately. Otherwise Smart Card computes ID and masked password. SC generates a random number RN_u, time stamp of user and now sends login request to S via a public channel. Now the authentication phase starts. When S receives login request from U_i, both mutually authenticate each other. After this both U_i and S establish a common shared secret key which is used in future for secure communication between them.

S receives login request at time TS_1^* and verifies maximum transmission delay. If verification is not successful, S terminates the request immediately. Otherwise S computes if KA' = KA and ensures that $ID_i' = ID_{i,}$. S searches for the pair (ID_i, r) in its database if this pair is found, S further computes and verifies other parameters. After verification S accepts the login request and considers the U_i as the authentic user. S generates a random number

RN_s and current timestamp RS_2. S creates SK*su* common secret key shared with the user for the current session. Finally, S sends the authentication request to the user via secure channel.

Upon receiving message from S, smart card SC verifies the timestamp TS_2 and maximum transmission delay, U_i calculate current session key shared with S as SK_{us} which is same as SK_{su}. Using this session key Ui assumes that S is an authenticate server. Also, the current session key $SK_{us} = SK_{su}$ is mutually verified and established. Figure 2.8 shows the timing diagram of first benchmark.

2.8.2 2PAKEP Security Scheme

Several protocols have been suggested and developed by researchers for mutual authentication. The benchmark selected is one of the key protocols recently proposed in the 2019 for D2D authentication and key agreement (Xu et al., 2019). The proposed scheme introduces a secure lightweight mutual authentication scheme using asymmetric encryption. In this proposed algorithm, Multi-factor authentication parameters which includes (Biometrics, Nonce, Timestamp) OTP= BIOMETRICS & NONCE(R) are used which ensures better security and help mitigating replay attacks. Despite these strengths, the proposed scheme has also some threats and weaknesses. In the proposed scheme, ID is sent as plain text without encryption on open channel, this can result in location -identity reveal attack as explained by (Loreti et al., 2018; Airehrour et al., 2018). Data is shared un-encrypted on open channel which can cause modification of timestamp resulting in replay attack. Also, this can lead to Man- in- the -Middle attack as described by (Diallo et al., 2017; Spiekermann et al., 2015; Chen & Zhao 2012).

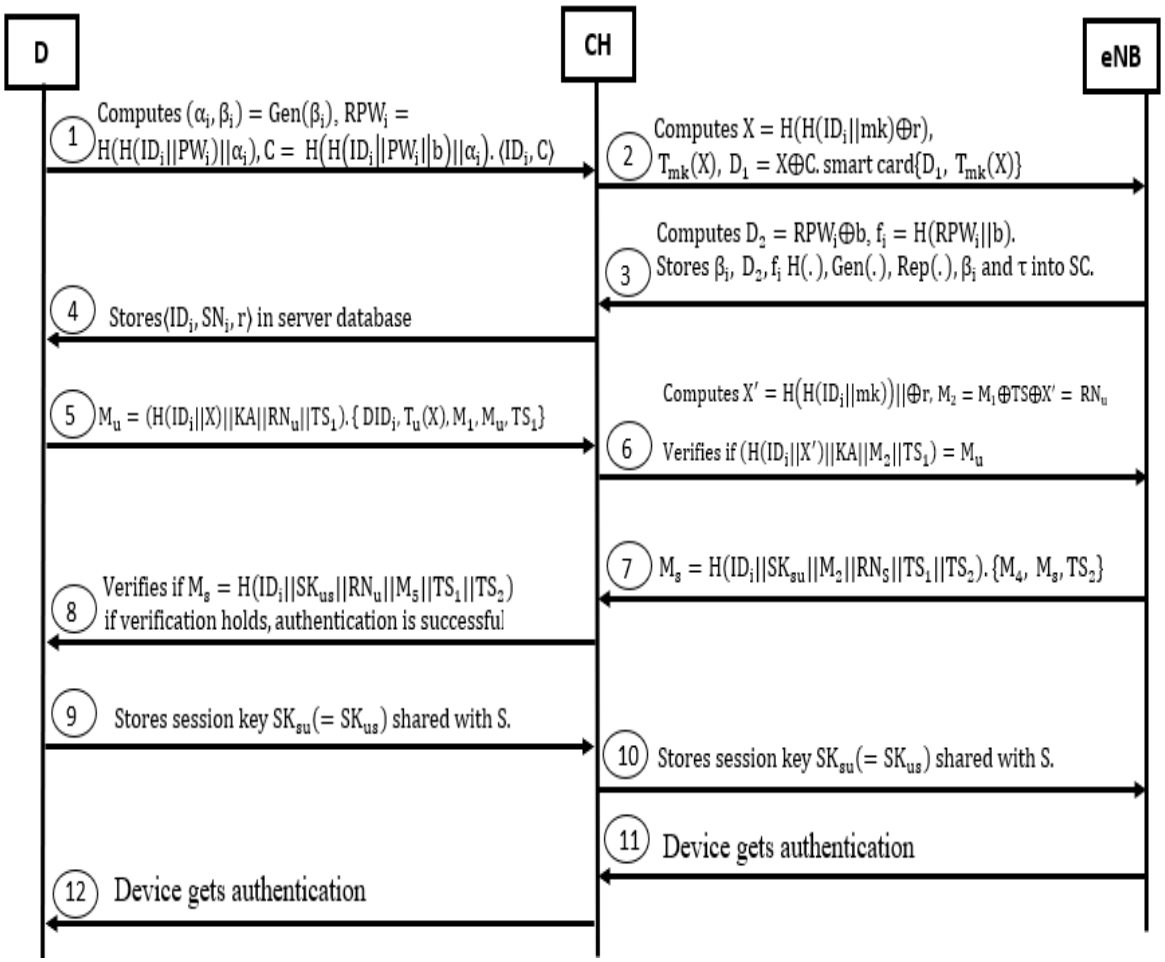

Figure 2.8: Chaotic Timing Diagram

Non-encrypted data is received at Cluster Head. This can result in Rouge relay attack by Nenvani and Gupta (2016), Yim (2016). An attacker duplicates a legitimate node in rogue relay attack which may lead to disturbance in service. The proposed scheme uses Single Hash function when sending data on open channel, this can result in Weak-collision attack by (Nimela et al., 2019; Kishore & Kappor, 2016; Mendel et al., 2009).

The proposed scheme uses some assumptions. The server is assumed to be secure and trusted node. The adversary can obtain exchanged data, modify it and replay data. It is assumed that Server Node is not physically protected. An adversary can steal information from its memory. Using the well-known Dolev-Yao threat model, it is assumed that the two

parties communicate in an insecure channel. This scheme has three phases: the initialization phase, the registration phase, and the authentication phase. The initialization phase and the registration phase are executed by the System Administrator (SA) in a secure environment. In the authentication phase, the Server Node (SN) exchanges information with the server on insecure channel and performs authentication and key agreement scheme.

First, SA generates a master key K_{ser} for the server and stores it in server memory. Then SA starts process for registering SNs and APs. For each SN, the server generates a unique identity ID_{SN}, a unique r and a random P_{Ks}. The SA stores tuple (IDSN, ASN, BSN, PKs) in the SN's memory, and stores ASN , X and PKs as a tuple <ASN , X, PKs> in the server's memory. The server may store multiple such tuples. For each AP and the server generates a unique identity ID_{AP} and stores in server's memory. In authentication phase, SN generates random number n1 and timestamp t1. SN Computes and sends message (ASN; S1; S2; t1) to the AP. The AP forwards this message to the server by placing its ID_{AP}. The server checks ID_{AP} and A_{SN} in its database if check fails terminates the session. Also checks the maximum transmission delay. The server also Generates n2, timestamp t2, and a unique r+ and Sends message (S3; S4; S5; S6; t2; IDAP) to the AP. The AP forwards the information received by the server to the SN and drops its identity IDAP. SN checks the validity of t2, computes random number n2 and checks if Checks S6*? = S6. Computes Ks and replaces the parameters A_{SN}; B_{SN}; P_{Ks} with the parameters A^{+}_{SN}; B^{+}_{SN}; P^{+}_{Ks} respectively in its memory. Store session key Ks. Figure 2.9 shows the timing diagram of two-factor timing diagram.

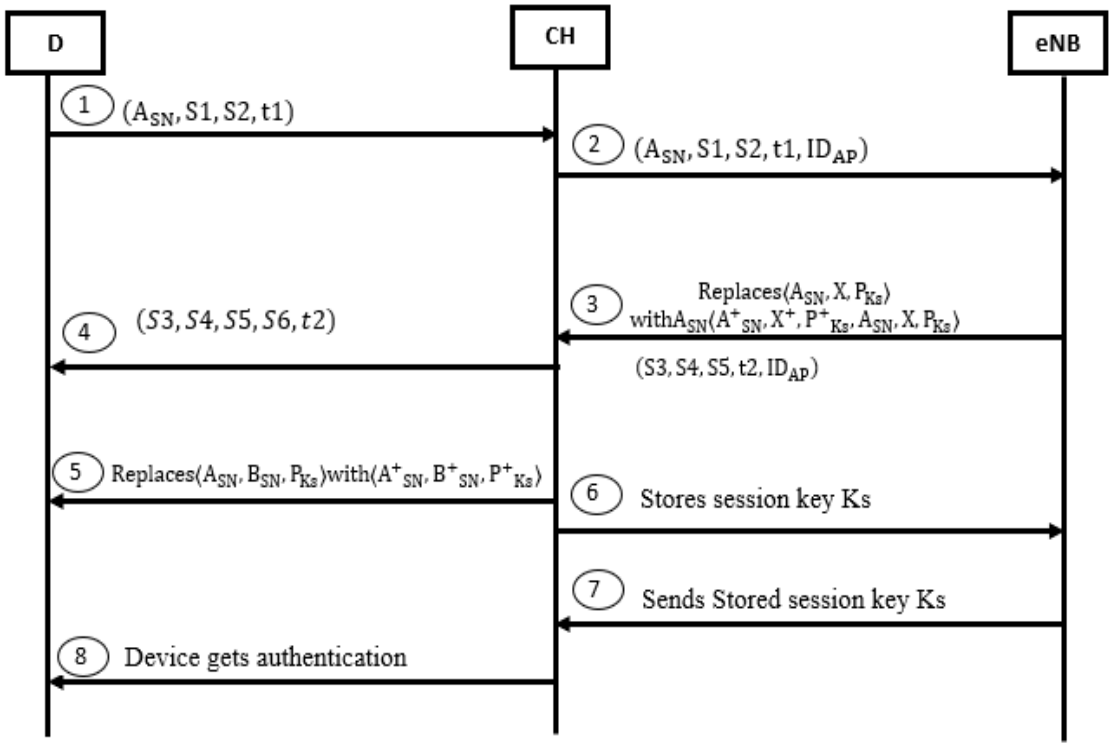

Figure 2.9: 2PAKEP Timing Diagram

2.8.3 Two Factor Security Scheme

This is another benchmark in which mutual authentication scheme is proposed by Park et al. (2018). It is based on multi factors like password, random number and smart card. The proposed scheme uses some assumptions that all devices are synchronized, and the open channel is secure. This scheme helps in mitigating replay attack and provides Multi-level Hashing for rigorous authentication and integrity. The proposed scheme also provides Quantum attack safety.

Despite above strengths of the proposed scheme in this paper, the proposed algorithm is still vulnerable to some threats and weaknesses. In the proposed scheme, ID is sent as plain text without encryption on open channel, this can result in location -identity reveal attack as explained by Loreti and Airehrour (2018). When data is sent from user to Cluster

Head, an unencrypted message is sent on open channel which can cause modification of message and hence can result in Man in the Middle attack (MITM) by Nenvani and Gupta (2016). Hash of data is sent without timestamp; this can cause Replay Attack Melki et al. (2019). Smart card is used as multi factor authentication. Smart card is vulnerable to Identity theft & Duplication of card threats by Amin et al. (2017) while also extra card reader is required.

In this scheme, the first phase is registration phase. The user Ui chooses IDi, Password PWi and random number ri. The user computes pseudo PIDi and Hash HIDi and send to Server S on secure channel. The Server S randomly chooses a number ni as it receives PIDi and Hash HIDi. The server S sends a smart card SC with parameters AIDi, ~n, and PWNi to the user Ui. The random number ri is stored when the user Ui receives the smart card SC. Finally, the secret parameters AIDi, ~n, PWNi and ri are stored on the smart card. Now the login phase starts. The user Ui inserts his smart card SC into the card reader and inputs IDi and computes HIDi and send this to server S. Server generates a random number ki. Server sends ki and SKDi to user. The card reader computes Ski and send HID'i to the smart card. The smart card SC checks HIDi = HID'i. If the condition is true the user Ui can put his password PWi, otherwise the session is terminated. The smart card checks if PWN'I = PWNi. If condition is true, then user moves to next step otherwise the session is terminated. The smart card SC chooses a random number ai and computes Ai. Then Ai, CIDi, MIDi, and Tu_1 are sent to the server S. Figure 2.10 shows a timing diagram of 2PAKEP.

Once verification phase starts. After receiving Ai, CIDi, MIDi, and Tu1, the server check time stamp maximum transmission delay. If Ts1 = Tu1, it means condition is true otherwise session is terminated. The server performs further steps. The server computes and

verifies if HIDi = HID'I, also checks MIDi = MID'I, if the equality holds, then the server moves to next step. The server chooses random number bi and computes Bi. Then, Bi, VIDi and Ts2 are sent to user Ui at time Ts2. The user Ui receives the above information at time Tu2. Ui checks Tu2 = Ts2 and VID'i = VIDi. If it is true, then proceed to further step. The User Ui sends WIDi and Tu2 to the server. When the server receives this information, it checks maximum transmission delay Ts3 = Tu2. When conditions hold, S computes and checks WIDi = H1(VIDi = H1(VIDi)||LBix || LBiy || Tu2 When the equality in this judgment equation holds, the authentication succeeds. Otherwise, the authentication fails, and the session is terminated.

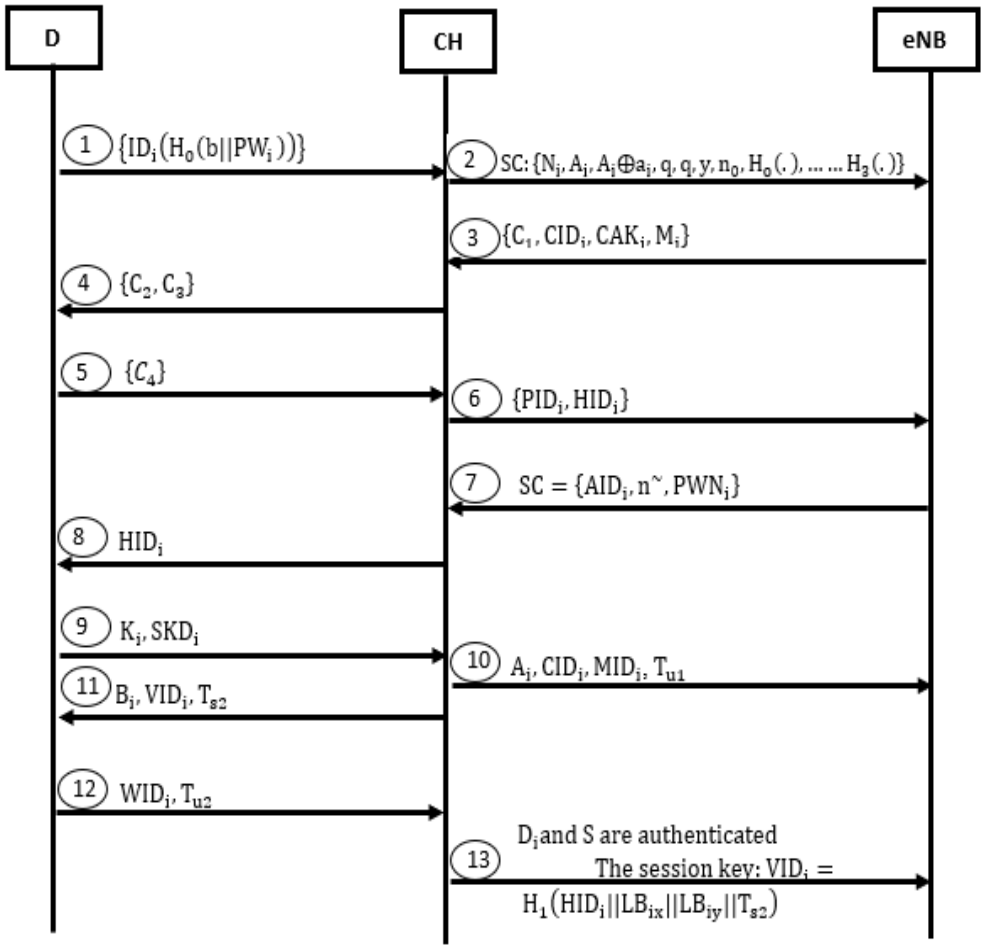

Figure 2.10: TwoFactor Timing Diagram

Table 2.2 defines the various attacks that have been mitigated by each benchmark protocol and the proposed protocol.

Table 2.2: Attacks Mitigated by Protocols

Attack Type	**LEMAP**	**2PAKEP**	**TwoFactor**	**Chaotic**
A Preimage attack	Yes	No	No	No
Man in The Middle Attack (MITM)	Yes	Yes	Yes	Yes
Impersonation Attacks	Yes	Yes	Yes	Yes
Privileged-Insider Attack	Yes	Yes	Yes	No
Mutual Authentication	Yes	Yes	Yes	Yes
Replay Attack	Yes	Yes	Yes	Yes
Password Change attack	No	Yes	Yes	Yes
Pass-the-Hash (PtH) attack	Yes	No	No	No
Masquerading Attack	Yes	No	No	No
Repudiation Attack	Yes	No	No	No
Denial of Service Attack.	Yes	No	No	No
Eavesdropping attack	Yes	No	No	Yes
Rogue Relay Attack	Yes	No	No	No
Malicious Card Reader Attack	No	No	Yes	Yes

2.9 Analysis Method

BAN logic proof was proposed by Burrows and Abadi in 1989. Authentication protocols are the basis of security in many systems. In order to ensure the correctness of a proposed scheme, many studies offer the BAN logic model to prove that their authentication protocols are effective, using many logic symbols and formula rules in the proof process.

The proposed scheme LEMAP has been verified and analyzed using BAN Logic in Chapter 4. The correctness of the proposed scheme is verified against different security threats and attacks. Benchmark protocols have also been analyzed using BAN Logic and compared with the proposed scheme. Compared with previous schemes, the main advantages of the proposed scheme are its low computation and communication cost, guaranteed security and better adaptability to actual client server communication environments.

2.10 Summary

In this Chapter, an overview of D2D communication, different modes of D2D communication, components of D2D and D2D security protocols have been discussed. The related work and research done so far in D2D security has also been discussed. The chapter also describes about security requirements for D2D communication and threats involved in D2D security. Detailed explanation is presented about security challenges and attacks in D2D. Security threats and proposed schemes in benchmarks have been explained. Threats mitigated by different benchmark protocols and the proposed scheme LEMAP is presented to show how much better security proposed algorithm offers. Benchmark algorithm and the proposed scheme all use ECC for lightweight and strong encryption. This chapter discusses the major security issues which can be faced and encountered using security algorithms and security requirements that any security algorithms should meet. Finally, this chapter

discusses the performance analysis method BAN Logic used in literature to validate and verify the correctness of the proposed security schemes.

CHAPTER 3

SYSTEM DESIGN FOR LEMAP SCHEME

3.1 Introduction

Device to Device (D2D) communication is a peer-to-peer communication mechanism between devices without need to communicate through intermediate node and operates in a highly open and dynamic environment. Despite many advantages of D2D multi-hop communication, there are many security challenges associated with D2D and it is vulnerable to several security threats. To securely operate D2D and to overcome challenges it faces, a multi-factor authentication security mechanism is required. This chapter aims to introduce a Lightweight ECC based Multi-Factor Authentication Protocol (LEMAP) for Device to Device Multi-hop Cellular Communication. The first section of the chapter discusses details of multi-factors which work combine to provide secure and lightweight security algorithm for secure D2D communication. Once mutual authentication is done using multi-factors and device trust is validated then devices can communicate securely over open channel. D2D communication allows devices to act as a non-transparent relay (NTR) where participating devices can decode and forward requests to the neighboring devices. Formal methods provide validation of proposed security mechanism. BAN logic has been used to analyze the secrecy of LEMAP and other benchmarking security schemes. D2D requires security schemes to be lightweight and computationally inexpensive. Computation and authentication cost have been analyzed to ensure minimal overhead for the proposed scheme. Introduction of computationally powerful computers has made most of the security schemes vulnerable to baseline attacks such as brute force or intelligent attacks. Pollard rho (Montenegro & Tetali, 2005) and Baby step Giant step (BsGs) methods are used to check

proposed scheme against quantum attacks. Security scheme is proven to be safe against any security attacks based on high power computational machines using Discrete Logarithm Problem (DLP).

3.2 System Model

To secure multi-hop D2D communication, Multi-Factor Authentication (MFA) layer and Security layer are introduced as sublayers to the main security layer. Block diagram of both layers is shown in Figure 3.1.

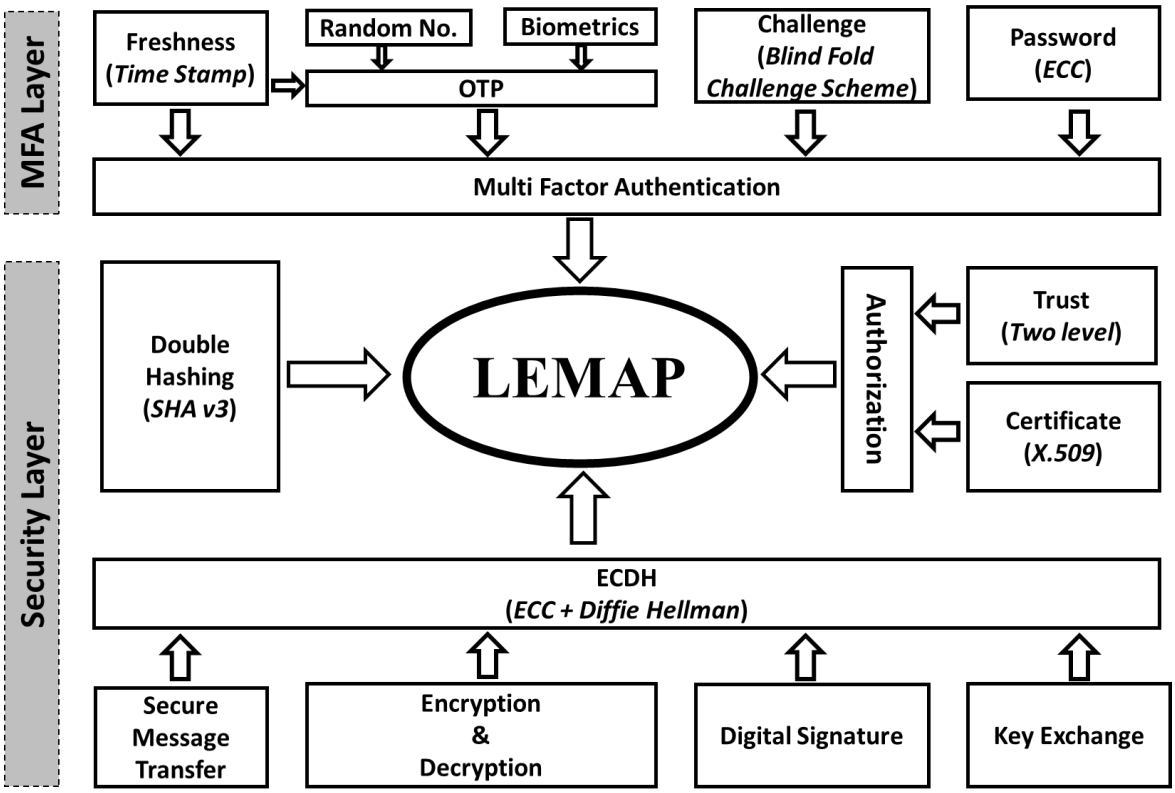

Figure 3.1: System Model of LEMAP

3.2.1 Multi Factor Authentication Layer

Multi Factor Authentication layer consists of five components timestamp, random number, biometric, challenge, and password. Pseudo IDs are assigned to devices to hide real

identity during communication so that adversary should not be able to know actual identity and location of the participating device. This helps to secure communication from identity reveal attack. Onetime password OTP is generated using random number with specific time validity. Each device must respond and solve OTP to get validity trust from eNB to communicate with other devices. If device is unable to respond OTP within the specific time period, its validity expires which secures communication from malicious attackers, who may try to use OTP after some time. Each request message is sent along with a challenge and every response is required to solve this challenge.

When a message is sent, its hashed challenge solution is also sent with the actual message. When a receiver receives the message and solves the challenge, it decrypts the already sent solved challenge and compares it with the solution. If solved challenge and sent challenge solution are same, the message is considered valid and trusted. Blind Fold Challenge Scheme is used in this algorithm to create challenge and challenge solution. Timestamp is another factor used to check the message freshness and verify it for replay attack. If the message is responded within the timestamp, the message is considered fresh and free of any replay attack. All these above-mentioned factors combine to develop LEMAP a multi-factor authentication protocol to secure D2D communication with less authentication overhead and lightweight in communication and computation cost.

3.2.2 Security Layer

Security layer comprises four major components which include secure message transfer, encryption & decryption, digital signature and key exchange. Key creation module is responsible for the creation of a secret point for communication. For this purpose, Elliptic Curve Cryptography (ECC) based algorithm is used. Secure Key transmission is very

important part of secure D2D communication where secret keys generated using ECC are securely transmitted. Elgamal algorithm is used for sharing key and sending encrypted messages. For message freshness, timestamp is used and this will secure the message from replay attack. To secure messages from MITM attack, a simple Challenge-Response authentication scheme is used. This scheme will also ensure the delivery of the message. To ensure integrity of the message, double hashing technique is used which helps to avoid collisions in hash tables. The reason for collision is when two keys are hashed to the same index in a hash table. The double hashing is done using SHA v3 algorithm. In cryptographic module, encryption and decryption of data is done which is received at any device or eNB. ECC is used for both encryption and decryption, which is lightweight and has small key size.

3.2.3 LEMAP for D2D Multi-hop Communication

The following section explains the Lightweight ECC based Multi-Factor Authentication Protocol (LEMAP) for Device to Device Multi-hop Communication security scheme and its components in detail as well as the procedure to achieve mutual authentication and security.

3.2.4 Notations Table for LEMAP (NTL)

Table 3.1 describes complete set of notations with their symbols and description, used in LEMAP. D_1, D_2 represent two devices which want to communicate with each other. CH represents Cluster Head that works as relay agent to forward request and response messages to other devices. eNB is a Base Station with strong coverage and ability to provide authentication and authorizations to devices in its area. Challenge is denoted by Ch which is used in each message to give challenge to other device. DH represents double hash which is used in each request and response message. Onetime password is denoted by OTP used by

devices to verify its validation and to get authorization. PID is used for Pseudo Identity to hide real identity of devices during communication. P represents public key that is used in LEMAP communication to encrypt messages with public key of the receiver. Pr denotes private key to digitally sign the message sent to other devices.

Table 3.1: Notations Table for LEMAP

Name	Symbol	Description
D_1	Device 1	Device 1 participating in D2D communication
D_2	Device 2	Device 2 participating in D2D communication
CH	Cluster Head	Forwarding or relay device
eNB	Evolved node Base Station	Base Station with strong coverage and authorizations
Challenges	$\mathrm{Ch}_{\mathrm{CH}}$	Challenge of Cluster Head
	$\mathrm{Ch}_{\mathrm{CH}}{}'$	Challenge Solution of Cluster Head
	$\mathrm{Ch}_{\mathrm{D}_1}$	Challenge of device1
	$\mathrm{Ch}_{\mathrm{D}_1}{}'$	Challenge Solution of device1
	$\mathrm{Ch}_{\mathrm{D}_2}$	Challenge of Device2
	$\mathrm{Ch}'_{\mathrm{D}_2}$	Challenge solution of Device2
	$\mathrm{Ch}_{\mathrm{eNB}}$	Challenge of device1
	$\mathrm{Ch}_{\mathrm{eNB}}{}'$	Challenge Solution of eNB

Table 3.1 continued

Double Hash	DH	Double Hash
One Time Password	OTP_{CH}	Onetime Password of Cluster Head
	OTP'_{CH}	Onetime Password Solution of Cluster Head
	OTP_{D1}	Onetime password of device 1
	$OTP_{D1}{}'$	Onetime password solution of device1
	OTP_{D2}	Onetime password of device 2
	$OTP_{D2}{}'$	Onetime password solution of device 2
	OTP_{eNB}	Onetime Password of eNB
Pseudo Identities	PID_1	Pseudo ID of device1
	PID_2	Pseudo ID of device2
Public Keys	P_{D1}	Public key of device1
	P_{D2}	Public key of device2
	P_{CH}	Public key of Cluster Head
Private Keys	Pr_{CH}	Private key of Cluster Head
	Pr_{D_1}	Private key of device 1
	Pr_{D_2}	Private key of device 2
	Pr_{eNB}	Private key of eNB

Table 3.1 continued

Timestamp	$T_{S_{CH}}$	Timestamp of Cluster Head
	$T_{S_{eNB}}$	Timestamp of eNB
	$T_{S_{D_1}}$	Timestamp of device1
	$T_{S_{D2}}$	Timestamp of Device2
Validation of Device	VD_1, VD_2	Validating of device1 Validating of device2

3.2.5 Factors Attacks Mitigation Table for LEMAP (FMT)

Table 3.2 also called Factors Attack Mitigation Table for LEMAP (FMT) which explains different notations used in LEMAP with purpose and attacks mitigated by each factor. **Ch** Challenge is used for achieving mutual authentication which helps mitigating MITM attack allowing only legitimate cellular devices. **PID** is used to keep privacy and helps in mitigating Location Based Identity attack. $\mathbf{T_S}$ timestamp is used to check message freshness and helps to mitigate replay attack. **DH** Double Hashing is used to achieve integrity and helps to mitigate preimage and MITM attack. $\mathbf{P_r}$ Private key is used to achieve confidentiality and helps in mitigating impersonation attack. **OTP** Onetime password ensures device legitimacy and helps in mitigation of time-stamp exploitation attack. **P** Public key is used to encrypt data to secure data from masquerading attack. $\mathbf{VD_1}$ and $\mathbf{VD_2}$ Validate the devices to provide services to legitimate device only and help mitigation of rouge device attack.

Table 3.2: Factors Attacks Mitigation Table for LEMAP (FMT)

Notation	Purpose	Attack Mitigation
Ch	Challenge is used for mutual authentication	Mitigating malware attack allowing only legitimate cellular devices
PID	Pseudo ID is kept for achieving anonymity and privacy	Mitigating Location Based Identity attack
T_S	Time stamping is used for achieving freshness	Mitigating replay attack
Ch'	Challenge Solution is sent in order to ensure receiver is a genuine cellular device	Mutual Authentication
DH	Double Hashing is done to achieve • Integrity of message • Confusion and diffusion	Mitigating Pre-image attack or weak Collision attacks, MITM attack and modification data attack
P_r	Private key is used to digitally sign data and achieve confidentiality	Mitigating impersonation and MITM attack

Table 3.2 continued

OTP	One-time password ensures authentication, Renewal of registration, Biometrics. (Legitimacy of cellular device)	Mitigation of time-stamp exploitation attack, timing-window attack using replay attack and Whitewash attack
P	Public key is used to encrypt data	Issuing public key will ensure Mitigation of Masquerading attack
VD_1	Validate the device and provide services to legitimate device. Also provide authorization and validation at two levels	Mitigation of rouge device attack.
VD_2	Validate the device and provide services to legitimate device. Also provide authorization and validation at two levels	Mitigation of rouge device attack.

3.3 System Design for LEMAP

This protocol used ECC for encryption and decryption due to its small key size and lightweight while Elgamal is used for key exchange. This protocol has been developed by using multi factors like onetime password, random number, challenge and timestamp to get secure mutual authentication. ECC with Elgamal helps to reduce authentication overhead, communication cost and low computation cost. Hence LEMAP protocol ensures asymmetric

cryptography, authentication, integrity, confidentiality and quantum attack safety. The proposed algorithm LEMAP steps are explained in detail below:

3.3.1 Authentication Request Message (D1- CH)

Equation 3.1 is an Authentication Request Message (D1- CH) for communication between D_1 and D_2. The security threat associated with this message and how message is encrypted and decrypted is explained as below:

$$\left\{ \begin{array}{c} \left\{ \begin{array}{c} \mathrm{PID}_1, \mathrm{PID}_2, \mathrm{PICH}, \mathrm{PIeNB}, \ \mathrm{Ch}_{D_1-\mathrm{CH}}, \ \mathrm{T}_{S_{D_1}}, \\ \left[\mathrm{DH}\left(\ \mathrm{PID}_1, \mathrm{PID}_2, \mathrm{PICH}, \mathrm{PIENB}, \ \ \mathrm{Ch}_{D_1-\mathrm{CH}}{}', \ \mathrm{T}_{S_{D_1}} \right) \right]_{\mathrm{P}_{rD_1}} \end{array} \right\}_{\mathrm{P}_{CH}}, \\ \left\{ \mathrm{PID}_1, \mathrm{PID}_2, \mathrm{Ch}_{D_1-eNB}, \ \mathrm{T}_{S_{D_1}}, \left[\mathrm{DH}\left(\mathrm{PID}_1, \mathrm{PID}_2, \mathrm{Ch}_{D_1-eNB}{}', \mathrm{T}_{S_{D_1}} \right) \right]_{\mathrm{P}_{rD_1}} \right\}_{\mathrm{P}_{eNB}} \end{array} \right\}_{\mathrm{P}_{CH}} \quad \text{Equation 3.1}$$

3.3.1.1 Potential Security Threats

The Authentication Request Message in Equation 3.1 is vulnerable to the attacks such as rouge devices attack where a fake device can claim to be CH. The messages can be used to do preimage attack on other devices by getting a message forwarded from legitimate CH. Man-in-the-middle attack can be performed by CH where CH can sniff what is going on. All other major form of attacks as well as how they can be executed is discussed in detail in Chapter 2.

3.3.1.2 Encryption Mechanism

To counter above mentioned attacks, the request message in Equation 3.1 is prepared by using three credentials i.e. pseudo ID for devices D_1 and D_2, CH and eNB, challenge and timestamp. This message also contains double hash value of pseudo ID, solution of challenge and timestamp which is signed by the private key P_{rD1} of D_1. The message is signed by public key of CH. The second portion of the message contains authentication request for eNB

containing credentials pseudo ID, challenge for eNB and timestamp. The message contains double hash value of the pseudo ID, solution of challenge and timestamp which is signed by the private key of device D_1 and encrypted with public key of eNB. Later, the whole Auth-Req (D_1-CH) is encrypted by the public key of CH.

As discussed in Table 3.2, usage of pseudo ID, challenge and timestamps can mitigate identity reveal attack, malware attack, timestamps exploitation attack, replay attack, denial of service attack, man in the middle attack and data modification attack. While double hashing the solution of challenge, one-time password and timestamp can ensure integrity of the data which means no data is modified during the entire communication. This also mitigates weak collision attack, man in the middle attack and pre-image attack. The message also includes challenge solution during double hashing process. The purpose of sending challenge is to ensure receiver is a genuine cellular device which means it is not bot. It also minimizes the chances of denial of service attack and rogue device attack. The double hash value is encrypted by private key of the sender (eNB) to ensure the message is sent by the legitimate device thus mitigating the rogue device attack, impersonation attack. Finally, the entire message is encrypted by public key of the receiver to maintain the secrecy of the message.

3.3.1.3 Decryption Mechanism

Once CH receives the authentication request message in Equation 3.1 from D_1, it will use its own private key to decrypt the entire message. Once the message is decrypted, it will check given challenge, pseudo ID and timestamp. CH will address the challenge and check either the timestamp is fresh or not. Later, CH will decrypt double hash value of challenge solution, pseudo ID and timestamp. CH will utilize its own public key to decrypt the double hash value sent by D_1 in encrypted version. Later, CH will match both values encrypted and

sent by D_1 and double hashed values generated by CH, if both values are same it means the data integrity is maintained.

3.3.2 Authentication Request Message (CH- eNB)

Equation 3.2 is an Authentication Request Message (CH- eNB) from CH to eNB. CH forwards the request message from D1 to eNB. The security threat associated with this message and how message is encrypted and decrypted is explained as below:

$$\left\{ \begin{matrix} \left\{ \begin{matrix} \mathrm{PID}_1 , \mathrm{PID}_2 , \mathrm{Ch}_{\mathrm{D}_1-\mathrm{eNB}} , \mathrm{T}_{\mathrm{S}_{\mathrm{D}_1}} , \\ \left[\mathrm{DH} \left(\mathrm{PID}_1 , \mathrm{PID}_2 , \mathrm{Ch}_{\mathrm{D}_1-\mathrm{eNB}}{}' , \mathrm{T}_{\mathrm{S}_{\mathrm{D}_1}} \right) \right]_{\mathrm{P}_{\mathrm{rD}_1}} \end{matrix} \right\}_{\mathrm{P}_{\mathrm{eNB}}} , \\ \begin{matrix} \mathrm{PID}_1 , \ \mathrm{PID}_2 , \mathrm{PICH}, \mathrm{Ch}_{\mathrm{CH}-\mathrm{eNB}} , \ \mathrm{T}_{\mathrm{S}_{\mathrm{CH}}} , \\ \left[\mathrm{DH} \left(\mathrm{PID}_1 , \ \mathrm{PID}_2 , \mathrm{PICH}, \ \mathrm{Ch}_{\mathrm{CH}-\mathrm{eNB}}{}' , \mathrm{T}_{\mathrm{S}_{\mathrm{CH}}} \right) \right]_{\mathrm{Pr}_{\mathrm{CH}}} \end{matrix} \end{matrix} \right\}_{\mathrm{P}_{\mathrm{eNB}}} \quad \text{Equation 3.2}$$

3.3.2.1 Potential Security Threats

The Authentication Request Message (CH-eNB) in Equation 3.2 is vulnerable to attacks such as rouge devices attack where a fake device can claim to be CH. The attack such as Identity reveal attack is also possible in case of CH being compromised. Man-in-the-middle attack can be performed by CH where CH can sniff what is going on. All other major form of attacks as well as how they can be executed is discussed in detail in Chapter 2.

3.3.2.2 Encryption Mechanism

To counter already discussed attacks, the request message in Equation 3.2 is prepared by using three credentials i.e. challenge, pseudo ID of devices D_1, D_2 and timestamp. This message also contains double hash value of pseudo IDs, challenge solution and timestamp which is signed by the private key of device D_1 and encrypted with public key of eNB. The message second portion contains request message for eNB containing credentials pseudo IDs, challenge and timestamp. This message also contains double hash value of the pseudo

IDs, challenge solution and timestamp which is signed by the private key of CH. The whole message Auth-Req (CH-eNB) is encrypted with public key of eNB.

As discussed in Table 3.2, usage of pseudo ID, challenge, and timestamp can mitigate identity reveal attack, timestamp exploitation attack, replay attack, pre-image attack, man in the middle attack, rogue device attack, weak collision attack and data modification attack. While double hashing pseudo IDs, challenge solution and timestamp mitigate weak collision attack, replay attack, man in the middle attack and pre-image attack. The message also contains challenge solution during double hashing process. The purpose of sending challenge is to ensure the receiver is a genuine cellular device and it is not bot. This mitigates denial of service attack and rogue device attack. The double hashed value is encrypted by private key of the sender (CH) to ensure the message is sent by the legitimate device thus mitigating the impersonation attack. Finally, the entire message is encrypted by public key of the receiver to maintain secrecy of the message and mitigating masquerading attack.

3.3.2.3 Decryption Mechanism

Once CH receives the authentication request message in Equation 3.2 from D_1, it will use its own private key to decrypt the entire message. When the message is decrypted, it will check challenge given, pseudo ID and timestamp. CH will address the challenge and check either the timestamp is fresh or not. Later, CH will find double hash value of pseudo ID, challenge solution and timestamp. eNB will utilize its own public key to decrypt the double hash value sent by CH in encrypted version. Later, eNB will match both double hash value i.e generated by eNB and the value sent by CH, if both values match, it means the data integrity is maintained.

3.3.3 Authentication Response Message (eNB- CH)

Equation 3.3 is an Authentication Response Message (eNB- CH) from eNB to CH. CH gets OTP request message from eNB. The security threat associated with this message and how it is encrypted and decrypted is explained as below.

$$\left\{ \begin{matrix} \mathrm{Ch}_{\mathrm{eNB-CH}}\,,\ \mathrm{T}_{\mathrm{S}_{\mathrm{eNB}}}, \mathrm{OTP}_{\mathrm{eNB-CH}} \\ , \left[\mathrm{DH}\left(\mathrm{Ch}_{\mathrm{eNB-CH}}{}' \ , \mathrm{T}_{\mathrm{S}_{\mathrm{eNB}}}\ , \ \mathrm{OTP}_{\mathrm{eNB-CH}}{}' \right) \right]_{\mathrm{Pr}_{\mathrm{eNB}}} \end{matrix} \right\}_{\mathrm{P}_{\mathrm{CH}}} \qquad \text{Equation 3.3}$$

3.3.3.1 Potential Security Threats

Authentication Response Message (eNB- CH) Equation 3.3 is vulnerable to attacks such as rouge devices attack where a fake device can claim to be CH. Man-in-the-middle attack can be performed by CH where CH can sniff what is going on. All other major form of attacks as well as how they can be executed is discussed in detail in Chapter 2.

3.3.3.2 Encryption Mechanism

To counter above mentioned attacks, the response message Equation 3.3 is prepared by utilizing three credentials i.e. challenge, timestamp and one-time password. This message also contains double hash value of the challenge solution, one-time password and timestamp which is signed by the private key of eNB. Later the Auth-Res (eNB-CH) is encrypted by the public key of CH. As discussed in Table 3.2, usage of challenge, one-time password and timestamps can mitigate timestamps exploitation attack, replay attack, rogue device attack and man in the middle attack. While double hashing the challenge solution, one-time password and timestamp can ensure integrity of data which means there is no data modification during the entire communication. The double hash value is encrypted by private key of the sender (eNB) to ensure the message is sent by the legitimate user thus mitigating

the impersonation attack. The entire message is encrypted by public key of the receiver to maintain secrecy of the message.

3.3.3.3 Decryption Mechanism

Once CH receives the authentication response message from eNB, CH will use its own private key to decrypt the entire message. When the message is decrypted, it will check challenge given by eNB, OTP and timestamp. CH will solve the challenge, use OTP and check the timestamp freshness. Later, CH will utilize public key of eNB to decrypt the double hash values of challenge solution, one-time password and timestamp sent by eNB. If double hash values generated by CH and the other sent by eNB match, it means data integrity is maintained.

3.3.4 Authentication Request Message (CH- eNB)

Equation 3.4 is an Authentication Request Message (CH- eNB) from CH to eNB. CH solves OTP and sends request message to eNB for authentication. The security threat associated with this message and how message is encrypted and decrypted is explained as below.

$$\left\{ \begin{matrix} \mathrm{Ch}_{\mathrm{CH-eNB}},\ \mathrm{T}_{\mathrm{S}_{\mathrm{CH}}}\ ,\ \mathrm{OTP}'_{\mathrm{CH-eNB}}\ , \\ \left[\mathrm{DH}(\ \mathrm{Ch}'_{\mathrm{CH-eNB}}\ , \mathrm{T}_{\mathrm{S}_{\mathrm{CH}}}\ , \mathrm{OTP}'_{\mathrm{CH-eNB}}\ )\right]_{\mathrm{Pr}_{\mathrm{CH}}} \end{matrix} \right\}_{\mathrm{P}_{\mathrm{eNB}}} \qquad \text{Equation 3.4}$$

3.3.4.1 Potential Security Threats

The Authentication Request Message (CH-eNB) Equation 3.4 is vulnerable to attacks such as rouge devices attack where a fake device can claim to be CH. Man-in-the-middle attack can be performed by CH where CH can sniff what is going on. All other major form of attacks as well as how they can be executed is discussed in detail in Chapter 2.

3.3.4.2 Encryption Mechanism

To counter above mentioned attacks, the request message Equation 3.4 is prepared by utilizing three credentials i.e. challenge, timestamp and onetime password solution. This message also contains double hash value of the challenge solution, timestamp and onetime password solution which is signed by the private key of CH. The whole Auth-Req (CH-eNB) message is encrypted with public key of eNB. Onetime password solution is sent by CH to eNB to verify that if CH is a valid and legitimate device. While double hashing f onetime password solution, challenge solution and timestamp signed by private key of CH can ensure integrity of data which means data is not modified during transmission. This can also mitigate rouge device attack, replay attack, man in the middle attack, impersonation attack and pre-image attack. The double hash value is encrypted by private key of the sender (CH) to ensure the message is sent by the legitimate device hence mitigating impersonation attack. The entire message is encrypted by public key of the receiver eNB to maintain the secrecy of the message which mitigates masquerading attack.

3.3.4.3 Decryption Mechanism

Once eNB receives the authentication request message from CH, it will use its own private key to decrypt the entire message. When the message is decrypted, it will check onetime password solution, challenge given and timestamp freshness. eNB will utilize public key of CH to decrypt encrypted message sent by CH. Then eNB will match both double hash values i.e. generated by eNB and other sent by CH, if both values match it means data integrity is maintained.

3.3.5 Authentication Response Message

Equation 3.5 is an Authentication Response Message from eNB to CH. eNB sends OTP to D_1 and D_2. The security threat associated with this message and how message is encrypted and decrypted is explained as below.

$$\left\{ \begin{array}{c} \mathrm{PICH}|\mathrm{P_{CH}}\,,\ \mathrm{PID_1}|\mathrm{P_{D_1}}\,,\mathrm{PID_2}\left|\mathrm{P_{D_2}}\right|, \\ \mathrm{Ch_{eNB-CH}}\,,\ \mathrm{T_{S_{eNB}}}\,,\ \left[\mathrm{DH}\left(\mathrm{Ch_{eNB-CH}}'\,,\ \mathrm{T_{S_{eNB}}}\right)\right]_{\mathrm{P_{r_{eNB}}}}, \\ \left\{ \begin{array}{c} \mathrm{Ch_{eNB-D_1}}\,,\mathrm{T_{S_{eNB}}}\,,\mathrm{OTP_{eNB-D_1}} \\ ,\left[\mathrm{DH}(\ \mathrm{Ch_{eNB-D_1}}'\,,\mathrm{T_{S_{eNB}}}\,,\mathrm{OTP_{eNB-D_1}}'\ )\right]_{\mathrm{P_{r_{eNB}}}} \end{array} \right\}_{\mathrm{P_{D_1}}} \\ ,\left\{ \begin{array}{c} \mathrm{Ch_{eNB-D_2}}\,,\mathrm{T_{S_{eNB}}}\,,\mathrm{OTP_{eNB-D_2}} \\ ,\left[\mathrm{DH}\left(\mathrm{Ch_{eNB-D_2}}'\,,\mathrm{T_{S_{eNB}}}\,,\mathrm{OTP_{eNB-D_2}}'\right)\right]_{\mathrm{P_{r_{eNB}}}} \end{array} \right\}_{\mathrm{P_{D_2}}} \end{array} \right\}_{\mathrm{P_{CH}}} \quad \text{Equation 3.5}$$

3.3.5.1 Potential Security Threats

The Authentication Response Message (eNB-CH) Equation 3.5 is vulnerable to attacks such as rouge devices attack where a fake device can claim to be CH. Man-in-the-middle attack can be performed by CH where CH can sniff what is going on. All other major form of attacks as well as how they can be executed is discussed in detail in Chapter 2.

3.3.5.2 Encryption Mechanism

To counter above mentioned attack, the Authentication response message, Equation 3.5 is formed by utilizing three credentials i.e. challenge, timestamps and onetime password (OTP). This message also contains double hashed value of challenge solution and timestamp which is signed by the private key of eNB. The message also contains messages for devices D_1 and D_2 which consists of three credentials challenge, timestamp and onetime password. This messages also contain double hash value of the challenge solution, timestamp and

onetime password which is signed by the private key of eNB and encrypted with public key of devices D_1 and D_2. The whole message is encrypted by public key of CH.

As discussed in Table 3.2, challenge, timestamp and onetime password can mitigate rogue device attack, timestamp exploitation attack, replay attack, pre-image attack, weak collision attack, man in the middle attack and data modification attack. While double hashing the challenge solution, timestamp and onetime password can ensure integrity of the data which means no data is modified during entire communication and it also mitigates rogue device attack, man in the middle attack and pre-image attack. The messages for devices D_1 and D_2 contain onetime password (OTP) for both devices to solve to verify legitimacy. The double hash value is encrypted by private key of the sender (eNB) to ensure the message is sent by the legitimate device so mitigating the impersonation attack. The entire message is encrypted by public key of the receiver to maintain secrecy of the message and hence mitigating masquerading attack.

3.3.5.3 Decryption Mechanism

Once CH receives the authentication response message from eNB, it will use its own private key to decrypt the entire message. When the message is decrypted, it will check challenge given and timestamp to verify message freshness. Then CH will match both double hashed values i.e generated by CH and other sent by eNB, if both values match it means data integrity is maintained. CH cannot open messages for D_1 and D_2 sent by eNB. CH will forward these messages to devices D_1 and D_2.

3.3.6 Authentication Response Message

Equation 3.6 is an Authentication Response Message where CH forwards OTP to D_1. The security threat associated with this message and how message is encrypted and decrypted is explained as below.

$$\left\{ \begin{array}{c} \left\{ \begin{array}{c} \mathrm{Ch}_{\mathrm{eNB-D_1}}, \mathrm{T}_{\mathrm{S_{eNB}}}, \mathrm{OTP}_{\mathrm{eNB-D_1}} \\ , \left[\mathrm{DH}\left(\mathrm{Ch'}_{\mathrm{eNB-D_1}}, \mathrm{T}_{\mathrm{S_{eNB}}}, \mathrm{OTP}_{\mathrm{eNB-D_1}}{}' \right)\right]_{\mathrm{P_{reNB}}} \end{array} \right\}_{\mathrm{P_{D_1}}} \\ , \mathrm{Ch}_{\mathrm{CH-D_1}}, \mathrm{T}_{\mathrm{S_{CH}}} \\ , \left[\mathrm{DH}\left(\mathrm{Ch'}_{\mathrm{CH-D_1}}, \mathrm{T}_{\mathrm{S_{CH}}} \right)\right]_{\mathrm{Pr_{CH}}}, \mathrm{PICH}|\mathrm{P_{CH}}, \mathrm{PID_1}|\mathrm{P_{D_1}}, \mathrm{PID_2}|\mathrm{P_{D2}} \end{array} \right\}_{\mathrm{P_{D_1}}} \quad \text{Equation 3.6}$$

3.3.6.1 Potential Security Threats

The Authentication Response Message, Equation 3.6 is vulnerable to attacks such as rouge devices attack where a fake device can claim to be CH. Man-in-the-middle attack can be performed by CH where CH can sniff what is going on. All other major form of attacks as well as how they can be executed is discussed in detail in Chapter 2.

3.3.6.2 Encryption Mechanism

To counter above mentioned attacks, the response message, Equation 3.6 is prepared by utilizing two credentials i.e. challenge and timestamp. This message also contains double hashed value of challenge solution and timestamp which is signed by the private key of CH. The message also contains public keys of CH, D_1 and D_2. The first portion of the message is sent from eNB for D1 which contains challenge, timestamp and onetime password (OTP). This message is encrypted with public key of D_1.

As discussed in Table 3.2, challenge and timestamp can mitigate timestamp exploitation attack, replay attack, pre-image attack, weak collision attacks, man in the middle attack and modification of data attack. While double hashing the challenge solution and

timestamp can also mitigate weak collision attack, man in the middle attack and pre-image attack. The purpose of sending challenge is to ensure receiver is a genuine cellular device. The double hash value is encrypted by private key of the sender (CH) to ensure the message is sent by the legitimate user hence mitigating the impersonation attack. Finally, the entire message is encrypted by public key P_{D1} of the receiver to maintain the secrecy of the message and, hence mitigating masquerading attack.

3.3.6.3 Decryption Mechanism

Once the D_1 receives the authentication response message from CH, it will use its own private key to decrypt the entire message. When the message is decrypted, it will check challenge given, OTP and timestamp. D_1 will address the challenge, use OTP and check either the timestamp is fresh or not. Later, D_1 will check double hash values of challenge solution, timestamp and OTP sent by eNB and match with values sent by CH. If both values match, it means the data integrity is maintained.

3.3.7 Authentication Response Message

Equation 3.7 is an Authentication Response Message, where CH forwards OTP to D_2. The security threat associated with this message and how message is encrypted and decrypted is explained as below.

$$\left\{ \begin{array}{c} \left\{ \begin{array}{c} \mathrm{Ch}_{\mathrm{eNB-D_2}}, \mathrm{T}_{\mathrm{S_{eNB}}}, \mathrm{OTP}_{\mathrm{eNB-D_2}} \\ , \left[\mathrm{DH}\left(\mathrm{Ch}_{\mathrm{eNB-D_2}}, \mathrm{T}_{\mathrm{S_{eNB}}}, \mathrm{OTP}_{\mathrm{eNB-D_2}}\right)\right]_{\mathrm{Pr_{eNB}}} \end{array} \right\}_{\mathrm{P_{D_2}}}, \\ \mathrm{Ch}_{\mathrm{CH-D_2}}, \mathrm{T}_{\mathrm{S_{CH}}}, \left[\mathrm{DH}\left(\mathrm{Ch}_{\mathrm{CH-D_2}}', \mathrm{T}_{\mathrm{S_{CH}}}\right)\right]_{\mathrm{Pr_{CH}}} \\ , \mathrm{PICH}|\mathrm{P_{CH}}, \mathrm{PID_1}|\mathrm{P_{D_1}}, \mathrm{PID_2}|\mathrm{P_{D2}} \end{array} \right\}_{\mathrm{P_{D_2}}} \quad \text{Equation 3.7}$$

3.3.7.1 Potential Security Threats

The Authentication Response Message Equation 3.7 is vulnerable to attacks such as rouge devices attack where a fake device can claim to be CH. Man-in-the-middle attack can be performed by CH where CH can sniff what is going on. All other major form of attacks as well as how they can be executed is discussed in detail in Chapter 2.

3.3.7.2 Encryption Mechanism

To counter above mentioned attacks, the response message Equation 3.7 is prepared by utilizing two credentials i.e. challenge and timestamp. This message also contains double hash value of challenge solution and timestamp which is signed by the private key of CH. The message also contains public keys of CH, D_1 and D_2.

The first portion of the message is sent from eNB for D_2 which contains challenge, timestamp and onetime password (OTP). This message is encrypted with public key of D_2. As discussed in Table 3.2, challenge and timestamp can mitigate timestamp exploitation attack, replay attack, pre-image attack, weak collision attacks, man in the middle attack and modification of data attack. While double hashing the challenge solution and timestamp can also mitigate weak collision attack, man in the middle attack and pre-image attack. The purpose of sending challenge is to ensure receiver is a genuine cellular device. The double hash value is encrypted by private key of the sender (CH) to ensure the message is sent by the legitimate user hence mitigating the impersonation attack. Finally, the entire message is encrypted by public key P_{D2} of the receiver to maintain the secrecy of the message and thus mitigating masquerading attack.

3.3.7.3 Decryption Mechanism

Once the D_2 receives the authentication response message from CH, it will use its own private key to decrypt the entire message. When the message is decrypted, it will check challenge given, OTP and timestamp. D_2 will address the challenge, use OTP and check either the timestamp is fresh or not. Later, D_2 will check double hash values of challenge solution, timestamp and OTP sent by eNB and match with values sent by CH. If both values match, it means data integrity is maintained.

3.3.8 The Authentication Request Message

Equation 3.8 is an Authentication Request Message where D1 requests CH for authentication. The security threat associated with this message and how message is encrypted and decrypted is explained as below.

$$\left\{ \begin{matrix} \mathrm{Ch}_{\mathrm{D}_1-\mathrm{CH}}, \mathrm{T}_{\mathrm{s}_{\mathrm{D}_1}}, \left[\mathrm{DH}\left(\mathrm{Ch}_{\mathrm{D}_1-\mathrm{CH}}{}', \mathrm{T}_{\mathrm{s}_{\mathrm{D}_1}}\right)\right]_{\mathrm{Pr}_{\mathrm{D}1}}, \\ \mathrm{Ch}_{\mathrm{D}_1-\mathrm{eNB}}, \mathrm{T}_{\mathrm{s}_{\mathrm{D}_1}}, \left\{\left[\mathrm{DH}\left(\mathrm{Ch}_{\mathrm{D}_1-\mathrm{eNB}}{}', \mathrm{T}_{\mathrm{s}_{\mathrm{D}_1}}\right)\right]_{\mathrm{Pr}_{\mathrm{D}_1}}\right\}_{\mathrm{P}_{\mathrm{eNB}}} \end{matrix} \right\}_{\mathrm{P}_{\mathrm{eNB}}} \quad \text{Equation 3.8}$$

3.3.8.1 Potential Security Threats

The Authentication Response Message, Equation 3.8 is vulnerable to attacks such as rouge devices attack where a fake device can claim to be CH. Man-in-the-middle attack can be performed by CH where CH can sniff what is going on. All other major form of attacks as well as how they can be executed is discussed in detail in Chapter 2.

3.3.8.2 Encryption Mechanism

To counter above mentioned attacks, the request message Equation 3.8 is prepared by utilizing two credentials i.e. challenge and timestamp. This message also contains double hash value of challenge solution and timestamp which is signed by the private key of D_1.

The message also contains request message for eNB containing credentials challenge for eNB and timestamp. This message also contains double hash value of challenge solution and timestamp which is signed by the private key of device1 D_1. Later the Auth-Req (D_1-CH) is signed by the public key of eNB.

As discussed in Table 3.2, usage of challenge and timestamp can mitigate timestamps exploitation attack, replay attack, man in the middle attack and modification of data attack. While double hashing the challenge solution and timestamp signed by private key of the sender D1 can ensure integrity of the data which means no data is modified during the entire communication. This also mitigates weak collision attack, man in the middle attack and pre-image attack. The message also used solution of challenge during double hashing process. The purpose of sending challenge is to ensure receiver is a genuine cellular device which means it's not bot thus also minimizing the chances of denial of service attack. The double hash value is encrypted by private key of the sender D_1 to ensure the message is sent by the legitimate user thus mitigating the impersonation attack and lastly, the entire message is encrypted by public key of the receiver to maintain secrecy of the message.

3.3.8.3 Decryption Mechanism

Once CH receives the authentication request message from D_1, it will use its own private key to decrypt the entire message. Once the message is decrypted, it will check challenge given and timestamp. CH will address the challenge and check either the timestamp is fresh or not. Later, CH will check double hash values of challenge solution and timestamp. CH will utilize its own public key to decrypt the double hash value sent by D_1 in encrypted version. Later, CH matches both values, if double hashed values i.e. generated by CH and other sent by D_1 match, it means the data integrity is maintained.

3.3.9 Auth-Request Message

Equation 3.9 is an Authentication Request Message where D2 requests CH for authentication. The security threat associated with this message and how message is encrypted and decrypted is explained as below.

$$\left\{ \begin{matrix} \mathrm{Ch}_{\mathrm{D}_{2-\mathrm{CH}}}, \mathrm{T}_{\mathrm{s}_{\mathrm{D}_2}}, \left[\mathrm{DH}\left(\mathrm{Ch}_{\mathrm{D}_{2-\mathrm{CH}}}{}', \mathrm{T}_{\mathrm{s}_{\mathrm{D}_2}}\right)\right]_{\mathrm{Pr}_{\mathrm{D}_2}}, \\ \mathrm{Ch}_{\mathrm{D}_2-\mathrm{eNB}}, \mathrm{T}_{\mathrm{s}_{\mathrm{D}_2}}, \left\{\left[\mathrm{DH}\left(\mathrm{Ch}_{\mathrm{D}_1-\mathrm{eNB}}{}', \mathrm{T}_{\mathrm{s}_{\mathrm{D}_1}}\right)\right]_{\mathrm{Pr}_{\mathrm{D}_1}}\right\}_{\mathrm{P}_{\mathrm{eNB}}} \end{matrix} \right\}_{\mathrm{P}_{\mathrm{eNB}}} \quad \text{Equation 3.9}$$

3.3.9.1 Potential Security Threats

The Authentication Response Message, Equation 3.9 is vulnerable to is vulnerable to attacks such as rouge devices attack where a fake device can claim to be CH. Man-in-the-middle attack can be performed by CH where CH can sniff what is going on. All other major form of attacks as well as how they can be executed is discussed in detail in Chapter 2.

3.3.9.2 Encryption Mechanism

To counter above mentioned attacks, the request message is prepared by utilizing two credentials i.e. challenge and timestamp. This message also contains double hash value of challenge solution and timestamp which is signed by the private key of D_2. The message also contains request message for eNB with credentials challenge and timestamp. The message also contains double hash value of challenge solution and timestamp which is signed by the private key of device D_2. Later the Auth-Req is signed by the public key of eNB.

As discussed in Table 3.2, usage of challenge and timestamp can mitigate timestamps exploitation attack, replay attack and man in the middle attack. While double hashing the challenge solution and timestamp signed by private key of the sender D_2 can ensure integrity

of the data which means no data is modified during the entire communication, hence mitigating data modification attack. The message also used solution of challenge during double hashing process. The purpose of sending challenge is to ensure receiver is a genuine cellular device which also minimizes the chances of denial of service attack. The double hash value is encrypted by the private key of the sender (D_2) to ensure the message is sent by the legitimate user so mitigating the impersonation attack. The entire message is encrypted by public key of the receiver to maintain the secrecy of the message.

3.3.9.3 Decryption Mechanism

Once CH receives the authentication request message from D_2, it will use its own private key to decrypt the entire message. When the message is decrypted, it will check challenge given and timestamp. CH will address the challenge and check either the timestamp is fresh or not. Later, CH will check double hash values of challenge solution and timestamp. CH will use its own public key to decrypt the double hash value sent by D_2 in encrypted version. Later, CH will match both values, if double hash values i.e generated by CH and other sent by D_2 match, it means the data integrity is maintained.

3.3.10 Authentication Request Message

Equation 3.10 is an Authentication Request Message where CH forwards requests from D1 and D2 to eNB to validate both devices. The security threat associated with this message and how message is encrypted and decrypted is explained as below.

$$\left\{\begin{array}{l} Ch_{D_1-eNB}, T_{s_{D_1}}, \left\{\left[DH\left(Ch_{D_1-eNB}', T_{s_{D_1}}\right)\right]_{Pr_{D_1}}\right\}_{P_{eNB}}, \\ Ch_{CH-eNB}, T_{S_{CH}}, \left[DH\left(Ch_{CH-eNB}', T_{S_{CH}}\right), VD_1\right]_{Pr_{CH}} \\ Ch_{D_2-eNB}, T_{s_{D_2}}, \left\{\left[DH\left(Ch_{D_1-eNB}', T_{s_{D_1}}\right)\right]_{Pr_{D_1}}\right\}_{P_{eNB}}, \\ Ch_{CH-eNB}, T_{S_{CH}}, \left[DH\left(Ch_{CH-eNB}', T_{S_{CH}}\right), VD_2\right]_{Pr_{CH}} \end{array}\right\}_{P_{eNB}}$$

Equation 3.10

3.3.10.1 Potential Security Threats

The Authentication Request Message, Equation 3.10 is vulnerable to attacks such as rouge devices attack where a fake device can claim to be CH. Man-in-the-middle attack can be performed by CH where CH can sniff what is going on. All other major form of attacks as well as how they can be executed is discussed in detail in Chapter 2.

3.3.10.2 Encryption Mechanism

To counter above mentioned attacks, the request message is prepared by using two credentials i.e. challenge and timestamp. This message also contains double hash value of challenge solution and timestamp which is signed by the private key of D_1. The message also contains request message for eNB with credentials challenge for eNB from CH and timestamp. The message also contains double hash value of challenge solution, timestamp and D_1 validation request from CH to eNB (VD_1) which is signed by the private key of CH. Similarly, the message second portion contains double hash value of challenge solution, timestamp and D_2 validation request from CH to eNB (VD_2) which is signed by the private key of CH. Later the Auth-Req message is signed by the public key of eNB.

As discussed in Table 3.2, usage of challenge and timestamp can mitigate malware attack, timestamps exploitation attack, replay attack, man in the middle attack and modification of data attack. While double hashing the solution of challenge and timestamp

can mitigate weak collision attack, man in the middle attack and pre-image attack and ensure integrity of the data which means no data is modified during the entire communication. The message also used solution of challenge during double hashing process, the purpose of sending challenge is to ensure receiver is a genuine cellular device which means it's not bot hence it also mitigates the denial of service attack. The double hash value is encrypted by private key of the sender CH to ensure the message is sent by the legitimate device thus mitigating the impersonation attack. Device D_1 validation request VD_1 and device D_2 validation request VD_2 is sent to eNB to validate both devices D_1 and D_2 to communicate. This also mitigates rogue device attack. The entire message is encrypted by public key of the receiver to maintain the secrecy of the message.

3.3.10.3 Decryption Mechanism

Once eNB receives the authentication request message from CH, it will use its own private key to decrypt the entire message. When the message is decrypted, it will check challenge given and timestamp. eNB will address the challenge and check either the timestamp is fresh or not. Later, eNB will check double hash values of challenge solution from CH and timestamp. Then CH will match both values, if double hash values i.e generated by eNB and other sent by CH match, it means data integrity is maintained.

3.3.11 Authentication Response Message

Equation 3.11 is an Authentication Response Message where eNB validates both devices D_1 and D_2 and sends CH to forwards validation response to both devices. The security threat associated with this message and how message is encrypted and decrypted is explained as below.

$$\left\{\begin{array}{c} Ch_{eNB-CH}\,,\, T_{S_{eNB}}\,,\left[DH\left(Ch_{eNB-CH}{}',\, T_{S_{eNB}}\right)\right]_{Pr_{eNB}}, \\ \left\{Ch_{eNB-D_1}\,,\, T_{S_{eNB-D_1}}\,, VD_1{}'\left[DH\left(Ch_{eNB-D_1}{}',\, T_{S_{eNB-D_1}}, VD_1{}'\right)\right]_{Pr_{eNB}}\right\}_{P_{D_1}}, \\ \left\{Ch_{eNB-D_2}\,,\, T_{S_{eNB-D_2}}\,, VD_2{}'\left[DH\left(Ch_{eNB-D_2}{}',\, T_{S_{eNB-D_2}}, VD_2{}'\right)\right]_{Pr_{eNB}}\right\}_{P_{D_2}} \end{array}\right\}_{P_{CH}}$$

Equation 3.11

3.3.11.1 Potential Security Threats

The Authentication Response Message, Equation 3.11 is vulnerable to attacks such as rouge devices attack where a fake device can claim to be CH. Man-in-the-middle attack can be performed by CH where CH can sniff what is going on. All other major form of attacks as well as how they can be executed is discussed in detail in Chapter 2.

3.3.11.2 Encryption Mechanism

To counter above mentioned attacks, the request message is prepared by using two credentials i.e. challenge and timestamp. This message also contains double hash value of challenge solution and timestamp which is signed by the private key of eNB. The message second portion contains response message for D_1 and D_2 with credentials challenge, timestamp and devices D_1&D_2 validation solution VD_1', VD_2' from eNB. The message also contains double hash value of challenge solution, timestamp and validation solution VD_1', VD_2' which is signed by the private key of eNB and encrypted by public key of D_1 and D_2. Later, the Auth-Response message is encrypted by the public key of CH.

As discussed in Table 3.2, usage of challenge and timestamp can mitigate timestamp exploitation attack, replay attack, pre-image attack, man in the middle attack and modification of data attack. While double hashing the solution of challenge and timestamp can mitigate weak collision attack, man in the middle attack and pre-image attack. It also

ensures integrity of data which means no modification during the entire communication. The message also uses solution of challenge during double hashing process. The purpose of sending challenge is to ensure receiver is a genuine cellular device which means it's not bot so it mitigates denial of service attack. The double hash value is encrypted by private key of the sender (eNB) to ensure the message is sent by the legitimate device thus mitigating the impersonation attack. Device validation solution $VD_1'$and VD_2' are sent to mitigate rogue device attack. The entire message is encrypted by the public key of the receiver to maintain secrecy of the message.

3.3.11.3 Decryption Mechanism

Once CH receives the authentication response message from eNB, it will use its own private key to decrypt the entire message. Once the message is decrypted, it will check challenge given and timestamp. CH will address the challenge and check either the timestamp is fresh or not. Later, CH will check double hash values of challenge solution from eNB and timestamp. Then CH matches both values, if double hash values i.e. generated by CH and other sent by eNB match, it means data integrity is maintained.

3.3.12 Authentication Response Message

Equation 3.12 is an Authentication Response Message where eNB validates device D_1 and CH forwards it to devices D_1.

$$\left\{ \begin{matrix} \mathrm{Ch}_{\mathrm{CH}-\mathrm{D}_1}, \mathrm{T}_{\mathrm{S}_{\mathrm{CH}}}, \left[\mathrm{DH}\left(\mathrm{Ch}_{\mathrm{CH}-\mathrm{D}_1}', \mathrm{T}_{\mathrm{S}_{\mathrm{CH}}}\right)\right]_{\mathrm{Pr}_{\mathrm{CH}}}, \\ \mathrm{Ch}_{\mathrm{eNB}-\mathrm{D}_1}, \mathrm{T}_{\mathrm{S}_{\mathrm{eNB}-\mathrm{D}_1}} \\ , \mathrm{VD}_1' \left[\mathrm{DH}\left(\mathrm{Ch}_{\mathrm{eNB}-\mathrm{D}_1}', \mathrm{T}_{\mathrm{S}_{\mathrm{eNB}-\mathrm{D}_1}}, \mathrm{VD}_1'\right)\right]_{\mathrm{Pr}_{\mathrm{eNB}}} \end{matrix} \right\}_{\mathrm{P}_{\mathrm{D}_1}} \quad \text{Equation 3.12}$$

3.3.12.1 Potential Security Threats

The Authentication Response Message, Equation 3.12 is vulnerable to attacks such as rouge devices attack where a fake device can claim to be CH. Man-in-the-middle attack can be performed by CH where CH can sniff what is going on. All other major form of attacks as well as how they can be executed is discussed in detail in Chapter 2.

3.3.12.2 Encryption Mechanism

To counter above mentioned attacks, the request message Equation 3.12 is prepared by utilizing two credentials i.e. challenge and timestamp. This message also contains double hash value of challenge solution and timestamp which is signed by the private key of CH. The message second portion contains validation response message VD_1' for D_1 signed by private key of eNB. This message contains challenge, timestamp and solution of device validation VD_1'. Later the Auth-Response (CH-D_1) is signed and encrypted by public key of device D_1.

As discussed in Table 3.2, usage of challenge and timestamp can mitigate timestamp exploitation attack, replay attack, pre-image attack, man in the middle attack and modification of data attack. While double hashing the solution of challenge and timestamp can mitigate weak collision attack, man in the middle attack and pre-image attack. It also ensures integrity of data which means no data is modified during the entire communication. The message also uses solution of challenge during double hashing process. The purpose of sending challenge is to ensure receiver is a genuine cellular device which means it's not bot so it mitigates denial of service attack. The double hashed value is encrypted by private key of the sender (eNB) to ensure the message is sent by the legitimate device thus mitigating the impersonation attack. Device validation solution VD_1' is sent to mitigate rogue device

attack. The entire message is encrypted by the public key of the receiver to maintain secrecy of the message.

3.3.12.3 Decryption Mechanism

Once the D_1 receives the authentication request message from CH, it will use its own private key to decrypt the entire message. Once the message is decrypted, it will check challenge given and timestamp. D_1 will address the challenge and check either the timestamp is fresh or not. Later, D_1 will check double hash values of challenge solution from CH and timestamp. Then D_1 will match both values, if double hash values i.e generated by D_1 and other sent by CH match, it means the data integrity is maintained.

3.3.13 Authentication Response Message (CH-D_2)

Equation 3.13 is an Authentication Response Message (CH-D2) where eNB validates device D2 and CH forwards it to device D2.

$$\left\{ \begin{matrix} \mathrm{Ch}_{\mathrm{CH}-\mathrm{D}_2}, \mathrm{T}_{\mathrm{S}_{\mathrm{CH}}}, \left[\mathrm{DH}\left(\mathrm{Ch}_{\mathrm{CH}-\mathrm{D}_2}{}', \mathrm{T}_{\mathrm{S}_{\mathrm{CH}}}\right)\right]_{\mathrm{Pr}_{\mathrm{CH}}}, \\ \mathrm{Ch}_{\mathrm{eNB}-\mathrm{D}_2}, \mathrm{T}_{\mathrm{S}_{\mathrm{eNB}-\mathrm{D}_2}} \\ \mathrm{VD}_2{}' \left[\mathrm{DH}\left(\mathrm{Ch}_{\mathrm{eNB}-\mathrm{D}_2}{}', \mathrm{T}_{\mathrm{S}_{\mathrm{eNB}-\mathrm{D}_2}}, \mathrm{VD}_2{}'\right)\right]_{\mathrm{Pr}_{\mathrm{eNB}}} \end{matrix} \right\}_{\mathrm{P}_{\mathrm{D}_2}} \quad \text{Equation 3.13}$$

3.3.13.1 Potential Security Threats

The Authentication Response Message (CH-D_2) message Equation 3.13 is vulnerable to attacks such as rouge devices attack where a fake device can claim to be CH. Man-in-the-middle attack can be performed by CH where CH can sniff what is going on. All other major form of attacks as well as how they can be executed is discussed in detail in Chapter 2.

3.3.13.2 Encryption Mechanism

To counter above mentioned attacks, the request message Equation 3.13 is prepared by using two credentials i.e. challenge and timestamp. This message also contains double hash value of challenge solution and timestamp which is signed by the private key of CH. The message second portion contains validation response message VD_2' for D_2 signed by private key of eNB. This message contains challenge, timestamp and solution of device validation VD_2'. Later the Auth-Response (CH-D_2) is signed and encrypted by public key of device D_2.

As discussed in Table 3.2, usage of challenge and timestamp can mitigate timestamp exploitation attack, replay attack, pre-image attack, man in the middle attack and modification of data attack. While double hashing the solution of challenge and timestamp can mitigate weak collision attack, man in the middle attack and pre-image attack. It also ensures integrity of data which means data is not modified during the entire communication. The message also utilizes solution of challenge during double hashing process, the purpose of sending challenge is to ensure receiver is a genuine cellular device means it's not bot thus it mitigates denial of service attack. The double hash value is encrypted by private key of the sender (eNB) to ensure the message is sent by the legitimate device hence mitigating the impersonation attack. Device validation solution VD_2' is sent to mitigate rogue device attack. The entire message is encrypted by the public key of the receiver to maintain secrecy of the message.

3.3.13.3 Decryption Mechanism

Once D_2 receives the authentication request message from CH, it will use its own private key to decrypt the entire message. When the message is decrypted, it will check

challenge given and timestamp. D_2 will address the challenge and check either the timestamp is fresh or not. Later, D_2 will check double hash values of challenge solution from CH and timestamp. Then D_2 will match both values, if double hash values i.e generated by D_2 and other sent by CH match, it means data integrity is maintained.

3.4 Network Model

Figure 3.2 shows a basic network model where LEMAP will be executed and evaluated. There are three important components of our network model (a) eNB and CH (b) Trusted Devices D_1, D_2 and (c) Trudy. eNB has a specific coverage area in which several mobile devices exist. Devices in near proximity range can communicate with each other directly without communicating through BS as in traditional cellular network. There can be a malicious device "Trudy" in the communication range which can jeopardize communication between two devices. ECC algorithm is used for creation of a secret key for communication where Elgamal is used for sharing secret key information. For challenge scheme, this research introduces a basic challenge-response scheme with a timestamp. eNB is a fully trusted representative of the base station and can handle request from devices and update BS about device information. CH is a forward relay device which receives requests from devices in its area and forwards requests to eNB for authentication and authorization. eNB is able to transfer basic information of the ECC algorithm used for creation of keys. The eNB also forwards the NTL to all devices periodically to decrease the request for initial information. eNB updates the BS with information shared by devices such as trust value, block list and authorizes D2D communication and maintain the trust level between all devices. Elgamal is used for sharing key information.

The mobile devices shown in Figure 3.2 are small mobile phones that have D2D communication capabilities. These devices can calculate secret keys, hashing and perform verification without addition of any new hardware or software capabilities. They can also run a proximity range algorithm to locate neighboring devices. These devices also communicate with Cluster Head (CH) which works as non-transparent relay as they can receive the traffic from all neighboring devices and forward it to eNB. These devices have capability to block any traffic intended to them and can forward the message to CH. These devices are intelligent enough to create and solve challenges as well as create timestamp. Devices are also capable of creating and verifying hash based on SHA v3 algorithm. Trudy is malicious device which has super computation and memory capabilities which acts like a normal trusted device. Trudy is looking for any possible security attack such as DoS, eavesdropping or MITM. Thus, Trudy can be called as super malicious device. It has also established a base trust with eNB to participate in communication. Trudy can manage the entire security algorithm such as timestamp, basic challenge-response scheme, ECC and Elgamal.

The scenario in Figure 3.2 explains multi hop D2D communication where all devices are connected to eNB through CH that works as relay agent. Here Trudy shown in yellow can attack by pretending to be legitimate device. In figure 3.2 (a) there is a device D_1 that is reachable by eNB but can communicate to D_2 through CH which allows multi-hop communication. The scenario shows that there are other devices in the coverage area of other eNBs would like to do D2D communication with devices which are under eNBs coverage. The figure 3.2(b) shows the mode of multi-hop communication where all devices are under the control of eNB but they prefer to communicate directly. There is one Trudy device that

is part of the network and would like to compromise the secure communication. Figure 3.2(c) shows the out-of-range communication that is not part of this research study.

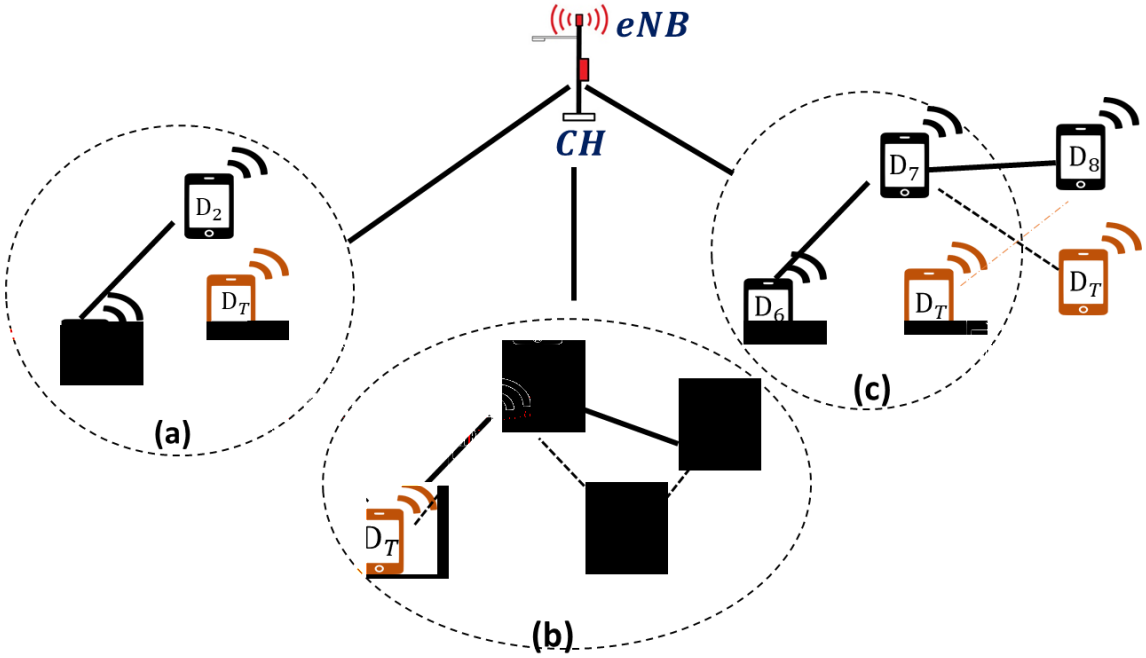

Figure 3.2: Network Model LEMAP Scheme

3.5 Performance Analysis

This research conducted a performance analysis using formal analysis to check authentication and correctness of the proposed scheme. Another dimension this research considered is mathematical analysis where communication cost, authentication overhead, and computation overhead have been calculated. Security requirements have been measured to check whether the proposed scheme covers all the recommended requirements. Security analysis has been performed to check the proposed scheme against quantum attacks.

3.5.1 Formal Analysis

The formal analysis is conducted using BAN Logic which requires security rules and goals should be well defined. Such as for message interpretation, message meaning rule is

considered while freshness of message is tested using nonce verification rule. Burrows, Abadi, and Needham (BAN) proposed a symbolic logic scheme for formal authentication of security protocol (Sufatrio & Yap, 2008; Cohen, 2005).

BAN logic is widely used for the formalization of security algorithms such as presented by Wang and Yu (2019) and Hafeez et al. (2017). Due to its practicability, there are several extensions that have been presented to address other security features which were not covered in base implementation. To use BAN logic, the authentication goals are first defined to represent authentication properties of the proposed algorithm. BAN logic is based on a set of notations to formally represent rules, devices and communication. BAN logic author suggests first converting the protocol into the formal model. All the assumptions required to achieve the authentication goals are proposed first formally. Then BAN logic system is applied on the assumptions and system model using inference rules to achieve the authentication goals. There are several rules for BAN logic, but this section will present only relevant rules used to achieve authentication goals. To check the interpretation of messages, the message meaning rule is applied. This message sets the basic trust about the authority of secret message sent.

$$\frac{D_1 \mid\equiv D_2 \xleftrightarrow{K_{D(1,2)}} D_1 \;, D_1 \triangleleft (\{M\}_{kS})}{D_1 \mid\equiv D_2 \mid\sim M} \qquad \text{Equation 3.14}$$

Equation 3.14 shows that D_1 believes that secret key $K_{D(1,2)}$ is pre-shared with D_2. This rule will infer that D_1believes that D_2 once sent a Message M that encrypted using kS. If it holds true, then it must be assumed that D_1 itself did not send the message. To verify the freshness of the message, the Freshness / Nonce verification belief rule is applied as shown in Equation 3.15.

$$\frac{D_1 \mid\equiv \#(M), D_1 \mid\equiv D_2 \mid\sim M}{D_1 \mid\equiv D_2 \mid\equiv (M)}$$ Equation 3.15

Equation 3.15 states that D_1 believes that Message M has been sent recently and D_1 also believes D_2 sent Message M once, thus this means D_1 believes that D_2 believes in Message M. This means the message is verified and has been sent recently. Jurisdiction rule is based on the belief of D_2 that if it has jurisdiction to create and send Message M, where D_1 believes in D_2 then D_1 will have trust in Message M such as shown in Equation 3.16.

$$\frac{D_1 \mid\equiv D_2 \Rightarrow M, D_1 \mid\equiv D_2 \mid\equiv M}{D_1 \mid\equiv (M)}$$ Equation 3.16

There are several properties D_1 or D_2 believes individually, if the individual believe is true for all then the above statements can also be true.

$$\frac{D_1 \mid\equiv M, D_1 \mid\equiv H}{D_1 \mid\equiv (M,H)}$$ Equation 3.17

Equation 3.17 says that if D_1 believes in M and D_2 believes on H then D_1 believes in both M and H. While vice versa also holds true such that if D_1 believes on the set of Messages M and Hash H then D_1 believes on M such as if $D_1 \mid \equiv (M, H)$ than $D_1 \mid \equiv M$. Similarly, if D_1 trust D_2 and D_2 believes that on Message M and Hash H then D_1 believes that M is sent by D_2 as shown in Equation 3.18.

$$\frac{D_1 \mid\equiv D_2 \mid\equiv (M, H)}{D_1 \mid\equiv D_2 \mid\equiv M}$$ Equation 3.18

Equation 3.18 states that if D_1 believes that D_2 once said M and H then D_1 believes D_2 believes that it sent once Message M shown in Equation 3.19.

$$\frac{D_1 \mid\equiv D_2 \mid\sim (M, H)}{D_1 \mid\equiv D_2 \mid\sim M}$$ Equation 3.19

For digital signature verification and validity, there is an extension in BAN logic made by Sufatrio and Roland. They proposed the idealization of certificates that means it must be trusted that communication made by eNB is secure and always protected by the signature. All eNB's are certified by BS and have a secure secret and public key pairs. This trust belief removes multiple verification steps between eNB and devices and thus allowing the actual communication to be safe and secure. The messages are signed using private key of sender to establish trust of the sender. As it is trusted that only the sender knows its own private key thus the only sender can send the message such as for device D_1 the Sign ${K_{D_1}}^{-1}$, M is done that refers to the message is signed by D_1 using its own secret private key ${K_{D_1}}^{-1}$.

$$\frac{D_2 \mid\equiv pk(D_1, PK_{D_1}), D_2 \mid\equiv \Pi(PK_{D_1}{}^{-1}), D_j \mid\sim D_1 \triangleleft \{M\}_{PK_{D_1}{}^{-1}}}{D_2 \mid\equiv D_1 \mid\sim M}$$ Equation 3.20

Equation 3.20 states that D_2 believes that D_1 has the public key PK_{D_1} that can be used for encryption of the message. D_2 also believes that D_1 has the private key ${PK_{D_1}}^{-1}$ that only is used by D_1 for digital signature. D_2 also believes that D_1 once sent a message M encrypted with the private key. As D_2 believes in the public key of D_1 then D_2 believes that D_1 once said M. While in Equation 3.20, it is stated that D_1 believes that D_2 has the public key PK_{D_2}. D_1 also believes that D_2 has the private key ${PK_{D_2}}^{-1}$ that will be used to look into a message that is encrypted with the public key of D_2 . D_2 also believes that D_1 once sent an encrypted message using its public kfey. Thus D_2 also believes that D_1 once sent a message M. This trust ensures that encrypted message can only be seen by D_2 while also ensuring that only the sender can send this message when merged with Equation 3.21.

$$\frac{D_1 \mid\equiv pk(D_2, PK_{D_2}), D_1 \mid\equiv \Pi(PK_{D_2}{}^{-1}), D_1 \triangleleft \{M\}_{PK_{D_2}}}{D_2 \mid\equiv D_1 \mid\sim M}$$

Equation 3.21

To verify the communication, the public keys are shared signed with eNB private key. For D2D communication, both devices select the secret point SP for communication and goal of the authentication protocol is to trust the secret point. The goal is shown as $D_1 \mid\equiv D_1 \overset{SP}{\longleftrightarrow} D_2$ that states D_1 believes that there is a secret key SP that is shared between D_1 and D_2 for secure communication. Similarly, the vice versa is also true that D_2 believes that there is a secret key SP that is shared between D_2 and D_1 for secure communication such as $D_2 \mid\equiv D_2 \overset{SP}{\longleftrightarrow} D_1$. Thus, both devices D_1 and D_2 believe in a secret point SP that will be used for secret communication. These extensions will allow security against replay attack, attack on confidentiality and integrity as well as an impersonation attack. These extensions are used in conjunction with old rules to establish trust. These rules are used in Chapter 4 for the verification of LEMAP and benchmark protocol to ensure that the proposed protocol is secure against mentioned attacks.

3.5.2 Communication Cost

Transferring small number of authentication messages in any security protocol is always a desired approach. The LEMAP analysis will be made through communication cost. There are three parts of the cost of LEMAP, cost of authentication, cost of calculation of solution of the hash and cost of updating the validation trust. Any kind of computation that is made for communication will be considered. For calculation of cost, we will use Capkun equation as shown in Equation 3.22.

$$C = h \sum_{Au=1}^{K} Sz_{Au}$$

Equation 3.22

Where in Equation 3.22, C is the total communication cost while h refers to several hops or devices in the communication. K is the total number of messages and Sz_{AU} is the size of each message involved in communication.

3.5.3 Mathematical Analysis

To calculate the number of authentication messages sent and received, authentication overhead is calculated. Authentication overhead is an extra load on the system as it is not part of actual communication. It is an inevitable requirement to achieve security in terms of legitimacy of the device. Usually all kind of communication requires authentication and response messages that may vary in number based on the requirements and system settings. (Turkanović et al., 2014; Yang et al., 2005). Some schemes are based on the usage of hardware or smart card to provide authentication (Lee & Chiu, 2005; Chang & Liao, 1994). This research aims to reduce the number of authentication messages as compared to other security schemes. Sharing the session points, calculation base points, finding curve, multiplication and creating key pairs has the major overhead. The computational complexity does not consider the memory usage for each of the components. While in computation overhead, the usage of memory, processing time for overhead and other resources required are noted down. There is a number of researches that have validated their algorithm based on computation overhead such as work by (Wang et al., 2017; Jun et al., 2016; Sedidi & Kumar, 2016; Zhang et al., 2016). In order to validate LEMAP algorithm against major mathematical and intelligent attacks, the validity of the algorithm is tested against these problems. Discrete Logarithm Problem (DLP) states that for any group G in LEMAP case Abelian group, we need to find x where all the information is known such as $a^x \bmod G = H$. In this case, G and H are known while there is need to find x. If x can be found in polynomial time, the algorithm is not secure while if it is proven that finding x is not possible in

polynomial time then the normal key guess attack will fail to result in the secure algorithm. In LEMAP case, ECC will be tested based on DLP (Wang et al., 2017; Javed et al., 2017; Miller, 2004).

Brute force attack (BFA) is one of the most successful attack until now. Several solutions have been presented which are mostly failed as it is very difficult to avoid this attack. But currently researchers are focusing on using the secret key that is very hard to guess and conduct a brute force attack. LEMAP algorithm will be tested against BFA and verified in terms of security based on NIST guidelines. A number of researches have proved their security scheme vigilant against BFA attack such as presented by (He et al., 2019; Javed et al., 2017; Militano et al., 2016; Abd-Elrahman et al., 2015; Raymond & Midkiff, 2008).

Pollard rho method was proposed in 1975 by John Pollard. It was based on finding the integer factor using small space as well time proportional to the square root of the time taken in BFA. Thus, making any algorithm reasonably less secure. There was an improvement made by Pollard and Bent in 1980 to use the greatest common divisor(gcd) based method making the guessing of key much faster (Gordon, 2011). LEMAP will be tested against Pollard rho extended method as used by (Javed et al., 2017).

A Baby step, Giant step (BsGs) is also called as meet in the middle algorithm that tends to find the secret key in less than the square root of time. For BsGs the given parameters are same that are $a^X \bmod G = H$. In which a, G and H are known where G is group point and H is the point of interest. Here the task is also to find X. The work goes as follows, we chose a point K and find all power of K such that from a^1, a^2 --- a^k. While we also find Ha^{-k}, Ha^{-2k}------Ha^{-rk}. BsGS algorithm has been used in proving many security algorithms

to be secure such as researchers conducted by (Bach & Sandlund, 2018; Mughal et al., 2018). LEMAP will be proven secure against BsGs attack. Key Space is one of the approaches of attack where it stores all possible combinations of keys in the database and then use a matching based scheme on the algorithm to find the right key. A key space is successful in case if the computation is just matching as all the key pairs are already made. There are number of researchers who used key space to prove the validity of their algorithm such as (Chen & Steinberger, 2014; Karthikeyan & Nesterenko, 2006). LEMAP will be proven secure against key space attack.

3.6 Summary

This chapter discusses overall concept of security and multi factor-based algorithm for D2D communication in both direct and multi-hop communication. LEMAP is a multi-hop D2D communication security algorithm based on ECC to ensure that multi factors are used for mutual authentication to ensure security and lightweight as compared to currently used algorithms. Elgamal scheme is used to share keys in this LEMAP algorithm. To provide message integrity as well as sender confirmation, signature-based hashing plus challenge scheme is introduced. The chapter provides detailed security aspects of LEMAP as well as computation analysis using multiple schemes. Formal analysis scheme is presented that validates LEMAP against major security attacks such as replay, MITM, impersonation and confidentiality attacks. Computation complexity as well as computation overhead is found to ensure that the algorithm is not expensive in terms of computation and can be used in traditional small devices. Security of algorithm is also validated using smart methods such as Pollard rho and BsGs against BFA.

CHAPTER 4

RESULT AND ANALYSIS

4.1 Introduction

In this chapter, the pseudo code of the proposed algorithm LEMAP has been explained and analysed. The validity and correctness of the proposed protocol LEMAP has been performed using formal method called BAN logic explained in Chapter 3. Communication cost and authentication overhead of the proposed protocol have been calculated. Communication cost is calculated using Capkun equation (Cohen, 2005) and compared with other benchmark protocols. The authentication overhead of the proposed protocol LEMAP is also evaluated and compared with selected benchmarks. Security analysis is performed to prove that the proposed algorithm is secure against brute force attacks, MITM, Replay, Rouge Relay and DoS attacks. Security analysis is performed using Pollard's rho method, Baby Step Giant Step and Key space methods. This is ensured that the proposed algorithm has met security requirements like privacy, confidentiality, integrity, traceability, revocability, non-repudiation and achieved mutual authentication is secure against different types of attacks.

4.2 Development of LEMAP Protocol

The algorithm of multi-hop LEMAP security protocol is explained which shows that LEMAP communication is a multi-hop protocol and achieves secure mutual authentication. Device D_1 requests to communicate with device D_2 and sends Auth-Req to CH to get authentication from eNB. CH receives request message and forwards to eNB where eNB first verifies the legitimacy of CH then responds CH with Auth-Res for D_1 and D_2. This type of communication makes LEMAP a multi-hop protocol. The algorithm with pseudo code of

each communicating message is explained below. Figure 4.1 shows a request message from D_1 to D_2. The message is sent to CH to forward it to eNB to get authentication to communicate with D_2. The message is sent to CH with timestamp, challenge and double hash of the message. The message is encrypted with private key of D_1. CH encrypts the whole message with its private key and forwards it to eNB.

Algorithm 4.1 – Authentication Request Message at D1(Sender)

Step 1: Get sender and receiver IDs through neighbour discovery

Step 2: Get Public key of

$CH=P_{CH}$; $eNB=P_{eNB}$; $D_2= P_{D2}$; $D_1= P_{D1}$

Step 3: Get Private key of $D_1=P_{rD1}$

Step 4: Generate challenge using Blind fold challenge scheme for

$CH_{D1\text{-}CH}$; $CH_{D1\text{-}eNB}$

Step 5: Solve the challenge for

$CH_{D1\text{-}CH} = CH_{D1\text{-}CH}'$

$CH_{D1\text{-}eNB} = CH_{D1\text{-}eNB}'$

Step 6: Generate timestamps T_{SD1}

Step 7: Compute

PID_1, PID_2, $CH_{D1\text{-}CH}$ and T_{SD1} = X

PID_1, PID_2, $CH_{D1\text{-}eNB}$ and T_{SD1} = Y

PID_1, PID_2, $CH_{D1\text{-}CH}'$ and T_{SD1} = X'

PID_1, PID_2, $CH_{D1\text{-}eNB}'$ and T_{SD1} = Y'

Double Hash value of X' = DH (X')

Double Hash value of Y' = DH (Y')

Step 8: Encrypt (D_1-CH) //Encryption for CH

$[DH (X')]_{PrD1}$

$[X, [DH (X')]_{PrD1}]_{PCH}$

Step 9: Encrypt (D_1- eNB) //Encryption for eNB

$[DH (Y')]_{PrD1}$

$[Y, [DH (Y')]_{PrD1}]_{PeNB}$

Step 10: AUTH-REQ: [Encrypt (D_1-CH) + Encrypt (D_1- eNB)]$_{PCH}$

Step 11: Send AUTH-REQ

Figure 4.1: Authentication Request Message at D1

The request message received at CH is decrypted by CH using its public key. CH matches the message received from D1 with hash of message, if both are equal then the message is valid and fresh otherwise discards the message. CH now sends the request to eNB by encrypting it with public key of eNB.

Algorithm 4.2 – Authentication Request Message at CH (Receiver & Sender)

Decryption:

Step 1: Get Public key of

$CH=P_{CH;}$ $eNB=P_{eNB;}$ $D_2= P_{D2;}$ $D_1= P_{D1}$

Step 2: Get Private key of CH =P_{rCH}

Step 3: Get AUTH-REQ (D_1-CH)

[Encrypt (D_1-CH) + Encrypt (D_1- eNB)]$_{PCH}$

Step 4: Compute using private key of CH (P_{rCH})

$[X, [DH (X')]_{PrD1}]_{PCH}$ + $[Y, [DH (Y')]_{PrD1}]_{PeNB}$

Step 5: Select relevant message portion

$[X, [DH (X')]_{PrD1}]_{PCH}$

Step 6: Compute using private key of CH (P_{rCH})

$X, [DH (X')]_{PrD1}$

Step 7: Compute using public key of D_1 (P_{D1})

X, DH (X')

Step 8: Compute DH(X) & compare with DH (X')

Proceed if Condition Satisfied else discard message

Encryption:

Step 9: Generate challenge using Blind fold challenge scheme for

$CH_{CH\text{-}eNB}$

Step 10: Solve the Challenge for: $CH_{CH\text{-}eNB} = CH_{CH\text{-}eNB}'$

Step 11: Generate timestamps T_{SCH}

Step 12: Compute

PID_1, PID_2, $CH_{CH\text{-}eNB}$ and T_{SCH} = X

PID_1, PID_2, $CH_{D1\text{-}CH}'$ and T_{SD1} = X'

Double Hash value of X' = DH (X')

Step 13: Encrypt (CH - eNB)

$[DH (X')]_{PrCH}$;$[X, [DH (X')]_{PrCH}]_{PeNB}$

Step 14: Get Encrypt $(D_1$- eNB)$]_{PCH}$ from STEP 3

Step 15: AUTH-REQ: $[$Encrypt (D1- eNB) + Encrypt (CH - eNB)$]_{PeNB}$

Step 16: Send AUTH-REQ

Figure 4.2: Authentication Request Message at CH

Algorithm 4.3 – Authentication Challenge Response Message at eNB

DECRYPTION:

Step 1: Get Public key of

$CH=P_{CH}$;$eNB=P_{eNB;}$ $D_2= P_{D2;}$ $D_1= P_{D1}$

Step 2: Get Private key of eNB $=P_{reNB}$

Step 3: Get AUTH-REQ (CH- eNB):

$[\text{Encrypt } (D_1\text{- eNB}) + \text{Encrypt (CH - eNB)}]_{PeNB}$

Step 4: Compute using private key of eNB (P_{reNB})

$[Y, [DH (Y')]_{PrD1}]_{PeNB}+[X, [DH (X')]_{PrCH}]_{PeNB}$

Step 5: Select Relevant Message Portion

$Y, [DH (Y')]_{PrD1} +[X, [DH (X')]_{PrCH}$

Step 6: Compute using public key of D1 (P_{D1})

Y, [DH (Y')

Step 7: Compute DH(Y) & Compare with DH (Y')

Proceed if Condition Satisfied (both same values) else discard message

Step 8: Compute using public key of CH (P_{CH})

X, DH (X')

Step 9: Compute DH(X) & Compare with DH (X')

Proceed if Condition Satisfied (both same values) else discard message

Encryption

Step 1: Generate challenge using Blind fold challenge scheme for

CH_{eNB-CH}

Step 2: Solve the Challenge for

$CH_{eNB-CH} = CH_{eNB-CH}'$

Step 3: Generate timestamps T_{SeNB}

Step 4: Generate $OTP_{eNB\text{-}CH}$

Step 5: Solve $OTP_{eNB\text{-}CH} = OTP_{eNB\text{-}CH}'$

Step 6: Compute

$CH_{eNB\text{-}CH}, T_{SeNB}, OTP_{eNB\text{-}CH} = X$

$CH_{eNB\text{-}CH}', T_{SeNB}, OTP_{eNB\text{-}CH}' = X'$

Double Hash value of X'= DH (X')

Encryption for CH

Step 7: Encrypt (eNB-CH)

$[DH (X')]_{PreNB}$

$[X, [DH (X')]_{PreNB}]_{PCH}$

Step 8: AUTH-CHALLENGE-REQ: Encrypt (eNB-CH)

Step 9: Send AUTH-CHALLENGE-REQ

Figure 4.3: Authentication Challenge Response Message at eNB

In Figure 4.2, eNB receives encrypted request message from CH. It decrypts this message using its private key and then matches the actual message with hash of the message. If the match is equal then it considers message is valid and fresh, otherwise discards the message. As the message is sent from CH, eNB responds CH with onetime password to validate either CH is a legitimate device. eNB encrypts the message including challenge, timestamp and onetime password with its private key and public key of CH and sends to CH.

Algorithm 4.4 – Authentication Challenge Response Message at CH

DECRYPTION:

Step 1: Get AUTH-CHALLENGE-REQ

Encrypt (eNB-CH) = [X, [DH (X')] PreNB]PCH

Step 2: Compute using private key of CH (PrCH)

X, [DH (X')] PreNB

Step 3: Compute using public key of eNB

X, DH (X')

Step 4: Compute DH(X) & Compare with DH (X')

Proceed if Condition Satisfied (both same values) else discard message

Encryption

Step 5: Generate challenge using Blind fold challenge scheme for

CHCH-eNB

Step 6: Solve the Challenge for

CHCH-eNB = CHCH-eNB'

Step 7: Generate timestamps TSCH

Step 8: Solve OTPeNB-CH

Step 9: Compute

$Ch_{CH-eNB}, T_{S_{CH}}$, OTP'_{CH-eNB}= X

Ch'_{CH-eNB} , $T_{S_{CH}}$, OTP'_{CH-eNB}= X'

Double Hash value of X'= DH (X')

Step 10: Encrypt (CH-eNB)

[DH (X')] PrCH

[X, [DH (X')] PrCH]PeNB

Step 11: AUTH-CHALLENGE-RESP: Encrypt (CH-eNB)

Step 12: Send AUTH-CHALLENGE-RESP

Figure 4.4: Authentication Challenge Response Message at CH

In Figure 4.3, CH receives response message from eNB to solve the onetime password. CH first decrypts the message with private key, solves the challenge and matches it with hashed message. If the solved challenge and timestamp are same, it considers the message is a valid and fresh message and there is no replay attack. CH solves the onetime password and sends request to eNB with challenge and timestamp. CH encrypts the message with its private key and public key of eNB.

In Figure 4.4, the response message from eNB is received at CH. CH decrypts whole message using its private key. Later CH decrypts the challenge and timestamp message using public key of eNB. Then CH solves challenge and compares the hashed message with actual message. If both messages are equal, then proceeds otherwise discards the message. After decryption CH sees encrypted messages for D1 and D2 and then forwards these messages to both devices with new challenge and timestamp.

In Figure 4.5, D1 receives response message from CH. D1 decrypts the whole message using its private key. Then decrypts the challenge message from CH using public key of CH. D1 solves the challenge, checks timestamp and compares this with hash of message. If both are equal, it accepts otherwise discards the message. D1 decrypts the message sent by eNB using its public key and finds a onetime password to be solved to prove its legitimacy.

Algorithm 4.5 – Authentication Response Message at eNB

Decryption:

Step 1: Get AUTH-CHALLENGE-RESP

Encrypt (CH-eNB) = [X, [DH (X′)] P_{rCH}]P_{eNB}

Step 2: Compute using private key of eNB (P_{reNB})

X, [DH (X′)] P_{rCH}

Step 3: Compute using public key of CH

X, DH (X')

Step 4: Compute DH(X) & Compare with DH (X′)

Proceed if Condition Satisfied (both same values) else discard message

Encryption:

Step 5: Generate challenge using Blind fold challenge scheme for

CH_{eNB-CH}; $CH_{eNB-D1;}CH_{eNB-D2}$

Step 6: Solve the Challenge for

$CH_{eNB-CH} = CH_{eNB-CH}'$; $CH_{eNB-D1} = CH_{eNB-D1}'$;$CH_{eNB-D2} = CH_{eNB-D2}'$

Step 7: Generate timestamps T_{SeNB}

Step 8: Generate

OTP_{eNB-D1} ; OTP_{eNB-D2}

Step 9: solve :$OTP_{eNB-D1} = OTP_{eNB-D1}'$; $OTP_{eNB-D2} = OTP_{eNB-D2}'$

Step 10: Compute

CH_{eNB-CH}, T_{SeNB} = X; CH_{eNB-CH}', T_{SeNB} = X ′

CH_{eNB-D1}, T_{SeNB} OTP_{eNB-D1} =Y; CH_{eNB-D1}' , T_{SeNB} , OTP_{eNB-D1}' = Y′

CH_{eNB-D2} T_{SeNB} , OTP_{eNB-D2} =Z; CH_{eNB-D2}', T_{SeNB}, OTP_{eNB-D2}' = Z′

Double Hash value of X′= DH (X′)

Double Hash value of $Y' = DH(Y')$

Double Hash value of $Z' = DH(Z')$

Step 11: Encrypt (eNB-CH)

$[DH(X')]_{PreNB:}\ [X, [DH(X')]_{PreNB}]P_{CH}$

Encrypt (eNB-D1)

$[DH(Y')]_{PreNB:}\ [Y, [DH(Y')]_{PreNB}]P_{D1}$

Encrypt (eNB-D2)

$[DH(Z')]_{PreNB;}\ [Z, [DH(Z')]_{PreNB}]P_{D2}$

Step 12: AUTH-RES: [Encrypt (eNB-CH)+Encrypt (eNB- D1) + Encrypt (eNB- D2)]

P_{CH}

Step 13: Send AUTH-RES (eNB- CH)

Figure 4.5: Authentication Challenge Response Message

In Figure 4.6, D1 decrypts message received from CH using its private key, solves challenge and checks timestamp. Then compares this with hash of message that is encrypted with private key of CH. D1 decrypts the message using public key of CH. Then compares both messages, if messages are same, it considers message is valid and there is no replay attack. Then D1 send request message to CH by encrypting with its private key.

Algorithm 4.6 – Authentication Challenge Response Message at CH

Decryption:

Step 1: Get AUTH-RES (eNB-CH):

[Encrypt (eNB-CH)+Encrypt (eNB- D_1) + Encrypt (eNB- D_2)] P_{CH}

Step 2: Compute using private key of CH (P_{rCH})

$[X, [DH(X')]_{PreNB}]P_{CH} + [Y, [DH(Y')]_{PreNB}]P_{D1} + [Z, [DH(Z')]_{PreNB}]P_{D2}$

Step 3: Select Relevant Message Portion

$[X, [DH (X')]_{PreNB}]P_{CH}$

Step 4: Compute using private key of CH (P_{rCH})

$X, [DH (X')]P_{reNB}$

Step 5: Compute using public key of eNB (P_{eNB})

$X, [DH (X')]$

Step 6: Compute DH(X) & Compare with DH (X′)

Proceed if Condition Satisfied (both same values) else discard message

Encryption

Step 7: Generate challenge using Blind fold challenge scheme for

CH_{CH-D1} ; CH_{CH-D2}

Step 8: Solve the Challenge for

$CH_{CH-D1} = CH_{CH-D1}'$; $CH_{CH-D2} = CH_{CH-D2}'$

Step 9: Generate timestamp T_{SCH}

Step 10: Compute

CH_{CH-D1} , T_{SCH} =X ; CH_{CH-D1}' , T_{SCH} =X′

CH_{CH-D2} , T_{SCH} =Y ; CH_{CH-D2}' , T_{SCH} =Y′

Double Hash value of X′= DH (X′)

Double Hash value of Y′= DH (Y′)

Step 11: Encrypt ($CH - D_1$)

$[DH (X')]P_{rCH}$

$[X, [DH (X')]_{PrCH}]_{PD1}$

Step 12: Encrypt ($CH - D_2$)

$[DH (Y')]P_{rCH}$

$[Y, [DH (Y')]_{PrCH}]_{PD2}$

Step 13: Get Encrypt (eNB- D_1) from STEP 2

Step 14: Get Encrypt (eNB- D_2) from STEP 2

Step 15: AUTH-RSP (CH – D_1): [Encrypt (CH – D_1) + Encrypt (eNB- D_1)] $_{PD1}$

Step 16: AUTH-RSP (CH – D_2): [Encrypt (CH – D_2) + Encrypt (eNB- D_2)] $_{PD2}$

Step 17: Send AUTH RESP (CH- D_1) and (CH- D_2)

Figure 4.6: Authentication Challenge Response Message for CH

In Figure 4.7, D2 decrypts message received from CH using its private key, solves challenge and checks timestamp. Then compares this with hash of message that's is encrypted with private key of CH. D2 decrypts the message using public key of CH. Then compares both messages, if messages are same, it considers message is valid and there is no replay attack. Then D2 send request message to CH by encrypting with its private key.

Algorithm 4.7 – Authentication Challenge Request Message at D_1

Decryption:

Step 1: Get AUTH-RES (CH- D_1)

[Encrypt (CH – D_1) + Encrypt (eNB- D_1)] $_{PD1}$

Step 2: Compute using Private key of D_1

[Encrypt (CH – D_1) + Encrypt (eNB- D_1)]

$[X, [DH (X')]_{PrCH}]_{PD1} + [Y, [DH (Y')]_{PreNB}]P_{D1}$

$X, [DH (X')]_{PrCH} + Y, [DH (Y')]_{PreNB}$

Step 3: Compute using Public key of CH & Public key of eNB

X, DH (X') + Y, DH (Y')

Step 4: Compute DH(X) & DH (Y) and Compare with DH (X') and DH (Y') respectively

Proceed if condition satisfied (both same values) else discard message

Encryption

Step 1: Generate challenge using Blind fold challenge scheme for

$CH_{D1\text{-}CH}$; $CH_{D1\text{-}eNB}$

Step 2: Solve the Challenge for

$CH_{D1\text{-}CH} = CH_{D1\text{-}CH}'$; $CH_{D1\text{-}eNB} = CH_{D1\text{-}eNB}'$

Step 3: Generate timestamps T_{SD1}

Step 4: Compute

$CH_{D1\text{-}CH}$,$T_{SD1} = X$; $CH_{D1\text{-}CH}'$,$T_{SD1} = X'$

$CH_{D1\text{-}eNB}$,$T_{SD1} = Y$; $CH_{D1\text{-}eNB}'$,$T_{SD1} = Y'$

Double Hash value of X'= DH (X')

Double Hash value of Y'= DH (Y')

Step 5: Encrypt (D_1-CH): $[DH (X')]_{PrD1}$; X, $[DH (X')]P_{rD1}$

Step 6: Encrypt (D_1- eNB)

$[DH (Y')]_{PrD1;}$ $[Y, [DH (Y')]_{PrD1}]_{PeNB}$

Step 7**:** AUTH-REQ-RSP: [Encrypt (D_1-CH) + Encrypt (D_1- eNB)] P_{CH}

Step 8**:** Send AUTH-REQ-RSP

Figure 4.7: Authentication Challenge Request Message at D1

Algorithm 4.8 – Authentication Challenge Request Message at D2

Decryption:

Step 1: Get AUTH-RES (CH- D_2)

// [Encrypt (CH – D_2) + Encrypt (eNB- D_2)] $_{PD2}$

Step 2: Compute using Private key of D_2

// [Encrypt (CH – D_2) + Encrypt (eNB- D_2)]

//Y, [DH (Y')]$_{PrCH}$ + Z, [DH (Z')]$_{PreNB}$

Step 3: Compute using Public key of CH & Public key of eNB

// Y, [DH (Y') + Z, [DH (Z')

Step 4: Compute DH(Y) & DH (Z) and Compare with DH (Y') and DH (Z') respectively

//Proceed if condition satisfied (both same values) else discard message

Encryption

Step 1: Generate challenge using Blind fold challenge scheme for

$CH_{D2\text{-}CH}$; $CH_{D2\text{-}eNB}$

Step 2: Solve the Challenge for

$CH_{D2\text{-}CH} = CH_{D1\text{-}CH}'$; $CH_{D2\text{-}eNB} = CH_{D1\text{-}eNB}'$

Step 3: Generate timestamps T_{SD1}

Step 4: Compute

$CH_{D2\text{-}CH}$,$T_{SD1} = X$; $CH_{D2\text{-}CH}'$,$T_{SD1} = X'$

$CH_{D2\text{-}eNB}$,$T_{SD1} = Y$; $CH_{D2\text{-}eNB}'$,$T_{SD1} = Y'$

Double Hash value of X'= DH (X')

Double Hash value of Y'= DH (Y')

Step 5: Encrypt (D1-CH)

[DH (X')]$_{PrD1;}$ X, [DH (X')]P_{rD1}

Step 6: Encrypt (D2- eNB)

$[DH(Y')]_{PrD1;}\ [Y, [DH(Y')]_{PrD1}]_{PeNB}$

Step 7: AUTH-REQ-RSP: [Encrypt (D_2-CH) + Encrypt (D_2- eNB)]P_{CH}

Step 8: Send AUTH-REQ-RSP

Figure 4.8: Authentication Challenge Request Message at D2

In Figure 4.8, CH gets AUTH-REQ message from D1 and D2 encrypted with public key of CH. CH decrypts this message using its private key. If encrypted message and hash of message both are same, it proceeds further otherwise discard the message. CH generates the challenge and challenge solution for eNB. Also generates new timestamp for eNB. Then encrypts the request message containing validation request VD1 and VD2 for both devices with its private key and sends to eNB.

Algorithm 4.9 – Authentication Request Message at CH (D_1 & D_2)

Decryption:

Step 1: Get AUTH-REQ-RSP:

$[\text{Encrypt } (D_1\text{-CH}) + \text{Encrypt } (D_1\text{- eNB})]_{PCH}$

Step 2: Decrypt Using Private key of CH

$\text{Encrypt } (D_1\text{-CH}) + \text{Encrypt } (D_1\text{- eNB})$

Step 3: Select Relevant Message Portion

$\text{Encrypt } (D_1\text{-CH}) = X, [DH(X')]_{PrD1}$

Step 4: Compute using public key of D_1 (P_{D1})

$X, [DH(X')]$

Step 5: Compute DH(X) & Compare with DH (X′)

Proceed if Condition Satisfied (both same values) else discard message

Encryption:

Step 6: Generate the Challenge for

$$CH_{CH\text{-}eNB} = CH_{CH\text{-}eNB}'$$

Step 7: Solve the Challenge for

$$CH_{CH\text{-}eNB} = CH_{CH\text{-}eNB}'$$

Step 8: Generate timestamps T_{SCH}

Step 9: Generate device validation request VD1 for D1

Step 10: Compute using private key of CH (P_{rCH})

$$CH_{CH\text{-}eNB}, T_{SCH} = X; CH_{CH\text{-}eNB}, T_{SCH} = X'$$

Double Hash value of $X' = DH(X')$

Step 11: Encrypt (CH - eNB)

$$[DH(X')]_{PrCH}$$

$$[X, [DH(X')]_{PrCH}$$

Step 12: Get Encrypt (D1- eNB) from Step 2

Step 13: AUTH-REQ-RSP: [Encrypt (CH - eNB) + Encrypt (D_1- eNB)]$_{PeNB}$

Step 14: Send AUTH-REQ-RSP (CH- eNB)

Figure 4.9: Authentication Request Message for D1 & D2

In Figure 4.9, Auth-Req message is received at eNB. eNB decrypts the message using its private key and checks the hash of message with actual message. If both are same, then proceeds otherwise discards the message. eNB decrypts the message sent from D1 and D2 and verifies the validation solution sent by both devices. Then eNB creates new challenge and timestamp for both devices D1 and D2, encrypts the message with public key of CH with challenge and timestamp and sends this response message to CH.

Algorithm 4.10 – Authentication Response Message at eNB

Decryption:

Step 1: Get AUTH-REQ-RSP: [Encrypt (CH - eNB) + Encrypt (D_1- eNB)]$_{PeNB}$

Step 2: Compute using private key of eNB

Encrypt (CH - eNB) + Encrypt (D_1- eNB)

Step 3: Select message Encrypt (CH - eNB) : [X, [DH (X′)]$_{PrCH}$

Step 4: Compute using Public key of CH ;X, [DH (X′)

Step 5: Compute DH(X) & Compare with DH (X′)

Proceed if Condition Satisfied (both same values) else discard message

Step 6: Select message Encrypt (D_1- eNB): [Y, [DH (Y')]$_{PrD1}$]$_{PeNB}$

Step 7: Compute using Private key of eNB : Y, [DH (Y')]$_{PrD1}$

Step 8: Compute using Public key of D_1 : Y, [DH (Y')]

Step 9: Compute DH(Y) & Compare with DH (Y′)

Proceed if Condition Satisfied (both same values) else discard message

Encryption

Step 10: Generate challenge using Blind fold challenge scheme for

$CH_{eNB\text{-}CH;}$ $CH_{eNB\text{-}D1;}$ $CH_{eNB\text{-}D2}$

Step 11: Solve the Challenge for

$CH_{eNB\text{-}CH} = CH_{eNB\text{-}CH}'$; $CH_{eNB\text{-}D1} = CH_{CH\text{-}D1}'$; $CH_{eNB\text{-}D2} = CH_{CH\text{-}D2}'$

Step 12: Generate timestamp T_{SeNB}

Step 13: Generate authorization VD_1' for D1 and VD_2' for D2

Step 14: Compute

$CH_{eNB\text{-}CH}$, $T_{SeNB} = X$; $CH_{eNB\text{-}CH}'$, $T_{SeNB} = X'$

$CH_{eNB\text{-}D1}$, T_{SeNB} $VD1' = Y$; $CH_{eNB\text{-}D1}'$, T_{SeNB}, $VD1' = Y'$

$CH_{eNB\text{-}D2}$ T_{SeNB} , VD2′ =Z; $CH_{eNB\text{-}D2}$′, T_{SeNB}, VD2′= Z′

Double Hash value of X′= DH (X′)

Double Hash value of Y′= DH (Y′)

Double Hash value of Z′= DH (Z′)

Step 15: Encrypt AUTH RESP (eNB-CH), (eNB- D_1) and (eNB- D_2)

ENCRYPTION for CH, D1 and D2

Step 16: Encrypt (eNB-CH) ; [DH (X′)]P_{reNB} ; X, [DH (X′)]P_{reNB}

Encrypt (eNB-D_1); [DH (Y′)] P_{reNB} ;[Y, [DH (Y′)] P_{reNB}]P_{D1}

Encrypt (eNB-D_2); [DH (Z′)] $P_{reNB;}$ [Z, [DH (Z′)] P_{reNB}] P_{D2}

Step 17: AUTH-RES:

[Encrypt (eNB - CH)] +Encrypt (D_1- eNB) + Encrypt (D_2- eNB)] P_{CH}

Step 18: Send AUTH-RES (eNB-CH)

Figure 4.10: Authentication Response Message at eNB

Algorithm 4.11 – Authentication Response Message from at CH

Decryption:

Step 1: Get AUTH-RES:

[Encrypt (eNB - CH)] + Encrypt (D1- eNB) + Encrypt (D2- eNB)] P_{CH}

[X, [DH (X′)]P_{reNB} +[Y, [DH (Y′)]$_{PrD1}$]P_{eNB}+ [Z, [DH (Z′)]P_{rD2}]P_{eNB}]P_{CH}

Step 2: Compute using private key of CH (P_{rCH})

[X, [DH (X′), VD_1']P_{reNB} +[Y, [DH (Y′)]$_{PrD1}$]P_{eNB}+ [Z, [DH (Z′)]P_{rD2}]P_{eNB}]

Step 3: Select Relevant Message Portion: [X, [DH (X′), VD_1']P_{reNB}

Step 4: Compute using public key of eNB(P_{eNB}): X, [DH (X′), VD_1']

Step 5: Compute DH(X) & Compare with DH (X′)

Proceed if Condition Satisfied (both same values) else discard message

Step 6: Select Relevant Message Portion: [M, [DH (M′), VD_2']P_{reNB}

Step 7: Compute using public key of eNB(P_{eNB}): M, [DH (M′), VD_2']

Step 8: Compute DH(M) & Compare with DH (M′)

Proceed if Condition Satisfied (both same values) else discard message

Encryption

Step 1: Get Private key of D1 =$P_{rD1;}$ Get Private key of D2 =P_{rD2} ;

Get Public key of eNB =P_{eNB} ; Get Public key of CH =P_{CH}

Step 2: Generate challenge using Blind fold challenge scheme for

Ch_{CH-eNB}; Ch_{eNB-D1};Ch_{eNB-D2}

Step 3: Solve the Challenge for

$Ch_{CH-eNB} = Ch_{CH-eNB}'$;$Ch_{eNB-D1} = Ch_{eNB-D1}'$;$Ch_{eNB-D2} = Ch_{eNB-D2}'$

Step 4: Generate timestamp T_{SCH} and T_{SeNB}

Step 5: Compute

Ch_{eNB-D_1} , $T_{S_{eNB}}$=M ; Ch_{eNB-D_1}', $T_{S_{eNB}} = M'$

Ch_{CH-eNB} , T_{SCH}, VD_1 =X;Ch_{CH-eNB}' , T_{SCH} , VD_1=X′

Ch_{eNB-D_2} , $T_{S_{eNB}}$=N; Ch_{eNB-D_2}', $T_{S_{eNB}}$ =N′

Ch_{CH-eNB} , T_{SCH}, VD_2 =Y;Ch_{CH-eNB}' , T_{SCH} , VD_2=Y′

Double Hash value of X′= DH (X′): Double Hash value of Y′= DH (Y′)

Double Hash value of M′= DH (M′):Double Hash value of N′= DH (N′)

Step 6: AUTH RESP (CH- D1) and (CH- D2)-: ENCRYPTION for D1and D2

Step 7: Encrypt (eNB-D1) :[DH (M′)]$P_{reNB;}$ M, [DH (M′)]P_{renB}]$P_{D1;}$

Encrypt (CH-eNB);[DH (X′)]$P_{rCH;}$[X, [DH (X′)]P_{rCH}]P_{D1}

Encrypt (eNB-D2): $[DH\ (N')]P_{reNB:}\ [N, [DH\ (N')]P_{renB}]P_{D2}$

Encrypt (CH-eNB): $[DH\ (Y')]P_{rCH:}[Y, [DH\ (Y')]P_{rCH}]P_{D2}$

Step 8: Send AUTH RESP (CH- D1) and (CH- D2)

Figure 4.11: Authentication Response Message at CH

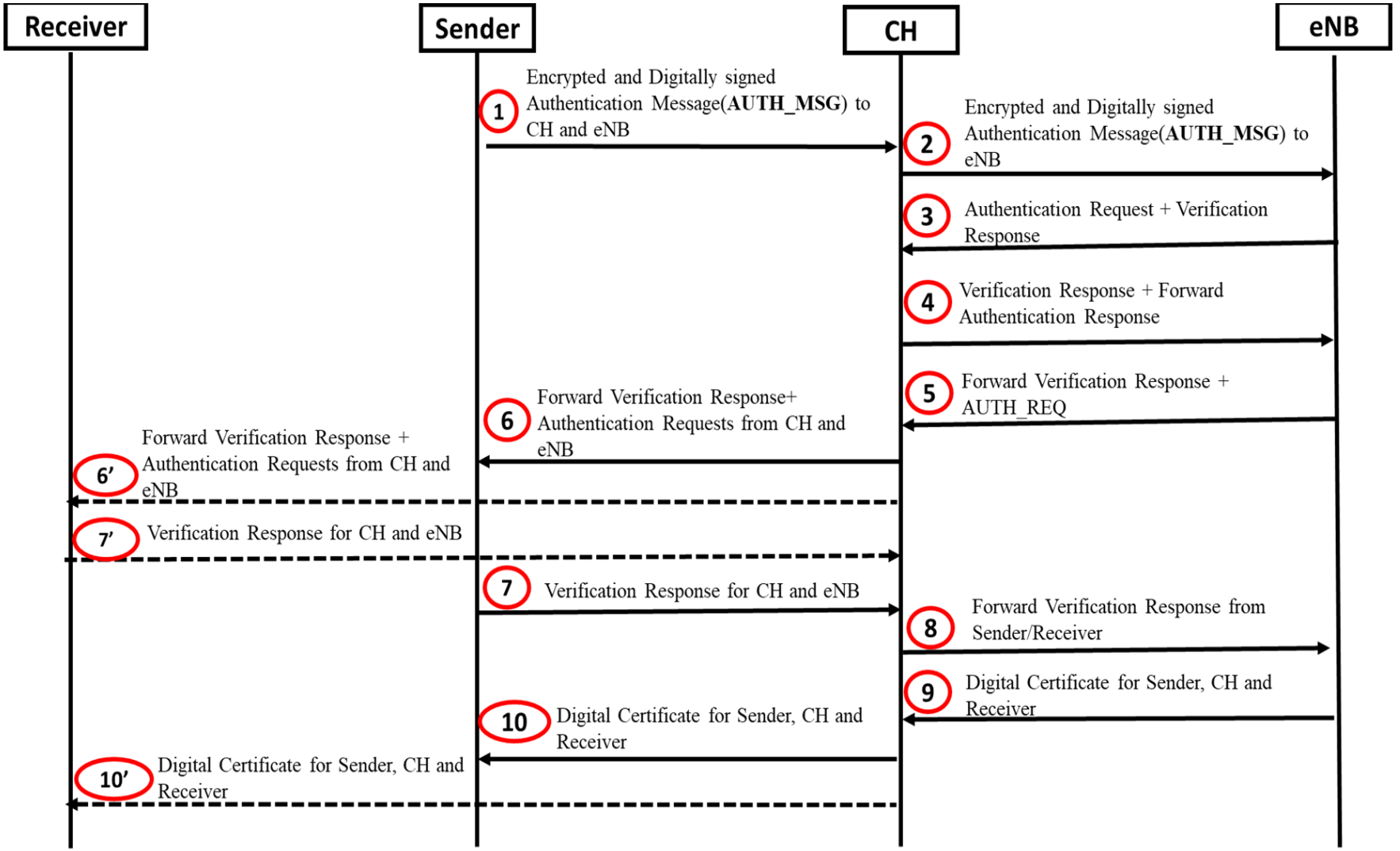

Figure 4.12: LEMAP Timing Diagram with Messages Flow

Figure 4.10 shows the message received at eNB and Figure 4.11 shows the message received at CH, AUTH-RESP message is received at CH. CH decrypts the message using its private key and compares the actual message with hash of message. If both are same, then proceeds otherwise discards the message. Then CH generates challenge for D1 and D2 with new timestamp and validation solution VD1′ and VD2′ from eNB, encrypts the messages for both D1 and D2 with their public keys and send AUTH-RESP messages to both devices.

Figure 4.12 shows the complete message flow between sender, CH, eNB and receiver. All these messages are explained in detailed from Figure 4.1 to Figure 4.11. It is also to be considered that all messages are digitally signed and encrypted by the private key of sender of the message and public key of receiver respectively. All the verification schemes are based on MFA and includes blind fold challenge scheme too. In Figure 4.12, authentication request refers to message send with challenge while authentication response refers to solution of authentication request and a new challenge. The certificates are only issued with limited validity time by eNB.

4.3 Formal Analysis

In this section, formal verification of authentication protocols has been carried out. The proposed algorithms LEMAP with the desired authentication goals will be verified using well known method Burrows, Abadi, and Needham (BAN) Logic. The use of basic notation, inference rules, authentication goals and initial assumptions are first elaborated. For simplicity, Msg stands for the messages and CH is used for Cluster Head. D_1 and D_2 stands for Device (1) and Device (2) respectively and eNB stands for evolved Node Base Station. K_p is used for a private key while P_k is used for the public key. During the analysis, device

(1) will be used as D_1 and device (2) will be used as D_2. The basic notations are defined in Table 4.1.

Table 4.1: BAN Logic Messages Interpretations

No	Formal Message	Interpretation of Formal Message
1	$D_1 \mid\equiv$ Msg	D_1 believes Msg;
2	$D_1 \triangleleft$ Msg	D_1 sees Msg;
3	$D_1 \mid\sim$ Msg	D_1 once said Msg;
4	$D_1 \Rightarrow$ Msg	D_1 has jurisdiction over Msg;
5	# Msg	Msg is fresh
6	{Msg} $K_{D(1,2)}$	Msg is encrypted with $K_{D(1,2)}$
7	$D_1 \overset{K_{D(1,2)}}{\longleftrightarrow} D_2$	D_1 and D_2 have a secret key of $K_{D(1,2)}$
8	$\rho_k (D_1, K_{D_1})$	D_1 has associated a good public key K_{D_1}
9	$\Pi(K_{D_1}^{-1})$	D_1 has a good private key $K_{D_1}^{-1}$
10	K_{CH}	Public key of CH
11	K_{eNB}	Public key of eNB

The various inference rules are listed below; the lists above are the inference rules of the. Burrows, Abadi, and Needham (BAN) Logic. There are many rules but below mentioned are used commonly. A detailed description of the rules can be found in (Sufatrio & Yap, 2008; Cohen, 2005).

$$\frac{D_1 \mid\equiv D_2 \xleftrightarrow{K_{D(1,2)}} D_1 , D_1 \triangleleft (\{Msg\}_k \text{ Signed } D_2)}{D_1 \mid\equiv D_2 \mid\sim Msg} \quad \text{Equation 4.1}$$

In Equation 4.1, D_1 believes that D_2 and D_1 have already shared the secret key. D_1 has also seen the signed message Msg that is signed by D_2. Thus D_1 believes that D_2 have once sent a message Msg.

$$\frac{D_1 \mid\equiv \#(Msg),\ D_1 \mid\equiv D_2 \mid\sim Msg}{D_1 \mid\equiv D_2 \mid\equiv (Msg)} \quad \text{Equation 4.2}$$

In Equation 4.2, D_1 believes that message Msg is fresh and D_1 believes that D_2 have sent once sent a message Msg. Thus D_1 believes that D_2 believes in message Msg.

$$\frac{D_1 \mid\equiv D_2 \Rightarrow Msg,\ D_1 \mid\equiv D_2 \mid\equiv Msg}{D_1 \mid\equiv (Msg)} \quad \text{Equation 4.3}$$

In Equation 4.3, D_1 believes that D_2 have jurisdiction over message Msg. D_1 believes that D_2 believes in message Msg. Thus D_1 believes in message Msg.

$$\frac{D_1 \mid\equiv Msg,\ D_1 \mid\equiv Rsp}{D_1 \mid\equiv (Msg, Rsp)} \quad \text{Equation 4.4}$$

In Equation 4.4, D_1 believes in message Msg. D_1 also believes in message response Rsp. Thus D_1 believes in both messages Msg and response Rsp also as Equation 4.5.

$$\frac{D_1 \mid\equiv (Msg, Rsp)}{D_1 \mid\equiv Msg} \quad \text{Equation 4.5}$$

D_1 believes in both message Msg and response Rsp. Thus D_1 believes in message Msg or vice versa D_1 believes in response Rsp shown as $D_1 \mid\equiv Rsp$.

$$\frac{D_1 \mid \equiv \ D_2 \mid \equiv \ (Msg, Rsp)}{D_1 \mid \equiv \ D_2 \mid \equiv Msg}$$ Equation 4.6

In Equation 4.6, D_1 believes that D_2 believes in both message Msg and response Rsp. Thus D_1 believes that D_2 believes in message Msg or vice versa D_1 believes that D_2 believes in message response Rsp written as $D_1 \mid \equiv \ D_2 \mid \equiv$ Rsp.

$$\frac{D_1 \mid \equiv \ D_2 \mid \sim (Msg, Rsp)}{D_1 \mid \equiv \ D_2 \mid \sim Msg}$$ Equation 4.7

In Equation 4.7, D_1 believes that D_2 once send both messages Msg and response Rsp. Thus D_1 believes that D_2 one sends message Msg.

$$\frac{D_1 \mid \equiv \ P_{D2}\left(D_2, \ K_{D_2}\right), D_1 \mid \equiv \Pi(K_{SS}^{-1}), D_2 \lhd \{\{Msg\}K_{D_2}\}P_{D_1}}{D_1 \mid \equiv \ D_2 \mid \sim Msg}$$ Equation 4.8

Equation 4.8 states that, D_1 believes that P_{D_2} is the public key of D_2 and it has the corresponding private key K_{D_2}. D_1 also believes that it has a secure private session key that can decrypt the message. D_2 has jurisdiction to encrypt the message with its own private key K_{D_2} and then encrypt with the public key of D_1 that is P_{D_1}. Thus D_1 believes that D_2 once sent message Msg. If D_1 has jurisdiction to do the message signature send to D_2. Thus D_1 has jurisdiction over message Msg such as $D_1 \lhd$ Msg.

$$\frac{D_1 \lhd \mu\left(Msg, \ K_{D(1,2)}\right)}{D_1 \lhd Msg}$$ Equation 4.9

Equation 4.9 shows, D_1 has jurisdiction to take the encryption of message Msg using the shared private key. Thus D_1 has jurisdiction over message Msg and vice versa, D_2 also

has jurisdiction to use a shared private key and thus D_2 has jurisdiction over message Msg also shown in Equation 4.10.

$$\frac{D_1 \triangleleft \mu\left(Msg,\ K_{D(1,2)}{}^{-1}\right)}{D_1 \triangleleft Msg} \qquad \text{Equation 4.10}$$

Equation 4.11 shows D_1 has jurisdiction to decrypt the message Msg using the shared private key. Thus D_1 has jurisdiction over message Msg and vice versa, D_2 also has jurisdiction to decrypt the message using a shared private key and thus D_2 has jurisdiction over message Msg.

$$\frac{D_1 |\equiv\ D_2| \equiv \Delta(t_1,\ t_2),\ D_1|\equiv\ D_2|\sim(\theta(t_1,\ t_2),\ Msg)}{D_1|\equiv\ D_2| \equiv (Msg)} \qquad \text{Equation 4.11}$$

D_1 believes that D_2 selects a good time interval that is between t1 and t2. D_1 believes in D_2 that D_2 once sent message Msg and that is between time interval t1 and t2. Thus D_1 believes that D_2 believes in message Msg as stated in Equation 4.11. Before analysing and verifying the authentication protocols, D2D security goals need to be clearly defined.

4.4 Analysis of LEMAP Protocol

In 2PAKEP benchmark protocol, assumptions are used to prove the security goals. The first assumption is that public and private keys have been distributed before start of communication. Second assumption is that the message reaches within timestamp and due time is not expired. Third assumption is that channel is secure, and no attack can occur on the transmission. The proposed algorithm LEMAP is introduced where these assumptions are handled by introducing sound security principles and techniques.

4.4.1 Authentication Goals

This section elaborates the desired authentication goals to be achieved for multi-hop Device to Device (D2D) authentication protocols. Following authentication goals will be needed in LEMAP to prove that secure mutual authentication is achieved. In authentication goals, it is believed that all devices in communication shared a secret key and achieved the required goals to achieve secure mutual authentication.

All goals have been formulated in Equation 4.12 to Equation 4.15. In Equation 4.12, D_1 believes in CH and shared a secret key with CH. Similarly, in Equation 4.13 CH believes in D_1 and shared a secret key with D_1. Hence it is clear from equations, D_1 and CH both must have shared secret keys to get authentication.

$$\textbf{Goal 1}: D_1 |\equiv CH \overset{SK}{\leftrightarrow} D_1 \qquad \text{Equation 4.12}$$

$$\textbf{Goal 2}: CH |\equiv D_1 \overset{SK}{\leftrightarrow} CH \qquad \text{Equation 4.13}$$

In Equation 4.14, D_2 believes in CH and shared a secret key with CH. Similarly, CH believes in D_2 and shared a secret key with D_2. Thus, it is clear from Equations 4.14 and 4.15, D_2 and CH both must have shared secret keys to get authentication.

$$\textbf{Goal 3}: D_2 |\equiv CH \overset{SK}{\leftrightarrow} D_2 \qquad \text{Equation 4.14}$$

$$\textbf{Goal 4}: CH |\equiv D_2 \overset{SK}{\leftrightarrow} CH \qquad \text{Equation 4.15}$$

In Equation 4.16, CH believe in eNB and shared a secret key with eNB and in Equation 4.17, eNB believes at CH and shared a secret key with CH. So being a multi-hop scenario, D_1, CH and eNB all shared secret keys to get secure mutual authentication.

Goal 5: $CH|\equiv eNB \overset{SK}{\leftrightarrow} CH$ Equation 4.16

Goal 6: $eNB|\equiv eNB \overset{SK}{\leftrightarrow} CH$ Equation 4.17

4.4.2 Assumptions

The following assumptions are considered to prove that our proposed algorithm LEMAP achieves secure mutual authentication. For instance, it is assumed that all participating devices in communication shared a secret key. And all messages received at the destination are always fresh. It is assumed that D1 believes CH and has jurisdiction over request message X and CH believes in eNB and has jurisdiction over response message Y. All assumptions have been formulated in Equation 4.18 to Equation 4.27.

A1**:** $D_1 \overset{SK}{\leftrightarrow} CH$ Equation 4.18

A2: $CH \overset{SK}{\leftrightarrow} eNB$ Equation 4.19

A3: $CH|\equiv \#Ts$ Equation 4.20

A4: $eNB|\equiv \#Ts$ Equation 4.21

A5: $D_1 |\equiv CH \Rightarrow X$ Equation 4.22

A6: $CH |\equiv eNB \Rightarrow Y$ Equation 4.23

A7: $D_1, D_2|\equiv CH |\equiv D_{1,} D_2 \overset{SK}{\leftrightarrow} CH$ Equation 4.24

A8: $CH |\equiv D_{1,} D_2|\equiv CH \overset{SK}{\leftrightarrow} D_1, D_2$ Equation 4.25

A9: $CH |\equiv eNB |\equiv eNB \overset{SK}{\leftrightarrow} CH$ Equation 4.26

A10: $eNB |\equiv CH |\equiv eNB \overset{SK}{\leftrightarrow} CH$ Equation 4.27

Let us start with the analysis of authentication request message and the idealization of the message as given below. Each communication message is explained and analysed using BAN logic to verify security goals and how it helps to avoid different types of attacks.

Equation 4.28 indicates that the authentication request message is sent from D_1 to D_2. This message contains pseudo IDs, the timestamp, challenge and hash of the message. The message is sent to CH and contains a message for eNB which is a trusted powerful device and all devices are registered with eNB already defined in Chapter 3. The double hash message is signed with the private key of the sender device D_1. An adversary cannot generate the message and modification attack will not work even if the message is read and modified. This will also help to avoid DoS attack as well. The message contains timestamp which will prevent any replay attack. The temporal secret key SK is also sent so that a secret point of communication can be established without sharing any private key.

4.4.3 Idealization of Authentication Request Message (D1-CH)

Equation 4.28 shows that the authentication request message is sent from D_1 to CH. This message contains pseudo IDs of D_1 , D_2 , CH and eNB, timestamp, challenge, and hash of the message. The message second portion contains a message for eNB.

$$X_1 = \left\{ \begin{array}{c} \left\{ \begin{array}{c} PID_1,\ PID_2,PICH,\ PIeNB,\ Ch_{D_1-CH},\ T_{S_{D_1}},\\ \left[DH\left(\ PID_1,PID_2,\ \ PICH,\ PIENB,\ \ Ch_{D_1-CH}{}',\ T_{S_{D_1}}\right)\right]_{P_{rD_1}} \end{array} \right\}_{P_{CH}} , \\ \left\{ \begin{array}{c} PID_1,\ PID_2,Ch_{D_1-eNB},\ T_{S_{D_1}},\\ \left[DH\left(PID_1,\ PID_2,\ Ch_{D_1-eNB}{}',T_{S_{D_1}}\right)\right]_{P_{rD_1}} \end{array} \right\}_{P_{eNB}} \end{array} \right\}_{P_{CH}}$$

Equation 4.28

After applying message meaning rule Equation 4.1 on Equation 4.28, Equation 4.29 is achieved. X_1 is request message from D_1 to CH. The message is encrypted with private key of D1 which helps to secure message from impersonation attack. Pseudo IDs are used which make it secure from identity reveal attack.CH believes K_{D1} is a public key of D_1 and is legitimate one which is also shared with eNB. CH also believes that only D_1 has access to its private key and D_1 can only use its private key. CH believes that the message request is sent by $D_1$with challenge and timestamp and only D_1 can sign the hash of the contents. Thus CH believes that D_1 created a message, hash and signed it. From the above discussion, it can be seen that CH believes that both keys are good and the message was sent by the legitimate device D_1, thus security goals 4.12 and 4.13 are verified.

$$\frac{CH \mid \equiv \ \rho k\,(D_1, K_{D1}),\ \ CH \mid \equiv \ \Pi(K_{D1}^{-1}),\ \ CH \ \lhd \{\ S\,(X_1), D_1\}K_{CH}}{CH \mid \equiv \ D_1 \mid \sim X_1} \qquad \text{Equation 4.29}$$

By applying freshness or nonce verification Equation 4.2 on statement 4.29, Equation 4.30 is achieved which shows that if CH believes in the freshness of the hashed message sent by D_1 and believes that the message is once sent by D_1 then CH has to believe in the message sent by D_1. From this equation, it is proven that goals are accomplished.

$$\frac{CH \mid \equiv \ \neq X_1, CH \mid \equiv D_1 \mid \sim X_1}{CH \ \mid \equiv \ D_1 \mid \equiv X_1} \qquad \text{Equation 4.30}$$

By applying jurisdiction rule Equation 4.3 on Equation 4.30, Equation 4.31 is accomplished. The equation states that if CH believes that D_1 has jurisdiction over the hashed message and it also believes that D_1 believes in that hashed message then CH has to believe in that message.

$$\frac{CH \mid \equiv D_1 \Rightarrow X_1, CH \mid \equiv D_1 \mid \equiv X_1}{CH \mid \equiv D_1 \mid \equiv X_1}$$ Equation 4.31

From above inference rules, it is observed that CH has full believe on authentication request message sent by the legitimate D_1 which leads towards the authenticity of the message. The message Equation 4.28 contains request from D_1 and signature. P_{CH} is the session public key of CH, X1 is the message request from D_1containing pseudo IDs, challenge and timestamp. The message is signed and encrypted by private key of D_1. Pr_{D_1} is private key of D_1 and P_{D1} is Public key of D_1. The hashed message contains secret private key inside it. If the message is replayed by some attacker within the same timestamp, there will be no impact as the SP cannot be calculated by the attacker which proves that message is safe from replay attack. If the message hash is not equal to actual message or message timestamp is not valid, the message will be discarded.

4.4.4 Idealization of Authentication Request Message (CH-eNB)

Equation 4.32 shows that the authentication request message is sent from CH to eNB. This message contains pseudo IDs of D_1 , D_2 and CH, the timestamp, challenge and hash of the message. The message is sent to eNB which is a trusted powerful device and all devices are registered with the eNB already defined in Chapter 3. The double hashed message is signed with the private key of the sender that is CH. Thus, an adversary cannot generate the message and modification attack will not work even if the message is read and modified. This will also help to avoid MITM and DoS attack as well. The message contains timestamp which will prevent any replay attack. The temporal public key P_{eNB} is also sent so that a secret point of communication can be established without sharing any private key.

$$X_2 = \left\{ \begin{Bmatrix} PID_1,\ PID_2,\ Ch_{D_1-eNB},\ T_{S_{D_1}},\\ \left[DH\left(PID_1,\ PID_2,\ Ch_{D_1-eNB}{}',\ T_{S_{D_1}}\right)\right]_{Pr_{D_1}} \end{Bmatrix}_{P_{eNB}}, \right. \\ \left. \begin{matrix} PID_1,\ PID_2, PICH, Ch_{CH-eNB},\ T_{S_{CH}},\\ \left[DH(PID_1,\ PID_2, PICH,\ Ch_{CH-eNB}{}', T_{S_{CH}},)\right]_{Pr_{CH}} \end{matrix} \right\}_{P_{eNB}}$$ Equation 4.32

$$\frac{eNB \mid \equiv \rho k\ (CH, K_{CH}),\ \ CH \mid \equiv \Pi(K_{CH}^{-1}), eNB \lhd \{\ S\ (X_2), CH\} K_{eNB}}{eNB \mid \equiv CH \mid \sim X_2}$$ Equation 4.33

After applying message meaning rule Equation 4.1 on Equation 4.32, Equation 4.33 is achieved. eNB believes K_{CH} is a public key of CH and is legitimate one which is also shared with eNB. eNB also believes that only CH have access to its private key K_{CH}^{-1} and CH can only use its private key. eNB believes that the message request is sent by CH with pseudo ID, challenge and timestamp. It is believed that only CH can sign the hash of the contents. Thus eNB believes that CH created a message, hash and signed it with its private key. From the above discussion, it can be seen that eNB believes that both keys are good and the message was sent by the legitimate CH, thus security goals are verified.

By applying freshness or nonce verification Equation 4.2 on statement Equation 4.33, Equation 4.34 is achieved which shows that if eNB believes in the freshness of the hashed message sent by CH and believes that the message is once sent by CH then eNB has to believe in the message sent by CH. From this equation, it is proven that goals are accomplished.

$$\frac{eNB \mid \equiv\ \neq X_2, eNB \mid \equiv CH \mid \sim X_2}{eNB\ \mid \equiv CH \mid \equiv X_2}$$ Equation 4.34

By applying jurisdiction rule Equation 4.3 on Equation 4.34, Equation 4.35 is accomplished. The equation states that if eNB believes that CH have jurisdiction over the

hashed message and it also believes that CH believes in that hashed message then eNB must believe in that message.

$$\frac{eNB \mid \equiv\ CH \Rightarrow X_2, eNB \mid \equiv CH \mid \equiv X_2}{eNB\ \mid \equiv\ CH \mid \equiv X_2} \qquad \text{Equation 4.35}$$

From above different inference rules, it is observed that eNB has full belief on authentication request message sent by the legitimate CH which leads towards the authenticity of the message. The message Equation 4.32 that is sent contains request from CH. Where P_{eNB} is the public key of eNB, Pr_{CH} is the private key of CH and P_{CH} is the Public key of CH. It indicates that the message request is fully encrypted and is signed by the P_{eNB}. Moreover, the double hash also contains the secret private key inside it, even if the message is replayed by some attacker within the same timestamp, there will be no impact as the SP cannot be calculated by the attacker. If the message hash is not equal or message time is not valid the message will be discarded.

4.4.5 Idealization of Authentication Response Message (eNB-CH)

Equation 4.36 shows that the authentication response message is sent from eNB to CH. This message contains challenge, timestamp, onetime password and double hash of the message. The message is sent from eNB to CH.

$$Y_1 = \left\{ \begin{matrix} Ch_{eNB-CH}\ ,\ T_{S_{eNB}}, OTP_{eNB-CH}\ , \\ \left[DH\left(Ch_{eNB-CH}{}'\ ,\ T_{S_{eNB}}\ ,\ OTP_{eNB-CH}{}'\ \right)\right]_{Pr_{eNB}} \end{matrix} \right\}_{P_{CH}} \qquad \text{Equation 4.36}$$

By applying message meaning rule Equation 4.1 on statement 4.36, then Equation 4.37 is achieved.

$$\frac{CH \mid \equiv \ \rho k\,(eNB, K_{CH}),\ \ eNB \mid \equiv \ \Pi(K_{eNB}^{-1}),\ \ CH \ \lhd \{\ S\,(Y_1), eNB\}K_{CH}}{CH \mid \equiv \ eNB \mid \sim Y_1} \qquad \text{Equation 4.37}$$

CH believes K_{CH} is a public key of CH and is legitimate one which is also shared with eNB. CH also believes that only eNB have access to its private key K_{eNB}^{-1} and eNB can only use its private key. CH believes that the message response is sent by eNB with challenge, timestamp and onetime password. It is believed that only eNB can sign the hash of the contents. Thus CH believes that eNB created a message, hash and signed it with its private key. From the above discussion, it can be seen that CH believes that both keys are good and the message was sent by eNB, thus security goals are verified.

By applying freshness or nonce verification Equation 4.2 on statement 4.37, Equation 4.38 is achieved that shows that if eNB believes in the freshness of the hashed message sent by eNB and believes that the message is once sent by eNB then CH has to believe in the message sent by the eNB. From this equation, it is proved that goals are accomplished.

$$\frac{CH \mid \equiv \ \neq Y_1, CH \mid \equiv eNB \mid \sim Y_1}{CH \ \mid \equiv eNB \mid \equiv Y_1} \qquad \text{Equation 4.38}$$

By applying jurisdiction rule Equation 4.3 on Equation 4.38, Equation 4.39 is accomplished. The equation states that if eNB believes that CH have jurisdiction over the hashed message and it also believes that CH believes in that hashed message then eNB has to believe in that message.

$$\frac{CH \mid \equiv \ eNB \Rightarrow Y_1, CH \mid \equiv eNB \mid \equiv Y_1}{CH \ \mid \equiv \ eNB \mid \equiv Y_1} \qquad \text{Equation 4.39}$$

From above different inference rules, it is observed that CH has full believe on authentication response message sent by eNB which leads towards the authenticity of the

message. The message that is sent contains response from eNB and signature such as shown in Equation 4.36.

Equation 4.36 is the response message from eNB, P_{CH} is the public key of CH. Pr_{eNB} is the private key of eNB and P_{eNB} is the Public key of eNB. It indicates that the message response is fully encrypted and is signed by eNB. Also the hash contains the secret private key inside it, even if the message is replayed by some attacker within the same timestamp, there will be no impact as the SP cannot be calculated by the attacker. If the message hash calculated by CH is not equal or message time is not valid, the message will be discarded. Secret key (Onetime password OTP) verification is sent to check legitimacy of device.

4.4.6 Idealization of Authentication Request Message (CH-eNB)

The Equation 4.40 shows that the authentication request message is sent from CH to eNB. This message contains challenge, timestamp and onetime password solution. The message is sent to eNB from CH.

$$X_3 = \left\{ \begin{matrix} Ch_{CH-eNB}, T_{S_{CH}}, OTP'_{CH-eNB}, \\ \left[DH\left(Ch'_{CH-eNB}, T_{S_{CH}}, OTP'_{CH-eNB} \right)\right]_{Pr_{CH}} \end{matrix} \right\}_{P_{eNB}} \quad \text{Equation 4.40}$$

After applying message meaning rule Equation 4.1 on Equation 4.40, Equation 4.41 is achieved.

$$\frac{eNB \mid \equiv \rho k\,(CH, K_{CH}),\ \ CH \mid \equiv\ \Pi(K_{CH}^{-1}),\ \ eNB \lhd \{\ S\,(X_2), CH\}K_{eNB}}{eNB \mid \equiv\ CH \mid \sim X_2} \quad \text{Equation 4.41}$$

Equation 4.41 shows that eNB believes K_{CH} is a public key of CH and is legitimate one which is also shared with eNB. eNB also believes that only CH has access to its private key K_{CH}^{-1} and CH can only use its private key. eNB believes that the message request is sent

by CH with challenge, timestamp and onetime password solution. It is believed that only CH can sign the hash of the contents. Thus eNB believes that CH created a message, hashed and signed it with its private key. From the above discussion, it can be seen that eNB believes that both keys are good and the message was sent by CH verifying that CH is a legitimate device by replying onetime password solution, thus security goals are verified.

By applying freshness or nonce verification (Equation 4.2) on statement 4.41, Equation 4.42 is achieved that shows that if eNB believes in the freshness of the hashed message sent by CH and believes that the message is once sent by CH then eNB has to believe in the message sent by the CH. From this equation, it is proven that goals are accomplished.

$$\frac{\text{eNB} \mid\equiv \ \neq X_2, \text{eNB} \mid\equiv \text{CH} \mid\sim X_2}{\text{eNB} \ \mid\equiv \text{CH} \mid\equiv X_2} \qquad \text{Equation 4.42}$$

By applying jurisdiction rule Equation 4.3 on Equation 4.42, Equation 4.43 is achieved. The equation states that if eNB believes that CH have jurisdiction over the hashed message and it also believes that CH believes in that hashed message then eNB must believe in that message.

$$\frac{\text{eNB} \mid\equiv \ \text{CH} \Rightarrow X_2, \text{eNB} \mid\equiv \text{CH} \mid\equiv X_2}{\text{eNB} \ \mid\equiv \ \text{CH} \mid\equiv X_2} \qquad \text{Equation 4.43}$$

From above different inference rules, it is observed that eNB has full believe on authentication request message sent by the legitimate CH which leads towards the authenticity of the message. In the message Equation 4.40 P_{eNB} is the public key of eNB, $X_3 = \text{Msg. req}$ is the message request. $KP_{CH}{}^{-1}$ is the private key of CH and K_{CH} is the Public key of CH. OTP'_{CH-eNB} is onetime password solution solved by CH which validates that CH is a legitimate device. This also indicates that the message request is fully encrypted and is

signed by CH. Moreover, the hash also contains the secret private key inside it, even if the message is replayed by some attacker within the same timestamp, there will be no impact as the SP cannot be calculated by the attacker. If the message hash is not equal or message time is not valid the message will be discarded. $\mathrm{OTP}'_{\mathrm{CH-eNB}}$ is solution of onetime password that was sent from eNB to CH to verify its legitimacy that is also called as secret key in BAN Logic. Hence using assumptions 4.26 and 4.27, authentication goals 4.16 and 4.17 have been achieved as secret key is successfully shared between CH and eNB and both devices have been validated. This not only achieves secure mutual authentication between CH and eNB but also MITM, replay and DoS attacks have been mitigated.

4.4.7 Idealization of Authentication Response Message (eNB-CH)

The Equation 4.44 shows that the authentication response message is sent from eNB to CH. This message contains challenge, timestamp and onetime password for both devices D_1 and D_2 and double hash of the message. The message is sent from eNB to CH.

$$\mathrm{Y}_2 = \left\{ \begin{array}{c} \mathrm{PICH}|\mathrm{P}_{\mathrm{CH}},\ \mathrm{PID}_1|\mathrm{P}_{\mathrm{D}_1},\ \mathrm{PID}_2|\mathrm{P}_{\mathrm{D}_2}|, \\ \mathrm{Ch}_{\mathrm{eNB-CH}},\ \mathrm{T}_{\mathrm{S}_{\mathrm{eNB}}},\ \left[\mathrm{DH}\left(\mathrm{Ch}_{\mathrm{eNB-CH}}',\ \mathrm{T}_{\mathrm{S}_{\mathrm{eNB}}}\right)\right]_{\mathrm{P}_{\mathrm{r}_{\mathrm{eNB}}}}, \\ \left\{ \begin{array}{c} \mathrm{Ch}_{\mathrm{eNB-D}_1},\ \mathrm{T}_{\mathrm{S}_{\mathrm{eNB}}},\mathrm{OTP}_{\mathrm{eNB-D}_1}, \\ \left[\mathrm{DH}\left(\mathrm{Ch}_{\mathrm{eNB-D}_1}',\ \mathrm{T}_{\mathrm{S}_{\mathrm{eNB}}},\mathrm{OTP}_{\mathrm{eNB-D}_1}'\right)\right]_{\mathrm{P}_{\mathrm{reNB}}} \end{array} \right\}_{\mathrm{P}_{\mathrm{D}_1}} \\ , \left\{ \begin{array}{c} \mathrm{Ch}_{\mathrm{eNB-D}_2},\ \mathrm{T}_{\mathrm{S}_{\mathrm{eNB}}},\mathrm{OTP}_{\mathrm{eNB-D}_2}, \\ \left[\mathrm{DH}\left(\mathrm{Ch}_{\mathrm{eNB-D}_2}',\ \mathrm{T}_{\mathrm{S}_{\mathrm{eNB}}},\ \mathrm{OTP}_{\mathrm{eNB-D}_2}'\right)\right]_{\mathrm{P}_{\mathrm{reNB}}} \end{array} \right\}_{\mathrm{P}_{\mathrm{D}_2}} \end{array} \right\}_{\mathrm{P}_{\mathrm{CH}}} \qquad \text{Equation 4.44}$$

After applying message meaning rule Equation 4.1 on Equation 4.44, Equation 4.45 is achieved.

$$\frac{\mathrm{CH}\mid\equiv\ \rho\mathrm{k}\ (\mathrm{eNB},\mathrm{K}_{\mathrm{CH}}),\ \ \mathrm{eNB}\mid\equiv\ \Pi(\mathrm{K}_{\mathrm{eNB}}^{-1}),\ \ \mathrm{CH}\ \lhd\{\ \mathrm{S}\ (\mathrm{Y}_2),\mathrm{eNB}\}\mathrm{K}_{\mathrm{CH}}}{\mathrm{CH}\mid\equiv\ \mathrm{eNB}\mid\sim\mathrm{Y}_2} \qquad \text{Equation 4.45}$$

CH believes that K_{CH} is a public key of CH and is legitimate one which is also shared with eNB. CH also believes that only eNB have access to its private key K_{eNB}^{-1} and eNB can only use its private key. CH believes that the message response is sent by eNB with challenge, timestamp and onetime password. It is believed that only eNB can sign the hash of the contents. Thus CH believes that eNB created a message, hashed and signed it with its private key. From the above discussion, it can be seen that CH believes that both keys are good and the message was sent by eNB, thus security goals are verified.

By applying freshness or nonce verification Equation 4.2 on statement 4.45, Equation 4.46 is achieved that shows that if eNB believes in the freshness of the hashed message sent by eNB and believes that the message is once sent by eNB then CH has to believe in the message sent by eNB. From this equation, it is proved that goals are accomplished.

$$\frac{CH \mid\equiv \neq Y_2, CH \mid \equiv eNB \mid\sim Y_2}{CH \mid \equiv eNB \mid\equiv Y_2} \qquad \text{Equation 4.46}$$

By applying jurisdiction rule Equation 4.3 on statement 4.46, Equation 4.47 is achieved. The equation states that if eNB believes that CH have jurisdiction over the hashed message and it also believes that CH believes in that hashed message then eNB must believe in that message.

$$\frac{CH \mid \equiv eNB \Rightarrow Y_2, CH \mid \equiv eNB \mid\equiv Y_2}{CH \mid \equiv eNB \mid\equiv Y_2} \qquad \text{Equation 4.47}$$

From above different inference rules, it is observed that CH has full believe on authentication response message sent by eNB which leads towards the authenticity of the message. The message Equation 4.44 that is sent contains response from eNB and signature. P_{CH} is the public key of CH, $Msg.\,res = Y2$ is the message response. Pr_{eNB} is the private key

of eNB and P_{eNB} is the Public key of eNB. It indicates that the message response is fully encrypted and is signed by the eNB. Moreover, the hash also contains the secret private key inside it, even if the message is replayed by some attacker within the same timestamp, there will be no impact as the SP cannot be calculated by the attacker. If the message hash is not equal or message time is not valid the message will be discarded.

4.4.8 Idealization of Authentication Response Message (CH-D1)

This message also contains message for D2. In Figure 4.48, D2 receives response message from CH. D2 decrypts the whole message using its private key. Then decrypts the challenge message from CH using public key of CH. D2 solves the challenge, checks timestamp, and compares this with hash of message. If both are equal, it accepts otherwise discards the message. D2 decrypts the message sent by eNB using its public key and finds a onetime password to be solved to prove its legitimacy.

Equation 4.48 shows that the authentication response message is sent from CH to D_1. This message contains challenge, timestamp, onetime password for device D_1 and double hash of the message.

$$Y_{3=}\left\{\begin{matrix}\left\{\begin{matrix}Ch_{eNB-D_1}\,,\ T_{S_{eNB}}\,, OTP_{eNB-D_1}\,,\\ \left[DH(\ Ch'_{eNB-D_1},\ T_{S_{eNB}}\,, OTP_{eNB-D_1}{}'\ )\right]_{Pr_{eNB}}\end{matrix}\right\}_{P_{D_1}}\\ ,\ Ch_{CH-D_1},\ T_{S_{CH}}\,,\ \left[DH(\ Ch'_{CH-D_1}\,, T_{S_{CH}}\ )\right]_{Pr_{CH}}\,,\\ PICH|P_{CH}\,,\ \ PID_1|P_{D_1},\ PID_2|P_{D2}\end{matrix}\right\}_{P_{D_1}} \qquad \text{Equation 4.48}$$

After applying message meaning rule Equation 4.1 on Equation 4.48, Equation 4.49 is achieved.

$$\frac{D_1 \mid \equiv\ \rho k\,(CH, K_{D1}),\ \ CH \mid \equiv\ \Pi(K_{CH}^{-1}),\ \ D_1\ \lhd \{\ S\,(Y_3), CH\}K_{D_1}}{D_1 \mid \equiv\ CH \mid \sim Y_3} \qquad \text{Equation 4.49}$$

D_1 believes K_{D_1} is a public key of D_1 and is legitimate one which is also shared with CH. D_1 also believes that only CH has access to its private key K_{CH}^{-1} and CH can only use its private key. D_1 believes that the message response is sent by CH with challenge, timestamp and onetime password. It is believed that only CH can sign the hash of the contents. Thus D_1 believes that CH created a message, hash and signed it with its private key. From the above discussion, it can be seen that D_1 believes that both keys are good, and the message was sent by CH, security goals are verified. By applying freshness or nonce verification (Equation 4.2) on statement 4.49, Equation 4.50 is achieved that shows that if D_1 believes in the freshness of the hashed message sent by CH and believes that the message is once sent by CH then D_1 has to believe in the message sent by the CH. From this equation, it is proved that goals are accomplished.

$$\frac{D_1 \mid\equiv\ \neq Y_3, D_1 \mid \equiv CH \mid\sim Y_3}{D_1\ \mid \equiv CH \mid\equiv Y_3} \qquad \text{Equation 4.50}$$

By applying jurisdiction rule Equation 4.3 on Equation 4.50, Equation 4.51 is accomplished. The equation states that if CH believes that D_1 have jurisdiction over the hashed message and it also believes that D_1 believes in that hashed message then CH has to believe in that message.

$$\frac{D_1 \mid \equiv\ CH \Rightarrow Y_3, D_1 \mid \equiv CH \mid\equiv Y_3}{D_1\ \mid \equiv\ CH \mid\equiv Y_3} \qquad \text{Equation 4.51}$$

From above different inference rules, it is observed that D_1 has full believe on authentication response message sent by CH which leads towards the authenticity of the

message. The message Equation 4.48 that is sent contains response from CH. P_{D_1} is the public key of D_1, $Msg.res = Y3$ is the message response from CH. Pr_{eNB} is the private key of eNB. It indicates that the message response is fully encrypted and is signed by CH. Moreover, the hash also contains the secret private key inside it, even if the message is replayed by some attacker within the same timestamp, there will be no impact as the SP cannot be calculated by the attacker. If the message hash is not equal or message time is not valid the message will be discarded. Secret key (OTP) verification is sent to check legitimacy of device. Thus, using assumptions 4.24 and 4.25, authentication goals 4.12, 4.13, 4.14, 4.15 have been achieved as secret key is successfully shared between D1 and CH and both devices have been validated. This not only achieves secure mutual authentication between D1 and CH but also MITM, replay and DoS attacks have been mitigated.

4.4.9 Idealization of Authentication Response Message (CH-D2)

Equation 4.52 shows that the authentication response message is sent from CH to D_2. This message contains challenge, timestamp, onetime password for device D_2 and double hash of the message.

$$Y_4 = \left\{ \begin{array}{c} \left\{ \begin{array}{c} Ch_{eNB-D_2}\,,\ T_{S_{eNB}}\,, OTP_{eNB-D_2}\,, \\ \left[H\left(Ch_{eNB-D_2}\,,\ T_{S_{eNB}}\,,\ OTP_{eNB-D_2}\right)\right]_{Pr_{eNB}} \end{array} \right\}_{P_{D_2}}, \\ Ch_{CH-D_2}\,, T_{S_{CH}}\,,\ \left[DH\left(Ch_{CH-D_2}{}',\ T_{S_{CH}}\right)\right]_{Pr_{CH}}, \\ PICH|P_{CH}\,,\ PID_1|P_{D_1},\ PID_2|P_{D2} \end{array} \right\}_{P_{D_2}} \quad \text{Equation 4.52}$$

After applying message meaning rule Equation 4.1 on Equation 4.52, Equation 4.53 is achieved.

$$\frac{D_2 \mid \equiv\ \rho k\left(CH, K_{D_2}\right),\ \ CH \mid \equiv\ \Pi(K_{CH}^{-1}),\ \ D_2\ \lhd \{\ S\left(Y_3\right), CH\} K_{D_2}}{D_2 \mid \equiv\ CH \mid \sim Y_3} \quad \text{Equation 4.53}$$

D_2 believes K_{D_2} is a public key of D_2 and is legitimate one which is also shared with CH. D_2 also believes that only CH has access to its private key and CH can only use its private key. D_2 believes that the message response is sent by CH with challenge, timestamp and onetime password. It is believed that only CH can sign the hash of the contents. Thus D_2 believes that CH created a message, hashed and signed it with its private key. From the above discussion, it can be seen that D_2 believes that both keys are good and the message was sent by CH; thus security goals are verified.

By applying freshness or nonce verification Equation 4.2 on statement 4.53, Equation 4.54 is achieved that shows that if D_2 believes in the freshness of the hashed message sent by CH and believes that the message is once sent by CH then D_2 has to believe in the message sent by CH. From this equation, it is proved that goals are accomplished.

$$\frac{D_2 \mid\equiv \neq Y_4, D_2 \mid\equiv CH \mid\sim Y_4}{D_2 \mid\equiv CH \mid\equiv Y_4} \qquad \text{Equation 4.54}$$

By applying jurisdiction rule Equation 4.3 on Equation 4.54, Equation 4.55 is accomplished. The equation states that if CH believes that D_2 have jurisdiction over the hashed message and it also believes that D_2 believes in that hashed message then CH has to believe in that message.

$$\frac{D_2 \mid\equiv CH \Rightarrow Y_4, D_2 \mid\equiv CH \mid\equiv Y_4}{D_2 \mid\equiv CH \mid\equiv Y_4} \qquad \text{Equation 4.55}$$

From above different inference rules, it is observed that D_2 has full believe on authentication response message sent by CH which leads towards the authenticity of the message. The message Equation 4.52 that is sent contains response from CH and signature. P_{D_2} is the public key of D_2, $Msg.res = Y4$ is the message response from CH. Pr_{eNB} is the

private key of eNB and P_{eNB} is the Public key of eNB. It indicates that the message response is fully encrypted and is signed by eNB. Moreover, the hash also contains the secret private key inside it, even if the message is replayed by some attacker within the same timestamp, there will be no impact as the SP cannot be calculated by the attacker. If the message hash is not equal or message time is not valid the message will be discarded. Secret key (OTP) verification is sent to check legitimacy of device. Thus, using assumptions 4.24 and 4.25, authentication goals 4.12, 4.13, 4.14, 4.15 have been achieved as secret key is successfully shared between D2 and CH and both devices have been validated. This not only achieves secure mutual authentication between D2 and CH but also MITM, replay and DoS attacks have been mitigated.

4.4.10 Idealization of Authentication Request Message (D1-CH)

Equation 4.56 shows that the authentication request message is sent from D_1 to CH. This message contains challenge, timestamp, and hash of the message. The message is sent to CH to forward it to eNB.

$$X_4 = \left\{ \begin{matrix} Ch_{D_1-CH}\,,\ T_{s_{D_1}}\,,\ \left[DH\left(Ch_{D_1-CH}{}',\ T_{s_{D_1}}\right)\right]_{Pr_{D_1}}\,, \\ Ch_{D_1-eNB}\,,\ T_{s_{D_1}}\,,\ \left\{\left[DH\left(Ch_{D_1-eNB}{}',\ T_{s_{D_1}}\right)\right]_{Pr_{D_1}}\right\}_{P_{eNB}} \end{matrix} \right\}_{P_{eNB}} \quad \text{Equation 4.56}$$

After applying message meaning rule Equation 4.1 on Equation 4.56, Equation 4.57 is achieved.

$$\frac{CH \mid\equiv\ \rho k\left(D_1, K_{D_1}\right),\ \ CH\mid\equiv\ \Pi(K_{D1}^{-1}),\ \ CH\ \lhd\{\ S\ (X_4), D_1\}K_{CH}}{CH\mid\equiv\ D_1\mid\sim X_4} \quad \text{Equation 4.57}$$

CH believes K_{D1} is a public key of D_1 and is legitimate one which is also shared with eNB. CH also believes that only D_1 has access to its private key K_{D1}^{-1} and D_1 can only use its private key. CH believes that the message request is sent by D_1 with challenge and timestamp and only D_1 can sign the hash of the contents. Thus CH believes that D_1 created a message, hash and signed it. From the above discussion, it can be seen that CH believes that both keys are good, and the message was sent by the legitimate device D_1, thus security goals are verified.

By applying freshness or nonce verification Equation 4.2 on statement 4.57, Equation 4.58 is achieved that shows that if CH believes in the freshness of the hashed message sent by D_1 and believes that the message is once sent by D_1 then CH has to believe in the message sent by the D_1. From this equation, it is proven that goals are accomplished.

$$\frac{CH \mid\equiv \neq X_4, CH \mid\equiv D_1 \mid\sim X_4}{CH \mid\equiv D_1 \mid\equiv X_4} \qquad \text{Equation 4.58}$$

By applying jurisdiction rule Equation 4.3 on Equation 4.58, Equation 4.59 is accomplished. The equation states that if CH believes that D_1 has jurisdiction over the hashed message and it also believes that D_1 believes in that hashed message then CH has to believe in that message.

$$\frac{CH \mid\equiv D_1 \Rightarrow X_4, CH \mid\equiv D_1 \mid\equiv X_4}{CH \mid\equiv D_1 \mid\equiv X_4} \qquad \text{Equation 4.59}$$

From above different inference rules, it is observed that CH has full believe on authentication request message sent by the legitimate device D_1 which leads towards the authenticity of the message. The message Equation 4.56 contains request from D_1. Pr_{D_1} is the private key of D_1, $Msg.req$ is the message request from D_1 containing challenge and

timestamp. The message is fully signed and encrypted by private key of D_1. This shows that the message request is fully encrypted and is signed by D_1. Moreover, the hash also contains the secret private key inside it, even if the message is replayed by some attacker within the same timestamp, there will be no impact as the SP cannot be calculated by the attacker. If the message hash is not equal or message time is not valid the message will be discarded.

4.4.11 Idealization of Authentication Request Message (D2-CH)

Equation 4.60 shows that the authentication request message is sent from D_2 to CH. This message contains challenge, timestamp and hash of the message. The message is sent to CH to forward it to eNB.

$$X_5 = \left\{ \begin{matrix} Ch_{D_{2-CH}}, T_{s_{D_2}}, \left[DH\left(Ch_{D_{2-CH}}', T_{s_{D_2}}\right)\right]_{Pr_{D_2}}, \\ Ch_{D_2-eNB}, T_{s_{D_2}}, \left\{\left[DH\left(Ch_{D_1-eNB}', T_{s_{D_1}}\right)\right]_{Pr_{D_2}}\right\}_{P_{eNB}} \end{matrix} \right\}_{P_{eNB}} \quad \text{Equation 4.60}$$

After applying message meaning rule Equation 4.1 on Equation 4.60, Equation 4.61 is achieved.

$$\frac{CH \mid\equiv \rho k\left(D_2, K_{D_2}\right),\ CH \mid\equiv \Pi\left(K_{D_2}^{-1}\right),\ CH \lhd \{S(X_5), D_2\}K_{CH}}{CH \mid\equiv D_2 \mid\sim X_5} \quad \text{Equation 4.61}$$

CH believes K_{D_2} is a public key of D_2 and is legitimate one which is also shared with eNB. CH also believes that only D_2 have access to its private key and D_2 can only use its private key. CH believes that the message request is sent by $D_2$with challenge and timestamp and only D_2can sign the hash of the contents. Thus CH believes that D_2created a message, hashed and signed it. From the above discussion, it can be seen that CH believes that both keys are good and the message was sent by the legitimate D_2, thus security goals are verified.

By applying freshness or nonce verification Equation 4.2 on statement 4.61, Equation 4.62 is achieved that shows that if CH believes in the freshness of the hashed message sent by D_2 and believes that the message is once sent by D_2 then CH has to believe in the message sent by D_2. From this equation, it is proven that goals are accomplished.

$$\frac{CH \mid\equiv \neq X_5, CH \mid\equiv D_2 \mid\sim X_5}{CH \mid\equiv D_2 \mid\equiv X_5} \qquad \text{Equation 4.62}$$

By applying jurisdiction rule Equation 4.3 on Equation 4.62, Equation 4.63 is accomplished. The equation states that if CH believes that D_2 have jurisdiction over the hashed message and it also believes that D_2 believes in that hashed message then CH has to believe in that message.

$$\frac{CH \mid\equiv D_2 \Rightarrow X_4, CH \mid\equiv D_2 \mid\equiv X_4}{CH \mid\equiv D_2 \mid\equiv X_4} \qquad \text{Equation 4.63}$$

From above different inference rules, it is observed that CH has full believe on authentication request message sent by the legitimate D_2 which leads towards the authenticity of the message. The message Equation 4.60 contains request from D_2 and signature. Pr_{D_2} is the private key of D_2, Msg. req is the message request from D_2containing challenge and timestamp. The message is fully signed and encrypted by private key of D_2. This shows that the message request is fully encrypted and is signed by D_2. Moreover, the hash also contains the secret private key inside it, even if the message is replayed by some attacker within the same timestamp, there will be no impact as the SP cannot be calculated by the attacker. If the message hash is not equal or message time is not valid the message will be discarded.

4.4.12 Idealization of Authentication Request Message (CH-eNB)

The Equation 4.64 shows that the authentication request message is sent from CH to eNB. This message contains challenge, timestamp, validation request from D_1 and D_2 and hash of the message. The message is sent to eNB.

$$X_6 = \left\{ \begin{array}{l} Ch_{D_1-eNB}, T_{s_{D_1}}, \left\{ \left[DH\left(Ch_{D_1-eNB}', T_{s_{D_1}}\right)\right]_{Pr_{D_1}} \right\}_{P_{eNB}}, \\ Ch_{CH-eNB}, T_{S_{CH}}, \left[DH\left(Ch_{CH-eNB}', T_{S_{CH}}\right), VD_1\right]_{Pr_{CH}} \\ Ch_{D_2-eNB}, T_{s_{D_2}}, \left\{ \left[DH\left(Ch_{D_1-eNB}', T_{s_{D_2}}\right)\right]_{Pr_{D_2}} \right\}_{P_{eNB}}, \\ Ch_{CH-eNB}, T_{S_{CH}}, \left[DH\left(Ch_{CH-eNB}', T_{S_{CH}}\right), VD_2\right]_{Pr_{CH}} \end{array} \right\}_{P_{eNB}} \quad \text{Equation 4.64}$$

After applying message meaning rule Equation 4.1 on Equation 4.64, Equation 4.65 is achieved.

$$\frac{eNB \mid \equiv \rho k\,(CH, K_{CH}),\ \ CH \mid \equiv\ \Pi(K_{CH}^{-1}),\ eNB\ \lhd \{\ S\,(X_6), CH\}K_{eNB}}{eNB \mid \equiv\ CH \mid \sim X_6} \quad \text{Equation 4.65}$$

eNB believes K_{CH} is a public key of CH and is legitimate one which is also shared with eNB. eNB also believes that only CH has access to its private key K_{CH}^{-1} and CH can only use its private key. eNB believes that the message request is sent by CH with challenge, timestamp and device validation request VD_1 and VD_2 for bothe devices D1 and D2. It is believed that only CH can sign the hash of the contents. Thus eNB believes that CH created a message, hashed and signed it with its private key. From the above discussion, it can be seen that eNB believes that both keys are good and the message was sent by the legitimate CH, thus security goals are verified.

By applying freshness or nonce verification Equation 4.2 on statement 4.65, Equation 4.66 is achieved that shows that if eNB believes in the freshness of the hashed message sent

by CH and believes that the message is once sent by CH then eNB has to believe in the message sent by the CH. From this equation, it is proven that goals are accomplished.

$$\frac{eNB \mid\equiv \neq X_6, eNB \mid \equiv CH \mid\sim X_6}{eNB \mid \equiv CH \mid\equiv X_6} \quad \text{Equation 4.66}$$

By applying jurisdiction rule Equation 4.3 on Equation 4.66, Equation 4.67 is accomplished. The equation states that if eNB believes that CH has jurisdiction over the hashed message and it also believes that CH believes in that hashed message then eNB must believe in that message.

$$\frac{eNB \mid\equiv CH \Rightarrow X_6, eNB \mid \equiv CH \mid\equiv X_6}{eNB \mid\equiv CH \mid\equiv X_6} \quad \text{Equation 4.67}$$

From above different inference rules, it is observed that eNB has full believe on authentication request message sent by the legitimate CH which leads towards the authenticity of the message. The message Equation 4.64 that is sent contains request from CH. P_{eNB} is the public key of eNB, $Msg.req$ is the message request. Pr_{CH} is the private key of CH and P_{CH} is the Public key of CH. It shows that the message request is fully encrypted and is signed by eNB. Moreover, the hash also contains the secret private key inside it, even if the message is replayed by some attacker within the same timestamp, there will be no impact as the SP cannot be calculated by the attacker. If the message hash is not equal or message time is not valid the message will be discarded.

4.4.13 Idealization of Authentication Response Message (eNB-CH)

Equation 4.68 shows that the authentication response message is sent from eNB to CH. This message contains challenge, timestamp and validation solution for both devices D_1 and D_2 and double hash of the message. The message is sent from eNB to CH.

$$Y_5 = \left\{ \begin{array}{l} Ch_{eNB-CH}\,,\ T_{S_{eNB}}\,,\left[DH\left(Ch_{eNB-CH}{}',\ T_{S_{eNB}}\right),\ VD_1{}'\right]_{Pr_{eNB}}\,, \\ \left\{Ch_{eNB-D_1}\,,\ T_{S_{eNB-D_1}}\,,\left[DH\left(Ch_{eNB-D_1}{}',\ T_{S_{eNB-D_1}}\right)\right]_{Pr_{D_1}}\right\}_{P_{eNB}}\,, \\ Ch_{eNB-CH}\,,\ T_{S_{eNB}}\,,\left[DH\left(Ch_{eNB-CH}{}',\ T_{S_{eNB}}\right),\ VD_2{}'\right]_{Pr_{eNB}}\,, \\ \left\{Ch_{eNB-D_2}\,,\ T_{S_{eNB-D_2}}\,,\ \left[DH\left(Ch_{eNB-D_2}{}',\ T_{S_{eNB-D_2}}\right)\right]_{Pr_{D_2}}\right\}_{P_{eNB}} \end{array} \right\}_{P_{CH}} \quad \text{Equation 4.68}$$

After applying message meaning rule Equation 4.1 on Equation 4.68, Equation 4.69 is achieved.

$$\frac{CH \mid \equiv\ \rho k\,(eNB, K_{CH}),\ \ eNB \mid \equiv\ \Pi(K_{eNB}^{-1}),\ \ CH\ \lhd \{\ S\,(Y_5), eNB\}K_{CH}}{CH \mid \equiv\ eNB \mid \sim Y_5} \quad \text{Equation 4.69}$$

CH believes K_{CH} is a public key of CH and is legitimate one which is also shared with eNB. CH also believes that only eNB have access to its private key K_{eNB}^{-1} and eNB can only use its private key. CH believes that the message response is sent by eNB with challenge, timestamp and device validation solution for D_1 and D_2. It is believed that only eNB can sign the hash of the contents. Thus, CH believes that eNB created a message, hashed and signed it with its private key. From the above discussion, it can be seen that CH believes that both keys are good and the message was sent by eNB, thus security goals are verified. By applying freshness or nonce verification Equation 4.2 on statement 4.69, Equation 4.70 is achieved that shows that if eNB believes in the freshness of the hashed message sent by eNB and believes that the message is once sent by eNB then CH has to believe in the message sent by the eNB. From this equation, it is proved that goals are accomplished.

$$\frac{CH \mid\equiv\ \neq Y_5, CH \mid \equiv eNB \mid\sim Y_5}{CH\ \mid \equiv eNB \mid\equiv Y_5} \quad \text{Equation 4.70}$$

By applying jurisdiction rule Equation 4.3 on Equation 4.70, Equation 4.71 is accomplished. The equation states that if eNB believes that CH have jurisdiction over the hashed message and it also believes that CH believes in that hashed message then eNB has to believe in that message.

$$\frac{CH \mid \equiv eNB \Rightarrow Y_5, CH \mid \equiv eNB \mid \equiv Y_5}{CH \mid \equiv eNB \mid \equiv Y_5} \quad \text{Equation 4.71}$$

From above different inference rules, it is observed that CH has full believe on authentication response message sent by eNB which leads towards the authenticity of the message. The message Equation 4.68 contains response from eNB. P_{CH} is the public key of CH, $Msg.res = Y5$ is the message response. Pr_{eNB} is the private key of eNB and P_{eNB} is the Public key of eNB. This indicates that the message response is fully encrypted and is signed by eNB. Moreover, the hash also contains the secret private key inside it, even if the message is replayed by some attacker within the same timestamp, there will be no impact as the SP cannot be calculated by the attacker. If the message hash is not equal or message time is not valid the message will be discarded.

4.4.14 Idealization of Authentication Response Message (CH-D1)

Equation 4.72 shows that the authentication response message is sent from CH to D_1. This message contains challenge, timestamp, and validation response from D_1 and hash of the message. The message is sent to D_1.

$$Y_6 = \left\{ \begin{matrix} \left\{ Ch_{eNB-D_1}\ ,\ T_{S_{eNB}}\ ,\ \left[DH\left(Ch_{eNB-D_1}{}',\ T_{S_{eNB}} \right) \right]_{Pr_{eNB}} \right\}_{P_{D_1}}\ , \\ Ch_{CH-eNB}\ ,\ T_{S_{CH}}\ , VD_1', \left[DH\left(Ch_{CH-eNB}{}',\ T_{S_{CH}}\ ,\ VD_1' \right) \right]_{Pr_{CH}} \end{matrix} \right\}_{P_{D_1}} \quad \text{Equation 4.72}$$

After applying message meaning rule Equation 4.1 on Equation 4.72, Equation 4.73 is achieved.

$$\frac{D_1 \mid \equiv\ \rho k\,(CH, K_{D1}),\ \ CH \mid \equiv\ \Pi(K_{CH}^{-1}),\ \ D_1\ \lhd \{\ S\,(Y_6), CH\}K_{D_1}}{D_1 \mid \equiv\ CH \mid \sim Y_6} \qquad \text{Equation 4.73}$$

D_1 believes K_{D_1} is a public key of D_1 and is legitimate one which is also shared with CH. D_1 also believes that only CH have access to its private key and CH can only use its private key. D_1 believes that the message response is sent by CH with challenge, timestamp and onetime password. It is believed that only CH can sign the hash of the contents. Thus D_1 believes that CH created a message, hash and signed it with its private key. From the above discussion, it can be seen that D_1 believes that both keys are good, and the message was sent by CH, thus security goals are verified.

By applying freshness or nonce verification Equation 4.2 on statement 4.73, Equation 4.74 is achieved that shows that if D_1 believes in the freshness of the hashed message sent by CH and believes that the message is once sent by CH then D_1 has to believe in the message sent by the CH. From this equation, it is proved that goals are accomplished.

$$\frac{D_1 \mid\equiv\ \neq Y_6, D_1 \mid \equiv CH \mid\sim Y_6}{D_1\ \mid \equiv CH \mid\equiv Y_6} \qquad \text{Equation 4.74}$$

By applying jurisdiction rule Equation 4.3 on Equation 4.74, Equation 4.75 is accomplished. The equation states that if CH believes that D_1 have jurisdiction over the hashed message and it also believes that D_1 believes in that hashed message then CH has to believe in that message.

$$\frac{D_1 \mid \equiv\ CH \Rightarrow Y_6, D_1 \mid \equiv CH \mid\equiv Y_6}{D_1\ \mid \equiv\ CH \mid\equiv Y_6} \qquad \text{Equation 4.75}$$

From above different inference rules, it is observed that D_1 has full believe on authentication response message sent by CH which leads towards the authenticity of the message. The message Equation 4.72 contains response from CH. P_{D_1} is the public key of D_1, $Msg.res = Y6$ is the message response from CH. Pr_{CH} is the private key of CH and P_{CH} is the Public key of CH. It indicates that the message response is fully encrypted and is signed by CH. Moreover, the hash also contains the secret private key inside it even if the message is replayed by some attacker within the same timestamp, there will be no impact as the SP cannot be calculated by the attacker. If the message hash is not equal or message time is not valid the message will be discarded. VD_1' is device validation solution to verify that D1 is a legitimate device to get authentication from eNB, hence mitigating rogue relay and DoS attack.

4.4.15 Idealization of Authentication Response Message (CH-D2)

Equation 4.76 shows that the authentication response message is sent from CH to D_2. This message contains challenge, timestamp, and validation response from D_2 and hash of the message. The message is sent to D_2.

$$Y_7 = \begin{Bmatrix} \left\{ Ch_{eNB-D_2}\,,\ T_{S_{eNB}}\,,\ \left[DH\left(\,Ch_{eNB-D_2}\,',\ T_{S_{eNB}}\right)\right]_{Pr_{eNB}} \right\}_{P_{D_2}}\,, \\ Ch_{CH-eNB}\,,\ \ T_{S_{CH}}\,, VD_2', \left[DH\left(Ch_{CH-eNB}{}'\,,\ T_{S_{CH}}\,,\ VD_2'\right)\right]_{Pr_{CH}} \end{Bmatrix}_{P_{D_2}} \qquad \text{Equation 4.76}$$

After applying message meaning rule Equation 4.1 on Equation 4.76, Equation 4.77 is achieved.

$$\frac{D_2 \mid \equiv\ \rho k\,(CH, K_{D2}),\ \ CH \mid \equiv\ \Pi(K_{CH}^{-1}),\ D_2\ \lhd \{\ S\,(Y_7), CH\}K_{D_2}}{D_2 \mid \equiv\ CH \mid \sim Y_7} \qquad \text{Equation 4.77}$$

D_2 believes K_{D_2} is a public key of D_2 and is legitimate one which is also shared with CH. D_2 also believes that only CH have access to its private key and CH can only use its private key. D_2 believes that the message response is sent by CH with challenge, timestamp and onetime password. It is believed that only CH can sign the hash of the contents. Thus D_2 believes that CH created a message, hashed and signed it with its private key. From the above discussion, it can be seen that D_2 believes that both keys are good and the message was sent by CH, thus security goals are verified.

By applying freshness or nonce verification Equation 4.2 on statement 4.77, Equation 4.78 is achieved that shows that if D_2believes in the freshness of the hashed message sent by CH and believes that the message is once sent by CH then D_2 has to believe in the message sent by the CH. From this equation, it is proved that goals are accomplished.

$$\frac{D_2 \mid\equiv \neq Y_7, D_2 \mid \equiv CH \mid\sim Y_7}{D_2 \mid \equiv CH \mid\equiv Y_7} \qquad \text{Equation 4.78}$$

By applying jurisdiction rule Equation 4.3 on Equation 4.78, Equation 4.79 is accomplished. The equation states that if CH believes that D_2 have jurisdiction over the hashed message and it also believes that D_2 believes in that hashed message then CH has to believe in that message.

$$\frac{D_2 \mid\equiv CH \Rightarrow Y_7, D_2 \mid \equiv CH \mid\equiv Y_7}{D_2 \mid\equiv CH \mid\equiv Y_7} \qquad \text{Equation 4.79}$$

From above different inference rules, it is observed that D_2 has full believe on authentication response message sent by CH which leads towards the authenticity of the message. The message Equation 4.76 contains response from CH and signature. P_{D_2} is the public key of D_2, $Msg.res = Y7$ is the message response from CH. Pr_{CH} is the private key

of CH and P_{CH} is the Public key of CH. This shows that the message response is fully encrypted and is signed by the CH. Moreover, the hash also contains the secret private key inside it, even if the message is replayed by some attacker within the same timestamp, there will be no impact as the SP cannot be calculated by the attacker. If the message hash is not equal or message time is not valid the message will be discarded. VD_1' is device validation solution to verify that D1 is a legitimate device to get authentication from eNB, hence mitigating rogue relay and DoS attack.

It is seen from above all different inference rules, D1, D2 and CH has full believe on authentication response message and all its credentials specially secret key (OTP) that it is sent by the legitimate eNB which leads towards the authenticity and secrecy of the message. So, authentication goals 4.12, 4.13, 4.14, 4.15, 4.16 and 4.17 have been achieved and assumptions 4.24, 4.25, 4.26, 4.27 have been verified. LEMAP protocol is fully secure against MITM attack, replay attack, DoS attack, impersonation attack and rogue device attack. Thus, it can be concluded that the proposed LEMAP multi-hop authentication protocol has achieved secure mutual authentication and is fully secure against the given attacks.

4.5 Mathematical Analysis

In this section, mathematical analysis has been performed on LEMAP and other benchmark algorithms. Firstly, communication cost of proposed algorithm is calculated to find out whether proposed algorithm is lightweight as compared to other benchmark algorithms and it is found how much LEMAP is improved over benchmarks algorithms in communication cost. Authentication overhead of the proposed algorithm is also computed to find the overhead of messages over the network.

4.5.1 Total Communication Cost

To find total communication of the proposed algorithm, Capkun cost equation explained in Section 3.6 has been used. Communication cost is computed by considering the messages sent. The communication cost is computed for LEMAP and all three benchmark algorithms.

4.5.1.1 Total Communication Cost of LEMAP Protocol

In the proposed algorithm, D_1 wants to communicate with D_2. CH is Cluster Head which forwards request to eNB. Several messages flow from D_1 to D_2 through CH and eNB. To compute the total cost, this research considers all operations that contribute to secure communication. In the proposed algorithm LEMAP, there are five operations that are part of secure communication between D_1 and D_2 as presented in Chapter 3. The messages sent in LEMAP protocol consist of pseudo ID, challenge, double hash, timestamp and one time password that is denoted by SZ_{PID}, SZ_{CH}, SZ_{DH} , SZ_{TS} and SZ_{OTP} respectively. Communication cost for each message is calculated as below. Initial message is sent from D_1 to CH that contains request from D_1 for communication with D_2. The message sent includes pseudo IDs, challenge, timestamp, and double hash. Thus the total size of message will be sum of pseudo IDs, challenge, timestamp and double hash that is sum of $\{SZ_{PID}, SZ_{CH}, SZ_{TS}, SZ_{DH}\}$. Thus, the size of authentication message can be computed in Equation 4.80.

$$m_{L1} = 6 * \sum_{Au=1}^{k=1} Sz_{PID} + 2 * \sum_{Au=1}^{k=1} \{Sz_{CH} + Sz_{TS} + Sz_{DH}\} \quad \text{Equation 4.80}$$

Second message is sent from CH to eNB, that forwards request of D_1 for communication with D_2. The communication cost for message 2 is shown in Equation 4.81.

$$m_{L2} = 5 * \sum_{Au=1}^{k=1} Sz_{PID} + 2 * \sum_{Au=1}^{k=1} \{Sz_{CH} + Sz_{TS} + Sz_{DH}\}$$ Equation 4.81

Third message is response message from eNB to CH to verify CH first that includes OTP as well. The communication cost for message 3 is shown in Equation 4.82.

$$m_{L3} = \sum_{Au=1}^{k=1} \{Sz_{CH} + SZ_{OTP} + Sz_{TS} + Sz_{DH}\}$$ Equation 4.82

Fourth message is sent from CH to eNB where CH verifies OTP. The communication cost for message 4 is shown in Equation 4.83.

$$m_{L4} = \sum_{Au=1}^{k=1} \{Sz_{CH} + SZ_{OTP} + Sz_{TS} + Sz_{DH}\}$$ Equation 4.83

Message number five is a response message from eNB to CH where eNB sends OTP for D_1 and D_2. The communication cost for message 5 is shown in Equation 4.84.

$$m_{L5} = 3 * \sum_{Au=1}^{k=1} Sz_{PID} + 3 * \sum_{Au=1}^{k=1} \{Sz_{CH} + Sz_{TS} + Sz_{DH}\} + 2 * \sum_{Au=1}^{k=1} Sz_{OTP}$$

Equation 4.84

This is response message from CH to D_1 where CH forwards OTP for D_1. The communication cost for message number 6 is shown in Equation 4.85.

$$m_{L6} = 3 * \sum_{Au=1}^{k=1} Sz_{PID} + 2 * \sum_{Au=1}^{k=1} \{Sz_{CH} + Sz_{TS} + Sz_{DH}\} + \sum_{Au=1}^{k=1} Sz_{OTP}$$

Equation 4.85

This is response message from CH to D_2 where CH forwards OTP for D_2. The communication cost for message number 7 is shown in Equation 4.85 and summed up in Equation 4.86.

$$m_{L7} = 3 * \sum_{Au=1}^{k=1} Sz_{PID} + 2 * \sum_{Au=1}^{k=1} \{Sz_{CH} + Sz_{TS} + Sz_{DH}\} + \sum_{Au=1}^{k=1} Sz_{OTP}$$

Equation 4.86

In message number 8, D_1 sends a forwarding message to CH for eNB. The communication cost for message 7 is shown in Equation 4.87.

$$m_{L8} = 2 * \sum_{Au=1}^{k=1} \{Sz_{CH} + Sz_{TS} + Sz_{DH}\}$$

Equation 4.87

In ninth message, D_2 sends a forwarding message to CH for eNB. The communication cost for message 7.1 is shown in Equation 4.88.

$$m_{L9} = 2 * \sum_{Au=1}^{k=1} \{Sz_{CH} + Sz_{TS} + Sz_{DH}\}$$

Equation 4.88

This is request message from CH to eNB where CH requests validation of devices D_1 and D_2. The communication cost for message 8 is shown in Equation 4.89.

$$m_{L10} = 4 * \sum_{Au=1}^{k=1} \{Sz_{CH} + Sz_{TS} + Sz_{DH}\}$$

Equation 4.89

Tenth message is a response message from eNB to CH where eNB sends validation solution of devices D_1 and D_2. The communication cost for message number eleven is shown in Equation 4.90.

$$m_{L11} = 4 * \sum_{Au=1}^{k=1} \{Sz_{CH} + Sz_{TS} + Sz_{DH}\}$$

Equation 4.90

Eleventh message is a response message from eNB to CH containing validation solution which is forwarded to D_1. The communication cost for message number 11 is shown in Equation 4.91 and Equation 4.92.

$$m_{L12} = 2 * \sum_{Au=1}^{k=1}\{Sz_{CH} + Sz_{TS} + Sz_{DH}\}$$ Equation 4.91

Message number 12 is a response message from eNB to CH containing validation solution which is forwarded to D_2. The communication cost for the message is shown in Equation 4.25 and Equation 4.26.

$$m_{L13} = 2 * \sum_{Au=1}^{k=1}\{Sz_{CH} + Sz_{TS} + Sz_{DH}\}$$ Equation 4.92

After looking into above Equation 4.80 to Equation 4.92 and for Message 1 to message 10.1, cumulative computation of communication cost has been calculated as follows.

$$AuthM_L = a * \sum_{Au=1}^{k=5} Sz_{PID} + b * \sum_{Au=1}^{k=13}\{Sz_{CH} + Sz_{TS} + Sz_{DH}\} + a * \sum_{Au=1}^{k=5} Sz_{OTP}$$

Equation 4.93

Where a, b are constants

Equation 4.93 can be simplified as Equation 4.94

$$AuthM_L = a * \sum_{Au=1}^{k=5}\{Sz_{PID} + Sz_{OTP}\} + b * \sum_{Au=1}^{k=13}\{Sz_{CH} + Sz_{TS} + Sz_{DH}\}$$

Equation 4.94

That can be simply stated as

$$AuthM_L = \alpha * \sum_{Au=1}^{k=\beta}\{Sz_{PID} + Sz_{OTP} + Sz_{CH} + Sz_{TS} + Sz_{DH}\}$$ Equation 4.95

Equation 4.95 shows message for one hop communication where α shows the number of individual messages and β represents the number of messages transmitted. As shown that LEMAP can handle two hops communication but for this, an extra component is added that is the addition of forwarding request. So, for each hop, forwarding request Sz_{FR} will be added as shown in Equation 4.96

$$\mathrm{AuthM_L}(2) = 2h\left(\alpha * \sum_{Au=1}^{k=\beta} \{Sz_{PID} + Sz_{OTP} + Sz_{CH} + Sz_{TS} + Sz_{DH} + Sz_{FR}\}\right)$$

Equation 4.96

Where h is number of hops.

Similarly, for three hops the Equation 4.97 will be as below

$$\mathrm{AuthM_L}(3) = 3h\left(\alpha * \sum_{Au=1}^{k=\beta} \{Sz_{PID} + Sz_{OTP} + Sz_{CH} + Sz_{TS} + Sz_{DH} + Sz_{FR}\}\right)$$

Equation 4.97

Thus, for multi-hop scenario, the Equation 4.98 will be hop times the equation.

$$\mathrm{AuthM_L\ (h)} = \mathrm{h}\left(\alpha * \sum_{\mathrm{Au=1}}^{\mathrm{k=\beta}} \{\mathrm{Sz_{PID}} + \mathrm{Sz_{OTP}} + \mathrm{Sz_{CH}} + \mathrm{Sz_{TS}} + \mathrm{Sz_{DH}} + \mathrm{Sz_{FR}}\}\right)$$

Equation 4.98

For more than one hops, the number of messages transmitted is multiple of number of hops. In multi-hop scenario, total authentication cost is equal to number of hops multiplied by number of messages in single hop plus number of messages in forwarding request as shown in Equation 4.99.

$$\mathrm{AutM_{D2D}\ (h)} = \text{number of hops} * (\text{Single hop message} + \text{forwarding request})$$

Equation 4.99

4.5.1.2 Total Communication Cost of Chaotic Authentication Protocol

Chaotic map based user authentication is the first benchmark protocol that has also been used ECC for D2D communication (Zhang et al., 2016). In this protocol, user starts communication with server and requires authentication from the server. There are several messages exchanged between user and server to complete authentication. The communication cost of each message and total authentication cost is explained below.

Equation 4.100 shows that the first message contains five items ID of the device, password, biometric, random number, and hashed message. The Equation 4.100 shows the cost of this message.

$$\mathrm{M_{1Chaotic}} = \sum_{\mathrm{Au=1}}^{\mathrm{k=1}} (\mathrm{Sz_{ID}} + \mathrm{Sz_{PW}} + \mathrm{Sz_B} + \mathrm{Sz_b}) + 2 * \sum_{\mathrm{Au=1}}^{\mathrm{k=1}} (\mathrm{Sz_H})$$

Equation 4.100

Equation 4.101 shows that the second message containing three items random number, master key and hashed message. The cost of message is shown in Equation 4.101.

$$\mathrm{M_{2Chaotic}} = \sum_{\mathrm{Au=1}}^{\mathrm{k=1}}(\mathrm{Sz}_b + \mathrm{Sz}_{mk} + \mathrm{Sz_H}) \qquad \text{Equation 4.101}$$

Equation 4.102 shows that third message contains password, biometric, random number and hashed message. The cost of message three is shown in Equation 4.102.

$$\mathrm{M_{3Chaotic}} = \sum_{\mathrm{Au=1}}^{\mathrm{k=1}}(\mathrm{SZ_{PW_{,}}} + \mathrm{SZ_{\mathbf{b}}} + \mathrm{SZ_B} + \mathrm{SZ_H}) \qquad \text{Equation 4.102}$$

Equation 4.103 shows that message contains device ID, serial number of a smart card and random number.

$$\mathrm{M_{4Chaotic}} = \sum_{\mathrm{Au=1}}^{\mathrm{k=1}}(SZ_{ID} + SZ_{SN_i} + SZ_{\boldsymbol{b}}) \qquad \text{Equation 4.103}$$

Equation 4.104 shows that message five contains ID, password, biometric, random number, hashed message, and timestamp. The cost of message five is shown in Equation 4.104.

$$\mathrm{M_{5Chaotic}} = 2 * \sum_{\mathrm{Au=1}}^{\mathrm{k=1}}(SZ_{ID}) + \sum_{\mathrm{Au=1}}^{\mathrm{k=1}}(SZ_{PW} + SZ_b + SZ_B + SZ_{RN_u} + SZ_{TS}) + 4 * \sum_{\mathrm{Au=1}}^{\mathrm{k=1}}(SZ_H)$$

Equation 4.104

Equation 4.105 shows that message six contains ID, timestamp, random number and hashed message. The cost of message six is shown in Equation 4.105.

$$\mathrm{M}_{6\mathrm{Chaotic}} = 2 * \sum_{\mathrm{Au}=1}^{\mathrm{k}=1} (SZ_{ID} + SZ_H) + \sum_{\mathrm{Au}=1}^{\mathrm{k}=1} (SZ_{RN_u} + SZ_{TS} + SZ_b) \quad \text{Equation 4.105}$$

Equation 4.106 shows that message seven contains random number of users, random number for server, timestamp, session key and hashed message. The cost of message seven is shown in Equation 4.106.

$$\mathrm{M}_{7\mathrm{Chaotic}} = 2 * \sum_{\mathrm{Au}=1}^{\mathrm{k}=1} (SZ_{TS}) + \sum_{\mathrm{Au}=1}^{\mathrm{k}=1} (SZ_{RN_u} + SZ_{RN_s} + SZ_H + SZ_{SK})$$

Equation 4.106

Equation 4.107 shows that message eight contains random number, session key and hashed message. The cost of message eight is shown in Equation 4.107.

$$\mathrm{M}_{8\mathrm{Chaotic}} = 2 * \sum_{\mathrm{Au}=1}^{\mathrm{k}=1} (SZ_H) + \sum_{\mathrm{Au}=1}^{\mathrm{k}=1} (SZ_{RN_s} + SZ_{SK}) \quad \text{Equation 4.107}$$

Equation 4.108 shows that message nine contains random number of user, hashed message, timestamp and session key. The cost of message nine is shown in Equation 4.108.

$$\mathrm{M}_{9\mathrm{Chaotic}} = \sum_{\mathrm{Au}=1}^{\mathrm{k}=1} (SZ_H + SZ_{RN_u} + SZ_{TS} + SZ_{SK}) \quad \text{Equation 4.108}$$

Equation 4.109 shows that message ten contains session key. The cost of message nine is shown in Equation 4.109.

$$\mathrm{M}_{10\mathrm{Chaotic}} = \sum_{\mathrm{Au}=1}^{\mathrm{k}=1} (SZ_{SK}) \quad \text{Equation 4.109}$$

Total cost of protocol is sum of all messages.

$M_{TotalChaotic}=M_{1Chaotic}+M_{2Chaotic}+M_{3Chaotic}+M_{4Chaotic}+M_{5Chaotic}+M_{6Chaotic}+M_{7Chaotic}+M_{8Chaotic}+M_{9Chaotic}+M_{10Chaotic}$

Thus, by adding equations from Equation 4.100 to Equation 4.109.

$$\text{AuthM}_{\text{TotalChoatic}} = \sum_{Au=1}^{k} Sz_{ID} + \sum_{Au=1}^{k} Sz_{PW} + \sum_{Au=1}^{k} Sz_{B} + \sum_{Au=1}^{k} Sz_{H} + \sum_{Au=1}^{k} Sz_{b} + \sum_{Au=1}^{k} Sz_{mk} + \sum_{Au=1}^{k} Sz_{SN_i} + \sum_{Au=1}^{k} Sz_{RN_u} + \sum_{Au=1}^{k} Sz_{RN_s} + \sum_{Au=1}^{k} Sz_{SK} \quad \text{Equation 4.110}$$

The cost of total communication will be ten messages for key sharing. Now to perform multi-hop, the messages have to be transferred to eNB as messages approval are done through eNB. So, if the node is one hop away then it will be as shown in Equation 4.111. The square is made because both nodes will have to run the authentication scheme with eNB while the last h is the messages transferred from the sender device to destination device.

$$\text{TotalCost}_{\text{auth}}(2) = (2 * \text{AuthM}_{\text{TotalChoatic}}) + 3 \quad \text{Equation 4.111}$$

Similarly, for three hops the cost will be as follows. Equation 4.112 shows that messages will be tripled as each node will have to establish key exchange with eNB while the last h represents the final message transferred.

$$\text{TotalCost}_{\text{auth}}(3) = (3 * \text{AuthM}_{\text{TotalChoatic}}) + 3 \quad \text{Equation 4.112}$$

Thus, the cost of *n* hops will be *n* times as shown in Equation 4.113.

$$\mathrm{TotalCost_{auth}(n)} = (\mathrm{n} * \ \mathrm{AuthM_{TotalChoatic}}) + (\mathrm{n} + 3) \qquad \text{Equation 4.113}$$

4.5.1.3 Total Communication Cost of TwoFactor Authentication Protocol

Twofactor authentication Protocol is another benchmark protocol that has been presented for two factor authentication security of D2D communication along with key exchange (Wanjun Xion et al., 2018). The messages transferred between device and eNB, and communication cost for each message is shown below.

Equation 4.114 shows that the first message contains five items ID of the device, password, biometric, random number and hashed message.

$$\mathrm{M_{1F}} = \ \sum_{\mathrm{Au=1}}^{\mathrm{k=1}}(\mathrm{Sz_{ID}} + \mathrm{Sz_{PW}} + \ \mathrm{Sz_{RN}}) + 2 * \sum_{\mathrm{Au=1}}^{\mathrm{k=1}}(\mathrm{Sz_{H}}) \qquad \text{Equation 4.114}$$

Equation 4.115 shows that the second message contains five items ID of the device, password, smart card, random number and hashed message.

$$\mathrm{M_{2F}} = \ 2 * \sum_{\mathrm{Au=1}}^{\mathrm{k=1}}(\mathrm{Sz_{ID}}) + \sum_{\mathrm{Au=1}}^{\mathrm{k=1}}(\mathrm{Sz_{PW}} + \ \mathrm{Sz_{SC}} + \mathrm{Sz_{RN}}) + 7 * \sum_{\mathrm{Au=1}}^{\mathrm{k=1}}(\mathrm{Sz_{H}}) \qquad \text{Equation 4.115}$$

Equation 4.116 shows that the third message contains four items ID of the device, password, smart card, and random number.

$$M_{3F} = \sum_{Au=1}^{k=1}(Sz_{ID} + Sz_{PW} + + Sz_{SC} + Sz_{RN})$$ Equation 4.116

Equation 4.117 shows that this message contains ID of the device and hashed message only.

$$M_{4F} = \sum_{Au=1}^{k=1}(Sz_{ID} + Sz_{H})$$ Equation 4.117

Equation 4.118 shows that this message contains three items random number, session key and hashed message.

$$M_{5F} = \sum_{Au=1}^{k=1}(Sz_{RN} + Sz_{SK}) + 2 * \sum_{Au=1}^{k=1}(Sz_{H})$$ Equation 4.118

Equation 4.119 shows that this message contains items password, session key, random number, timestamp and hashed message.

$$M_{6F} = \sum_{Au=1}^{k=1}(Sz_{PW} + Sz_{SK} + Sz_{TS}) + 8 * \sum_{Au=1}^{k=1}(Sz_{H}) + 2 * \sum_{Au=1}^{k=1}(Sz_{RN})$$

Equation 4.119

Equation 4.120 shows that this message contains random number, timestamp and hashed message.

$$M_{7F} = 4 * \sum_{Au=1}^{k=1}(Sz_{H}) + SZ_{RN} + 3 * \sum_{Au=1}^{k=1}(Sz_{TS})$$ Equation 4.120

Equation 4.121 shows that this message contains ID of the device, timestamp and hashed message.

$$M_{8F} = 2 * \sum_{Au=1}^{k=1}(Sz_{ID} + SZ_{H} + Sz_{TS})$$ Equation 4.121

Equation 4.122 shows that this message contains ID of the device, timestamp, session key and hashed message.

$$\mathrm{M_{9F}} = 2 * \sum_{\mathrm{Au=1}}^{\mathrm{k=1}} (\mathrm{Sz_{ID}} + SZ_H + \mathrm{Sz_{TS}}) + SZ_{SK} \quad \text{Equation 4.122}$$

After looking into all above Equation 4.114 to Equation 4.122 starting from message 1-9, computation of communication cost has been added in Equation 4.123.

$$\mathrm{AuthM_F} = \sum_{\mathrm{Au=1}}^{\mathrm{k=6}} \mathrm{Sz_{ID}} + \sum_{\mathrm{Au=1}}^{\mathrm{k=4}} \mathrm{Sz}_{PW} + \sum_{\mathrm{Au=1}}^{\mathrm{k=2}} \mathrm{Sz_{SC}} + \sum_{\mathrm{Au=1}}^{\mathrm{k=8}} \mathrm{Sz}_H + \sum_{\mathrm{Au=1}}^{\mathrm{k=5}} \mathrm{Sz_{RN}} + \sum_{\mathrm{Au=1}}^{\mathrm{k=3}} \{\mathrm{Sz_{TS}} + \mathrm{Sz_{SK}}\} \quad \text{Equation 4.123}$$

Equation 4.124 shows cost for two hops.

$$\mathrm{TotalCost_{auth}}(2) = (2 * \mathrm{AuthM_F}) + 3 \quad \text{Equation 4.124}$$

Similarly, for three hops the cost will be as follows. Equation 4.125 shows that messages will be tripled as each node will have to establish key exchange with eNB while the last h represents the final message transferred

$$\mathrm{TotalCost_{auth}}(3) = (3 * \mathrm{AuthM_F}) + 3 \quad \text{Equation 4.125}$$

Thus, the cost of n hops will be n times as shown in Equation 4.126.

$$\mathrm{TotalCost_{auth}}(\mathrm{n}) = (\mathrm{n} * \mathrm{AuthM_F}) + (\mathrm{n} + 3)) \quad \text{Equation 4.126}$$

4.5.1.4 Total Communication Cost of 2PAKEP Protocol

Equation 4.127 shows that the first message contains items ID of the device, password, random number and hashed message.

$$M_{1P} = \sum_{Au=1}^{k=1} (Sz_{ID} + Sz_{PW}) + 2 * \sum_{Au=1}^{k=1} (Sz_{H} + Sz_{RN}) \quad \text{Equation 4.127}$$

Equation 4.128 shows that this message contains items ID of the device and hashed message.

$$M_{2P} = \sum_{Au=1}^{k=1} (Sz_{ID}) + 3 * \sum_{Au=1}^{k=1} (Sz_{H}) \quad \text{Equation 4.128}$$

Equation 4.129 shows that this message contains items ID of the device, smart card, private key of server and hashed message.

$$M_{3P} = \sum_{Au=1}^{k=1} (Sz_{ID} + SZ_{SC} + SZ_{ds}) + 2 * \sum_{Au=1}^{k=1} (Sz_{H}) \quad \text{Equation 4.129}$$

Equation 4.130 shows that this message contains items ID of the device, password, random number, timestamp and hashed message.

$$M_{4P} = \sum_{Au=1}^{k=1} (Sz_{ID} + SZ_{PW} + SZ_{RN} + SZ_{TS}) + 5 * \sum_{Au=1}^{k=1} (Sz_{H}) \quad \text{Equation 4.130}$$

Equation 4.131 shows that this message includes ID of the device, private key of server, random number, session key, time stamp and hashed message.

$$M_{5P} = \sum_{Au=1}^{k=1} (Sz_{ID} + SZ_{ds} + SZ_{RN} + SZ_{SK}) + 4 * \sum_{Au=1}^{k=1} (Sz_{H}) + 2 * \sum_{Au=1}^{k=1} (Sz_{TS})$$

Equation 4.131

Equation 4.132 shows that this message includes session key, time stamp, secure one-way derivation function and hashed message.

$$\mathrm{M_{6P}} = 2 * \sum_{Au=1}^{k=1}(Sz_H + SZ_{TS} + SZ_{SK}) + SZ_{kdf}$$ Equation 4.132

Equation 4.133 shows that this message includes session key, time stamp, secure one-way derivation function and hashed message.

$$\mathrm{M_{7P}} = \sum_{\mathrm{Au=1}}^{\mathrm{k=1}}(\mathrm{Sz_H} + SZ_{kdf} + SZ_{TF}) + 2 * \sum_{\mathrm{Au=1}}^{\mathrm{k=1}}(\mathrm{Sz_{SK}})$$ Equation 4.133

Equation 4.134 shows that this message includes ID of device, password, and hashed message.

$$\mathrm{M_{8P}} = \sum_{\mathrm{Au=1}}^{\mathrm{k=1}}(SZ_{ID_,} + SZ_{PW} + SZ_H)$$ Equation 4.134

Equation 4.135 shows that this message includes ID of device, password, and hashed message.

$$\mathrm{M_{9P}} = \sum_{\mathrm{Au=1}}^{\mathrm{k=1}}(SZ_{ID_,} + SZ_{PW}) + 4 * \sum_{\mathrm{Au=1}}^{\mathrm{k=1}}(SZ_{H_,})$$ Equation 4.135

Equation 4.136 shows that this message includes ID of device, password, smart card and session key.

$$\mathrm{M_{10P}} = \sum_{\mathrm{Au=1}}^{\mathrm{k=1}}(SZ_{PW} + SZ_{SC} + SZ_{RN} + SZ_{SK})$$ Equation 4.136

After looking into all the above Equations starting from 4.127 - 4.136 for Message 1-10, for computation of communication cost Equations 4.127 to Equation 4.136 have been selected.

$$\mathrm{AutM_{PTotal}} = \mathrm{M_{1P+}M_{2P}} + \mathrm{M_{3P}} + \mathrm{M_{4P}} + \mathrm{M_{5P}} + \mathrm{M_{6P}} + \mathrm{M_{7P}} + \mathrm{M_{8P}} + \mathrm{M_{9P}} + \mathrm{M_{10P}}$$

Equation 4.137 shows total communication cost of each message is calculated.

$$\mathrm{AutM_{PTotal}} = \sum_{\mathrm{Au=1}}^{\mathrm{k=6}} \mathrm{Sz_{ID}} + \sum_{\mathrm{Au=1}}^{\mathrm{k=4}} \mathrm{Sz_{PW}} + \sum_{\mathrm{Au=1}}^{\mathrm{k=2}} \mathrm{Sz_{SC}} + \sum_{\mathrm{Au=1}}^{\mathrm{k=9}} \mathrm{Sz_{H}}$$

$$+ \sum_{\mathrm{Au=1}}^{\mathrm{k=4}} \mathrm{Sz_{RN}} + \sum_{\mathrm{Au=1}}^{\mathrm{k=4}} \mathrm{SZ_{TS}} + \sum_{\mathrm{Au=1}}^{\mathrm{k=4}} \mathrm{SZ_{SK}} + \sum_{\mathrm{Au=1}}^{\mathrm{k=2}} \mathrm{SZ_{kdf}} + \sum_{\mathrm{Au=1}}^{\mathrm{k=2}} \mathrm{SZ_{ds}} \qquad \text{Equation 4.137}$$

Total cost for two hops is calculated as in Equation 4.138.

$$\mathrm{TotalCost_{auth}}(2) = (2 * \ \mathrm{AuthM_{2PAKEP}}) + 3 \qquad \text{Equation 4.138}$$

Similarly, for three hops the cost will be as follows. Equation 4.139 shows that messages will be tripled as each node will have to establish key exchange with eNB while the last h represents the final message transferred.

$$\mathrm{TotalCost_{auth}}(3) = (3 * \ \mathrm{AuthM_{2PAKEP}}) + 3 \qquad \text{Equation 4.139}$$

Thus, the cost of *n* hops will be *n* times as shown in Equation 4.140.

$$\mathrm{TotalCost_{auth}}(\mathrm{n}) = (\mathrm{n} * \ \mathrm{AuthM_{2PAKEP}}) + (\mathrm{n} + 3) \qquad \text{Equation 4.140}$$

4.5.2 Computational Overhead

To compute the computational overhead this research used base D2D testing model already used for benchmarking protocols (Chaotic, TwoFactor, and 2PAKEP). In order to compute, initially the cost was computed for a single hop and later the same model was applied on multi-hop scenario. The total communication cost is based on the communication costs computed in Section 4.3.1, Section 4.3.2, Section 4.3.3 and Section 4.3.4. The computer costs are then compared to find the scheme that bears least traffic over the network.

Table 4.2: Total Communication Overhead

Security Algorithm	No of Hops									
	2	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**
LEMAP	74.9	149.8	224.7	299.6	374.5	449.4	524.3	599.2	674.1	749
Chaotic	232.5	310	387.5	465	542.5	620	697.5	775	852.5	930
TwoFactor	262.5	350	437.5	525	612.5	700	787.5	875	962.5	1050
2PAKEP	220.2	293.6	367	440.4	513.8	587.2	660.6	734	807.4	880.8

In Table 4.2, detailed calculation of eleven hops is shown. It is observed that the communication cost increases with the increase of number of hops for all D2D security algorithms. Figure 4.13 shows in detail the effect of increasing communication cost with number of hops. Figure 4.1 shows that our proposed algorithm LEMAP has the least communication cost as compared to other benchmark protocols. The third protocol 2PAKEP is some better in terms of communication cost but its cost is twice more than LEMAP. While Chaotic and Two Factor protocols have much higher communication cost as compared to LEMAP. By looking at Figure 4.14, we conclude that LEMAP has the lowest communication cost as compared to other benchmark protocols. This results in lower traffic load on network and makes D2D communication easier in all types of networks.

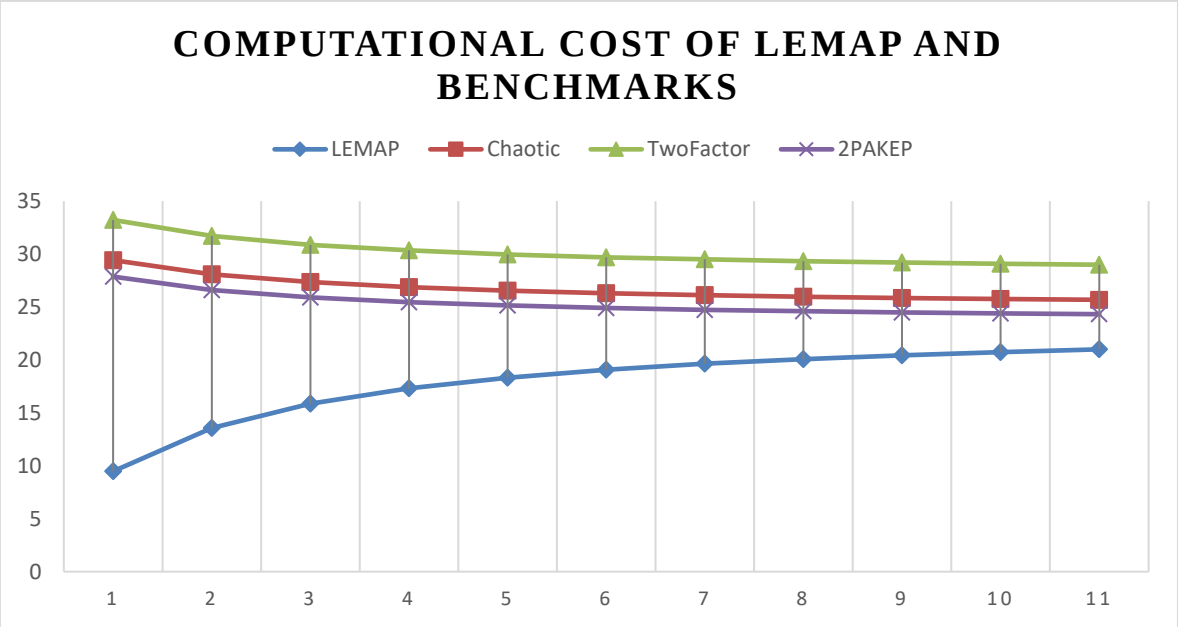

Figure 4.13: Effect of Increasing Hops on Computational Cost

4.5.3 Authentication Overhead

There are many things to be considered in secure D2D communication but one of the key concerns is authentication overhead which means how many messages are sent before actual transmission. Authentication overhead is not part of actual communication, but it is required for authentication of secure authorization and communication. Almost every security algorithm generates authentication overhead. LEMAP has been designed to get least possible authentication overhead. All of them are reflected in Figure 4.14.

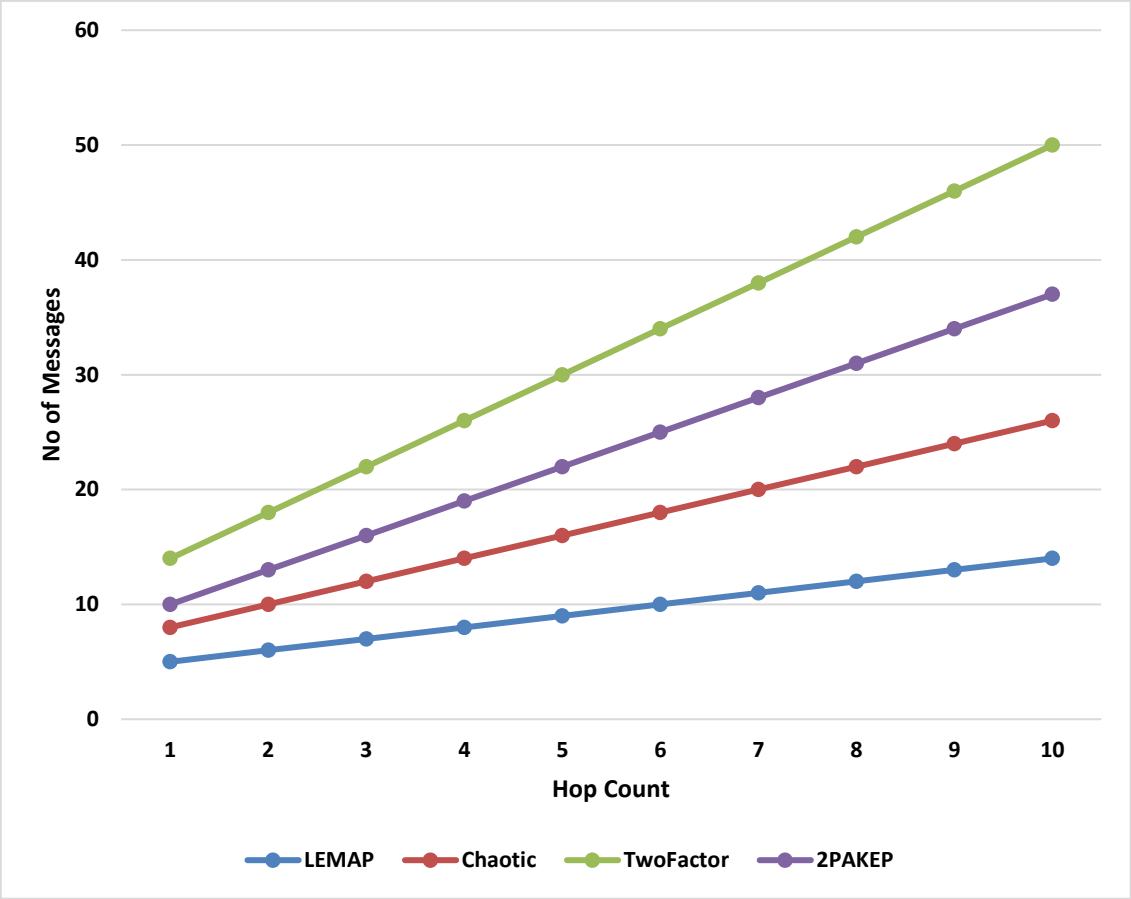

Figure 4.14: Authentication Overhead with Multi-hop

The overhead of LEMAP is $5 + h$ where h is a number of hops. For Chaotic the authentication overhead is $8 + 2(h)$. In Chaotic algorithm, the number of messages will be eight for each node. To find the authentication overhead of TwoFactor protocol, the authentication overhead is $14 + 4(h)$, where it is maximum due to extra sharing of messages. The 2PAKEP benchmark authentication overhead is $10 + 3(h)$. To understand the effect, Figure 4.13 shows the effect of increasing hop on several authentication messages that is drawn on the basis of the authentication overhead of LEMAP and other security techniques.

4.6 Simulation Analysis of LEMAP Protocol

The proposed protocol LEMAP is a distributed non-transparent relay protocol for D2D communication that is why this research adapted NCTUns discrete event simulator. For proof of study LEMAP protocol has been implemented using IEEE802.16j topology network as shown in Figure 4.15.

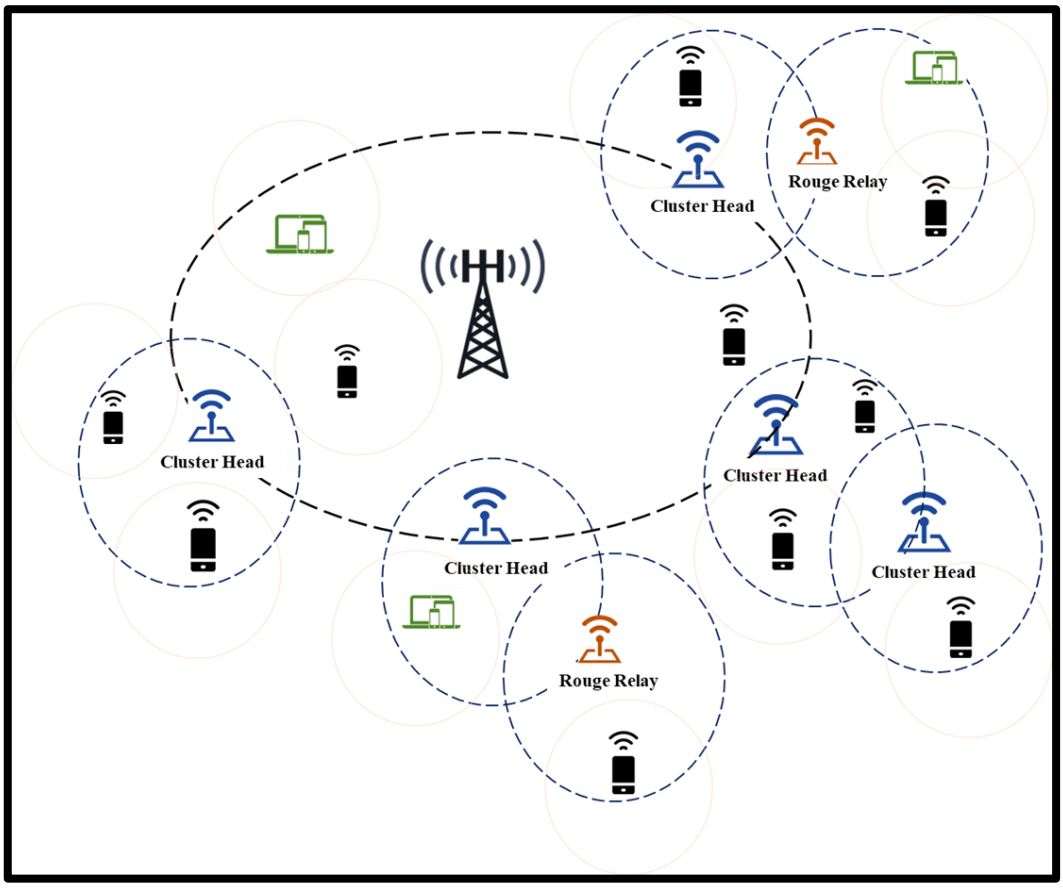

Figure 4.15: Simulation Setup of LEMAP and other Benchmarking Protocols

Figure 4.15 shows a setting of tested simulation in NCTUns 6.0 version where there is a replay attacker as well as multiple devices communicating in a setting with each other as D2D under jurisdiction of one eNB. The performance study has been conducted in NCTUns to find the effect of Packet Delivery Ratio, Packet Overhead and Processing times

while comparing it with increasing the number of hops and non-transparent relays with and without presence of attackers. For simulation four protocols LEMAP, TwoFactor, Chaotic and 2PAKEP were analysed and tested. While implementation of devices, this research considers the relaying devices as mobile devices having 1GHZ processor with 500 MB RAM and 4GB ROM. This research does not include battery power or time as it was not considered as part of this study. The details are shown in Table 4.3.

Table 4.3: Device / Relaying Node Computational Parameters

Device Properties	**Size/Processing**
CPU	1GHZ
Cores	1
RAM	1GB
ROM	4GB
Cache	100 MB embedded with Processor
Battery Time	direct power

Each device has similar capabilities and even the attacking nodes to simplify the analysis, the attacking result thus may be affected if the illegitimated devices are high computation power devices. Each algorithm while their implementation within this research comparison considers the following properties and sizes to have fair comparison as shown in Table 4.4.

Table 4.4: Message Sizes Declaration of Different Message Parts

Communicating Message Part	Size
Hashing (SHA - V3)	512 bits
Message	4096 bits
Timestamp	512 bits
Nonce/ Random Number	128 bits
Pseudo-Identifier	512 bits
Identifier	128 bits
Private key	512 bits
Public key	512 bits
Smart Key Identifier	512 bits
Biometrics	1026 bits
XOR of two Data	512 bits
Session Key	512 bits

This research selected NCTUns as explained for computation of results. In order to compute the results each execution or experimental setup logs were maintained. The log file contains the nodeID that sent the packet, time when the packet was sent, the destination ID and packet number. The rogue relays were using poison distribution in their behaviour so the packets that were not forwarded were random. Rogue relays were only present in case of with attacker scenario. To compute the time to receive a packet, the time when packet was sent and time when packet was received. If there is a missing packet number at receiver and

it was sent from destination it is considered as packet drop. For computation of overhead the packet size was defined already based on IEEE 802.11 and allowed permission in simulator. The computation time to create packet was logged along with packet number and total time to compute the packet. The following section provides a complete details about the achieved results.

4.6.1 Packet Delivery Ratio without Attacker

Figure 4.16 shows the effect of packet delivery ratio (PDR) without attacker in the network. Result shows that the proposed protocol LEMAP has the highest packet delivery ratio as the number of packets increase per message. In figure 4.16, packet delivery ratio is set as per million bit/second. 2PAKEP protocol PDR falls as number of packets are increased. TwoFactor protocol's PDR further lowers with increase of packets per message and the Chaotic scheme shows the lowest PDR among all protocols. In LEMAP, number of packets sent per message are smaller in size and require low computation while other two schemes 2PAKEP and TwoFactor have slightly lower PDR due to the computational complexity and new hash generation for forwarding message.

It is seen as the number of nodes increased, the PDR drops around 0.014% which is because of increase in overhead caused by devices and nodes verification time. LEMAP as stated earlier perform better than 2PAKEP by around 0.07 millisecond and perform better than TwoFactor by 0.02 millisecond. The PDR of Chaotic is lower by 0.05 millisecond as compared to LEMAP. This change in difference is caused because of number of changes in LEMAP as compared to other protocols such as embedding the acknowledgement inside normal messages and multi-factor verification at once. This difference is quite significant as more nodes adds to the communication and request for data shown as packet rate. This may

reach a difference of 0.92 millisecond for Choatic where Chaotic takes more time. Secondly, TwoFactor takes slightly higher time than LEMAP that is 0.76 millisecond. Least difference is of 2PAKEP which takes 0.07 millisecond lower than TwoFactor due to short message size. Overall, the whole algorithm performs better in ideal situation when there is no security breach.

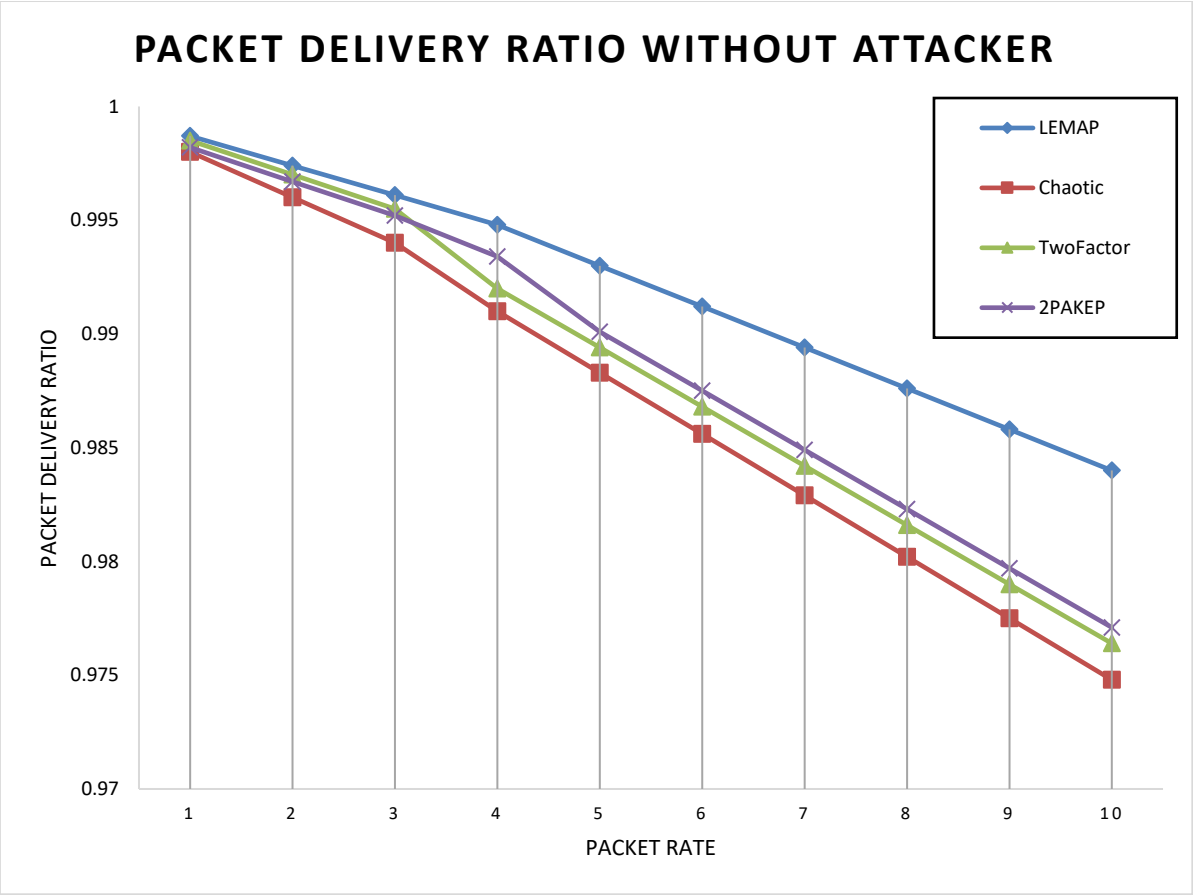

Figure 4.16: PDR without Attacker for Proposed and Benchmarking Protocols

4.6.2 Packet Delivery Ratio with Attacker

When attacker enters the network, the performance of LEMAP is still better than other benchmark protocols as it does not allow any MITM and replay attack and thus have higher throughput as compared to TwoFactor and 2PAKEP which do not handle replay

attack as shown in Figure 4.17. And the packet rate is still lower than packet delivery ratio without attacks. Here attackers are considered as rouge devices or nodes that are jeopardizing the communication by not forwarding the communication, re-authentication request and non-receipt of packet request.

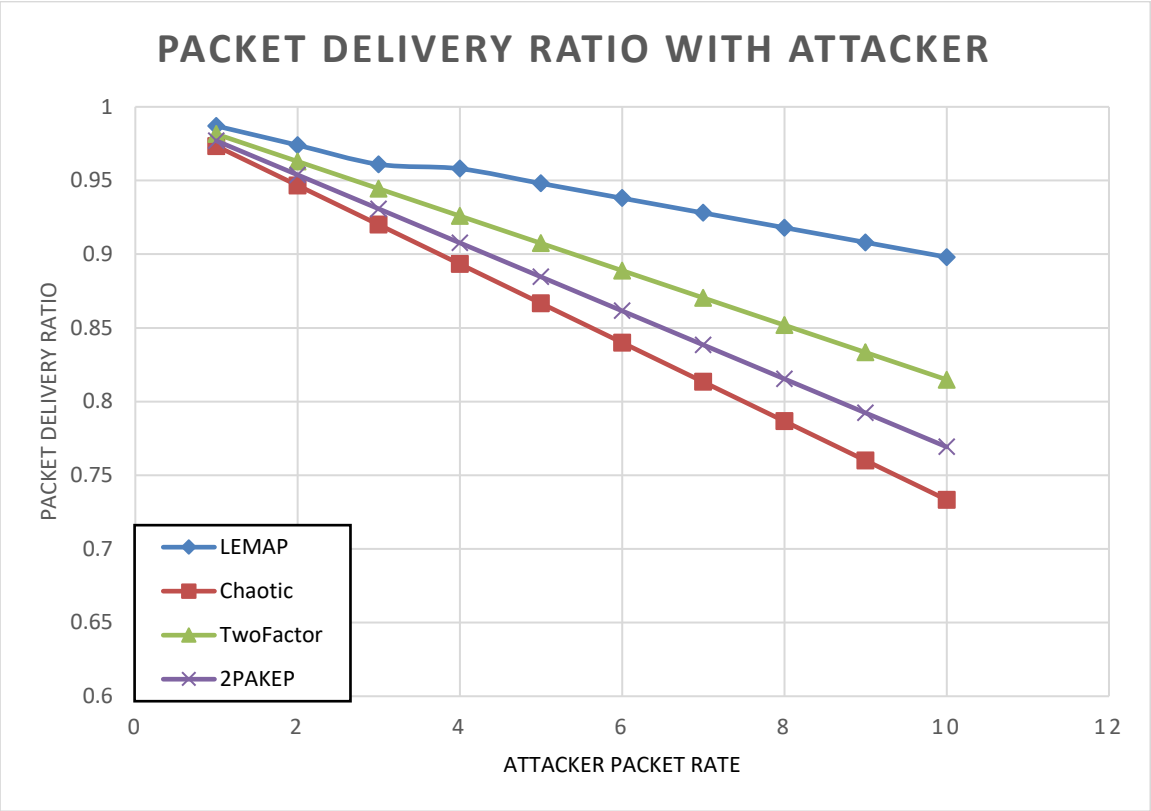

Figure 4.17: PDR with Attacker for Proposed and Benchmarking Protocols

It is observed that all benchmarks and the proposed algorithm has a slight effect on PDR such as Chaotic is 0.13 millisecond slower than LEMAP while TwoFactor is slower than Chaotic by 0.042 milliseconds. Least difference is with 2PAKEP which is 0.01 milliseconds. When the illegitimate packets get higher in number, the PDR drops which is around maximum for Chaotic that takes 0.31 milliseconds higher time than LEMAP because of non-availability of certification validation option. Second highest difference is with 2PAKEP that is around 0.272 milliseconds. TwoFactor takes 0.26 milliseconds higher than

LEMAP. The PDR drop is significant in other benchmarks due to re-verification request, acknowledgment/NACK at one packet and trust. LEMAP itself only have effect of 0.08 millisecond that keeps the PDR above 87% achievement level that can help it in selection as one of potential candidate for D2D communication.

4.6.3 Effect of Packet Overhead

Packet Overhead is total time taken to send packets over network that is time taken from source to the destination. In Figure 4.18, it is obvious from simulation results that proposed protocol LEMAP performs better compared to other benchmark protocols. In LEMAP, initial packets are hello packets which are smaller in size so they can be transmitted easily. LEMAP scheme shows better approach as compared to 2PAKEP and Chaotic.

The quick drop shows several authentication messages starting to validate the authentication. Complete diagram is shown in Figure 4.18. The packet overhead of LEMAP is lower as it reduces the packet size as compared to other schemes with added security and mitigation of various attacks. The Packet Overhead is the highest with TwoFactor which is 6.875% while 2PAKEP has the lowest packet overhead that is 2.675% higher than LEMAP. Chaotic takes 4.84% higher than LEMAP. All schemes have lower packet overhead as they use hash and drop all the illegitimate packets. Secondly, LEMAP uses double hash scheme of multi-factors that make it slightly better than other schemes in catching the illegitimate packets.

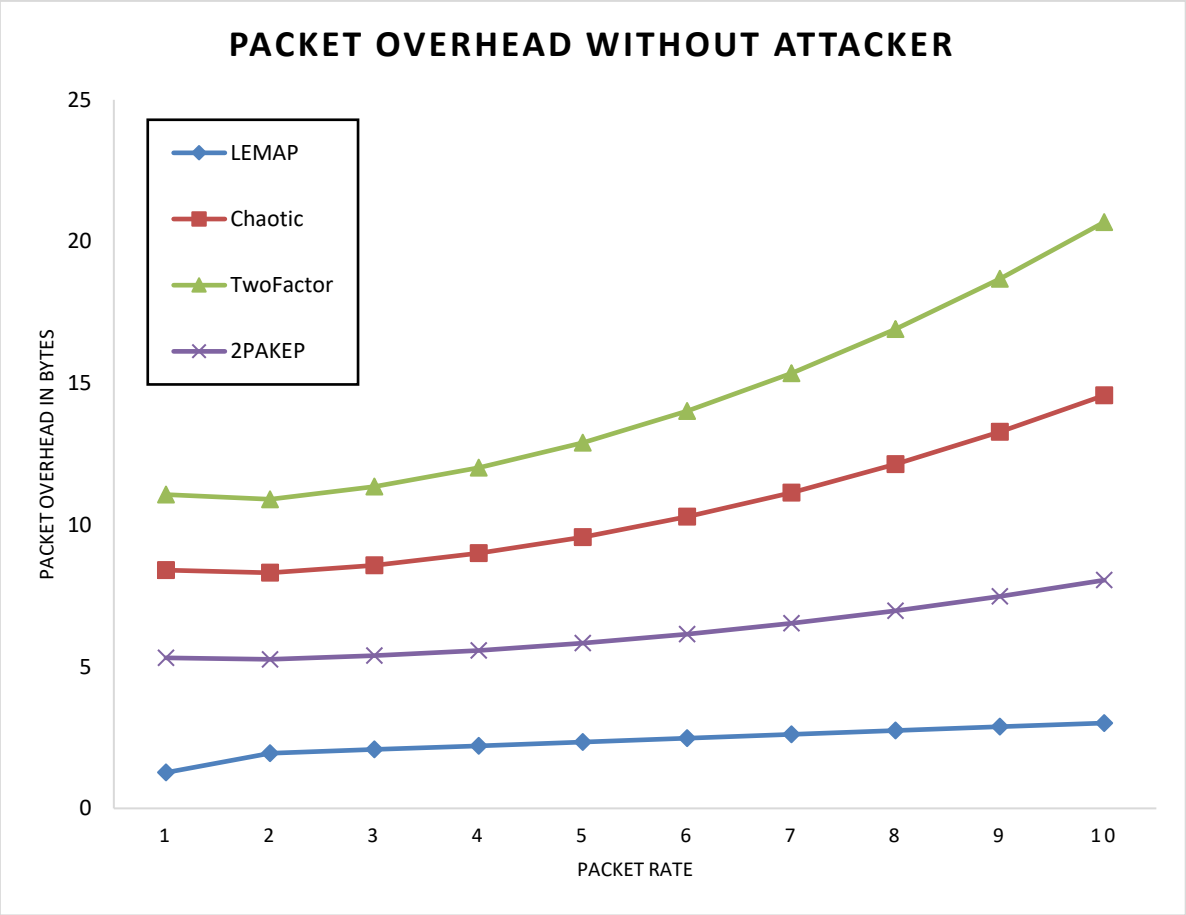

Figure 4.18: PO without Attacker for Proposed and Benchmarking Protocols

If an attacker enters the network, the packet overhead is lower as the multi-factor authentication scheme allows the LEMAP to skip the packet exchange due to introduction of multi-factor. If the attacker succeeds in its attack, the attacking node will be blocked. Results of packet overhead with attacker is shown in Figure 4.19. When the attacker is introduced all schemes can mitigate the attacks with usage of hash and timestamp but still the packet drop will cause an extra packet that results in increase of packet overhead. The packet overhead further increases with introduction of attacker by 1.54% in LEAMP as compared to without attacker approach. Highest packet overhead is with TwoFactor that is 9.1% as compared to LEMAP approach due to increase in packet drop rate as well as creation of re-authentication packets. Second highest packet overhead is 6.0% of Chaotic as

compared to LEMAP while 2PAKEP has 4.18% higher packet overhead as compared to LEMAP. LEMAP uses trust validation, multi-factor authentication and double hash which lead to a slight improvement in performance of the proposed algorithm. Current schemes also provide security at same level but packet size increases significantly that causes lower PDR and higher computation cost.

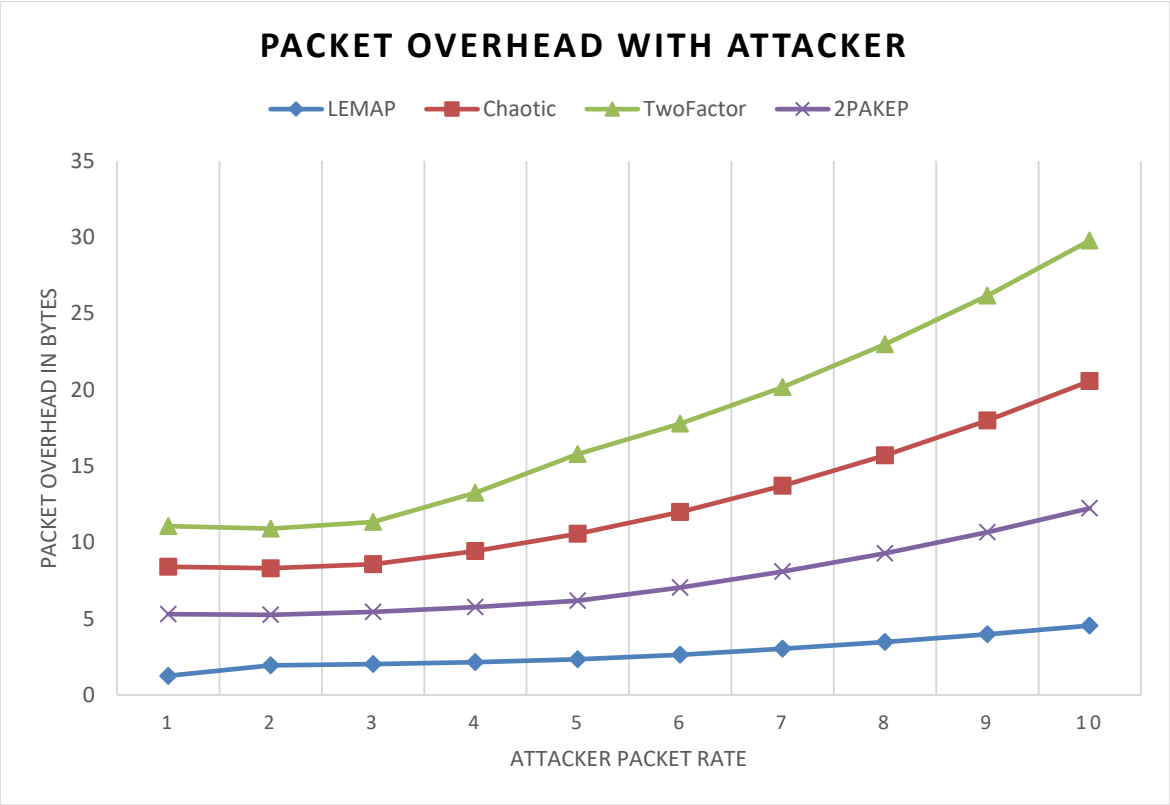

Figure 4.19: PO with Attacker for Proposed and Benchmarking Protocols

4.6.4 Comparison of Processing Time

Processing time in security algorithm is an important parameter as it is directly related to delivery rate or ratio. If processing time is higher, it can result in higher processing cost as well as will be difficult to be adapted in small scale devices. From simulation analysis, processing time of proposed protocol LEMAP is far lower than other benchmark algorithms. Processing time of LEMAP remains lower even the number of packets is increasing. It is

better than 2PAKEP in terms of lower processing cost while other benchmark algorithms have higher processing cost. The processing cost is fully dependant on computation operations, encryption or decryption operations and time to compute signature. The complete graph is shown in Figure 4.20. The processing time of LEMAP is lower as compared to 2PAKEP by 6.9%, while chaotic takes 22.37% more time as compared to LEMAP. The time taken by other algorithm is higher due to their increased data size, separate processing of hash, timestamps and key exchange. The maximum processing time is taken by TwoFactor that is 28.4% due to extra processing of two hashes.

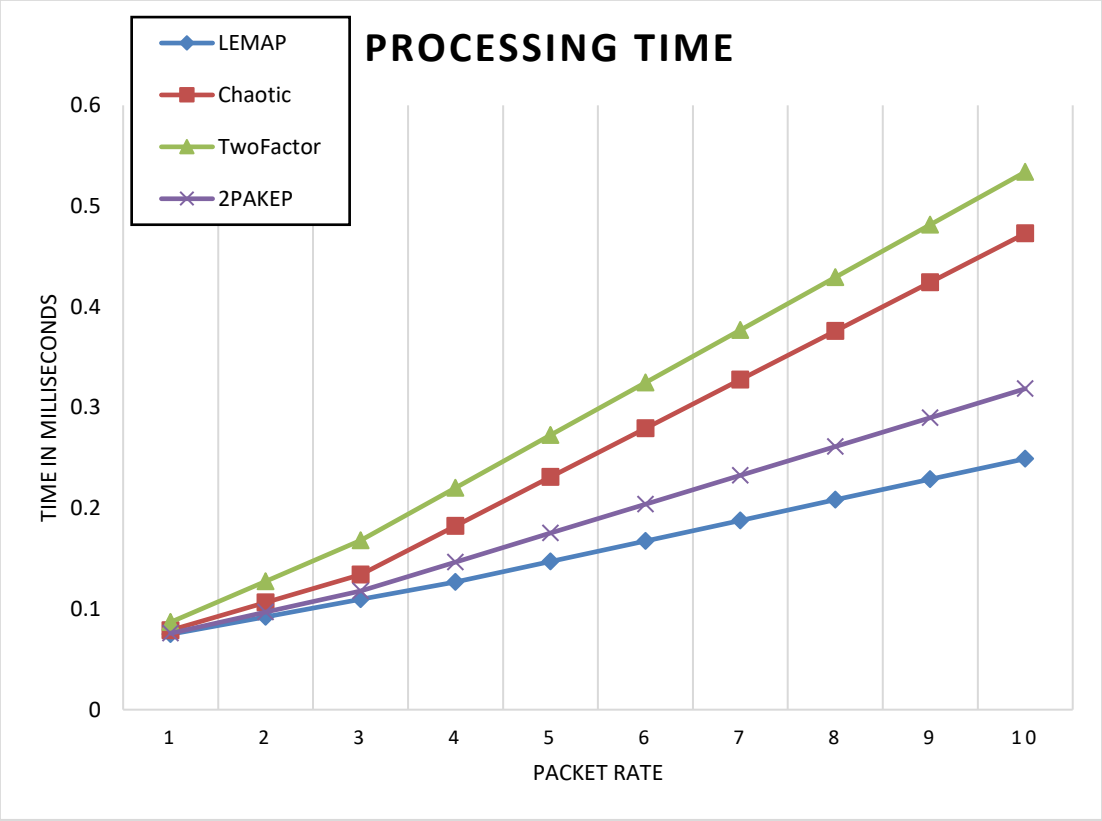

Figure 4.20: Processing Time for Proposed and Benchmarking Protocols

4.6.5 Effect of Increasing Rogue Relay Station

As mentioned in literature, with the increase of number of rouge relays, the number of attacks also increase and when rouge relays number increases as compared to legitimate

relays, the traffic cannot be transmitted which may result near to DoS attack. LEMAP still ensures that no illegitimate traffic can pass through the network due to prior registration and mutual authentication. The effect of rouge relays and PDR is shown in Figure 4.21. The delay in LEMAP scheme is due to reverification of nodes and sending the verification message again.

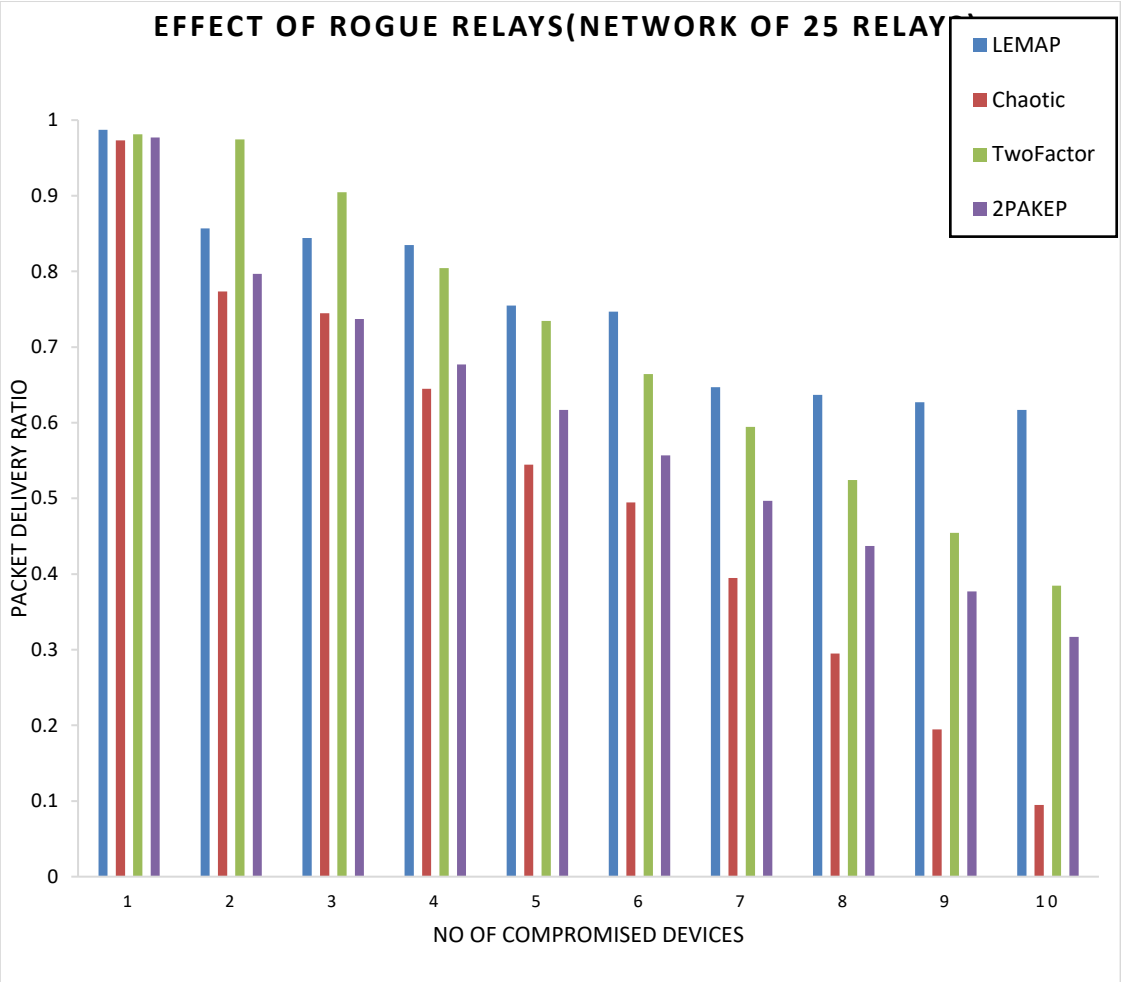

Figure 4.21: Effect of Increasing Rogue Relays in terms of Traffic Simulation

In this experiment we considered total 25 nodes out of which maximum 10 nodes are compromised. This research did not check the effect of rouge relays above 10 as one of the benchmarks, Chaotic PDR has dropped to non-acceptable level. There is significant drop in

LEMAP PDR that is around 50% but this experiment considered only 25 nodes in the experiment out of which 40% are compromised. This is better as trust and validation model is embedded in proposed algorithm while another benchmark lacks this feature. The only acceptable level is at 25% for other benchmarks where chaotic has PDR of 49.47%, 2PAKEP has PDR to 55.62% and TwoFactor has PDR of 66.48% while LEMAP has PDR of around 75%. Later, the drop in PDR of these benchmarks as already explained due to no feedback on transmission and non-consideration of more than two or three nodes being compromised.

4.6.6 Discussion on Computational Security Analysis

The proposed protocol LEMAP validates that it is lightweight than other benchmark protocols. Packet delivery ratio of LEMAP protocol is better than other protocols in both situations either the network is without attacker or the attacker is present in the network but still LEMAP results in high data rate and increased performance. Results have proved that packet overhead of LEMAP is also better than other protocols due to hello packets of smaller size. Processing time of LEMAP is also proved that it is lower than other protocols even with increased number of packets. But in case of vulnerable situation and normal traffic, one of the other benchmark protocols perform better that is TwoFactor. It provides PDR up to 73% when there are around 25% of rouge relays but as the percentage of rouge nodes increases, the effect on transmission failure increases. One of the reasons is that trust validation and certificate are not part of current security schemes especially in selected benchmarks. Secondly, this research considers that devices can be compromised in bulk as compared to existing security schemes that only considers two devices communication with third device acting as an attacker. Authentication overhead of LEMAP is the lowest and hence it can be adapted even in case of vulnerable situation such as increase in rouge relays or DoS attacks.

4.7 Security Analysis

As presented in Chapter 3, the algorithm must be secure enough to work against any intelligent as well as brute force attack. This section will check LEMAP against well-known computation attacks. DLP is one of the kinds of trapdoor function that is easy to calculate but extremely hard to get back the original. DLP is proven computationally challenging than factorization problem used in RSA or DH algorithms (Gupta, K., & Silakari, 2011; Steinfeld & Zheng, 2000). LEMAP falls into a category where finding the key is extremely hard. LEMAP is based on ECC cryptography where the key size is proposed to be 512 bits for the key selection between devices and eNB while the session key is considered to be 384 bits. It is proved that LEMAP has lower authentication overhead as compared to other protocols. ECC is based on finite cyclic group F_c for a primitive element α and another primitive element β where both α and $\beta \in F_c$. DLP is finding the integer k where k satisfies the following criteria $\alpha^k \equiv \beta$ or $k = \log_\alpha \beta$. The above criteria are referred to as DLP. Now in terms of ECC, we need to find multiplicative inverse k while α, β and F_c. There are several ways to conduct DLP check. This research uses BFA, Pollard's rho method and BsGs method.

4.7.1 Brute Force Attack

In BFA, we must find the *K* time point multiplication with the base point F_c such that α is achieved. While the Elliptic curve works on an elliptic equation that makes the rotation with *K*, so the complexity becomes more when the key is rotated β times. Even if this communication is cracked, the proposed algorithm uses a session-based encryption using the same algorithm making the cracking to be done for each session. Thus, the complexity will

add up for each session and vice versa; if one of the sessions is hacked (that is not possible) so only the session communication may become compromised and not the rest of sessions.

Usually, the attacker will start with $K = 1$ then $K = 2$ and so on. If the size of $K = 4$, bits then about after 4096 tries we can find K. But for our case, as the key is 512 bits large so it is practically impossible to conduct attack as required possible attempts will be 2^{15360}. Or even attack 384 bits that means will be 2^{7680} cannot be performed. Suppose we have world fast supercomputer that is IBM AC922 costing 200 million USD (Vetter, Nohria, & Santos, 2019). The speed of single processing on this processor is 2^{50} per second as explained around 3 billion cycles to be executed. Thus, around 1536 years will be required to do the cracking process or we can buy 1536 computers to crack the key in one year. We have set the refresh time to be one week for the main key while for the session the smaller key will change after each session. This proves that brute force attack is highly expensive (around 3 trillion USD) that is almost impossible to conduct. Moreover, we have taken the assumption that each key can be computed in one second while it requires more computation for higher key size.

4.7.2 Pollard's Rho Method

It is a better and intelligent way to attack the setup as it supports parallelization and random walk. Pollard's rho method reduced the permutations by a square root. So only possible keys will reduce the number of permutations by one equation to 2^{15359}, hence requiring almost the same time as the BFA. The Pollard rho method will fail here also. LEMAP is a generic algorithm and allows the key size to be increased according to the security required such as if the key size is 1024 then we have almost $2^{1966080}$ combinations that will require above 1.9 million years.

4.7.3 Baby Step Giant Step

Baby Step Giant Step (BsGs) is also a method that required an intelligent approach to crack the security key in two directions. It also reduced the number of efforts to $\sqrt{N}$ times thus making it half the size. In BsGs, the task is to find k where k > $\sqrt{\#\beta}$ and compute $\alpha = g^k$ where g is from 0 to k. As still, it is computationally expensive even after $\sqrt{\#\beta}$, it will just reduce the efforts by half providing surety to have complexity such as DLP.

4.7.4 Keyspace

A keyspace is a way to store all possible permutation of pair and then just compare the results. As mentioned in Section 4.6.2, we are considering that key calculation takes a second, so the validity holds true for the key space. This means we require to 215359×29, which is equal to 215368 bits or 215365 bytes to store the key or 215347 petabytes minimum, and making it extremely complex for parallel processing even through IBM AC922. The comparison will require still the same amount as calculated in Section 4.6.3 as a minimum.

4.7.5 Key size

Key size is an important feature of ECC that makes it extremely usable in the current era where supercomputers can compute billions of computations per second. ECC is based on elliptic curves and thus their key size complexity is far higher than that of RSA and DH. Table 4.5 shows the detailed comparison of the key size of ECC as compared to symmetric key as well as RSA and DH. Table 4.5 is made on values reported by (Javed et al., 2017) (Bos et al., 2009).

Table 4.5: Key Size Comparison

RSA/DH	ECC	AES (symmetric)
1024	160	80
2048	224	112
3072	256	128
7680	384	192
15360	512	256
1966080	1024	512

It is clear from Table 4.5 that ECC consumes lesser space than traditional asymmetric algorithm and double the space than symmetric algorithm. Thus, ECC provides the best approach to date to use for small devices. As storage, processing and security of ECC are highly better than any other security algorithms in practice.

4.7.6 Processing Cost

The major cost in all ECC based algorithms is pairing, multiplication and encryption as mentioned by (Wang et al., 2017; Zhang et al., 2017; Kim et al., 2016; Zhang et al., 2016). Thus, we will only consider these operations as a major contributor to computations. In Section 4.6, we have considered the world's fastest and most powerful supercomputer for the attack but in practice, the purpose of LEMAP is to be executed on small devices such as mobile phones or small computing devices. Thus in order to consider the computation cost or processing cost we will use benchmark device power such as used by (Vetter et al., 2018; Zhang et al., 2016) where each implementation was executed on Intel 3.0 GHz Pentium processor. The curve used in the implementation was of 6 degrees with 160-bit size. The

processing time took around 4.5 ms on average for pairing while 0.6 ms for multiplication. Approximately, for a 1 GHZ processor, the time taken will be 13.5 ms as 4.5 * 3 over 1 GHz is 13.5 ms. The time to compute the exponential computation is 1.8 ms for a 1 GHz processor. For ECC the encryption and decryption are also multiplication thus it will be a multiple of multiplication steps. While in case of symmetric encryption, it takes 0.5 ms for performing the encryption on a 1 Ghz processor. Let t_p the cost of pairing while t_m the cost of multiplication and t_e be the cost of encryption as shown in Table 4.6.

Table 4.6: Processing time of Computing Operations

Operation	**Notation used**	**Time required in milliseconds**
Encryption symmetric	t_e	0.5 ms
Encryption asymmetric ECC	t_m	1.8 ms
Multiplication encryption	t_m	1.8 ms
Pairing time	t_p	13.5 ms

For Chaotic protocol, there are several operations required such as one exponential computation and encryption operation thus requiring $23t_h + 4t_m + 5t_p + t_e$ in time. There are twenty-three hashing operations requiring $23t_h$ in time. For key generation, there are encryption using public and private keys of the session thus requiring $4t_m$ in time. For pairing five operations are required $5t_p$.Thus in total, it will be $23t_h + 4t_m + 5t_p + t_e$. For TwoFactor protocol, each node must calculate their key and two hashing operations, thus requiring $2t_m$ time cost. With each random number, there is a key generation that requires $2t_m$time in cost. There are four signature generation operations that will require $6t_p$ time. There are two encryption operations thus requiring $2t_m$ time. Hence, it will be $29t_h + 2t_m + 6t_p$ in total. For 2PAKEP protocol, there are thirty-one hashing operations for each key so

it requires $31t_h$ in time. Each device must generate the key pairs which will require $2t_m$ in time. There are five signature operations requiring $5t_p$ in time. For each encryption, it will require t_m time. Merging the messages in signature encryption will require $2t_e$ operation time in cost. Thus, in total $31t_h + t_m + 5t_p + 2t_e$ will be time cost.

For LEMAP protocol, there are twenty-two encryption operations requiring $22t_e$ time cost. There are twenty-one hash operations that will require $21t_h$time. Thus, in total, we will need $21t_m + 22t_e + 21t_h$ as the time cost. Table 4.7 shows the computation cost of each security algorithm along with LEMAP.

Table 4.7: Computation Cost

Security Protocol	Computation Cost
LEMAP	$21t_m + 22t_e + 21t_h$
Chaotic	$23t_h + 4t_m + 5t_p + t_e$
TwoFactor	$29t_h + 2t_m + 6t_p$
2PAKEP	$31t_h + t_m + 5t_p + 2t_e$

Figure 4.22 shows the computation cost for each of major operations such as signature, encryption, hashing and random number calculation. It is clearly observed that in terms of signature operation, LEMAP performs better than other benchmark algorithms. In case of encryption, the time cost of LEMAP and 2PAKEP is the lowest while in case of other operations cost of LEMAP and Chaotic have the lowest. Thus overall, the time cost of proposed LEMAP is the lowest as compared to selected benchmark algorithms. It also provides better security as compared to existing security algorithms.

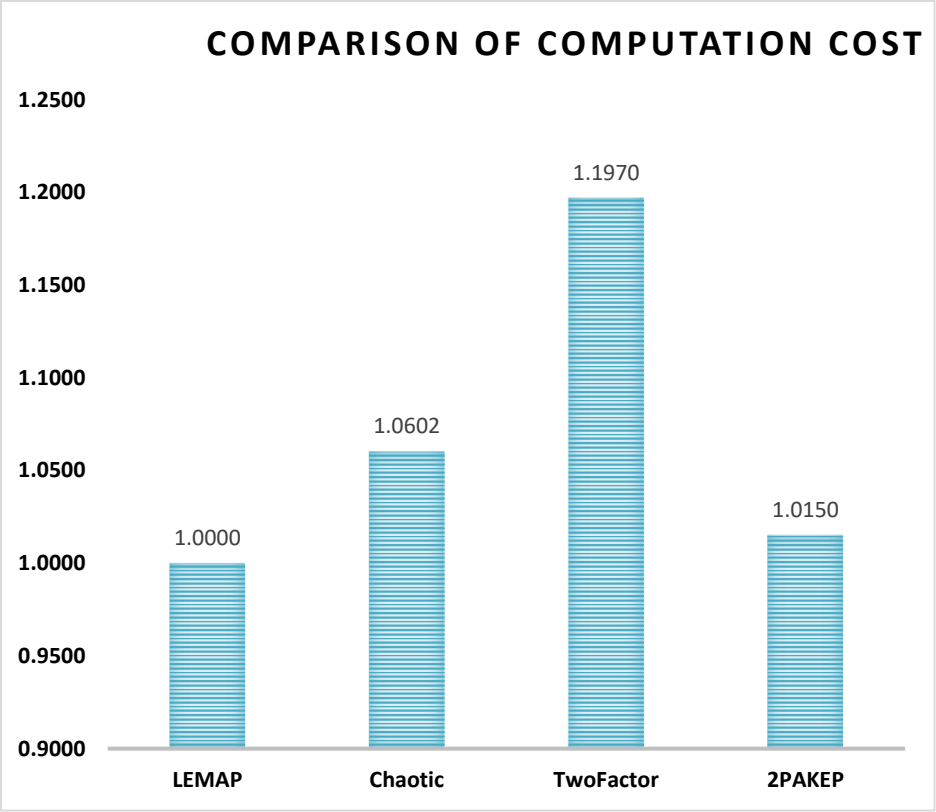

Figure 4.22: Ratio of Computational Cost

4.8 Verification of Security Requirements

Due to the aforementioned threats, a secure D2D communications system should fulfil the following security requirements (Zhao et al., 2016; Renauld et al., 2011), whether they are assisted, controlled or autonomous: Authentication, identification of communicating parties must be checked. There are a number of researches that verify their algorithm for security required as presented by (Javed et al., 2019; Ni et al., 2018; Sharma et al., 2018; Yao et al., 2016; Shukla , 2015; Hu & Evans, 2003).

4.8.1 Data Confidentiality

To verify data confidentiality of LEMAP algorithm, there is need to check whether the data sent can be read by anyone other than receiving party. For this, we can check

Equation 3.12 and Equation 3.13, both the equations are encrypted with the public keys of receiving party such as Equation 3.12 is encrypted with P_{D_1} while in Equation 3.13 whole message is encrypted with P_{D_2}. Thus, the message is fully confidential and can only be decrypted using the private key of the receiving devices.

4.8.2 Data Integrity

To verify the integrity of LEMAP algorithm, we need to check whether the data sent is received as it was sent or can be modified. To achieve integrity, each message is encrypted with sender private key and hash of the message is also sent with actual message. If the hash sent is not equal to the hash message received, then we have a modification in the message. Thus, the messages sent are fully secure against any integrity lost as any modification will be detected.

4.8.3 User Privacy

To verify the privacy of LEMAP algorithm, there is a need to check whether we can find out the real identities of the user. In D2D communication, knowing the real identities of all users can cause various privacy issues as well as real security threats thus the real identities must be kept private. Secondly, no participating device can know who is talking to whom. These identities are only known to eNB thus no other device can know the real identity.

4.8.4 Traceability

To verify the traceability of LEMAP algorithm, we need to check whether the sender can deny that he has not sent the message. For this, all hash is encrypted with the secret key

of the sender as shown in Equation 3.10. Also, the returning hash is signed by the receiving party to make sure the reception and message generator.

4.8.5 Non-repudiation

To verify the non-repudiation attack where the devices can be blocked by not forwarding their message to other devices. There is a check in the protocol that asks to not stop any forwarding request as explained in Section 3.8.5. But users can also block other users to send their traffic to them if they feel any malicious behaviour. It only applies if they are legitimate recipient, they must forward the request. It will result in stoppage of all kind of block hole attacks where a device can pretend to be the service provider and provide no services or wrong services. The feedback mechanism will result in information about any malicious activity by devices.

4.8.6 Mutual Authentication and Key Agreement

To ensure that there is no impersonation attack, all users are registered with the network and get their public key registered at registration authority. MFT contains all registration records of validated devices and is shared with eNB. Thus, the validation trust on keys is already established. Also, the devices establish their own session keys for communication that will provide confidentiality as well as secrecy from key-escrow issues. Impersonation attack cannot be made on LEMAP algorithm as the private key is only known to the registered device.

4.9 Summary

This chapter focused on the validity and proof of LEMAP using formal and mathematical proof. The formal verification of any security algorithm is performed to validate the security algorithm. BAN Logic has been used for proving the LEMAP security

against MITM attack, integrity attack and freshness attack. Results show that the proposed protocol offers much better security. The communication cost of the proposed algorithm is calculated and compared with benchmark algorithms and it is seen that computation overhead generated by LEMAP is less than benchmark algorithms. LEMAP can be used on small devices as it's light-weight protocol. The computation overhead for LEMAP is much lower than benchmark algorithms which creates less traffic over the network, almost half kilobits. The LEMAP is verified against major security requirements such as confidentiality, integrity, availability, traceability and non-repudiation. LEMAP has another contribution in terms of reducing the message overhead to only two messages achieving better security than benchmark security algorithms. Computation complexity is also tested for LEMAP as there are several attacks such as BFA or BsGs on any security algorithm. To avoid these attacks, the security algorithms must be proven DLP and it is proven that LEMAP is secure against these attacks. Multi Factor authentication scheme will block any device acting maliciously thus making the security at two folds. To find what will be time taken to use these security algorithms, an average cellular device is chosen for testing. It is observed LEMAP takes less time in computation resulting in low computation cost as compared to benchmark security algorithms.

CHAPTER 5

CONCLUSION AND RECOMMENDATIONS

5.1 Introduction

Device-to-Device (D2D) communication has emerged as a prominent and latest technology for cellular devices to provide high spectrum efficiency, high data rate and best resources consumption in future cellular networks without supervision of central network node. By utilizing and getting benefit of proximity services, D2D technology offers communicating devices for efficient utilization of available resources, improved data rates, low latency and increased system capacity. The research community is actively and aggressively investigating D2D communication to get full advantages of this technology from IoT devices to emergency services. D2D communication allows device direct communication without communicating to Base Station BS and eNB to provide higher data rate, extended coverage with low signal or even no coverage. D2D communication is mostly associated with 5G and has revolutionized communication aspects due to its associated benefits. However, with advantages of D2D, direct communication without authorization from Base Station BS or eNB has also confronted some challenges as communication occurs on open unsecure channel. Communication security and data privacy are one of the top concerns in D2D communication as existing cellular network security approaches cannot be applied on small cellular devices.

This research has developed a Lightweight ECC-Based Multi-factor Authentication Protocol (LEMAP) D2D to ensure security in multi-hop D2D communication. The developed security algorithm not only provides better security but also decreases extra communication and computation overhead due to its lightweight mechanism. Consequently,

it will reduce processing and storage burden on small devices. There is a lot of research on security algorithms for single hop D2D, while multi-hop D2D research is still at beginning stage.

This research focused on both multi-hop D2D as well as direct D2D communication. The algorithm is based on ECC and multi-factor authentication of devices as multi-hop D2D allows miniature cellular devices to act as decode and forwarding relay. To secure communication between two devices, LEMAP protocol has been developed. The following section summarizes the contributions that our developed protocol will provide as well as the last section have listed future research directions.

5.2 Significant Contribution of LEMAP

There are several contributions of LEMAP algorithm including security, privacy with low communication and computation cost. LEMAP is based on Eliptic Curve Cryptography (ECC) and Elgamal to keep the algorithm lightweight with flexible small key size depending upon security requirements of communication and provide security for both D2D as well as multi-hop communication. As we have used ECC in LEMAP, this provides better security as compared to current famous asymmetric algorithms such as RSA and DH. And being lightweight in nature, it is the best suited for miniature cellular devices having low computation power and storage. Elgamal is used for sharing messages and key information over the network. The validation trust of devices and multi-factor authentication characteristics of LEMAP do not allow an adversary to control devices with its malicious activity.

5.2.1 Privacy

User and device privacy are very important in D2D cellular communication. The issue of user privacy where an adversary can monitor or sniff network traffic to get identity of participating devices and perform some malicious activity may result in severe security issues. In the proposed algorithm LEMAP, identity privacy issue has been resolved by using pseudo identities instead of real IDs to secure identities to avoid any identity reveal attack. This ensures communicating devices to remove the risk of any identity theft and rogue device attack.

5.2.2 Integrity and Confidentiality

Integrity is one of the important aspects of security where it is ensured that data transmitted by the sender is received at the receiver must be same without any tampering or modification during transmission. LEMAP has solved this issue by using SHA-v3 hashing technique approved by NIST. All messages by sender are sent to the receiver with their hash. The received message is compared with its hash at the receiver side to ensure if it is the same or modified during transmission. In case of any change in content is detected by matching hash of the message and the actual message, the message is discarded.

Another important security aspect is confidentiality. The message or data confidentiality is very important so that the contents of the message can only be read by an authorized and legitimate receiver. To achieve confidentiality, proposed algorithm LEMAP has used technique to encrypt all messages by the sender's private key and receiver's public key. Only the authorized and legitimate receiver can decrypt the encrypted message and read its contents. If an adversary sniffs the communication and succeeds in getting the same message to start communication with receptive device and replays the message. LEMAP has

solved this issue by using timestamp in messages which ensures message freshness. When an adversary reuses the message after some time, the receiver checks message freshness by using timestamp and discards the message if timestamp is not valid.

5.2.3 Non-Repudiation

Non-Repudiation is a security aspect in which it is ensured that an individual or user could not deny being the originator of a message. A digital signature along with message is used to get non- repudiation for secure D2D communication. For each message, a digital certificate is issued and digitally signed by a trusted Certificate Authority or CA, and its hash value is encrypted with a private key also held by that same trusted CA. To achieve non-repudiation in LEMAP, each message is digitally signed with sender's private key that is only known to the sender. In this way sender cannot deny the sent message and Non-Repudiation is achieved.

5.2.4 Multi-factor Authentication

MFA is a security mechanism in which an individual or device requires two or more credentials to get mutual authentication and authorization. LEMAP is based on multi-factor authentication scheme that includes Pseudo ID, challenge, timestamp and onetime password to get mutual authentication and authorization for communication. The devices are not authorized until challenge and onetime password is verified, then get validation trust from eNB for communication.

5.2.5 Authentication Overhead

Authentication overhead is a term used in security algorithms which requires minimum number of messages for authentication and reduced communication cost. LEMAP offers minimal possible authentication overhead as compared to other base line algorithms.

To reduce the number of messages over the network, LEMAP has introduced challenge scheme merged with timestamps. So, response as well as freshness and integrity of messages is maintained.

5.2.6 Formal Validation

To measure the correctness and the verification of LEMAP algorithm, formal analysis has been conducted using BAN LOGIC. The results of analysis show that LEMAP is secure against masquerading attack, replay attack, rogue device attack, identity reveal attack and denial of service attack by using SHA v3 and timestamp when sent using a digital signature. It also proves that challenge scheme combined with digital signature avoids Man in The Middle (MITM) attack as well as impersonation attack. As compared to benchmark protocols, the proposed scheme LEMAP mitigates a greater number of attacks and security threats. Due to incapability of handling above mentioned attacks, benchmark protocols are potentially vulnerable to security threats. According to analysis using communication cost calculation, it is observed that LEMAP just takes one extra message per hop while other approaches require at least double time messages for each hop. The ratio on computational cost of LEMAP is 2% better than 2PAKEP security protocol while 6.02% better than Chaotic map-based protocol and 19.70% better than TwoFactor Authentication. Thus, the results clearly indicate that communication cost of LEMAP outperforms other benchmarks by at least more than 2%. The verification of security requirements for D2D communication is an important aspect which proves that either the developed security algorithm can mitigate the attacks which has been claimed. LEMAP achieves data privacy, integrity and confidentiality with non- repudiation. Another important feature of LEMAP is mutual authentication using multi factor approach where devices get authentication and authorization only after applying

multi factors and message freshness is also verified using timestamp. If any of the factor is not verified, device cannot get authentication and communication is not allowed.

5.2.7 Quantum Attack Safe

With the advent of new powerful devices and quantum computing, generic security approaches cannot be useful. One of the options is to increase the key size but it will increase computation and storage space requirement. LEMAP is based on ECC crypto system, so it eliminates this problem as the key size of 512 in ECC is equivalent to 15360 bits of RSA or DH. Moreover, ECC is based on lattice-based cryptography and chances of any kind of Brute Force Attack (BFA) or intelligent attack is computationally very hard to succeed. We have performed DLP analysis to validate the security of LEMAP and it is evident from the approach that it cannot be attacked even through an intelligent attack as proven DLP is validated. LEMAP is a flexible security algorithm that allows multiple key size based on the security provision requirements. The current recommendation is 512-bit keys for network-based authentication and 384-bit key size for session-based communication.

Computation cost is one of the key factors in security algorithms, it is always higher in security protocols making it undesirable to be adopted in D2D communication. LEMAP uses ECC with Elgamal approach and is lighter in computation as compared to Chaotic map based, Two factor Authentication and 2PAKEP protocols. LEMAP takes time in the signature calculation that is 54 ms for a session key calculation as compared to 2PAKEP and Chaotic that takes 22.22% of more time in computation. The other benchmark algorithm Two Factor Authentication takes 38.06% more time in computation than LEMAP making it highly usable in D2D communication. 2PAKEP has 15.81% higher computation cost.

5.3 Future Work

LEMAP is a trustworthy and adaptive security protocol due to its lightweight nature, multi factor authentication characteristic and allows scalability of its key size. There are many features that require attention in D2D communication. LEMAP provides a multi-hop security approach but still there are several research directions that can be adapted such as:

- Multi-hop approach for security is still at its infancy. LEMAP provides an adaptive approach but still performance evaluation and in-premise security analysis of multi-hop for LEMAP can be a potential research direction.
- The multi-factor authentication scheme introduced in LEMAP is adaptive approach which must handle any kind of malicious behavior, in-future the multi-level biometric testing and profiling can be a potential future research direction.
- D2D multi-hop allows communication outside network area. This research does not address in detail about disastrous or catastrophic areas where there is lack of infrastructure deployment. The deployment testing can be a potential research area.
- Multi-hop D2D networks will enhance network diversity, but on the other side a complex billing system and specialized hardware will be required which can increase the cost as dual interface is required for users to switch to either D2D or cellular. This area can be future research to decrease the D2D infrastructure deployment cost.

Physical-layer security is becoming popular rapidly in D2D communications. D2D channels privacy and secrecy may be improved by optimal way of power control and mode selection and how device mobility affects key rate in channel-based key agreement schemes is a promising future research direction

REFERENCES

Abbasi, I. A., Khan, A.S., & Ali, S. (2018). Dynamic multiple junction selection based routing protocol for VANETs in city environment. *Applied Sciences 8*(5), 687.

Abbasi, I. A., Khan, A.S., & Ali, S. (2018). A reliable path selection and packet forwarding routing protocol for vehicular ad hoc networks. *EURASIP Journal on Wireless Communications and Networking, 2018*(1), 1-19.

Abbasi, I. A., & Shahid Khan, A. (2018). A review of vehicle to vehicle communication protocols for VANETs in the urban environment. *Future Internet, 10*(2), 14.

Abd-Elrahman, E., Ibn-Khedher, H., Afifi, H., & Toukabri, T. (2015). Fast group discovery and non-repudiation in D2D communications using IBE. In *2015 International Wireless Communications and Mobile Computing Conference,* (pp. 616–621).

Adam, N., Tapparello, C., & Heinzelman, W. (2019). Infrastructure vs. Multi-Hop D2D Networks: Availability and Performance Analysis. In *2019 International Conference on Computing, Networking and Communications,* (pp. 735–740).

Aggarwal, D., Joux, A., Prakash, A., & Santha, M. (2018). A new public-key cryptosystem via Mersenne numbers. In *Annual International Cryptology Conference,* (pp. 459–482).

Ahmad, I., Kumar, T., Liyanage, M., Okwuibe, J., Ylianttila, M., & Gurtov, A. (2017). 5G security: Analysis of threats and solutions. In *2017 IEEE Conference on Standards for Communications and Networking*, (pp. 193–199).

Ahmad, Z., Shahid Khan, A., Nisar, K., Haider, I., Hassan, R., Haque, M. R., Tarmizi, S., & Rodrigues, J. J. P. C. (2021). Anomaly Detection Using Deep Neural Network for IoT Architecture. *Applied Sciences, 11*(15), 7050.

Ahmad, Z., Shahid Khan, A., Wai Shiang , C., Abdullah, J. & Ahmad, F. (2021). Network intrusion detection system: A systematic study of machine learning and deep learning approaches. *Transactions on Emerging Telecommunications Technologies, 32*(1), e4150.

Andreev, S., Galinina, O., Pyattaev, A., Johnsson, K., & Koucheryavy, Y. (2014). Analyzing assisted offloading of cellular user sessions onto D2D links in unlicensed bands. *IEEE Journal on Selected Areas in Communications*, *33*(1), 67–80.

Aqeel, S., Shahid Khan, A., Ahmad, Z., & Abdullah, J. (2021). A comprehensive study on DNA based Security scheme Using Deep Learning in Healthcare. EDPACS, 1-17.

Asadi, A., Wang, Q., & Mancuso, V. (2014). A survey on device-to-device communication in cellular networks. *IEEE Communications Surveys & Tutorials*, *16*(4), 1801-1819.

Astely, D., Dahlman, E., Fodor, G., Parkvall, S., & Sachs, J. (2013). LTE release 12 and beyond. *IEEE Communications Magazine*, *51*(7), 154–160.

Baccelli, F., Khude, N., Laroia, R., Li, J., Richardson, T., Shakkottai, S., & Wu, X. (2012). On the design of device-to-device autonomous discovery. In *2012 Fourth International Conference on Communication Systems and Networks,* (pp. 1–9).

Bach, E., & Sandlund, B. (2018). Baby-step giant-step algorithms for the symmetric group. *Journal of Symbolic Computation, 85*, 55–71.

Balan, K., Khan, A.S., Julaihi, A.A., Tarmizi, S., Pillay, K.S., Abdulrazak, L., & Sallehudin, H. (2018). RSSI and Public Key Infrastructure based Secure Communication in Autonomous Vehicular Networks. *International Journal of Advanced Computer Science and Applications*, *9*(12), 298-304.

Bastiaan, M. (2015). Preventing the 51%-Attack: a Stochastic Analysis of Two Phase Proof of Work in Bitcoin. In *2015 Proceedings of the 22nd Twente Student Conference on IT*, (pp 1–10).

Borgohain, T., Kumar, U., & Sanyal, S. (2015). Survey of security and privacy issues of internet of things. *arXiv preprint arXiv*:1501.02211.

Bos, J., Kaihara, M., Kleinjung, T., Lenstra, A. K., & Montgomery, P. L. (2009). *On the Security of 1024-bit RSA and 160-bit Elliptic Curve Cryptography*.

Chang, C. C., & Liao, W. Y. (1994). A remote password authentication scheme based upon ElGamal's signature scheme. *Computers & Security*, *13*(2), 137–144.

Chan, K.Y., Abdullah, J.B., & Shahid, A. (2019). A framework for traceable and transparent supply chain management for agri-food sector in Malaysia using blockchain technology. *nternational Journal of Advanced Computer Science and Applications, 10*(11), 149-156.

Chen, S., & Steinberger, J. (2014). Tight security bounds for key-alternating ciphers. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, (pp. 327–350).

Cheng, C., Lu, R., Petzoldt, A., & Takagi, T. (2017). Securing the Internet of Things in a quantum world. *IEEE Communications Magazine, 55*(2), 116–120.

Cohen, M., & Dam, M. (2005). A completeness result for BAN logic. In *Methods for modalities*, *4*(1), 1–32.

Cote, G., Shirani, S., & Kossentini, F. (2000). Optimal mode selection and synchronization for robust video communications over error-prone networks. *IEEE Journal on Selected Areas in Communications*, *18*(6), 952–965.

Dholeshwar, A., & Salapurkar, D. (2016). A Survey on Bio-Inspired Proximity Discovery and Synchronization with Security Solutions for D2D Communications. *International Journal of Engineering Research and Technology*, *3*(3), 970–974.

Dildar, M. S., Khan, N., Abdullah, J. B., & Khan, A. S. (2017, March). Effective way to defend the hypervisor attacks in cloud computing. In *2017 2nd International Conference on Anti-Cyber Crimes*, (pp. 154-159).

Doppler, K., Ribeiro, C. B., & Kneckt, J. (2011). Advances in D2D communications: Energy efficient service and device discovery radio. In *2011 2nd international conference on wireless communication, vehicular technology, information theory and aerospace & electronic systems technology*, (pp. 1–6).

Doppler, K., Yu, C., Ribeiro, C. B., & Janis, P. (2010). Mode selection for device-to-device communication underlaying an LTE-advanced network. In IEEE Wireless Communication and Networking Conference, (pp. 1–6).

Droste, H., Zimmermann, G., Stamatelatos, M., Lindqvist, N., Bulakci, O., Eichinger, J., &

Tullberg, H. (2015). The METIS 5G architecture: A summary of METIS work on 5G architectures. In *2015 IEEE 81st Vehicular Technology Conference*, (pp. 1–5).

Fabian, B., Gürses, S., Heisel, M., Santen, T., & Schmidt, H. (2010). A comparison of security requirements engineering methods. *Requirements Engineering*, 15*(1),* 7–40.

Faria, D. B., & Cheriton, D. R. (2006). Detecting identity-based attacks in wireless networks using signalprints. In *Proceedings of the 5th ACM Workshop on Wireless Security,* (pp. 43–52).

Fauzi, A., & A. Khan (2017). Threats Advancement in Primary User Emulation Attack and Spectrum Sensing Data Falsification (SSDF) Attack in Cognitive Radio Network (CRN) for 5G Wireless Network Environment. *Journal of Telecommunication, Electronic and Computer Engineering*, *9*(2-10), 179-183.

Ferrus, R., & Sallent, O. (2014). Extending the LTE\/LTE-A Business Case: Mission-and Business-Critical Mobile Broadband Communications. *IEEE Vehicular Technology Magazine*, *9*(3), 47–55.

Fujdiak, R., Misurec, J., Mlynek, P., & Janer, L. (2016). Cryptograph key distribution with elliptic curve diffie-hellman algorithm in low-power devices for power grids. *Revue Roumaine des Sciences Techniques,* 84–88.

Gandotra, P., & Jha, R. K. (2016). Device-to-device communication in cellular networks: A survey. *Journal of Network and Computer Applications, 71,* 99–117.

Gandotra, P., Jha, R. K., & Jain, S. (2017). A survey on device-to-device (D2D)

communication: Architecture and security issues. *Journal of Network and Computer Applications*, *78*, 9–29.

Ghudayyer, M. B., Javed, Y., & Alenezi, M. (2017). A Security Perspective on Adoption and Migration to Mobile Cloud Technology. *International Journal on Informatics Visualization*, *1*(4), 143–149.

Gordon, D. (2011). Discrete logarithm problem. *Encyclopedia of Cryptography and Security*, *1*(1), 352–353.

Grossglauser, M., & Tse, D. N. (2002). Mobility increases the capacity of ad hoc wireless networks. *IEEE/ACM Transactions on Networking*, *10*(4), 477–486.

Gu, J., Bae, S. J., Choi, B. G., & Chung, M. Y. (2011). Dynamic power control mechanism for interference coordination of device-to-device communication in cellular networks. In *2011 Third International Conference on Ubiquitous and Future Networks*, (pp. 71-75).

Gupta, A., & Jha, R. K. (2015). A survey of 5G network: Architecture and emerging technologies. *IEEE Access*, *3*, 1206–1232.

Gupta, A., Jha, R. K., & Devi, R. (2018). Security architecture of 5g wireless communication network. *International Journal of Sensors Wireless Communications and Control*, *8*(2), 92–99.

Gupta, K., & Silakari, S. (2011). ECC over RSA for asymmetric encryption: A review. *International Journal of Computer Science Issues*, *8*(3), 370.

Habib, B., Cambou, B., Booher, D., & Philabaum, C. (2017). Public key exchange scheme that is addressable (PKA). In *2017 IEEE Conference on Communications and Network Security*, (pp. 392–393).

Habiba, U., & Hossain, E. (2018). Auction mechanisms for virtualization in 5G cellular networks: Basics, trends, and open challenges. *IEEE Communications Surveys & Tutorials*, *20*(3), 2264–2293.

Haddad, Z., Mahmoud, M., Taha, S., & Saroit, I. A. (2015). Secure and privacy-preserving AMI-utility communications via LTE-A networks. In *2015 IEEE 11th International Conference on Wireless and Mobile Computing, Networking and Communications*, (pp. 748–755).

Hafeez, I., Ding, A. Y., Antikainen, M., & Tarkoma, S. (2017). Toward Secure Edge Networks Taming Device to Device (D2D) Communication in IoT. *arXiv preprint arXiv*:1712.05958.

Hamzah, A. R. M., Fisal, M., Khan, A. S., Kamilas, S., & Hafizah, S. (2013). Distributed Multi-Hop Reservation Protocol for Wireless Personal Area Ultra-Wideband Networks. *Computers and Software*, *8*(6), 1294–1301.

Han, M. H., Kim, B. G., & Lee, J. W. (2012). Subchannel and transmission mode scheduling for D2D communication in OFDMA networks. In *2012 IEEE Vehicular Technology Technology Conference, (*pp. 1–5).

Haus, M., Waqas, M., Ding, A. Y., Li, Y., Tarkoma, S., & Ott, J. (2017). Security and privacy in device-to-device (D2D) communication: A review. *IEEE Communications*

Surveys & Tutorials, *19*(2), 1054–1079.

He, Y., Ren, J., Yu, G., & Cai, Y. (2019). D2D Communications Meet Mobile Edge Computing for Enhanced Computation Capacity in Cellular Networks. *IEEE Transactions on Wireless Communications*, *18*(3), 1750–1763.

Hossain, M. M., & Hasan, R. (2017). Boot-IoT: A Privacy-Aware Authentication Scheme for Secure Bootstrapping of IoT Nodes. In *IEEE Congress on Internet of Things*, (pp. 1–8).

Hsu, R. H., Lee, J., Quek, T. Q., & Chen, J. C. (2017). GRAAD: Group anonymous and accountable D2D communication in mobile networks. *IEEE Transactions on Information Forensics and Security*, *13*(2), 449–464.

Hu, L., & Evans, D. (2003, January). Secure aggregation for wireless networks. In *2003 Symposium on Applications and the Internet Workshops proceedings*, (pp. 384–391).

Huang, J., Liao, Y., Xing, C. C., & Chang, Z. (2019). Multi-Hop D2D Communications With Network Coding: From a Performance Perspective. *IEEE Transactions on Vehicular Technology*, *68*(3), 2270–2282.

Husain, S., Prasad, A., Kunz, A., Papageorgiou, A., & Song, J. (2014). Recent trends in standards related to the internet of things and machine-to-machine communications. *Journal of Information and Communication Convergence Engineering*, *12*(4), 228–236.

Imai, H., Okada, H., Yamazato, T., & Katayama, M. (2006). The effect of multipath hybrid

routing protocol in multihop cellular networks. In *2006 IEEE 17th International Symposium on Personal, Indoor and Mobile Radio Communications* (pp. 1–5).

Iqbal, A. M. & Khan, A. S. & Senin, A. (2015). Reinforcing the National Innovation System of Malaysia Based on University-Industry Research Collaboration: A System Thinking Approach. *International Journal of Management Sciences and Business Research, 4*(1), 6-15.

Iqbal, A. M., Aslan, A. S., & Khan, A. S. (2010). Innovation Oriented Constraints between University-industry Technological Collaboration. PROCEEDING ICPE-4 2010, 4, 24.

Iqbal, A. M., Iqbal, S., Khan, A. S., & Senin, A. A. (2013). A novel cost efficient evaluation model for assessing research-based technology transfer between university and industry. *Sains Humanika, 64*(2).

Iqbal, A. M., Shahid Khan, A., Kulathuramaiyer, N. & Senin, A. (2021). Blended system thinking approach to strengthen the education and training in university-industry research collaboration. *Technology Analysis & Strategic Management*, 1-14. 10.1080/09537325.2021.1905790.

Iqbal, A. M., Khan, S., Bashir, F., & Senin, A. (2015). Evaluating national innovation system of malaysia based on university-industry research collaboration: A system thinking approach. *Asian Social Science, 11*(13), 45.

Iqbal, A. M., Khan, A. S., Iqbal, S. P., & Senin, A. A. (2011). Designing of success criteria-based evaluation model for assessing the research collaboration between university

and industry. *International Journal of Business Research and Management, 2*(2), 59-73.

Iqbal, A. M., Khan, S., Parveen, S., & Senin, A. (2015). An efficient evaluation model for the assessment of university-industry research collaboration in Malaysia. *Research Journal of Applied Sciences, Engineering and Technology, 10*(3), 298-306.

Iqbal, A. M., Khan, S., & Senin, A. (2012). Determination of high impact evaluation metrics for evaluating the University-industry technological linkage. *International Journal of Physical and Social Sciences, 2*(4), 111-122.

Iqbal, S., Muhammad Iqbal, A., Khan, A. S., & Amat Senin, A. (2013). A Modern Strategy for the Development of Academic Staff Based on University-Industry Knowledge Transfer Effectiveness & Collaborative Research. *Sains Humanika, 64*(3).

Jambli, M. N., Khan, A. S., Lenando, H., Abdullah, J., & Suhaili, S. M. (2017). A Dynamic Energy Savvy Routing Algorithm for Mobile Ad-Hoc and Sensor Networks. *Advanced Science Letters, 23*(6), 5542-5546.

Jambli, M. N., Pillay, K. S., Julaihi, A. A., Khan, A. S., & Suhaili, S. M. (2017). A survey of cluster based routing protocols for mobile ad-hoc sensor network. *Journal of Telecommunication, Electronic and Computer Engineering, 9*(2-10), 71-78.

Jambli, M. N., Khan, A. S., & Shoon, S. C. (2016). A survey of VASNET framework to provide infrastructure-less green IoTs communications for data dissemination in search and rescue operations. *Journal of Electronic Science and Technology, 14*(3), 220-228.

Janis, P., Koivunen, V., Ribeiro, C., Korhonen, J., Doppler, K., & Hugl, K. (2009). Interference-aware resource allocation for device-to-device radio underlaying cellular networks. In *VTC Spring IEEE 69th Vehicular Technology Conference*, (pp. 1–5).

Javed, Y., & Khan, A. S. (2019). Major Security attacks in D2D Communication. *Ubiquitous Computing and Communication Journal*, *13*(1), 12–18.

Javed, Y., Khan, A. S., & Abbasi, M. A. K. (2019). Key Security attacks and their remedies in D2D Communication. *Ubiquitous Computing and Communication Journal*, *13*(1), 19–24.

Javed, Y., Khan, A. S., Hamid, A, Almuqhim, S., Khan, Z. I., & Abdullah, J. (2019). Securing TLS from MITM Incursion using Diffie-Hellman. In *2019 Second International Conference on Advanced Technologies, Computer Engineering and Science*, (pp. 407–411).

Javed, Y., Khan, A. S., Qahar, A., & Abdullah, J. (2017). EEoP: A lightweight security scheme over PKI in D2D cellular networks. *Journal of Telecommunication, Electronic and Computer Engineering*, *9*(3), 99–105.

Javed, Y., Khan, A. S., Qahar, A., & Abdullah, J. (2017). Preventing DoS Attacks in IoT Using AES. *Journal of Telecommunication, Electronic and Computer Engineering*, *9*(3), 55–60.

Javed, Y., Khan, S., Qureshi, B., & Chaudhry, J. (2015). Estimating Diabetic cases in KSA through search trends and Creating Cyber Diabetic Community. In *International Conference on Recent Advances in Computer Systems, Atlantis Press.*

Javed, Y., Khan, S., & Qahar, A. (2017). EEoP: A lightweight security scheme over PKI in D2D cellular networks. *Journal of Telecommunication, Electronic and Computer Engineering*, *9*(3-11), 99-105.

Ji, M., Caire, G., & Molisch, A. F. (2015). Fundamental limits of caching in wireless D2D networks. *IEEE Transactions on Information Theory*, *62*(2), 849–869.

Jiang, Y., Liu, Q., Zheng, F., Gao, X., & You, X. (2015). Energy-efficient joint resource allocation and power control for D2D communications. *IEEE Transactions on Vehicular Technology*, *65*(8), 6119–6127.

Kaleem, Z., Qadri, N. N., Duong, T. Q., & Karagiannidis, G. K. (2019). Energy-efficient device discovery in D2D cellular networks for public safety scenario. *IEEE Systems Journal*, *1*(1), 1–4.

Karthikeyan, S., & Nesterenko, M. (2005). RFID security without extensive cryptography. In *Proceedings of the 3rd ACM Workshop on Security of Ad Hoc and Sensor Networks*, (pp. 63–67).

Kato, N. (2016). On device-to-device (D2D) communication. *IEEE Network*, *30(3)*, 2–2.

Khan, A. S. & Lenando, H., & Nazim, J. M. (2016). Green Resource Allocation for Multiple OFDMA Based Networks: A Survey. *Journal of Electronic Science and Technology*, *14*(2), 170-182.

Khan, I. U., Tan, C. E., & Khan, A. S. (2014). The Enhanced Amplify-and-Forward Three Time Slots TDMA-Based Protocol Using Inter-Relay Communication Over Rician Fading Channel. *International Review on Computers and Software 9*(8), 1384-1391.

Khan, A. S., Abdullah, J., Zen, K., & Tarmizi, S. (2017). Secure and Scalable Group Rekeying for Mobile Multihop Relay Network. *Advanced Science Letters, 23*(6), 5242-5245.

Khan, A. S., Fisal, N., Esa, M., Kamilah, S., Zubair, S., Abu Bakar, W. M. (2014). Privacy Key Management Protocols and Their Analysis in Mobile Multihop Relay WiMAX Networks. *Security for Multihop Wireless Networks, 43*.

Khan, A. S., Fisal, N., Yusof, S. K. S., Ariffin, S. H. S., Maarof, N. N., & Abbas, M. (2010). Security Issues of Relay-Based IEEE 802.16 m Network. In *4th International Conference on Post Graduate Education, 26th-28th November 2010, Mid Velly Cititel Hotel, Kula Lumpur, Malaysia*.

Khan, A. S., Halikul, L., Jambli, M. N., & Thangaveloo, R. (2017). Mitigation of Non-Transparent Rouge Relay Stations in Mobile Multihop Relay Networks. *Advanced Science Letters, 23*(6), 5246-5250.

Khan, A. S., Lenando, H., & Abdullah, J. (2014). Lightweight message authentication protocol for mobile multihop relay networks. *International Review on Computers and Software, 9*(10), 1720-1730.

Khan, A., Khan, A. S., Qahar, A., Fauzi, A. H., & Javed, Y. (2018). Using Green and Emerging Technology. *Asian Journal of Information Technology, 17*(1), 23-51.

Khan, A. S. (2014). Secure and efficient distributed relay-based rekeying algorithm for group communication in mobile multihop relay network. *International Journal of Communication Networks and Information Security*, *6*(3), 189.

Khan, A. S., Ahmad, Z., Abdullah, J., & Ahmad, F. (2021). A Spectrogram image-based Network Anomaly Detection System using Deep Convolutional Neural Network. *IEEE Access*, *9*, 87079-87093.

Khan, A. S., Balan, K., Javed, Y., Abdullah, J., & Tarmizi, S. (2019). Secure trust-based blockchain architecture to prevent attacks in VANET. *Sensors 19*(22), 4954.

Khan, A. S., Fisal, N., Elshafie, H. E. A., EI Khalifa, F., Abbas, M., & Shaneshin, M. (2010). An Overview of Security Challenges of Next Generation Mobile Wimax IEEE 802.16 m Technology. In *3rd International Graduate Conference on Engineering, Science and Humanities*.

Khan, A. S., Fisal, N., & Hossain, S. (2009). Man-in-the-middle attack and possible solutions on Wimax 802.16 j. In *Proceedings of International Conference on Recent and Emerging Advance Technologies in Engineering*.

Khan, A. S., Fisal, N., Kamilah, S., & Abbas, M. (2010). Efficient distributed authentication key scheme for multi-hop relay in IEEE 802.16 j network. *International Journal of Engineering Science and Technology*, *2*(6), 2192-2199.

Khan, A. S., Fisal, N., Kamilah, S., Hafizah, S., Esa, M., Abu Bakar, Z., & Abbas, M. (2011). An Efficient Self-Organized Authentication and Key Management Scheme for

Distributed Multihop Relay-Based IEEE 802.16 Networks. *International Journal of Computer Science and Information Security, 9*(3), 30.

Khan, A. S., Fisal, N., Kamilah, S., A Rashid, R., & Abbas, M. (2011). Secure and Efficient Multicast Rekeying Approach For Non-Transparent Relay-based IEEE 802.16 Networks. *International Journal of Computer Applications, 975*, 8887.

Khan, A. S., Fisal, N., Ma'arof, N. N. M. I., EI Khalifa, F., Abbas, M., Elshafie, H. E. A. (2010). Provisioning of Public Key Infrastructure (PKI) and Security Issues in IEEE802. 16m Networks. In Proceedings 3rd International Graduate Conference of Engineering Science, and Humanity. 2-4 Nov 2010.

Khan, A. S., Fisal, N., Ma'arof, N. N. M. I., Khalifa, F. E. I., & Abbas, M. (2011). Security Issues and Modified Version of PKM Protocol in Non-Transparent Multihop Relay in IEEE 802. 16 j Networks. *International Review on Computers and Software*, *6*(1), 104-109.

Khan, A. S., Fisal, N., Maarof, N. N. M. I., EI Khalifa, F., Abbas, M., & Hashim, E. A. (2010). Security zone and key derivation management in centralized security control in wimax multihop relay system. In *Proceedings 3rd International Graduate Conference of Engineering Science, and Humanity*.

Khan, A. S., Fisal, N., Yusof, S. K. S., Ariffin, S. H. S., Esa, M., Maarof, N. N., & Abbas, M. (2010). An improved authentication key management scheme for multihop relay in IEEE 802.16 m networks. In *2010 IEEE Asia-Pacific Conference on Applied Electromagnetics*, (pp. 1-5).

Khan, A. S., Fisal, N., Bakar, Z. A., Salawu, N., Maqbool, W., Ullah, R., & Safdar, H. (2015). Secure Authentication and Key Management Protocols for Mobile Multihop WiMAX Networks. *Indian Journal of Science and Technology*, *7*(3), 282.

Khan, A. S., Javed, Y., Abdullah, J., & Zen, K. (2021). Trust-based lightweight security protocol for device to device multihop cellular communication (TLwS). *Journal of Ambient Intelligence and Humanized Computing*, 1-18. https://doi.org/10.1007/s12652-021-02968-6

Khan, N., & Abdullah, J. (2017). Security issues in 5G device to device communication. Inter*national Journal of Computer Science and Network Security*, *17*(5), 366.

Khan, N., Abdullah, J., & Khan, A. S. (2015). Towards vulnerability prevention model for web browser using interceptor approach. In *2015 9th International Conference on IT in Asia*, (pp. 1-5).

Khan, N., & Khan, A. S. (2017). Defending malicious script attacks using machine learning classifiers. *Wireless Communications and Mobile Computing, 2017*, 1-9. 10.1155/2017/5360472.

Khan, N., Abdullah, J., & Khan, A. S. (2017). A Dynamic Method of Detecting Malicious Scripts Using Classifiers. *Advanced Science Letters, 23*(6), 5352-5355.

Khan, N., Abdullah, J., & Khan A. S. (2017). A Taxonomy Study of XSS Vulnerabilities. *Asian Journal of Information Technology, 16*(2-5), 169-177.

Khan, S., Abdullah, J., Khan, N., Julahi, A. A., & Tarmizi, S. (2017). Quantum-elliptic curve cryptography for multihop communication in 5G networks. *International Journal of Computer Science and Network Security, 17*(5), 357-365.

Khan, S., Javed, Y., Abdullah, J., Nazim, J. M., & Khan, N. (2017). Security issues in 5G device to device communication. *International Journal of Computer Science and Network Security*, *17*(5), 366.

Khalid, Z., Fisal, N. B., Ullah, R., Safdar, H., Maqbool, W., Zubair, S., & Khan, A. S. (2013). M2M communication in virtual sensor network for SHAAL. *Jurnal Teknologi, 65*(1). 10.11113/jt.v65.1749.

Kim, J. Y., Hu, W., Shafagh, H., & Jha, S. (2016). SEDA: secure over-the-air code dissemination protocol for the Internet of Things. *IEEE Transactions on Dependable and Secure Computing*, *15*(6), 1041–1054.

Kim, S. H., & Han, S. J. (2012). Contour routing for peer-to-peer DTN delivery in cellular networks. *In 2012 Fourth International Conference on Communication Systems and Networks*, (pp. 1–9).

Le, D. N., Le Tuan, L., & Tuan, M. N. D. (2019). Smart-building management system: An Internet-of-Things (IoT) application business model in Vietnam. *Technological Forecasting and Social Change, 141*, 22–35.

Lee, J., Kim, Y., Kwak, Y., Zhang, J., Papasakellariou, A., Novlan, T., & Li, Y. (2016). LTE-advanced in 3GPP Rel-13/14: an evolution toward 5G. *IEEE Communications Magazine*, *54*(3), 36–42.

Lee, N. Y., & Chiu, Y. C. (2005). Improved remote authentication scheme with smart card. *Computer Standards & Interfaces*, *27*(2), 177–180.

Lenando, H., Gharin, A. H., Jambli, M. N., Abdullah, J., & Khan, A. S. (2015). Neighbor selection protocol for heterogeneous information dissemination in Opportunistic Networks. In *2015 9th International Conference on IT in Asia*, (pp. 1-7).

Lenando, H., Sian, G. S., Khan, A. S., Fauzi, A. H. (2014). Identify the best location to place data based on social interaction in opportunistic network. In *The 5th International Conference on Information and Communication Technology for the Muslim World*, (pp. 1-6).

Li, H., Wang, B., Song, Y., & Ramamritham, K. (2016). VeShare: A D2D infrastructure for real-time social-enabled vehicle networks. *IEEE Wireless Communications*, *23*(4), 96–102.

Li, S., Da Xu, L., & Zhao, S. (2018). 5G Internet of Things: A survey. *Journal of Industrial Information Integration*, *10*, 1–9.

Li, Y., Chi, K., Chen, H., Wang, Z., & Zhu, Y. (2017). Narrowband Internet of Things systems with opportunistic D2D communication. *IEEE Internet of Things Journal*, *5*(3), 1474–1484.

Li, Y., Wu, T., Hui, P., Jin, D., & Chen, S. (2014). Social-aware D2D communications: Qualitative insights and quantitative analysis. *IEEE Communications Magazine*, *52*(6), 150–158.

Lien, S. Y., Chien, C. C., Tseng, F. M., & Ho, T. C. (2016). 3GPP device-to-device

communications for beyond 4G cellular networks. *IEEE Communications Magazine*, *54*(3), 29–35.

Liu, J., Kato, N., Ma, J., & Kadowaki, N. (2014). Device-to-device communication in LTE-advanced networks: A survey. *IEEE Communications Surveys & Tutorials*, *17*(4), 1923–1940.

Liu, J., Zhang, S., Kato, N., Ujikawa, H., & Suzuki, K. (2015). Device-to-device communications for enhancing quality of experience in software defined multi-tier LTE-A networks. *IEEE Network*, *29*(4), 46–52.

Lu, P., Zhang, L., Liu, X., Yao, J., & Zhu, Z. (2015). Highly efficient data migration and backup for big data applications in elastic optical inter-data-center networks. *IEEE Network*, *29*(5), 36–42.

Mach, P., Becvar, Z., & Najla, M. (2018). Combined Shared and Dedicated Resource Allocation for D2D communication. *In 2018 IEEE 87th Vehicular Technology Conference*, (pp. 1–7).

Maikol, S. O., et al. (2021). A Novel Authentication and Key Agreement Scheme for Countering MITM and Impersonation Attack in Medical Facilities. *International Journal of Integrated Engineering*, *13*(2), 127-135.

Malandrino, F., Casetti, C., & Chiasserini, C. F. (2014). Toward D2D-enhanced heterogeneous networks. *IEEE Communications Magazine*, *52*(11), 94–100.

Mallana, M. F. B., Iqbal, A.M., Iqbal S., Khan A.S., Senin, A. A. (2013). The critical factors for the successful transformation of technology from developed to developing countries. *Jurnal Teknologi, 64*(3). https://doi.org/10.11113/sh.v64n3.76

Mao, W. (2003). *Modern cryptography: theory and practice*. Pearson Education India.

Meister, D. (2013). *Advanced data deduplication techniques and their application*, Doctoral dissertation, Johannes Gutenberg University Mainz.

Melki, L., Najeh, S., & Besbes, H. (2016). Interference management scheme for network-assisted multi-hop D2D communications. In *2016 IEEE 27th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications,* (pp. 1–5).

Militano, L., Orsino, A., Araniti, G., Nitti, M., Atzori, L., & Iera, A. (2016). Trusted D2D-based data uploading in in-band narrowband-IoT with social awareness. In *2016 IEEE 27th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications*, (pp. 1–6).

Mishra, P. K., & Pandey, S. (2016, August). A Method for Mode Selection in a dynamic network for Device-to-Device Communication for 5G. In *Proceedings of the International Conference on Informatics and Analytics* (pp. 1-6).

Monserrat, J. F., Mange, G., Braun, V., Tullberg, H., Zimmermann, G., & Bulakci, Ö. (2015). METIS research advances towards the 5G mobile and wireless system definition. *EURASIP Journal on Wireless Communications and Networking*, *2015*(1), 53.

Nardini, G., Stea, G., & Virdis, A. (2017). A fast and reliable broadcast service for LTE-

Advanced exploiting multihop device-to-device transmissions. *Future Internet*, *9*(4), 89.

Nasrallah, A., Thyagaturu, A. S., Alharbi, Z., Wang, C., Shao, X., Reisslein, M., & ElBakoury, H. (2018). Ultra-low latency (ULL) networks: The IEEE TSN and IETF DetNet standards and related 5G ULL research. *IEEE Communications Surveys & Tutorials*, *21*(1), 88–145.

Ni, J., Lin, X., & Shen, X. S. (2018). Efficient and secure service-oriented authentication supporting network slicing for 5G-enabled IoT. *IEEE Journal on Selected Areas in Communications*, *36*(3), 644–657.

Nisar, K., et al. (2021). Evolutionary Integrated Heuristic with Gudermannian Neural Networks for Second Kind of Lane–Emden Nonlinear Singular Models. *Applied Sciences*, *11*(11), 4725.

Nomikos, N., Charalambous, T., Krikidis, I., Skoutas, D. N., Vouyioukas, D., Johansson, M., & Skianis, C. (2015). A survey on buffer-aided relay selection. *IEEE Communications Surveys & Tutorials*, *18*(2), 1073–1097.

Noura, M., & Nordin, R. (2016). A survey on interference management for device-to-device (D2D) communication and its challenges in 5G networks. *Journal of Network and Computer Applications*, *71*, 130–150.

Ometov, A., Olshannikova, E., Masek, P., Olsson, T., Hosek, J., Andreev, S., & Koucheryavy, Y. (2016). Dynamic trust associations over socially-aware D2D technology: A practical implementation perspective. *IEEE Access*, *4*, 7692–7702.

Ometov, A., Zhidanov, K., Bezzateev, S., Florea, R., Andreev, S., & Koucheryavy, Y. (2015, August). Securing network-assisted direct communication: The case of unreliable cellular connectivity. In *2015 IEEE Trustcom/BigDataSE/ISPA* (Vol. 1, pp. 826–833).

Othmen, S., Zarai, F., Belghith, A., Obaidat, M. S., & Kamoun, L. (2017). Secure and Reliable Multi-Path Routing Protocol for Multi-Hop Wireless Networks. *Adhoc & Sensor Wireless Networks*, *36*(1-4), 127-147.

Oueis, J., Conan, V., Lavaux, D., Stanica, R., & Valois, F. (2017). Overview of LTE isolated E-UTRAN operation for public safety. *IEEE Communications Standards Magazine*, *1*(2), 98–105.

Panaousis, E., Alpcan, T., Fereidooni, H., & Conti, M. (2014). Secure message delivery games for device-to-device communications. In *International Conference on Decision and Game Theory for Security*, (pp. 195–215).

Phunchongharn, P., Hossain, E., & Kim, D. I. (2013). Resource allocation for device-to-device communications underlaying LTE-advanced networks. *IEEE Wireless Communications*, *20*(4), 91–100.

Qin, H., Mi, Z., Dong, C., Peng, F., & Sheng, P. (2016). An experimental study on multihop D2D communications based on smartphones. In *2016 IEEE 83rd Vehicular Technology Conference*, (pp. 1–5).

Raghothaman, B., Deng, E., Pragada, R., Sternberg, G., Deng, T., & Vanganuru, K. (2013, January). Architecture and protocols for LTE-based device to device communication. In *2013 International Conference on Computing, Networking and*

Communications, (pp. 895–899).

Rakshanda, J., & Yadav, M. R. L. (2013). Network Security: A Detailed Review. *International Journal for Technological Research in Engineering*, *4*(11), 2380-2384.

Ramadan, M., Li, F., Xu, C., Mohamed, A., Abdalla, H., & Ali, A. A. (2016). User-to-User Mutual Authentication and Key Agreement Scheme for LTE Cellular System. *Inetrnational Journal of Network Security*, *18*(4), 769–781.

Raya, M., Papadimitratos, P., & Hubaux, J. P. (2006). Securing vehicular communications. *IEEE Wireless Communications*, *13*(5), 8–15.

Raymond, D. R., & Midkiff, S. F. (2008). Denial-of-service in wireless sensor networks: Attacks and defenses. *IEEE Pervasive Computing*, *7*(1), 74–81.

Renauld, M., Standaert, F. X., Veyrat-Charvillon, N., Kamel, D., & Flandre, D. (2011). A formal study of power variability issues and side-channel attacks for nanoscale devices. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, (pp. 109–128).

Safdar, H., Fisal, N., Ullah, R., Maqbool, W., Asraf, F., Khalid, Z., Khan, A. S. (2013). Resource allocation for uplink M2M communication: A game theory approach. In *2013 IEEE Symposium on Wireless Technology & Applications*, (pp. 48–52).

Salawu, N., Syed Ariffin, S. H., Fisal, N., Ghazali, N. E., & Khan, A. S. (2013). A cost function algorithm for mobility load balancing in long term evolution networks. *Australian Journal of Basic and Applied Sciences*, *7*, 742-754.

Sakr, A. H., & Hossain, E. (2015). Cognitive and energy harvesting-based D2D communication in cellular networks: Stochastic geometry modeling and analysis. *IEEE Transactions on Communications*, *63*(5), 1867–1880.

Sedidi, R., & Kumar, A. (2016). Key exchange protocols for secure Device-to-Device (D2D) communication in 5G. In *2016 Wireless Days,* (pp. 1–6).

Shahid Khan, A., et al. (2009). Man-in-the-middle attack and possible solutions on wimax 802.16 j. In *Proceedings of International Conference on Recent and Emerging Advance Technologies in Engineering.*

Shalmashi, S., Miao, G., & Slimane, S. B. (2013). Interference management for multiple device-to-device communications underlaying cellular networks. In *2013 IEEE 24th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications,* (pp. 223–227).

Shaneshin, M., Khan, A. S., Ngah, R., Rahayu, Y., & Banitalebi A. (2010). An overview of Femtocell architecture in collaboration with Wimax application. In *3rd International Graduate Conference on Engineering, Science and Humanities*. School of Graduate Studues Universiti Teknologi Malaysia.

Sharma, V., You, I., Leu, F. Y., & Atiquzzaman, M. (2018). Secure and efficient protocol for fast handover in 5G mobile Xhaul networks. *Journal of Network and Computer Applications*, *102*, 38–57.

Shen, W., Yin, B., Cao, X., Cai, L. X., & Cheng, Y. (2016). Secure device-to-device communications over WiFi direct. *IEEE Network*, *30*(5), 4–9.

Shen, Z., Papasakellariou, A., Montojo, J., Gerstenberger, D., & Xu, F. (2012). Overview of 3GPP LTE-advanced carrier aggregation for 4G wireless communications. *IEEE Communications Magazine*, *50*(2), 122–130.

Shukla, V., Chaturvedi, A., & Srivastava, N. (2015). A new secure authenticated key agreement scheme for wireless (mobile) communication in an EHR system using cryptography. *Communication on Applied Electronics*, *3*(3), 16–21.

Steinfeld, R., & Zheng, Y. (2000). A signcryption scheme based on integer factorization. In *International Workshop on Information Security*, (pp. 308–322).

Sufatrio, A. & Yap, R. H. C. (2008). Extending BAN logic for reasoning with modern PKI-based protocols. In *2008 IFIP International Conference on Network and Parallel Computing*, (pp. 190–197).

Suo, H., Wan, J., Zou, C., & Liu, J. (2012). Security in the internet of things: a review. *In 2012 International Conference on Computer Science and Electronics Engineering*, (Vol. 3, pp. 648–651).

Ta, T., Baras, J. S., & Zhu, C. (2014). Improving smartphone battery life utilizing device-to-device cooperative relays underlaying LTE networks. In *2014 IEEE International Conference on Communications*, (pp. 5263–5268).

Tata, C., & Kadoch, M. (2014). Multipath routing algorithm for device-to-device communications for public safety over LTE heterogeneous networks. In *2014 1st International Conference on Information and Communication Technologies for Disaster Management*, (pp. 1–7).

Tehrani, M. N., Uysal, M., & Yanikomeroglu, H. (2014). Device-to-device communication in 5G cellular networks: challenges, solutions, and future directions. *IEEE Communications Magazine*, *52*(5), 86–92.

Theobald, L. M., Wu, D., Choi, K. W., & Han, Z. (2014). Device-to-device discovery for proximity-based service in LTE-advanced system. *IEEE Journal on Selected Areas in Communications*, *33*(1), 55–66.

Tran, K. A. (2018). Resource allocation in D2D communication in cellular mode. *Journal of Advanced Engineering and Computation*, *2*(3), 197–207.

Trifunovic, S., Distl, B., Schatzmann, D., & Legendre, F. (2011). WiFi-Opp: ad-hoc-less opportunistic networking. In *Proceedings of the 6th ACM Workshop on Challenged Networks*, (pp. 37–42).

Turkanović, M., Brumen, B., & Hölbl, M. (2014). A novel user authentication and key agreement scheme for heterogeneous ad hoc wireless sensor networks, based on the Internet of Things notion. *Ad Hoc Networks*, 20, 96–112.

Ullah, R., et al. (2013). Voronoi cell geometry based dynamic fractional frequency reuse for OFDMA cellular networks. In *2013 IEEE International Conference on Signal and Image Processing Applications*, (pp. 435-440).

Ullah, R., Fisal, N., Safdar, H., Maqbool, W., Khalid, Z., & Khan, A. S. (2013). Voronoi cell geometry based dynamic Fractional Frequency Reuse for OFDMA cellular networks. IEEE ICSIPA 2013 - IEEE International Conference on Signal and Image Processing Applications. 10.1109/ICSIPA.2013.6708046.

Vanganuru, K., Ferrante, S., & Sternberg, G. (2012). System capacity and coverage of a cellular network with D2D mobile relays. In *MILCOM 2012-2012 IEEE Military Communications Conference*, (pp. 1–6).

Vetter, S., Nohria, R., & Santos, G. (2018). IBM power system AC922 introduction and technical overview. *IBM Redbooks*.

Wallner, D., Harder, E., & Agee, R. (1999). *RFC2627: Key Management for Multicast: Issues and Architectures*. RFC 2627.

Wang, H., & Chu, X. (2012). Distance-constrained resource-sharing criteria for device-to-device communications underlaying cellular networks. *Electronics Letters*, *48*(9), 528–530.

Wang, L., Li, Z., Chen, M., Zhang, A., Cui, J., & Zheng, B. (2017). Secure content sharing protocol for D2D users based on profile matching in social networks. In *2017 9th International Conference on Wireless Communications and Signal Processing*, (pp. 1–5).

Wang, L., Tian, F., Svensson, T., Feng, D., Song, M., & Li, S. (2015). Exploiting full duplex for device-to-device communications in heterogeneous networks. *IEEE Communications Magazine*, *53*(5), 146–152.

Wang, P., & Yu, R. (2019). SMF-GA: Optimized Multitask Allocation Algorithm in Urban Crowdsourced Transportation. *Wireless Communications and Mobile Computing*, *1*(1), 1–13.

Wei, L., Hu, R. Q., Qian, Y., & Wu, G. (2014). Enable device-to-device communications

underlaying cellular networks: challenges and research aspects. *IEEE Communications Magazine*, *52*(6), 90–96.

Wen, S., Zhu, X., Zhang, X., & Yang, D. (2013). QoS-aware mode selection and resource allocation scheme for device-to-device (D2D) communication in cellular networks. In *2013 IEEE International Conference on Communications Workshops,* (pp. 101–105).

Wong, V. W. (Ed.). (2017). *Key technologies for 5G wireless systems*. Cambridge University Press.

Xiaolong, H., Huiqi, Z., Lunchao, Z., Nazir, S., Jun, D., & Khan, A. S. (2021). Soft Computing and Decision Support System for Software Process Improvement: A Systematic Literature Review. *Scientific Programming*, *2021*, 1–14.

Xi, W., Li, X. Y., Qian, C., Han, J., Tang, S., Zhao, J., & Zhao, K. (2014). KEEP: Fast secret key extraction protocol for D2D communication. In *2014 IEEE 22nd International Symposium of Quality of Service,* (pp. 350–359).

Yang, C. C., Wang, R. C., & Liu, W. T. (2005). Secure authentication scheme for session initiation protocol. *Computers & Security*, *24*(5), 381–386.

Yang, M. J., Lim, S. Y., Park, H. J., & Park, N. H. (2013). Solving the data overload: Device-to-device bearer control architecture for cellular data offloading. *IEEE Vehicular Technology Magazine*, *8*(1), 31–39.

Yao, J., Wang, T., Chen, M., Wang, L., & Chen, G. (2016). GBS-AKA: Group-based secure authentication and key agreement for M2M in 4G network. In *2016 International*

Conference on Cloud Computing Research and Innovations, (pp. 42–48).

Yilmaz, O. N., Li, Z., Valkealahti, K., Uusitalo, M. A., Moisio, M., Lundén, P., & Wijting, C. (2014, April). Smart mobility management for D2D communications in 5G networks. In *2014 IEEE Wireless Communications and Networking Conference Workshops*, (pp. 219–223).

Yin, W., Wen, Q., Li, W., Zhang, H., & Jin, Z. (2018). An anti-quantum transaction authentication approach in blockchain. *IEEE Access*, *6*, 5393–5401.

Zen, K., Javed, M., Lenando, H., Zen, H., & Khan, A. S. (2015). Intelligent coordinator selection mechanism (ICSM) for IEEE802. 15.4 Beacon-Enabled MAC protocol in mobile wireless sensor networks [J]. *International Review on Computers and Software*, *10*(2), 164.

Zhang, A., Wang, L., Ye, X., & Lin, X. (2016). Light-weight and robust security-aware D2D-assist data transmission protocol for mobile-health systems. *IEEE Transactions on Information Forensics and Security*, *12*(3), 662–675.

Zhang, A., Zhou, L., & Wang, L. (2016). *Security-aware device-to-device communications underlaying cellular networks*. Springer International Publishing.

Zhang, H., Liu, N., Chu, X., Long, K., Aghvami, A. H., & Leung, V. C. (2017). Network slicing based 5G and future mobile networks: mobility, resource management, and challenges. *IEEE Communications Magazine*, *55*(8), 138–145.

Zhang, K., & Xuemin, S. (2015). Security and privacy for mobile healthcare networks. *IEEE Wireless Communications*, *22*(4), 104–112.

Zhao, C., Yang, S., Yang, X., & McCann, J. A. (2016). Rapid, user-transparent, and trustworthy device pairing for d2d-enabled mobile crowdsourcing. *IEEE Transactions on Mobile Computing*, *16*(7), 2008–2022.

Zhou, Z., Dong, M., Ota, K., Wu, J., & Sato, T. (2014). Energy efficiency and spectral efficiency tradeoff in device-to-device (D2D) communications. *IEEE Wireless Communications Letters*, *3*(5), 485–488.

Zubair, S., Fisal, N., Abazeed, M. B., Salihu, B. A., & Shahid Khan, A. S. (2015). Lightweight distributed geographical: A lightweight distributed protocol for virtual clustering in geographical forwarding cognitive radio sensor networks. *International Journal of Communication Systems*, *28*(1), 1-18.

Zulhasnine, M., Huang, C., & Srinivasan, A. (2010). Efficient resource allocation for device-to-device communication underlaying LTE network. In *2010 IEEE 6th International Conference on Wireless and Mobile Computing, Networking and Communications*, (pp. 368–375).