Security Framework and Jamming Detection for Internet of Things

Babar, Sachin D.

Publication date:
2015

Document Version
Accepted author manuscript, peer reviewed version

SECURITY FRAMEWORK AND JAMMING DETECTION FOR INTERNET OF THINGS

A DISSERTATION

SUBMITTED TO THE DEPARTMENT OF

ELECTRONIC SYSTEM

OF

AALBORG UNIVERSITY

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS

FOR THE DEGREE OF

DOCTOR OF PHILOSOPHY

SACHIN DILIP BABAR

FEB 25, 2015

Supervisor:

Associate Professor Neeli R. Prasad, CTiF, Aalborg University, Aalborg, Denmark

The Assessment Committee:

Professor Josef Noll , Department of Informatics, University of Oslo, Norway

Professor Milica Pejanovic-Djurisic, Faculty of Electrical Engineering, University of Montenegro, Montenegro

Associate Professor Zheng-Hua Tan (Chairman), Department of Electronic Systems, Aalborg University, Denmark

Moderator:

Associate Prof. Albena D. Mihovska, Department of Electronic Systems, Aalborg University, Denmark

Date of Defence: Feb 25, 2015

ISBN: 978-87-7152-065-1

Dedicated to Almighty God SHREE GANESHA and My Beloved Parents

Abstract

The Internet of Things (IoT) consists of billions of people, things and services having the potential to interact with each other and their environment. This highly interconnected global network structure presents new types of challenges from a security, trust and privacy perspective. Hence, security for IoT will be a critical concern that must be addressed in order to enable several current and future applications. The resource constrained devices such as cell phones, PDAs, RFIDs, sensor nodes etc. are the part of IoT. Design process for securing these resource constrained devices is guided by factors like small form factor, good performance, low energy consumption, and robustness to attacks. These design constraints forces us to think of integrating the security features right in to the hardware and software parts of the devices which is also called as embedded security. The research concentrates on embedded security in perspective of software approaches. The IoT system become prone to different security attack, out of all that system is more prone to jamming attack. The goal of research is to design the embedded security framework for IoT and to model the jamming attack and design the defensive technique for Wireless Sensor Network (WSN)-based IoT.

The first part of the thesis proposes the embedded security framework for IoT. The research gives a detailed survey and analysis of embedded security especially in the area of IoT and proposes the security model and threat taxonomy for IoT. The research also highlights the need to provide in-built security in the device itself to provide a flexible infrastructure for dynamic prevention, detection, diagnosis, isolation, and countermeasures against successful breaches. The research proposes the embedded security framework as a feature of software/hardware co-design methodology.

The security framework for IoT also proposes the AES-GCM-based security protocol. The proposed protocol is divided into two components: first is the creation of capability and second component is an application of AES–GCM. AES-GCM is one of the latest authenticated encryption algorithms which provides both message encryption and authentication and can be a good option which will be suited for IoT. AES-GCM core uses a binary Galois Field Multiplier (GFM) for authentication; together with a high-performance AES counter mode cipher to provide high-speed encryption.

The next part of research addresses jamming attack, which is one of the most destructive security attack in the WSN-based IoT. Jamming attack jams the traffic in network by blocking the channel. The different kinds of jamming attack are modelled using unified modelling language (UML). The thesis uses the sequential- and activity- modelling UML approaches to model the behaviour of the jamming attacks. The behavioural modelling and analysis of jamming attack in realistic situations (e.g. sensing in industrial application by following all network rules), gives the clear understanding of jamming attack execution. The research also evaluated the different jamming attack under realistic situations and forms the guidelines to design the countermeasure for jamming attack. The analysis of jamming attack gives the possibility of new kind of jamming attack inside cluster-based network.

The research defines the novel threshold-based countermeasure for reactive jamming attack. The threshold-based jamming countermeasure (TJC) allows the attack into the network and starts its defensive mechanism once it detects the assaults in a network. It uses threshold based mechanism to detect the attack and to cure it. It first detects the jamming node, then informs all neighbouring node about jammer node. The simulation results show that TJC perform in better manner in existence of reactive jamming attack. It demonstrates good performance of TJC by varying traffic interval and number of malicious nodes in

network. The TJC algorithm is further modified for cluster-based intelligent jamming attack. It also shows good performance under the presence of jamming attack.

The research proposes the game-theory- based countermeasure for detecting different kind of jamming attacks in the network. First, the jamming game is modelled to understand the different moves during attack and non-attack conditions. The game theoretic solution is developed by understanding the game moves. The solution uses the different cross-layer features to design the countermeasures. The proposed detection mechanism shows better energy consumption, throughput, and delay in different realistic situations of network (e.g. varying- amount of traffic and number of malicious nodes) as compared to state-of-art solutions.

The research also contributes in key-management algorithm by proposing cluster-based key management algorithm. The algorithm focused on the management and maintenance of keys under cluster based mobile WSN network. The scheme consider two phases, first for key maintenance which establish the two private keys, home key for own cluster and foreign key when node moves from one cluster to another. The second phase maintain the keys when cluster head (CH) moves from one cluster to another. The proposed algorithm improves the efficiency of key management algorithm in terms of security, mobility, energy efficiency, and scalability of network. The simulation of scheme in different realistic situation shows that proposed solution shows less computational overheads, energy consumption, and delay as compared with state-of-art solution.

The outcome for PhD thesis is proposal for,

- IoT embedded security framework
- IoT threat taxonomy.
- Modelling of jamming attack and proposal for new kind of jamming attack
- Threshold-based countermeasure to detect reactive- and intelligent CH jamming attack.
- Game-theory-based countermeasure for detecting jamming attack by using cross-layer features.
- Efficient key management algorithm for managing the keys under cluster-based mobile WSN network.

In summary, this thesis addresses many important topics of embedded security with special focus on jamming attack detection and defence mechanism and on novel key management for mobile cluster-based WSN. The framework, methods, and techniques proposed in this thesis are, for the most part, applicable to the IoT networks and ubiquitous computing.

Keywords: Embedded security, Internet of Things, Security, Privacy, Wireless sensor networks (WSNs), behavioral modelling, activity modelling, sequential modelling, security attacks, Jamming attacks, media access control (MAC), game Theory, cluster, key management, mobility.

Abstrakt

Tingenes Internet (IoT) består af milliarder af mennesker, ting og tjenester med potentiale til at interagere med hinanden og deres omgivelser. Denne stærkt indbyrdes forbundne globale netværksstruktur præsenterer nye typer af udfordringer fra en sikkerhed, tillid og personlige perspektiv. Derfor vil sikkerhed for IoT være en kritisk bekymring, der skal løses for at aktivere flere aktuelle og fremtidige programmer. Ressourcen begrænset enheder såsom mobiltelefoner, PDA'er, RFID, sensor noder etc. er del af tingenes internet. Designproces for at sikre disse resource begrænset enheder er styret af faktorer som lille formfaktor, god ydeevne, lavt energiforbrug og robusthed til angreb. Disse design begrænsninger tvinger os til at tænke på at integrere sikkerhed funktioner ret i til hardware og software delene af enhederne, som kaldes også som integreret sikkerhed. Forskningen koncentrerer sig om integreret sikkerhed i perspektiv af software tilgange. IoT systemet blive udsat for forskellige sikkerhed angreb, ud af al denne ordning er mere udsat for jamming angreb. Målet med forskningen er design integreret sikkerhed rammerne for IoT og model jamming angreb og design den defensive teknik for trådløs Sensor netværk WSN-baserede IoT.

Den første del af afhandlingen foreslår integreret sikkerhedsramme for IoT. Forskningen giver en detaljeret undersøgelse og analyse af integreret sikkerhed især i området af tingenes internet og foreslår sikkerhed model og trussel taksonomien for IoT. Forskningen fremhæver også behovet for at levere indbygget sikkerhed i selve enheden til at levere en fleksibel infrastruktur for dynamisk forebyggelse, opdagelse, diagnose, isolation og modforanstaltninger mod vellykket overtrædelser. Forskningen foreslår integreret sikkerhedsramme som en funktion af software/hardware Co design metode.

Sikkerhedsmiljøet for IoT foreslår også en AES-GCM-baserede sikkerhedsprotokol. Den foreslåede protokol er opdelt i to komponenter: først er oprettelsen af kapacitet og anden komponent er en anvendelse af AES-GCM. AES-GCM er en af de nyeste godkendte krypteringsalgoritmer, der giver både besked kryptering og godkendelse og kan være en god mulighed, som vil være egnet til IoT. AES-GCM core bruger en binær Galois felt multiplikator (Feltmarskal) til godkendelse; sammen med en højtydende AES counter tilstand cipher at levere højhastigheds kryptering.

Den næste del af forskning adresser jamming angreb, som er en af de mest destruktive sikkerhed angreb i de WSN-baserede IoT. Jamming angreb syltetøj trafikken i netværket ved at blokere kanalen. De forskellige former for jamming angreb er modelleret ved hjælp af unified modelling language (UML). Afhandlingen bruger de sekventielle - og aktivitet - modellering UML tilgange til model adfærd jamming-angreb. Den adfærdsmæssige modellering og analyse af jamming angreb i realistiske situationer (fx sensing i industriel anvendelse ved at følge alle netværk regler), giver en klar forståelse af jamming angreb udførelse. Forskningen også evalueret forskellige jamming angrebet under realistiske situationer og former retningslinjer til at designe modtræk til jamming angreb. Analyse af jamming angreb giver mulighed for nye slags jamming angreb inde klynge-baseret netværk.

Forskningen definerer den roman tærskel-baserede modtræk til reaktiv jamming angreb. Den tærskel-baserede jamming modforanstaltning (TJC) giver mulighed for angrebet ind i netværket og starter sin defensive ordning, når det registrerer angrebene i et netværk. Det

bruger tærskel baseret mekanisme til at registrere angreb og helbrede den. Det første registrerer noden jamming, så oplyser alle tilstødende node om jammer node. Simuleringen resultaterne viser, at TJC udfører i bedre måde i eksistensen af reaktive jamming angreb. Det viser gode resultater af TJC af varierende trafik interval og antallet af ondsindede noder i netværk. TJC algoritme er yderligere ændret til klynge-baserede intelligente jamming angreb. Det viser også gode resultater under tilstedeværelse af jamming angreb.

Forskningen foreslår spillet-teori-baserede modtræk til påvisning af forskellige slags jamming angreb i netværket. Først, jamming spillet er modelleret til at forstå de forskellige bevægelser under angreb og ikke-angreb betingelser. De spil teoretisk løsning er udviklet af forståelse spillet flytter. Løsningen bruger forskellige cross-lag til at designe modforanstaltningerne. Den foreslåede detection mekanisme viser bedre energiforbrug, overførselshastighed og forsinkelse i forskellige realistiske situationer af netværk (f.eks. varierende mængde af trafik og antallet af ondsindede noder) i forhold til state-of-art løsninger.

Forskningen bidrager også i nøgleadministration algoritme ved at foreslå klynge-baserede nøglehåndtering algoritme. Algoritmen fokuseret på forvaltning og vedligeholdelse af nøglerne under klynge baseret ambulant WSN netværk. Ordningen overveje to faser, først til central vedligeholdelse, der etablerer to private nøgler, starttasten for egen klynge og fremmed nøgle når node flytter fra én klynge til en anden. Den anden fase opretholde nøglerne når klynge hovedet (CH) bevæger sig fra én klynge til en anden. Den foreslåede algoritme forbedrer effektiviteten af nøglehåndtering algoritme med hensyn til sikkerhed, mobilitet, energieffektivitet og skalerbarhed af netværk. Simulering af ordningen i forskellige realistiske situation viser, at løsningsforslag viser mindre beregningsmæssige overhead, energiforbrug og forsinkelse sammenlignet med state-of-art løsning.

Resultatet for ph.d.-afhandling forslag til,

- IoT integreret sikkerhedsramme
- IoT trussel taksonomi
- Modellering af jamming angreb og forslag til nye slags jamming angreb
- Tærskel baseret modtræk til at opdage reaktiv- og intelligent CH jamming angreb.
- Spilteori baseret modtræk til påvisning af jamming angreb ved hjælp af cross-lag funktioner.
- Effektiv nøglehåndtering algoritme til styring af nøgler under klynge-baserede mobile WSN netværk.

I Resumé omhandler denne afhandling mange vigtige emner af integreret sikkerhed med særlig fokus på jamming attack detection og forsvar mekanisme og roman nøglehåndtering for mobile klynge-baseret WSN. Ramme, metoder og teknikker, der foreslås i denne afhandling er for det meste gælder for tingenes internet netværk og allestedsnærværende computing.

Nøgleord: Integreret sikkerhed, tingenes Internet, sikkerhed, privatliv, trådløs sensornetværk (WSNs), adfærdsmæssige modellering, aktivitet modellering, sekventiel modellering, sikkerhed angreb, Jamming angreb, media access control (MAC), spilteori, klynge, nøglehåndtering, mobilitet.

Acknowledgements

We believe, “No matter how big or small an endeavor is, we do nothing in vacuum! We do it because of the supporting roles of many others”. Here I would like to express my thanks to all those who contributed in many ways to the success of this PhD study and made it an unforgettable experience for me.

Foremost, I would like to express my sincere gratitude to my Supervisors Associate Professor Dr. Neeli R. Prasad and Professor Ramjee Prasad for their guidance and continuous support both while I was considering to apply to Aalborg University as well as during my time here as a PhD student. I will be very grateful to them throughout my life for giving me the opportunity to work at CTiF and pursue my PhD here.

Archimedes once said, “Give me a firm place to stand upon and I can move the Earth”. This very platform is given to the PhD Students by Professor Ramjee Prasad to carry out our research work. In a nutshell, I would like to say that ‘Moments are cherished for the expressions they make’ and learning from such an esteemed personality is one of these moments.

I am very much thankful to my supervisor, Dr. Neeli R. Prasad, for guiding me through this work and keeping faith in me. This work would not have been possible without her guidance, support and encouragement. Under her guidance I successfully overcame many difficulties and learned a lot. I am deeply indebted to Dr. Neeli R. Prasad for her tireless and unconditional help and being a role model for me throughout the journey of research.

I am very thankful to Parikshit Mahalle for collaborating with me and his invaluable advice concerning the implementation of many publications. Furthermore, I am thankful to all my GISFI colleagues from the department for their continuous support and cooperation during these five years of PhD. I am also thankful to Jens Erik, Prof. Fleming, and Kirsten Jensen for making my stay at Aalborg, a memorable and comfortable. My special thanks to Mrs. Jyoti Prasad, Mr. Rajiv Prasad for making my stay much comfortable with their love and support. Their affection and care is memorable.

My PhD program at Aalborg University has been funded by Sinhgad Technical Education society (STES), Pune, India. I am indebted to Honourable founder president of STES, Prof. M. N. Navale, founder secretary of STES, Dr. Mrs. S. M. Navale, Dr. A.V. Deshpande, Dr. S. S. Inamdar, Dr. S. D. Markande, Dr. M. S. Gaikwad for their faith on me and inexplicable support. I am also very thankful to all my department colleagues at SIT, Lonavala especially Nitin Dhawas, Vilas Deotare and Pallavi Ahire for their kind support and help during these five years of my PhD.

I would like to thank my parents, sisters and brother-in-laws for supporting me and encouraging me with their best wishes. I owe a lot to my parents and sisters, who encouraged and helped me at every stage of my personal and academic life, and longed to see this achievement come true. Finally, I would like to thank my wife Sheetal. She was always there cheering me up and stood by me through the good times and bad. I would also like to thank my son Avaneesh for making me forget all the pressure with his innocent smile.

Last but not the least, I would like to also thanks to all those who directly and indirectly involved in building this thesis and research work.

TABLE OF CONTENTS

Contents

List of Figures

List of Tables

List of Acronyms

IoT	Internet of Things
PDA	Personal digital assistant
CH	Cluster Head
RFID	Radio Frequency Identification
WSN	Wireless Sensor Networks
MAC	Media Access Control
PKI	Public-key infrastructure
ARPANET	Advanced Research Projects Agency Network
PGP	Pretty Good Privacy
DoS	Denial of Service
SSO	Single sign-on
DHCP	Dynamic Host Configuration Protocol
GSM	Global System for Mobile Communications
UMTS	Universal Mobile Telecommunications System
WiMAX	Worldwide Interoperability for Microwave Access
PC	Personal computer
DRM	Digital Rights Management
AP	Access Point
AES	Advanced Encryption Standard
GCM	Galois/Counter Mode
BS	Base Station
TJC	Threshold-based Jamming Countermeasure
GPP	General purpose processors
ECC	Elliptical Curve Cryptography
ASIC	Application Specific Integrated Circuits
FPGA	Field Programmable Gate Array
SoC	System on Chip
IC	Integrated circui**t**
ID	Identifier
IPsec	Internet Protocol Security
OTP	One-Time-Programmable
JTAG	Joint Test Action Group
SEE	Secure Execution Environment
GF	Galois Field
UML	Unified Modeling Language
QoS	Quality of service
ACM	Access Control Matrix
ACL	Access Control List
CAC	Capability based Access Control
RTS	Request to Send
CTS	Clear to Send
LEACH	Low Energy Adaptive Clustering Hierarchy
AODV	Ad Hoc On Demand Distance Vector
UDP	User Datagram Protocol
NAV	Network Allocator Vector
EDDK	Energy-Efficient Distributed Deterministic Key Management
CMKMS	Cluster based Mobile Key Management Scheme

1

Introduction

The goal of this chapter is to explain the motivation, challenges and security requirements for Internet of Things (IoT). Key issues and milestones for different security architectures are explained in order to get the synopsis of the thesis. Goals and objects of research are elucidated in this chapter. The scientific contributions of this thesis are explained, and the details of related publications are provided. Finally, the outline of the thesis is provided to give an overview of the individual chapters.

1.1 Introduction

The Internet has undergone severe changes since its first launch in the late 1960s as an outcome of the ARPANET with number of users about 20% of the world population. "7 trillion wireless devices serving 7 billion people in 2017". This vision reflects the increasing trend of introducing micro devices and tools in future. The Future of internet i.e. Internet of Things(IoT) will pervade all aspects of our lives, capturing, storing, and communicating a wide range of sensitive and personal data anywhere anytime. With the objectives of IoT, all objects will be able to exchange information and, if necessary actively process information according to predefined schemes, which may or may not be deterministic. In such ambient environment not only user become ubiquitous but also devices and their context become transparent and ubiquitous. With the miniaturization of devices, increase of computational power, and reduction of energy consumption, this trend will continue towards IoT[1].

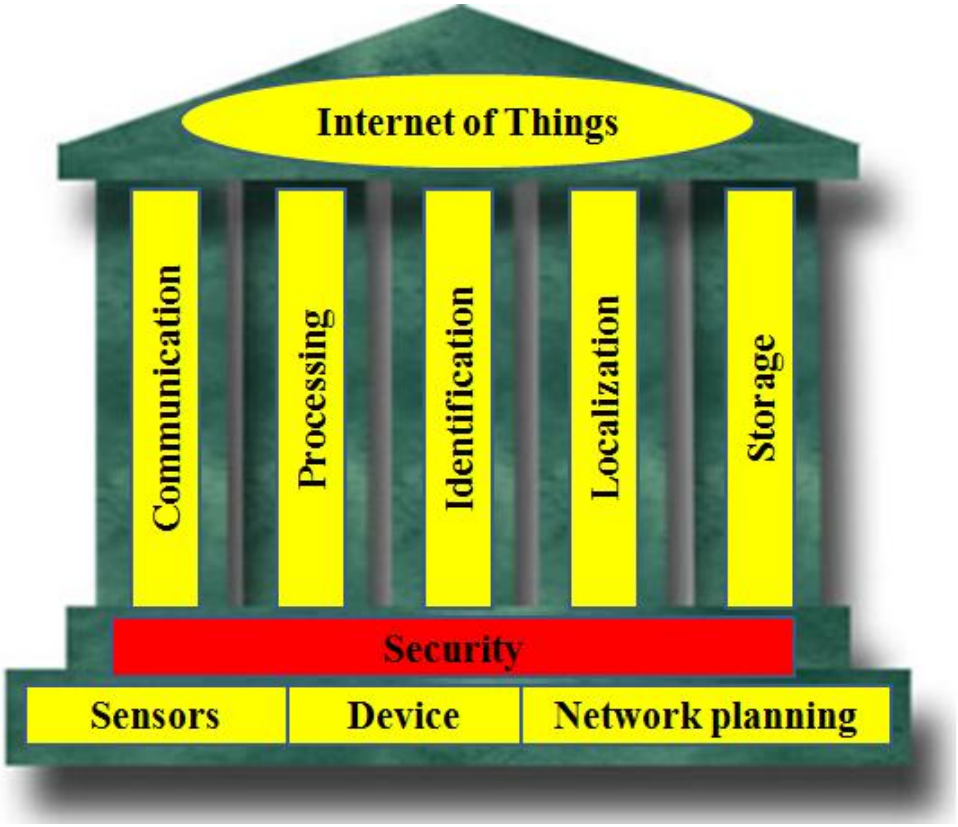

Figure 1.1: IoT Pillars

Figure 1.1 shows the house for IoT which is build from all the components required for communication and connectivity. Communication, data processing, identification, localization and storage will be the pillars for IoT which will enable any-to-any and anywhere connectivity. Security, Sensor device and network planning will be the base on which the pillars of IoT will reside. IoT will connect things to users, business and to other things using combination of wired and wireless connectivity. The effectiveness and efficiency of these

systems will be important and crucial which will enable new forms of connectivity which should be inexpensive with support to standard Internet protocols. Most of the devices in the IoT will be used in two broad areas:

1. Critical Infrastructure: power production/generation/distribution, manufacturing, transportation, etc.
2. Personal infrastructure: personal medical devices, automobiles, home entertainment and device control, retail, etc.

Critical infrastructure represents an attractive target for national and industrial espionage, denial of service and other disruptive attacks. Internet connected things that touch very sensitive personal information is the high priority targets for cyber criminals, identity theft and fraud. Both these areas will demand new technology requiring new approaches to security and a major change in the way security is architected, delivered and monitored.

IoT will demand new approaches to security like a secure lightweight operating system, scalable approaches to continuous monitoring and threat mitigation, and new ways of detecting and blocking active threats. One of the most challenging topics in such an interconnected world of miniaturized systems and sensors are security and privacy aspects. Having every 'thing' connected to the global future IoT and communication with each other, new security and privacy problems arise, e. g., confidentiality, authenticity, and integrity of data sensed and exchanged by 'things'. Due to manifold aspects that involves, security for IoT will be a critical concern that must be addressed in order to enable several current and future applications [2,3].

1.2 IoT Scenario and Objectives

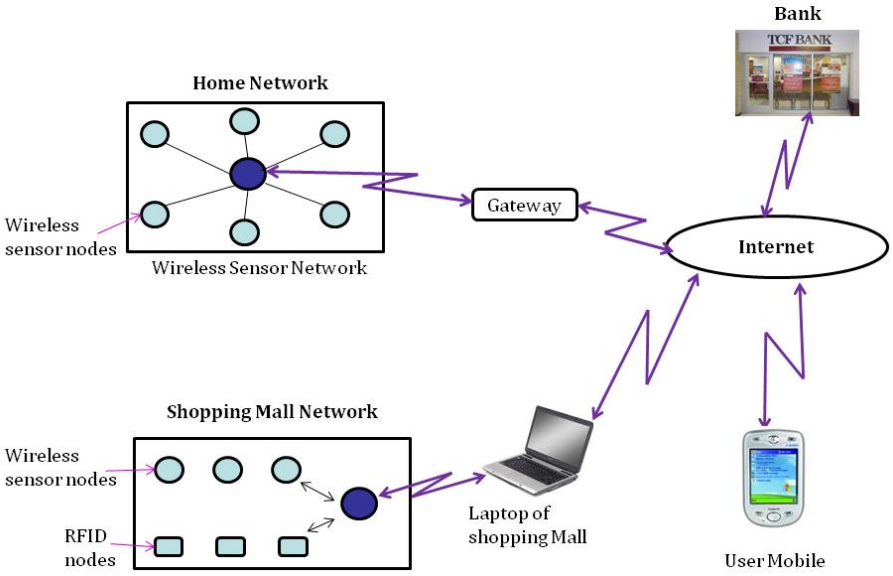

Figure 1.2: Virtual Shopping Scenario for IoT

Consider a virtual shopping scenario as shown in figure 1.2. Suppose you are at your office, and one of your family member demands for a matching sofa set for your hall. Because of

office constraints you cannot go to the shopping mall to do the needful. You also do not know about the size and color that will best suit your hall. Now to avoid the travelling back home and going to the shop, you can just call your home network through your mobile device sitting at your office and connect to your home network through different wireless technologies. The home network consists of multiple sensors/wireless devices. You can call in your home network and connect to the camera located in the home. You view the hall and take a remote picture of the hall from a suitable angle. On similar lines you can connect to the network of the shopping mall, and select the item that best suits your hall. After finalizing the item, now you do the payment by connecting to the bank and transfer the amount to the shopping mall store account.

By using different networks and devices as shown in figure 1.2 we have just left our homes, mobile and bank information open to hackers and thieves. Apart from the security present in the existing networks, we will have to focus on the security aspects of all the resource constrained devices involved in the communications. Existing networks are inadequate to meet the security needs of data sensitive applications. Hence in security terms we need to identify two areas which need to be secured i.e. network security and device security.
The IoT scenarios, like individual wireless device interfacing with internet, constellation of wireless devices, pervasive system and sensor network, are associated with new network service requirements that motivate rethinking of several Internet architecture issues. Several mobile/wireless features may require mechanisms that cannot be implemented through the conventional IP framework for the Internet, or if they can, may suffer from performance degradation due to the additional overhead associated with network protocols that were originally designed for static infrastructure computing [3]. We therefore discuss a set of objectives related to the networking requirements of the representative IoT scenarios identified earlier. Figure 1.3 shows the IoT Objectives followed by their description.

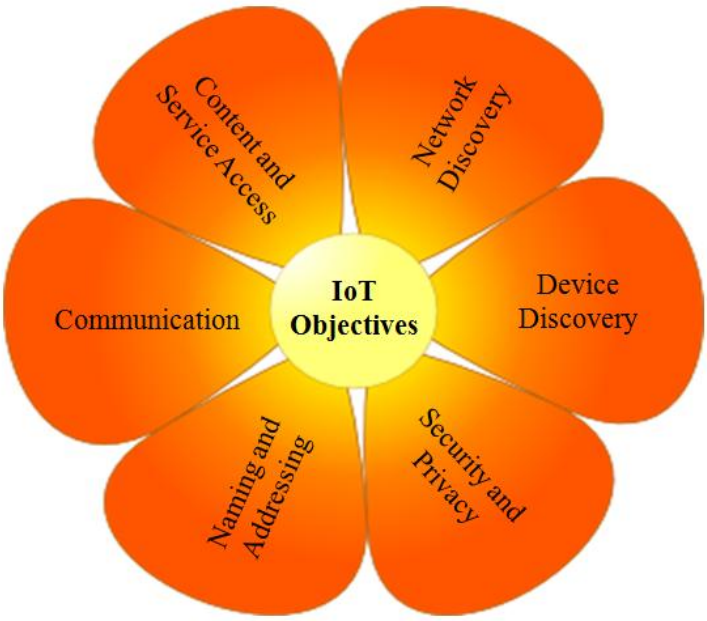

Figure 1.3: IoT Objectives

1. Naming and Addressing

Today's Internet addressing scheme is rather rigid; it is well suited to a static, hierarchical topology structure. It provides a very efficient way to label (and find) each device interface in this hierarchy. To support mobility and routing, the next generation Internet must provide ways to name and route to a much richer set of network elements than just attachment points. A clean architectural separation between name and routable address is a critical requirement for IoT[4,5].

2. Device Discovery and Network Discovery

The current Internet is text-dominated with relatively efficient search engines for discovering textual resources with manual configuration. An Internet dominated by unstructured information supplied from large numbers of sensor devices must support efficient mechanisms for discovering available sensor resources. The new architecture must support methods for the registration of a new sensor system in the broader network [6,7].

3. Content and Service access

A new architecture should provide data cleansing mechanisms that prevent corrupted data from propagating through the sensor network. In particular, services that maintain device calibration and monitor/detect adversarial manipulation of sensor devices should be integrated into sensor networks. This could be realized through obtaining context information, metadata, and statistical techniques to locally detect faulty inputs [6,8,9].

4. Communication

Wireless devices should be able to operate independently of the broader internet. In particular, there may be times during which the connection of a wireless device or, network to the internet is not available. During these times, wireless devices should be able to operate stably in modes disconnected from the rest of the infrastructure, as well as be able to opportunistically establish "local" ad-hoc networks using their own native protocols. In particular, this means that issues such as authorization and updating the device state should be seamless, with minimal latency [5,9].

5. Security and Privacy

Wireless networks can be expected to be the platform of choice for launching a variety of attacks targeting the new Internet. At the most basic level, wireless devices will likely have evolving naming and addressing schemes and it will be necessary to ensure that the names and addresses that are used are verifiable and authenticated. One parameter uniquely associated with wireless networks is the notion of location. Location information provided by the network should be trustworthy [9]. Additionally the architecture should provision hooks for future extensions to accommodate legal regulations.

1.3 Problem statement

This section describes the motivation and problem statement along with the hypothesis and the methodology.

1.3.1 Motivation and Problem Statement

The Internet of Things (IoT) consists of billions of people, things and services having the potential to interact with each other and their environment. This highly interconnected global network structure presents new types of challenges from a security, trust and privacy perspective. Hence, Security for IoT will be a critical concern that must be addressed in order to enable several current and future applications. The resource constrained devices such as cell phones, PDAs, RFIDs, sensor nodes etc. are the part of IoT. Design process for securing these devices is guided by factors like small form factor, good performance, low energy consumption, and robustness to attacks. Following are the challenges which need to be tackled in the world of pervasive devices.

- Management, scalability and heterogeneity of devices
- Networked knowledge and context
- Privacy, security and trust will have to be adapted to both devices and information

This will involve the development of highly efficient cryptographic algorithms and protocols that provide basic security properties such as confidentiality, integrity, and authenticity, as well as secure implementations for the various kinds of mostly resource constrained devices.

Embedded security is growing as a new dimension for resource constrained devices which will integrate the security features right in to the hardware and software parts of the devices. The research concentrates on embedded security in perspective of software services. The IoT system become prone to different security attack, out of all that, system is more prone to jamming attack.

The main goal of the research is to design the embedded security framework for IoT and design the security solutions to save from different jamming attacks and perform efficient key management in cluster based WSN.

To meet above challenges, the main research problem is divided into following sub problems,

- Propose the embedded security framework for IoT: The research gives a detailed survey and analysis of embedded security especially in the area of IoT and proposes the security model and threat taxonomy for IoT. The research also highlights the need to provide in-built security in the device itself to provide a flexible infrastructure for dynamic prevention, detection, diagnosis, isolation, and countermeasures against successful breaches. The research proposes the embedded security framework as a feature of software/hardware co-design methodology.
- Modelling of Jamming attacks and to design efficient defense mechanism against jamming attacks: The research modelled the different kinds of jamming attack using sequential and activity modelling, and proposed the different countermeasures to save from jamming attack. The research also proposed the new kind of jamming attack for cluster based network and suggested the solution for it.
- To specify and design optimized secure key management for WSN: The research proposes the optimized key management for cluster-based WSN by considering mobility of the nodes and cluster head (CH).

1.3.2 Hypothesis

It is hypothesized that the Threat Taxonomy for IoT, jamming attack modelling, jamming attack detection, defence mechanisms, and efficient key management will constitute the security framework for IoT. The research divides the main hypothesis into small hypothesis. It is hypothesized that the proposal for embedded security protocol takes into consideration the resource constraints of IoT devices i.e. battery life, processing power and computation time. The new threat taxonomy will identify the level of threats, to find mitigation on it. Modelling of jamming attack using UML based modelling is used to understand the behaviour of attack. Evaluation of jamming attack and new different possible attack on cluster based network is proposed. Threshold based and game theory based solutions to identify and mitigate the jamming attack is developed for cluster-based WSN. The key management solution is developed for cluster-based WSN by considering mobility in the network.

A comprehensive hypothesis comprises:

A. It is hypothesized that, the proposed threat taxonomy for IoT will address the security requirements in broader aspect and will be helpful for framing the security framework for IoT which takes into consideration the resource constraints of devices of IoT.
B. It is hypothesized that the proposed mutual authentication process based on AES-GCM will improve resistance to attack and efficiency of network in presence of attacks.
C. It is hypothesized that, the modelling of jamming attacks using UML approach gives the clear understanding of attack penetration and it will be useful for developing solution on jamming attack. It is also hypothesized that the modelling of jamming attack gives the notion to propose new possibility of attacks. The evaluation of jamming attack is performed by considering varying traffic rate and number of malicious nodes in the network.
D. Using the proposed threshold-based jamming countermeasure, it is hypothesized that the reactive jamming attack can be detected and mitigated, to enhance the security. It is also hypothesized that the approach considered will be efficient in realistic network conditions.
E. The game theory based solution for jamming detection and mitigation hypothesize that the cross-layer features will be useful to take secure moves during jamming game. It is also hypothesized that the proposed solution will be energy and delay efficient as compared with state-of-art solutions.
F. The last hypothesis is that the key management technique will help to build a more strong security framework but it should be modified according to current need of applications. The key management technique is developed by considering the mobility conditions of network for Mobile Cluster-Based WSN. The key management technique should require less communication and computation cost while managing the key.

The hypothesis addresses the consideration and assumption made for developing the secure framework and jamming detection for IoT. Therefore, dissertation work gives answers to the following questions through this research:

1. What is need of security framework for IoT?
2. How the threat taxonomy helps to address the level of threat?
3. What is need of attack modelling? How to do it? How it helps to develop attack detection and mitigation techniques?
4. Will the threshold-based decision lead to correct detection of attack?
5. Will the cross-layer features help to improve security decisions?
6. How the lightweight and efficient framework can be develop and applied to IoT security?
7. Will the proposed set of solutions help to make IoT secure against jamming attack?
8. How key management should be addressed in mobile Cluster-based WSN scenarios?

1.3.3 Methodology

The current research problem is divided into three phases as described in the problem statement. The understanding and conclusions of each phase has given motivation to address the next phase in better manner. The first phase of research is to develop the security framework and architecture for IoT performance enhancement. The security model and threat taxonomy for IoT is developed by understanding the available literature in the field. The defined threat taxonomy in research had motivated to extend the work in jamming attack, which is one of the disastrous attacks on WSN. The research had taken the understanding of the currently available approaches for jamming attacks and defined more simpler and understandable models for the jamming attacks. The research modelled the jamming attacks using activity- and sequential- modelling techniques. The research also defined the game theoretic model for playing a different kind of jamming game and given the secure moves to detect and avoid jamming situations in the network. In the last phase of research, the secure key management is developed for mobile nodes. The research is motivated from the current literature in secure key management where very few work addressed the management of keys under mobile environment. The research proposed the efficient key management technique under mobility and compared it with state-of-art available solution. The performance of each phase task is evaluated by using theory assisted designs and comparative simulation using widely used simulation tools in research community. The comparative simulations in thesis are performed by using NS-2 simulator, which is widely used simulator in the research community. The research mainly considers the energy efficiency, computational overheads, delay and throughput of system by varying the number of nodes, number of malicious nodes and traffic interval, which shows the correct efficiency and scalability of system. All the simulations of given solution are performed by considering IEEE 802.15.4 radio model. IEEE 802.15.4 is good for time-critical low power WSN. The research developed is majorly concentrating on industrial, home, and health applications of WSN. All these applications majorly considers low rate wireless personal area network (Low-WPAN).

1.4 Security Architectures

1.4.1 Security Frameworks

Security framework for IoT will mainly include architectures for providing and managing access control, authentication, and authorization. It will provide methods for controlling the identification and authentication of users and for administering which authenticated users are granted access to protected resources. Some of the existing frameworks described can be used to provide several functions as shown in Table 1.1.

Table 1.1 State of Art Evaluation

Sr. No.	Framework	Identity Certificate Management	Single Sign-on	Federated Identity	User-centric	Device Security
1	PKI[10]	√				
2	PGP[11]	√				
3	Kerberos[12]		√			
4	Windows Live ID[13]		√		√	
5	OpenID[14]		√		√	
6	Liberty Alliance[15]		√	√	√	
7	WS-Federation[16]		√	√		

1. Identity Certificate Frameworks

These frameworks allow users without prior contact to authenticate to each other and digitally sign and encrypt messages. They are based on identity certificates, which are certificates that bind a public key to an identity. Examples of identity certificate frameworks include Public Key Infrastructures (PKIs), and Pretty Good Privacy (PGP).

2. Single Sign-on

Single sign-on (SSO) allows users to be authenticated only once in a system. Users can then access all resources for which they have access permission without entering multiple passwords. Example of SSO frameworks include:

Kerberos: a distributed authentication service, which provides SSO within a single administrative domain.

Windows Live ID: an Internet-based SSO framework used by Microsoft applications and web services such as MSN messenger.

OpenID: an authentication framework that allows users to login to different web sites using a single digital identity, eliminating the need to have different usernames and passwords for each site.

Liberty Alliance: a consortium that aims to establish open standards, guidelines and best practices for federated identity management.
WS-Federation: a federated identity standard developed by Microsoft, IBM, VeriSign, BEA and RSA Security, which forms part of the Web Services Security framework.

3. Identity Federation

Federated Identity allows users of one security domain to securely access resources on another security domain, without the need for another user account. Users register with an authentication server in their own domain and other domains trust its assertions.

4. User-centric identity management

User-centric identity management is a design principle that focuses on usability and cost-effectiveness from the user's point of view. There are three main approaches to user-centric identity management that are managing multiple identities e.g. information cards [15], giving users a single identity e.g. OpenID and, lastly giving users control over access to their resources.

5. Device Security

The Device Security Framework includes device-resident security software as well as security capabilities delivered across the network. The device-resident software is embedded into devices at the time of manufacture. In order to provide security at the physical or execution level, we need to build our security solution based on secure execution environment (SEE). In this respect, Trusted Platform Module (TPM) by Atmel [17] and Trustzone by ARM [18] have done good amount of development in embedded platform security.

1.4.2 Key Properties of IoT

There are a number of key properties of IoT that create several issues for security and raises additional requirements for security[19]. These key properties are listed below:

Mobility: IoT devices are mobile and often generally connected to the Internet via a large set of providers.

Wireless: These devices typically connect to the rest of the Internet via a wide range of wireless links, including Bluetooth, 802.11, WiMAX, Zigbee and GSM/UMTS. With wireless communications, any nearby observer can intercept unique low-level identifiers that are sent in the clear, e.g., Bluetooth and 802.11 device addresses.

Embedded Use: Major IoT devices have a single use (e.g., blood pressure or heart monitors and household appliances). As a result, the detection of communication patterns unique to a specialized device allows users to be profiled[12].

Diversity: These devices span a range of computational abilities from full-fledged PCs to low-end RFID tags. Privacy designs must accommodate even the simplest of devices.

Scale: These devices are convenient, growing in number daily, and increasingly embed network connectivity into everyday settings. This makes it difficult for users to monitor privacy concerns.

1.4.3 High level security requirements

In business process, security requirements are described as shown in figure 1.4.

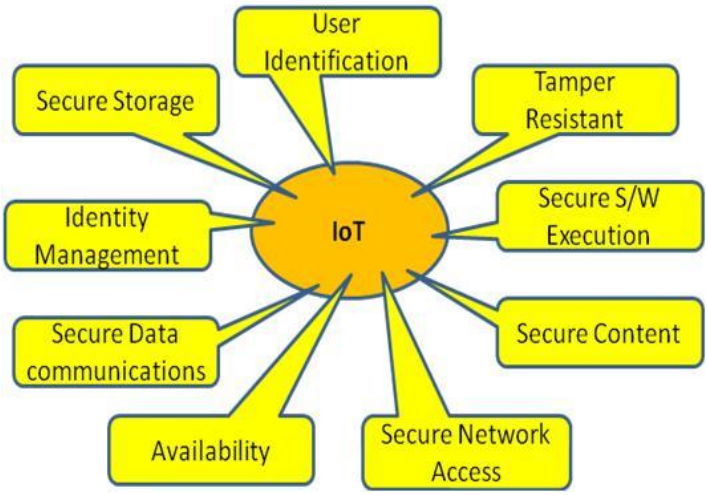

Figure 1.4: High level Security Requirements for IoT

Resilience to attacks: The system has to avoid single points of failure and should adjust itself to node failures.

Data authentication: As a principle, retrieved address and object information must be authenticated.

Access control: Information providers must be able to implement access control on the data provided.

Client privacy: Measures need to be taken that only the information provider is able to infer from observing the use of the lookup system, related to a specific customer; at least, inference should be very hard to conduct.

User identification: It refers to the process of validating users before allowing them to use the system.

Secure storage: This involves confidentiality and integrity of sensitive information stored in the system.

Identity Management: It is broad administrative area that deals with identifying individuals / things in a system and controlling their access to resources within that system by associating user rights and restrictions with the established identity.

Secure data communication: It includes authenticating communicating peers, ensuring confidentiality and integrity of communicated data, preventing repudiation of a communication transaction, and protecting the identity of communicating entities.

Availability: Availability refers to ensuring that unauthorized persons or systems cannot deny access or use to authorized users.

Secure network access: This provides a network connection or service access only if the device is authorized.

Secure content: Content security or Digital Rights Management (DRM) protects the rights of the digital content used in the system.

Secure execution environment: It refers to a secure, managed-code, runtime environment designed to protect against deviant applications.

Tamper resistance: It refers to the desire to maintain these security requirements even when the device falls into the hands of malicious parties, and can be physically or logically probed.

1.5 Security Model and Threat Taxonomy for IoT

This section presents the attack classification for IoT, identifies the threat taxonomy for IoT and based on the key properties and challenges proposes a cube structure security model for IoT.

1.5.1 Security attacks on IoT

The domain of security attacks on embedded device is increasing day by day. Following Figure 1.5 summarizes the attacks on IoT Systems [20-22].

1. Physical attacks

These types of attacks tamper with the hardware components and are relatively harder to perform because it requires expensive material. Some examples are de-packaging of chip, layout reconstruction, micro-probing, particle beam techniques, etc.

2. Side channel attacks

These attacks are based on "side channel Information" that can be retrieved from the encryption device that is neither the plaintext to be encrypted nor the ciphertext resulting from the encryption process. Encryption devices produce timing information that is easily measurable, radiation of various sorts, power consumption statistics, and more. Side channel attacks makes use of some or all of this information to recover the key the device is using. It is based on the fact that logic operations have physical characteristics that depend on the input data. Examples of side channel attacks are timing attacks, power analysis attacks, fault analysis attacks, electromagnetic attacks and environmental attacks.

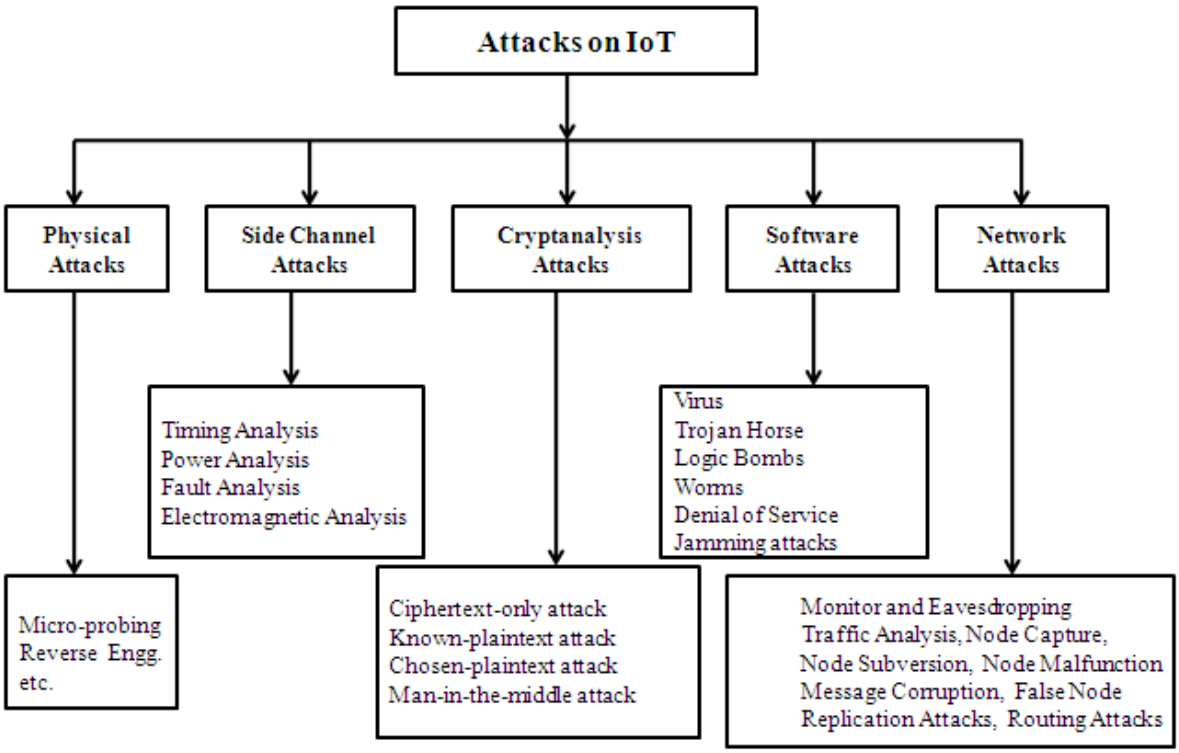

Figure 1.5: Attacks on IoT Devices

3. Cryptanalysis attacks

These attacks are focused on the ciphertext and they try to break the encryption, i.e. find the encryption key to obtain the plaintext. Examples of cryptanalysis attacks include ciphertext-only attack, known-plaintext attack, chosen-plaintext attack, man-in-the-middle attack, etc.

4. Software attacks

Software attacks are the major source of security vulnerabilities in any system. Software attacks exploit implementation vulnerabilities in the system through its own communication interface. This kind of attack includes exploiting buffer overflows and using trojan horse programs, worms or viruses to deliberately inject malicious code into the system. Jamming attack is the one of the ruinous invasion which blocks the channel by introducing larger amount of noise packets in a network. Jamming is the biggest threat to IoT where a network consists of small nodes with limited energy and computing resources. So it is very difficult to adopt the conventional anti jamming methods to implement over IoT.

5. Network Attacks

Wireless communications systems are vulnerable to network security attacks due to the broadcast nature of the transmission medium. Basically attacks are classified as active and passive attacks. Examples of passive attacks include monitor and eavesdropping, Traffic analysis, camouflage adversaries, etc. Examples of active attacks include denial of service attacks, node subversion, node malfunction, node capture, node outage, message corruption, false node, routing attacks, etc

1.5.2 Threat Taxonomy for IoT

IoT is coupled with new security threats and alters overall information security risk profile. Although the implementation of technological solutions may respond to IoT threats and vulnerabilities, security for IoT is primarily a management issue. Effective management of the threats associated with IoT requires a sound and thorough assessment of risk given the environment and development of a plan to mitigate identified threats [23]. Figure 1.6 presents threat taxonomy to understand and assess the various threats associated with the use of IoT.

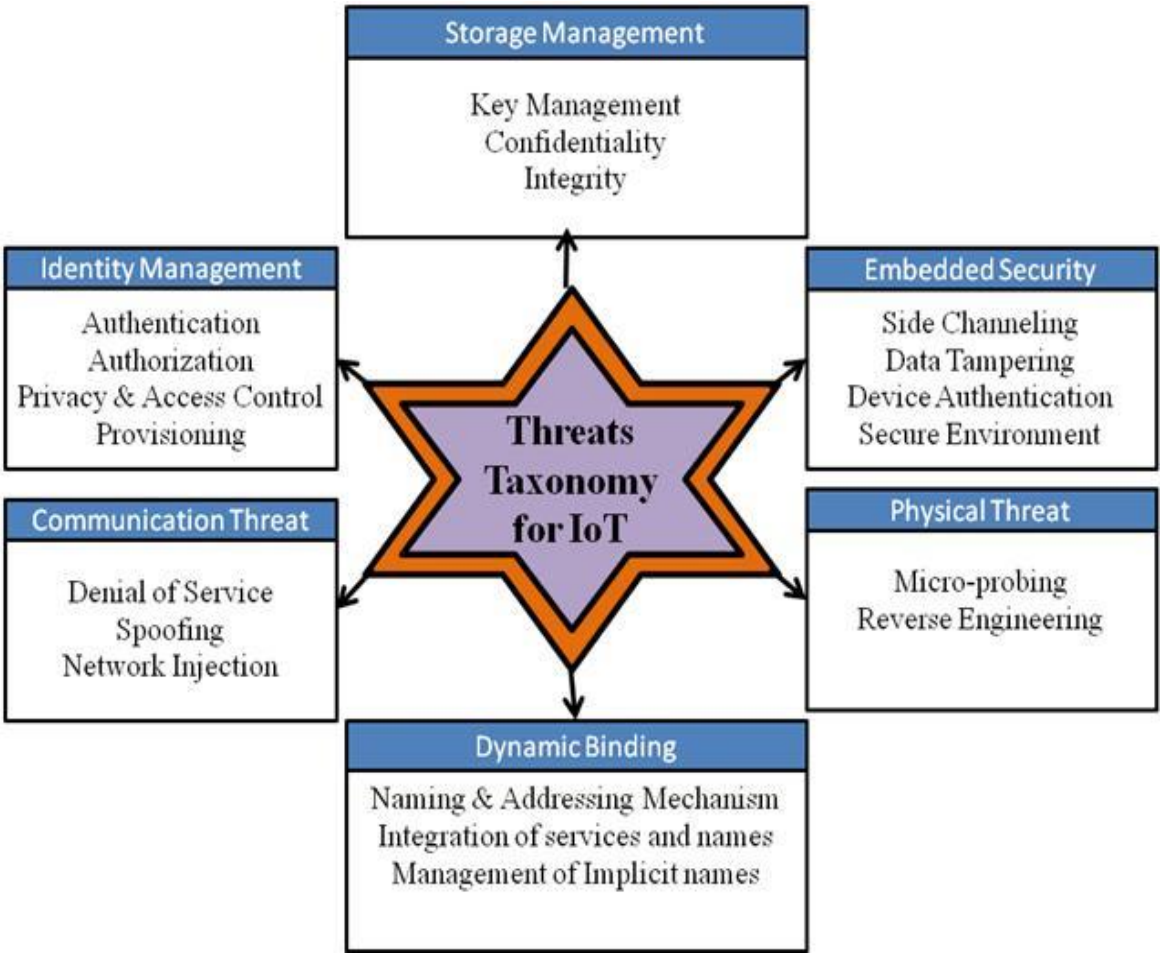

Figure 1.6: Threat Taxonomy for IoT

Identification covers determination of unique device/user/session with authentication, authorization, accounting and provisioning.

Communication threats covers a denial-of-service attack (DoS) and it occurs when an attacker continually bombards a targeted AP (Access Point) or network with bogus requests, premature successful connection messages, failure messages, and/or other commands.

Physical threat includes micro probing and reverse engineering causing serious security problem by directly tampering the hardware components. Some types of physical attack

require expensive material because of which they are relatively hard to perform. Some examples are: de-packaging of chip, layout reconstruction, micro-probing.

Embedded security threat model will span all the threats at physical and MAC layer. Security threats like device and data tampering, side channel analysis, bus monitoring, etc will be the concerns at device level.

Storage management has crucial impact on the key management to achieve confidentiality and integrity. We must also be careful in choosing which cryptographic components to use as the building blocks since, for example, the cipher texts for some public key encryption schemes can reveal identifying information about the intended recipient.

1.5.3 Security Model for IoT

The different possible attacks on IoT and the threat taxonomy give new challenges to security and privacy in end to end communication of things. Protection of data and privacy of things is one of the key challenges in the IoT. Lack of security measures will result in decreased adoption among users and therefore is one of the driving factors in the success of the IoT[24-27]. Figure 1.7 depicts the cube structure model for IoT.

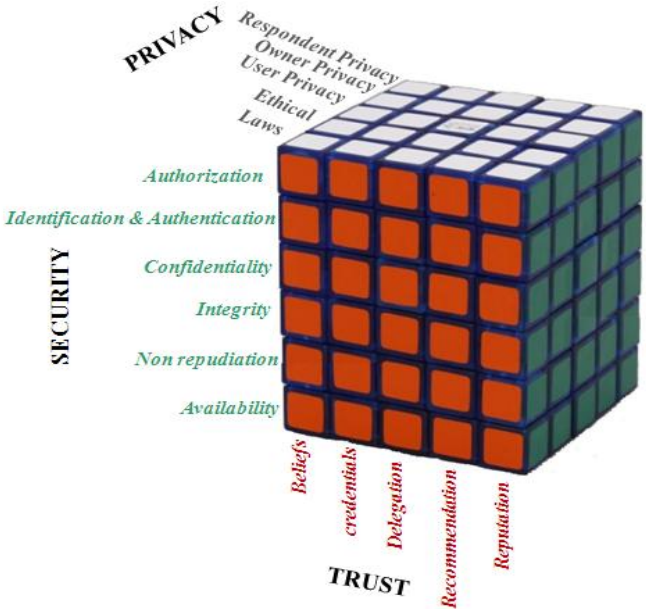

Figure 1.7: Security Model for IoT

Integrated and interrelated perspective on security, trust, privacy can potentially deliver an input to address protection issues in the IoT. Therefore, we have chosen a cube structure as a modelling mechanism for security, trust, and privacy in the IoT. A cube has three dimensions with the ability to clearly show the intersection thereof. Therefore, a cube is an ideal modelling structure for depicting the convergence of security, trust, and privacy for the IoT. In IoT access information, required to grant/reject access requests, is not only complex but also composite in nature. This is a direct result of the high level of interconnectedness between things, services, and people. It is clear that the type and structure of information required to grant/reject such an access request is complex and should address the following

IoT issues: security (authorization), trust (reputation), and privacy (respondent). The incremental deployment of the technologies that will make up the IoT must therefore provide adequate security and privacy mechanisms from the start. We must be sure that adequate security and privacy is available before the technology gets deployed and becomes part of our daily live.

1.6 Novelty and Contributions

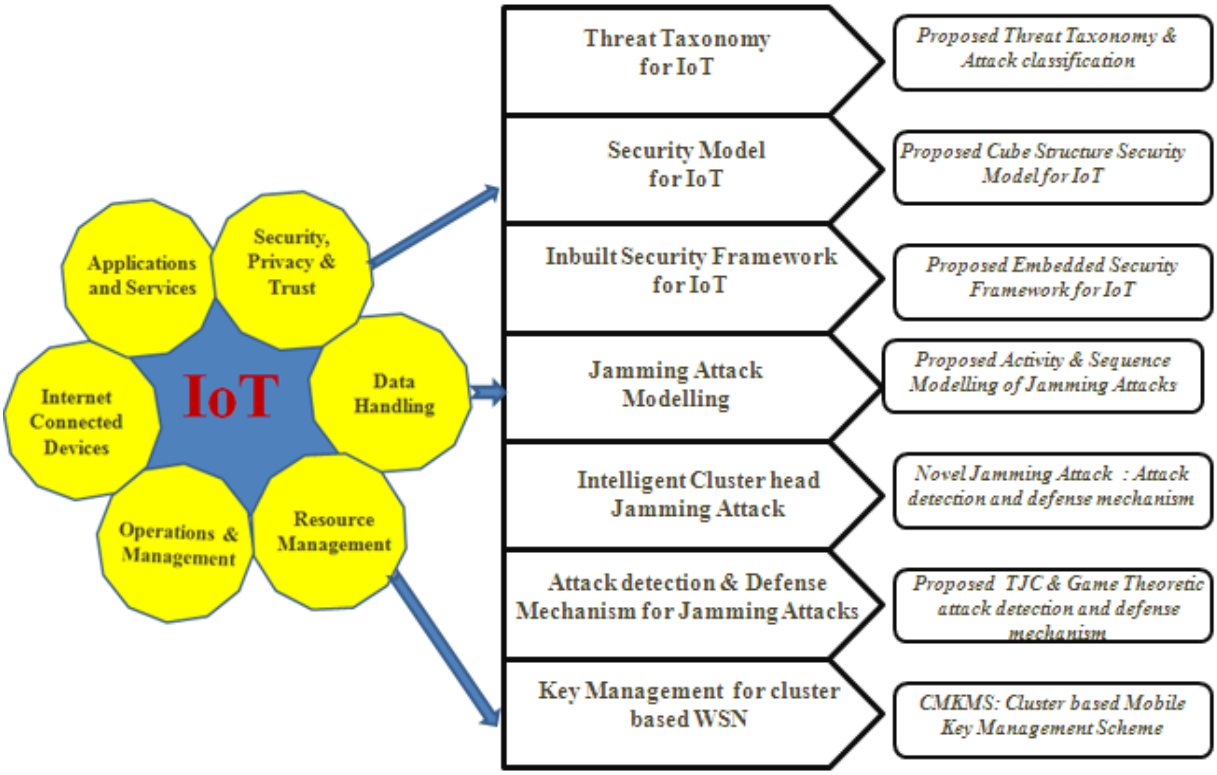

Figure 1.8: Problem Evolution and Thesis Contribution

The goal of this thesis is to design the security framework for IoT and design the security solutions to save from different jamming attacks and perform efficient key management in cluster based WSN. Major factors of influence are the energy consumption, delay, throughput and computational overheads for resource constrained devices in IoT. This study contributes to find out efficient attack detection and defense mechanism for jamming attack, which is the biggest threat in IoT. The thesis compares the performance evaluation of the proposed techniques with the existing state of art solution. The thesis also provides a novel key management scheme for cluster based mobile WSNs. Figure 1.8 provides an overview of the contributions presented in this thesis. The major contributions of thesis are as follows,

- Threat taxonomy for IoT
- Security model for IoT
- Security framework for IoT
- Jamming attack modelling
- Intelligent cluster head jamming attack
- Attack detection and defence mechanism against jamming attack
- Key management for cluster-based WSN

1. Threat Taxonomy for IoT

In this contribution of thesis the attack classification and threat taxonomy for IoT is proposed which will help to build the security framework for IoT. Security assessment for IoT is a tough problem, and attack classification and threat taxonomies will definitely aid in this process. The attack classification and threat taxonomy for IoT will be used as a framework for systematically examining new systems for similar but as yet unknown vulnerabilities. This Taxonomy relates to the needs of security model developers and will provide a more useful information tool for security analysts[28].

2. Security model for IoT

In this contribution, a cube structure security model for IoT is proposed which is derived from the key properties, challenges, attack classification and threat taxonomy for IoT. The IoT is an entirely new way of building out networks and services, so there is a need for a new security model for IoT which will take into considerations the security requirements and resource constraints of IoT. The security model organizes the security requirements and will help to propose an efficient protocol suitable for future internet[28-29].

3. Security Framework for IoT

In this contribution of thesis the design issues and the need of a different security framework for IoT which will take into consideration the limitations of resource constrained devices is analysed and structured. The challenges for embedded security framework are addressed and an enhanced security framework is proposed. The thesis also proposes the AES-GCM based embedded security protocol. Authenticated encryption is best suited concept for IoT that will provide both message encryption and authentication. Unique part of AES-GCM work is a novel approach of extending authentication and encryption with cryptographic capabilities[29,30].

4. Jamming Attack Modelling and Evaluation

In this contribution of thesis behaviour of different types of jamming attack are modelled using sequential and activity modelling approaches under unified modelling language (UML). The different types of jamming attacks consider here is constant-, deceptive-, random- and reactive- jamming. The behavioural modelling of these jamming attacks gives the clear understanding of jamming attack execution in the network, and it is useful tool to develop the defensive mechanism on it. The contribution also analysed the performance of the different jamming attack under varying network situations such as packet interval and number of malicious nodes in the network. The performance criteria measured here are delay, throughput and energy consumption[31,32].

5. Intelligent Cluster Head Jamming Attack

The thesis proposed the new kind of attack on cluster-based IoT network i.e. intelligent cluster head (CH) jamming attack. It is kind of reactive jamming attack, which mainly targets the CH. The attack is destructive because it attacks on CH, which is aggregating information

coming from other nodes in cluster and forwarding it to next CH or BS on path. The research also measured the performance of intelligent CH jamming attack with normal reactive attack. The result shows that intelligent CH jamming attack is more destructive as compared with normal reactive jamming attack[33].

6. Attack Detection and Defence Mechanism for Jamming Attack

The research developed the three different contributions under this title. The first contribution is development of the new countermeasure for reactive jamming attack. It suggests the Threshold based Jamming Countermeasure (TJC), which detects the jamming in network based on specific threshold value stored in the network. The simulation of TJC under different realistic situations shows that, the TJC helps to countermeasure the reactive jamming attack. The performance of TJC is measured by varying traffic interval, number of malicious nodes under static and mobile scenarios.

The second contribution is development of game theoretic modelling of jamming attack and detection mechanism. The jamming attack is modelled using game theory to understand different strategies of jammer in better manner. The new detection mechanism is developed to save from different jamming attack using cross-layer features. The proposed detection mechanism shows good energy consumption, throughput, and delay in different realistic situations of network. Its performance is compared with state-of-art optimal solutions on jamming.

The last contribution is to develop the countermeasure against intelligent CH jamming attack. The countermeasure is developed by modifying the TJC countermeasure for intelligent CH jamming attack. The proposed countermeasure is successful to detect and cure the attack during both inter- and intra- cluster communication [33,34].

7. Key Management for Cluster-based WSN

In this contribution of thesis new key management technique is developed for cluster-based WSN by considering mobility in the network. The scheme considers the two phases for managing the keys i.e. key establishment and key maintenance. The proposed algorithm improves the efficiency of key management algorithm in terms of security, mobility, energy efficiency and scalability of network. The simulation of scheme in different realistic situation shows that proposed solution shows less computational overheads, energy consumption and delay as compared with state-of-art solution [35].

1.7 Publications

The contributions have been, or are in the process of being, validated through peer-review and publication in journal and conference proceedings. The relevant publications are listed below:

A. Journal Publications

1. **Sachin D. Babar**, Neeli R. Prasad, Ramjee Prasad, "**Activity Modelling and Countermeasures on Jamming Attack**", Journal of Cyber Security and Mobility, Vol. 2, Issue no. 2, pp. 1-27, April 2013**.**
2. **Sachin D. Babar**, Neeli R. Prasad, Ramjee Prasad, "**CMKMS: Cluster based Mobile Key Management Scheme for Wireless Sensor Network**", International Journal of Pervasive computing and Communications (IJPCC) : Special Issue on Adaptive Security for IoT, Vol. 10, Issue 2, pp-196-211, April 2014.
3. **Sachin Babar**, Parikshit N Mahalle, Neeli R. Prasad and Ramjee Prasad, "**A Hash Key-based Key Management Mechanism for Cluster-based Wireless Sensor Network**", Journal of Information Security and Applications, Elsevier editorial system. (Submitted)

B. Conference Publications

1. **Sachin Babar**, Parikshit N. Mahalle, Antonietta Stango, Neeli R Prasad and Ramjee Prasad, **"Proposed Security Model and Threat Taxonomy for the Internet of Things (IoT),"** In proceedings of 3rd International Conference CNSA 2010, Book titled: Recent Trends in Network Security and Applications - Communications in Computer and Information Science, Springer Berlin Heidelberg, pp. 420 - 429 Volume: 89. Chennai – India, July 23-25, 2010.
2. **Sachin Babar**, Antonietta Stango, Neeli Prasad, Jaydip Sen and Ramjee Prasad, "**Proposed Embedded Security Framework for Internet of Things (IoT)**" , In proceedings of 2nd International Conference on Wireless Communications, Vehicular Technology, Information Theory and Aerospace & Electronic Systems, Wireless VITAE 2011, vol., no., pp.1-5, Feb. 28, 2011 - March 3, 2011.
3. **Sachin Babar**, Parikshit N Mahalle, Neeli R. Prasad and Ramjee Prasad, **"Proposed on Device Capability based Authentication using AES-GCM for Internet of Things (IoT),"** In proceedings of 3rd International ICST Conference on Security and Privacy in Mobile Information and Communication Systems (Mobisec 2011), Aalborg – Denmark, May 17-19, 2011.
4. **Sachin D. Babar**, Neeli R. Prasad, Ramjee Prasad, "**Jamming Attack: Behavioral Modelling and Analysis**", In proceedings of the 3rd International Conference on Wireless Communications, Vehicular Technology, Information Theory and Aerospace & Electronic Systems, Wireless VITAE 2013, Princeton, New Jersey, USA, June 24-26, 2013.

5. **Sachin D. Babar**, Neeli R. Prasad, Ramjee Prasad, "**Proposed Game Theoretic Modelling of Jamming Attack and Attack Detection Mechanism**" , In proceedings of the 16th International Symposium on Wireless Personal Multimedia Communications, WPMC 2013, Atlantic City, New Jersey, USA, June 24 - 27, 2013.
6. **Sachin D. Babar**, Neeli R. Prasad, Ramjee Prasad, "**Countermeasure for Intelligent Cluster-head Jamming Attack in Wireless Sensor Network**" , In the proceedings of the International Conference on Privacy and Security in Mobile Systems, PRISMS 2013, Atlantic City, New Jersey, USA, June 24 - 27, 2013.

C. Other Publications

1. Parikshit N. Mahalle, **Sachin Babar**, Neeli R Prasad and Ramjee Prasad, "**Identity Management Framework towards Internet of Things (IoT): Roadmap and Key Challenges**" In proceedings of 3rd International Conference CNSA 2010, Book titled: Recent Trends in Network Security and Applications - Communications in Computer and Information Science, Springer Berlin Heidelberg, pp. 430 - 439 Volume: 89. Chennai – India, July 23-25, 2010.

1.8 Thesis Outline

The following provides an outline of the thesis with a brief description of the individual chapters.

Chapter 2: Security Framework for IoT Security

This chapter introduces the concept of embedded security, its requirement, embedded security issues in IoT, challenges of embedded security framework design and presents enhanced embedded security framework. The chapter discusses the environment factor and security objectives for enhanced embedded security framework. The chapter also proposed the AES-GCM-based embedded security protocol. Authenticated encryption is best suited concept for IoT that will provide both message encryption and authentication. Unique part of AES-GCM work is a novel approach of extending authentication and encryption with cryptographic capabilities. The chapter also evaluates the proposed AES-GCM in terms of its mutual authentication process, resistance to attack and efficiency.

Chapter 3: Jamming Attack: Modelling and Evaluation

This chapter introduces the jamming attack that take place at physical layer, and its classification in detail. The chapter provides modelling of jamming attack using activity- and sequential- modeling approach. The chapter also evaluates the different jamming attack in variety of network situations. The chapter proposed new possibility of the jamming attack; intelligent cluster-based jamming attack and evaluated the performance impact of cluster-based jamming attack. The last part of chapter discussed the requirements to design efficient defense mechanism against jamming attack.

Chapter 4: Defense Mechanism against Jamming Attack

This chapter discusses the classification of jamming countermeasures, comparison of available countermeasures and derives the open issues to develop efficient countermeasures. Threshold-based Jamming Countermeasure (TJC) is discussed in the chapter with assumption made, working mechanism of algorithm and comparative simulation with result discussion. The chapter overviews the game theory for WSN security and proposed the game formulation for jamming attack. It also proposed the jamming detection mechanism based on game theory concept. The proposed game theory-based simulation is compared with state of art solutions. The counter measure is developed for the proposed intelligent cluster-based jamming attack and it is compared with existing solutions.

Chapter 5: Secure Key Management

This chapter illustrates the classification, comparison of secure key management techniques, and gives the requirements to develop the optimized secure key management technique. The chapter proposes cluster-based key management algorithm and discusses the system model, proposed key management scheme, and its performance evaluation.

Chapter 6: Conclusions and Future Work

This chapter provides the summary of the thesis, and discusses future research work.

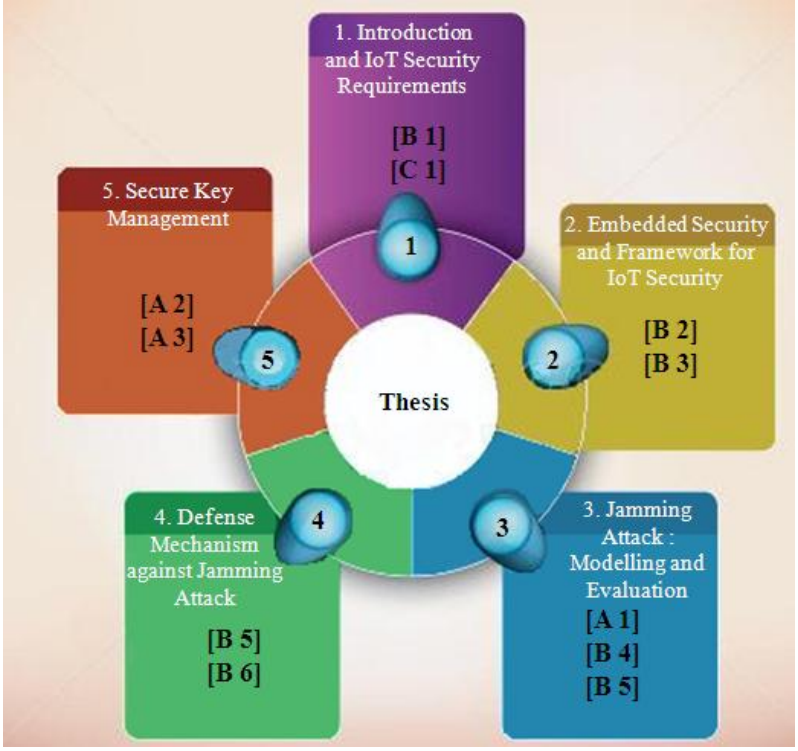

Figure 1.9: Thesis Organization

Following the research contributions agenda, the rest of this dissertation is divided into five self-contained parts as shown in Figure 1.9. An overview of the thesis and the chapter wise publications can also be seen from Figure 1.9, which shows the connection between individual chapters. [A], [B] , [C] shown in the Figure 1.9 refers to the list of publications mentioned in Section 1.6.

1.9 References

[1] Agrawal S. and Das M.L., "Internet of Things — A paradigm shift of future Internet applications" , International Conference on Engineering (NUiCONE), 2011 Nirma University , IEEE, vol., no., pp.1,7, 8-10 Dec. 2011.

[2] Silverajan, B. and Harju, J., "Developing network software and communications protocols towards the internet of things", In Proceedings of the Fourth international ICST Conference on Communication System Software and middleware (Dublin, Ireland, June 16 - 19, 2009). COMSWARE '09. ACM, New York, NY, 2009, 1-8.

[3] Wang C., Daneshmand M., Dohler M., Mao X., Hu R. Q., Wang H., "Guest Editorial - Special Issue on Internet of Things (IoT): Architecture, Protocols and Services", Sensors Journal, IEEE , vol.13, no.10, pp.3505,3510, Oct. 2013.

[4] Ivan Seskar, Kiran Nagaraja, Sam Nelson and Dipankar Raychaudhuri, "MobilityFirst Future Internet Architecture Project", Proceedings of the 7th Asian Internet Engineering Conference AINTEC 2011, ACM, New York, NY, USA.

[5] Adjie-Winoto W., Schwartz E., Balakrishnan H., and Lilley J., "The design and implementation of an intentional naming system", In Proceedings of the Seventeenth ACM Symposium on Operating Systems Principle, SOSP '99. ACM, New York, NY, 1999, 186-201.

[6] Beerliova Z., Eberhard F., Erlebach T., Hall A., Hoffmann M., Mihal'ak M., Ram L.S., "Network Discovery and Verification," IEEE Journal on Selected Areas in Communications, Vol.24, No.12, 2006, 2168-2181.

[7] Antonio J. Jara, Pablo Lopez, David Fernandez, Jose F. Castillo, Miguel A. Zamora, and Antonio F. Skarmeta,"Mobile digcovery: discovering and interacting with the world through the Internet of things", ACM, Personal Ubiquitous Computing. 18, 2 (February 2014), 323-338.

[8] Qiang Wei and Zhi Jin, " Service discovery for internet of things: A context-awareness perspective", Proceedings of the Fourth Asia-Pacific Symposium on Internetware (Internetware 2012), ACM, New York, NY, USA, , Article 25 , 6 pages.

[9] Y.-C. Hu and H. J. Wang, "Location Privacy in Wireless Networks",In Proceedings of the ACM SIGCOMM Asia Workshop, 2005, 1-5.

[10] Huston G., Michaelson G., Kent S., "Resource Certification - A Public Key Infrastructure for IP Addresses and AS's," GLOBECOM Workshops, 2009 IEEE , vol., no., pp.1,6, Nov. 30 2009-Dec. 4 2009.

[11] Loren M Kohnfelder, "Towards a Practical Public Key System" Thesis, 1978, http://dspace.mit.edu/bitstream/handle/1721.1/15993/07113748.pdf

[12] Neuman B.C, Tsaposo, "Kerberos: an authentication service for computer networks." IEEE Communications Magazine.Vol 32, Issue 9, Pages 33–38. Sep 1994.

[13] Ahmad Z., Manan J.A. , Sulaiman S., "Trusted Computing based open environment user authentication model," Advanced Computer Theory and Engineering (ICACTE), 2010 3rd International Conference on , vol.6, no., pp.V6-487,V6-491, 20-22 Aug. 2010.

[14] Huping Wang, Chunxiao Fan, Shuai Yang, JunweiZou, Xiaoying Zhang, "A New Secure OpenID Authentication Mechanism Using One-Time Password (OTP)," Wireless Communications, Networking and Mobile Computing (WiCOM), 2011 7th International Conference on , vol., no., pp.1,4, 23-25 Sept. 2011

[15] Chehab M.I., Abdallah A.E., "Architectures for identity management," International Conference for Internet Technology and Secured Transactions,ICITST 2009., vol., no., pp.1,8, 9-12 Nov. 2009.

[16] Ates M., Gravier C., Lardon J., Fayolle J., Sauviac B., "Interoperability between Heterogeneous Federation Architectures: Illustration with SAML and WS-Federation," Signal-Image Technologies and Internet-Based System, 2007. SITIS '07. Third International IEEE Conference on , vol., no., pp.1063,1070, 16-18 Dec. 2007.

[17] Schmitz J., Loew J., Elwell J., Ponomarev D., Abu-Ghazaleh N., "TPM-SIM: A framework for performance evaluation of Trusted Platform Modules", Design Automation Conference (DAC), 2011 48th ACM/EDAC/IEEE , vol., no., pp.236,241, 5-9 June 2011.

[18] Winter J., "Experimenting with ARM TrustZone -- Or: How I Met Friendly Piece of Trusted Hardware," IEEE 11th International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom) 2012 , vol., no., pp.1161-1166, 25-27 June 2012.

[19] Yuxi Liu, Guohui Zhou, "Key Technologies and Applications of Internet of Things" ,Fifth International Conference on Intelligent Computation Technology and Automation (ICICTA), 2012, vol., no., pp.197,200, 12-14 Jan. 2012.

[20] Kocher P., Lee R., McGraw G., and Raghunathan A., "Security as a new dimension in embedded system design", In proceedings of the 41st Annual Design Automation Conference, DAC '04. ACM, New York, NY, 2004, 753-760.

[21] Welch, D.; Lathrop, S., "Wireless security threat taxonomy," Information Assurance Workshop, IEEE Systems, Man and Cybernetics Society, 2003, 76-83

[22] Srivaths Ravi, AnandRaghunathan, Paul Kocher, Sunil Hattangady , "Security in embedded systems: Design challenges " , ACM Transactions on Embedded Computing Systems (TECS) , Volume 3, Issue 3, 2003, 1-30.

[23] Covington M.J., Carskadden R., "Threat implications of the Internet of Things", In proceedings of 5th International conference on Cyber Conflict (CyCon-2013), vol., no., pp.1,12, 4-7 June 2013.

[24] Zeng, Ling-yuan, "A security framework for internet of things based on 4G communication", 2nd International Conference on Computer Science and Network Technology (ICCSNT-2012) , vol., no., pp.1715,1718, 29-31 Dec. 2012.

[25] Hong Ning, Xuefeng Zheng, "A Security Framework for Internet of Things Based on SM2 Cipher Algorithm", Fifth International Conference on Computational and Information Sciences (ICCIS-2013) , vol., no., pp.13,16, 21-23 June 2013.

[26] Yong Wang, RamamurthyB., Yuyan Xue, Xukai Zou, "A security framework for wireless sensor networks utilizing a unique session key," 5th International Conference on Broadband Communications, Networks and Systems (BROADNETS 2008) , vol., no., pp.487,494, 8-11 Sept. 2008

[27] Jokhio S.H., Jokhio I.A., Kemp A.H., "Light-weight framework for security-sensitive wireless sensor networks applications", Wireless Sensor Systems, IET , vol.3, no.4, pp.298,306, December 2013.

[28] Sachin Babar, Parikshit N. Mahalle, Antonietta Stango, Neeli R Prasad and Ramjee Prasad, "Proposed Security Model and Threat Taxonomy for the Internet of Things (IoT)," In proceedings of 3rd International Conference CNSA 2010, Book titled: Recent Trends in Network Security and Applications - Communications in Computer

and Information Science 2010 Springer Berlin Heidelberg, pp. 420 - 429 Volume: 89. Chennai – India, July 23-25, 2010

[29] Sachin Babar, Antonietta Stango, Neeli Prasad, Jaydip Sen and Ramjee Prasad, "Proposed Embedded Security Framework for Internet of Things (IoT)" , In Proceedings of 2nd International Conference on Wireless Communications, Vehicular Technology, Information Theory and Aerospace &Electronic Systems, Wireless VITAE 2011, vol., no., pp.1-5, Feb. 28, 2011 - March 3, 2011

[30] Sachin Babar, Parikshit N Mahalle, Neeli R. Prasad and Ramjee Prasad, "Proposed on Device Capability based Authentication using AES-GCM for Internet of Things (IoT)," In proceedings of 3rd International ICST Conference on Security and Privacy in Mobile Information and Communication Systems (Mobisec 2011), Aalborg – Denmark, May 17-19, 2011.

[31] Sachin D. Babar, Neeli R. Prasad, Ramjee Prasad, "Activity Modelling and Countermeasures on Jamming Attack", Journal of Cyber Security and Mobility, Vol. 2, Issue no. 2, pp. 1-27, April 2013

[32] Sachin D. Babar, Neeli R. Prasad, Ramjee Prasad, "Jamming Attack: Behavioral Modelling and Analysis", 3rd International Conference on Wireless Communications, Vehicular Technology, Information Theory and Aerospace & Electronic Systems, Wireless VITAE 2013, IEEE, vol., no., pp.1,5, 24-27 June 2013.

[33] Sachin D. Babar, Neeli R. Prasad, Ramjee Prasad, "Countermeasure for Intelligent Cluster-head Jamming Attack in Wireless Sensor Network" , In the proceedings of the International Conference on Privacy and Security in Mobile Systems, PRISMS 2013, Atlantic City, New Jersey, USA, June 24 - 27, 2013

[34] Sachin D. Babar, Neeli R. Prasad, Ramjee Prasad, "Proposed Game Theoretic Modelling of Jamming Attack and Attack Detection Mechanism" , In proceedings of the 16th International Symposium on Wireless Personal Multimedia Communications, WPMC 2013, Atlantic City, New Jersey, USA, June 24 - 27, 2013

[35] Sachin D. Babar, Neeli R. Prasad, Ramjee Prasad, "CMKMS: Cluster based Mobile Key Management Scheme for Wireless Sensor Network", International Journal of Pervasive computing and Communications (IJPCC) : Special Issue on Adaptive Security for IoT, Vol. 10, Issue 2, pp-196-211, April 2014.

2

Security Framework for IoT

This chapter introduces the concept of embedded security for IoT. In this chapter, the embedded security issues and challenges for IoT is discussed. This chapter proposes the embedded security framework as a feature of software/hardware co-design methodology. The chapter also proposes the AES-GCM based embedded security protocol based on capability and authenticated encryption process. The chapter also evaluates the proposed AES-GCM in terms of its mutual authentication process, resistance to attack and efficiency.

2.1 Introduction

The IoT will consist of billions of digital devices, people, services and other physical objects having the potential to seamlessly connect, interact and exchange information about themselves and their environment. This will make our lives simpler through a digital environment that will be sensitive, adaptive, and responsive to human needs. It will combine the power of universal network connectivity with embedded systems, sensors, and actuators in the physical world. This new concept involves objects of our daily life, like clothes, cars, shopping carts, which will be able to reveal information about them, interact with each other and with the environment. IoT will therefore add an enormous range of new industrial opportunities to the software and hardware markets. Due to manifold aspects that involves, security for IoT will be a critical concern that must be addressed in order to enable several current and future applications [1].

Existing solutions cannot result in a complete solution and are often not integrated into the entire system. Sometimes these solutions violate the criteria that designers have taken into consideration from the beginning. These are subtle points that are not addressed by designers who tend to focus mainly on functionality and by companies that tend to focus on short term profits. All these reveal the importance of fundamental security solutions and the need for applied security. The main technical challenges for IoT will therefore include the design and integration of different technologies, as well as providing the necessary degree of security.

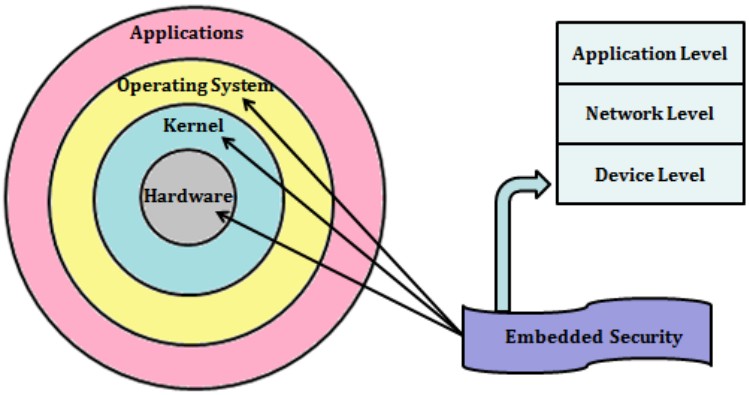

Figure 2.1: Structure of Embedded security

Embedded security means building security in from the start i.e. security features built into a device[2]. Embedded security is nowadays growing as new dimension which designers should consider throughout the design process, along with other metrics such as cost, performance, and power. Security features like physical tamper resistance, cryptography (keys, algorithms), platform (operating system elements) will be the concern for IoT systems. Embedded security can also be defined in another form as a new dimension which refers to what a device needs, to be part of a distributed computing system. Embedded security equips the device with a robust trusted element that serves numerous applications and services such as DRM (Digital Rights Management), commerce and device management. It makes security transparent to the end-user, improves the performance of the security solution and reduces its power consumption. Figure 2.1 shows the structure of embedded security. Embedded security will play its role at hardware level, kernel level and operating system level. A secure

kernel will provide secure interface between the operating system and the applications. The system architecture will provide a secure runtime execution environment.
This chapter gives a detailed survey and analysis of embedded security especially in the area of IoT. Together with the conventional security solutions, the chapter highlights the need to provide in-built security in the device itself to provide a flexible infrastructure for dynamic prevention, detection, diagnosis, isolation, and countermeasures against successful breaches. Based on this survey and analysis, the chapter defines the security needs taking into account computational time, energy consumption, and memory requirements of the devices. This chapter proposes the embedded security framework as a feature of software/hardware co-design methodology.
The chapter also introduces an authentication and encryption protocol which serves as a proof of concept for authenticating device using the Advanced Encryption Standard (AES) – Galois/ Counter Mode (GCM) as cryptographic primitive. Authenticated encryption is best suited concept for IoT that will provide both message encryption and authentication. Unique part of this work is a novel approach of extending authentication and encryption with cryptographic capabilities.

2.2 Related Works

The security for resource constrained devices always faces contradictories: it should provide a high level of security, manage several types of protocols and be flexible enough to support rapid evolution of security mechanisms and standards with limited silicon area and less energy consumption. The solution selected for security in these devices is always a question of trade-off between security, flexibility, performance, power consumption, and cost. Existing solutions to these problems are divided into three approaches as shown in figure 2.2 :

i) Software only Approach

This approach makes use of programmability of embedded general purpose processors for performing security operations. This approach reaches the demand in cost and flexibility but not in the power consumption and silicon area points of view. This approach sometimes leads to overwhelm the processing capacity of the embedded GPP (General purpose processors). However, one option to solve this problem is to use an optimized GPP. This means that the instruction set architecture of the processor contains some extra specific instructions which speed up some algorithm-operations or reduce the memory necessary to these operations. But this latest solution is not a global answer: it is efficient only for a limited number of algorithms (i.e. these which use the extra instructions). Concerning the parameter computation capacity, it can be evaluate more easily but it is generally not optimum. In the point of view of countermeasures against security attack, this approach can provide several solutions. In [3], a countermeasure against side-channel attack at software level is described.

ii) Hardware only Approach

This approach makes use of ASICs (Application Specific Integrated Circuits) to implement a given cryptography algorithm in hardware. This policy allow controlling precisely the

parameters energy, computation capacity and time constraints but it is generally not optimum for the flexibility and cost parameters. The FPGA solution allows to reach the flexibility demand but to the detriment of cost and sometimes energy. Research in the ASIC or SoC approaches is generally focused on the optimization of the basic security functions. In [4] a crypto-coprocessor, dedicated to the IPSec applications, is presented which speed up the basic security functions (authentication, confidentiality, integrity) with low power consumption. However, the countermeasures against security attacks are also explored. In [5] a new logic style for secure IC against differential power analysis is presented.

iii) Hybrid Approach

This approach is a combination of the two previous approaches. It optimizes the overall partitioning of functionality between HW and SW, as well as between the system host processor and security processor, to maximize overall processing efficiency while satisfying other design constraints. It is the best trade-off between efficiency and flexibility but it requires a clear vision of the complete system and a good communication between the hardware designers, the software designers and the security experts [6,7].

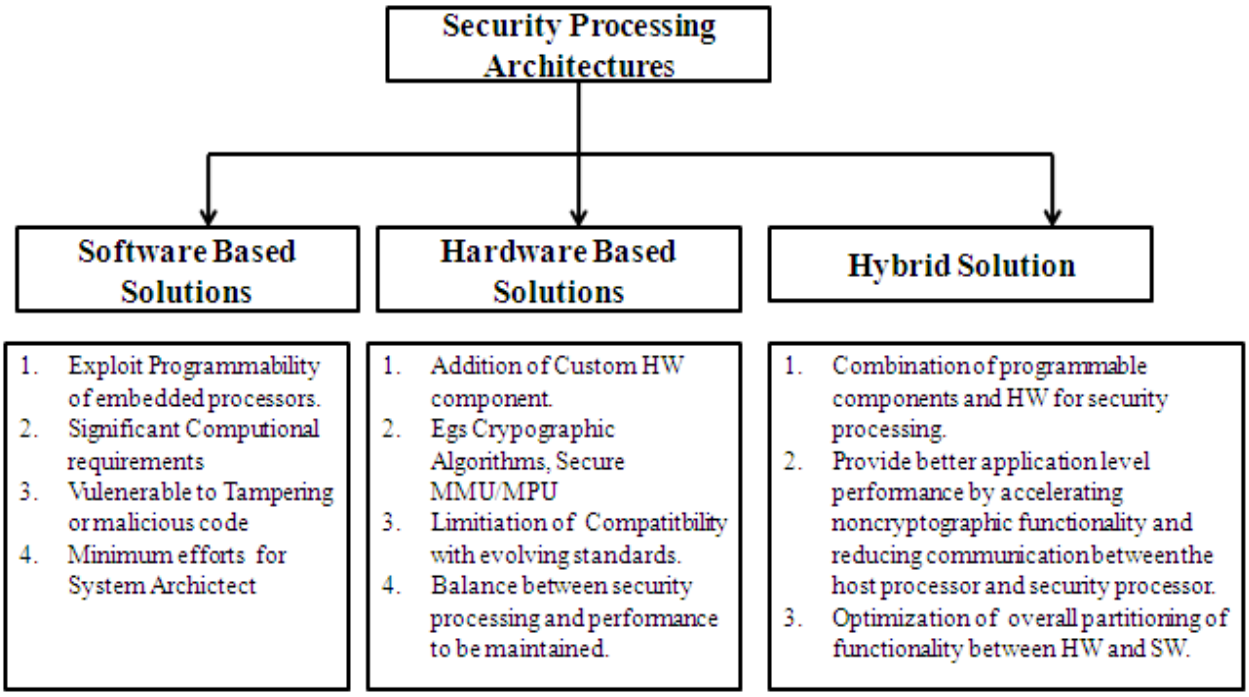

Figure 2.2: Classification of Security processing Architectures

The research on existing solutions is divided into two main topics: optimization of the basic security functions and countermeasures against security attacks. Table 2.1 presents the functionality comparison for existing solutions between these two topics for the publications used as references for this state of art evaluation. Optimization parameters like energy, computational time, memory requirement, flexibility, cost, reliability, etc are concerns for IoT. The major parameters of concern for resource constraint systems are energy, computational time and cost.

Table 2.1: Functionality comparison for existing solutions

Existing solutions[6-12]/ Comparison Parameters	Counter measures against attack			Optimization of the basic security functions			
	Side-channel	HW-attack	SW-attack	Energy Efficiency	Flexible	Computational time	cost
An FPGA Implementation of a Flexible Secure ECC Processor			√		√	√	
HW-SW Implementation of Public-Key Cryptography for Wireless Sensor Networks			√		√		√
Implementing Embedded Security on Dual-Virtual-CPU Systems			√	√			√
A security approach for off-chip memory in embedded microprocessor systems			√				√
A compiler-hardware approach to software protection for embedded systems		√	√		√		
Embedded security: New trends in personal recognition systems		√	√				√
A data-driven approach for embedded security			√				

All solutions discussed basically focus on to speed up the basic security functions and it does not provide solutions against the majority of the security attacks. So, there is a need for an embedded security framework and architecture which will move security considerations from a function-centric perspective to system architecture (HW-SW) design issue.

2.3 Embedded Security Issues in IoT

This section discusses the building blocks for Embedded Security and presents the issues and challenges in IoT.

2.3.1 Building Blocks for Embedded Security

Embedded security means building security in from the start i.e. security features built into a device. Some of the major building blocks for embedded security for IoT is listed below [13, 14]:

1. Cryptographic Algorithms: These are basically the essential building block of a robust security solution. Two types of algorithms generally used are symmetric cryptography and public key cryptography. The unusual design constraints placed on embedded devices require a new lightweight, highly efficient, easy to deploy cryptography scheme that provides high levels of security while minimizing memory, execution speed requirements, and power requirements. ECC is an essential methodology for meeting these requirements of embedded designs and can serve as a good alternative for embedded security.
2. Secure Storage: Cryptographic algorithms require keys as their basis for operation. Since the algorithms are published and known to all, including to potential attackers, protecting the secrecy of the key is an important issue for security. Secure Storage essentially deals with protecting access to keys and other pieces of data. Secure storage also needs to be persistent, such that items are not lost during power cycles. Examples of persistent storage are on-chip ROM memory, on-chip One-Time-Programmable (OTP) technology, as well as off-chip flash memory.
3. Secure Boot: The purpose of secure boot is to bring the system to a known and trusted state. The secure boot routine is a ROM-based routine, so that an attacker cannot intercept the procedure. Additional features are required in order to provide a complete secure boot solution. These include the ability for software update at any point in time i.e a software version revocation mechanism for system advancement to a new version of the software image with prevention of roll-back to an older version is a must.
4. Secure JTAG: The JTAG interface is a debugging interface for chips. It is used primarily during development and manufacturing, but also used to help debug errors that are found in the course of the lifetime of the system. The JTAG interface is potentially exploitable by attackers, who can try to read internal registers or memories.
5. Secure Execution Environment (SEE): It refers to a processing unit which is capable of executing applications in a protected manner. The building blocks of an SEE are : a secure processor (either a dedicated processor or one capable of supporting a secure mode) which is hardware compartmentalized from the non-secure mode, secure code and data memory (most likely dedicated on-chip RAMs) and a secure kernel for providing the interface between hardware and software.

2.3.2 Issues and challenges

Making embedded devices secure is not only protecting resources and assets but also providing opportunities for new services and new businesses in the optic of IoT. The new applications of IoT bring also new security issues:

- With the advent of IoT, cost of the embedded device will be cheap with higher degree of networking. Networking, sharing resources and holding sensitive assets exposes these devices to a growing potential risks.
- Traditional or conventional security solutions that exist are not feasible for majority of the devices involved in IoT because of the power, computation speed and memory limitations. The common characteristics of embedded devices - mobile and resource constrained systems─ enforce researchers to take a new look at current solutions.
- Application of IoT in areas such as health care, avionics, or car industry where humans are involved raises the issue of safety. For example, the violation of integrity and availability of an artificial hearth, brake of a car and navigation system of an airplane may have disastrous consequences. Attacks are turning from digital-data attacks to human attacks.
- Legal usage of various applications / devices for financial gain will require security for revenue protection.
- There will be many new applications or business models that strongly depend on the security requirements. e.g., pay-TV, video on demand or time-limited services.
- IoT systems are able to track, sense and capture a huge amount of data, such as location, status of a user, and personal data. There will be a huge amount of data to manage and protect.
- The secure identification of device is a major concern for a large number of applications, considering the software download or digital rights, for example.

2.4 Enhanced embedded security framework

The basic embedded security framework should consider the following things:

1. Environment factor: With respect to the environment in which the devices operate determine the assumptions, threats, vulnerabilities, attacks, and required policies for secure functioning.
2. Security Objectives: Determine your device's security objectives. Consider the data (assets) or operation it will protect and which threats from step 1 require countermeasures.
3. Requirements: Determine your functional security requirements.

The basic idea for framing the security architecture for IoT is to utilize security mechanisms and protocols effectively and to start off with a design that takes security into consideration from the start of requirements gathering to maintenance as seen in figure 2.3 following the software development life cycle.

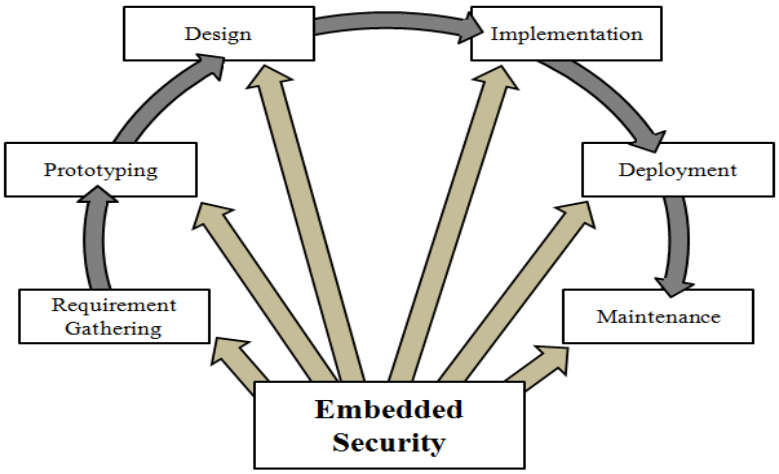

Figure 2.3: Embedded security design steps

For building the embedded security Framework for IoT, we also need to look at all of the tradeoffs between performance, cost, and security. Unfortunately, these three concepts are almost always directly at odds with one another. More performance means the cost goes up, lowering the cost means lowering security and performance, and implementing higher security means performance will decrease. Hardware software based security architecture for IoT is proposed which should be the best trade off for cost/efficiency or security/performance as shown in figure 2.4.

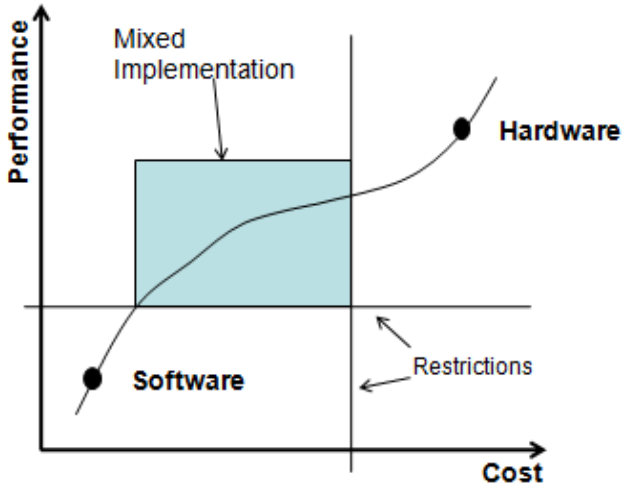

Figure 2.4: Hardware Software Security implementation performances

Figure 2.5 illustrates the proposed embedded security framework and architecture for IoT. The architecture can be divided into hardware and software level with lightweight standardized protocols supporting at the physical and MAC layer. The level of security within the device will vary depending on the nature of the protected content and kind of application. The architecture should provide physical protection to secret keys by keeping the components like secure ROM, which is handling the secret keys, inside the secure SoC. The Secure Boot loader should ensure that the device boots up with the genuine OS or firmware with right process privileges. Secure ROM, secure runtime execution environment, secure memory management unit are the prime focus for inbuilt security. Also rich operating system with necessary security functionality, secure kernel interface and compatible standardized security

protocols for IoT system will contribute towards the secure security architecture and framework for IoT. Thus the secure architecture is based on three main components i.e. secure software management, secure hardware blocks and secure communications inside the processor. If any one of these is missing device security cannot be achieved.

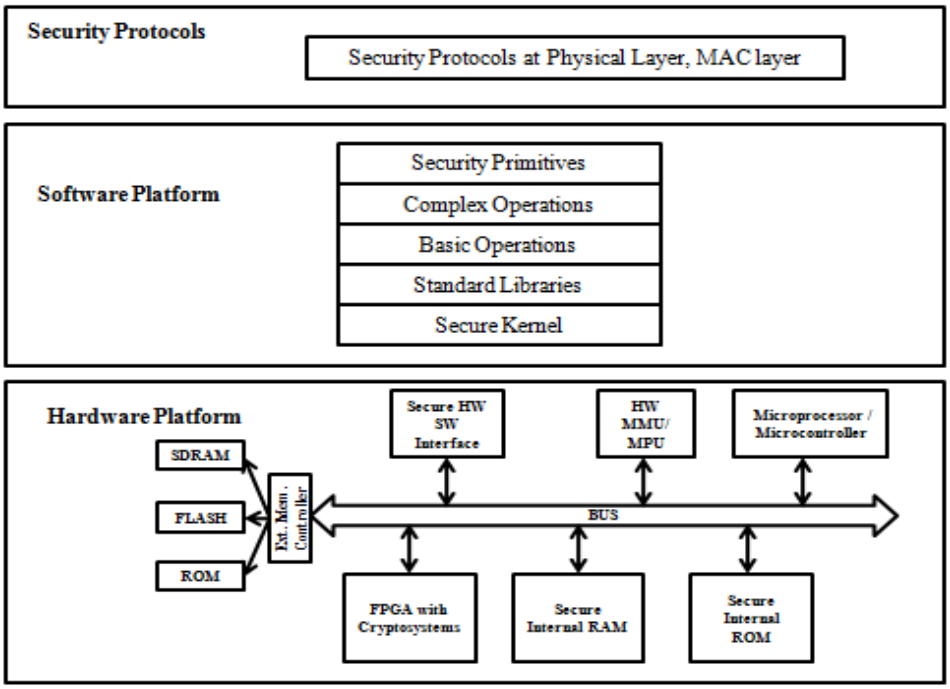

Figure 2.5: Embedded security framework and architecture

A cost effective designs use a mixture of hardware and software to accomplish overall security goals. This provides sufficient motivation for attempting a synthesis-oriented approach to achieve security system implementations having both hardware and software components. Such an approach would benefit from a systematic analysis of design trade-offs that is common in synthesis while also creating cost effective systems.

Following are the key features of the security framework :

1. Lightweight cryptography: Optimized Cryptographic algorithms and hardware architecture for extreme low power, memory and processing requirements.
2. Physical Security: Trusted Platform module which will take into account the vulnerabilities of the hardware device at physical level.
3. Standardized security Protocols: Development of standardized protocols which are both lightweight with respect to communication and cryptographic computations.
4. Secure operating systems: Rich operating systems with a secure kernel which will ensure a secure communication inside the processor by providing secure runtime execution environment, secure booting, secure content, etc.
5. Future application Areas: Understanding the technical, economic, social context of a given application area, in order to develop security solutions which are appropriate and acceptable.
6. Secure Storage: Protect the sensitive information stored in RAM / ROM and secondary storage and efficient key management.
7. Protection against different attacks.

2.5 Authentication Schemes for IoT

Devices like RFID or sensor node themselves have no access control function, so they can freely obtain information from each other. As a result, an authentication as well as authorization scheme must be established between devices so as to achieve the security goals for IoT. In RFID, tag security issue related to the scenario, like the communication between a tag and a reader which is by radio, anyone can access the tag and obtains its output, i.e. attackers can eavesdrop on the communication channel between tags and readers, which is a cause of consumer's apprehension. So the authentication scheme employed in RFID must be able to protect the data passing between the tag and the reader, i.e. the security solution itself should have some kind of encryption capability.

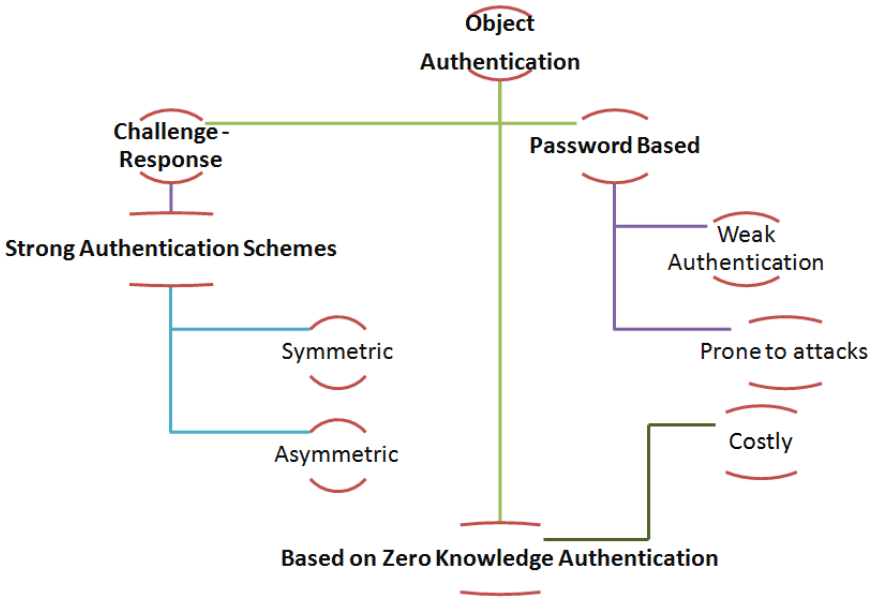

Figure 2.6: Authentication Scheme

Authentication is related to secure identification of devices in which there is need for verification of identity possession. Every act of an access control will enable authentication process. So, secure identity establishment is promising in nomadic IoT which is prone to many threats [15, 16]. Authentication with encryption can solve all of the former mentioned security threats in IoT scenario like RFID and sensor Networks applications.

Broadly there are three authentication schemes: password systems (weak authentication), challenge-response authentication (strong authentication), and customized and zero-knowledge authentication [17]. Password systems offers a weak level of security and zero-knowledge techniques are often related to "strong" mathematical problems which are very costly in calculation and implementation. So we aim for the second type, the challenge-response techniques, which are broadly used. There are asymmetric and symmetric challenge-response techniques. The disadvantage of asymmetric authentication methods is that they are very time consuming and costly to implement in hardware. So, they are not the first choice for resource constraints devices. This classification is shown in figure 2.6.

2.6 AES-GCM based embedded security protocol

This section presents an Authenticated encryption scheme that will best suited for IoT and also proposes on device capability based authentication and access control protocol for IoT. Further it also evaluates the proposed protocol in terms of its mutual authentication process, resistance to attack and efficiency.

2.6.1 Authentication and Encryption using AES-GCM

Authenticated encryption is evolving as a relatively new concept that will provide both message encryption and authentication which can be adapted for embedding security in device. AES-GCM is one of the latest authenticated encryption algorithms providing both confidentiality and authenticity suitable for hardware implementation. AES-GCM accepts four inputs namely symmetric key, Initialization vector (IV), Plaintext and an optional field for authenticating data. The output of AES-GSM is the cipher text and the message. The Initialization Vector (IV) is generated by the device performing the authenticated encryption operation. It can also be a nonce within the scope of any authenticated encryption key with uniqueness. Repeating nonce for two different messages encrypted with the same key destroys the security properties. The optional additional authenticated data can be used to authenticate plaintext packet headers. AES-GCM makes use of the AES block cipher in counter mode to provide encryption. When used properly, counter mode provides strong confidentiality [18]. GCM uses universal hashing in the finite field $GF(2^w)$ for generating a message authentication code (MAC). The additional merit of using $GF(2^w)$ is that the computation cost of multiplication under $GF(2^w)$ is less than integer multiplication. AES-GCM provides high security suitable for hardware implementation. Therefore, the use of AES-GCM is the best solution for resource constrained device to meet the security needs of IoT devices [19, 20]. Implementing AES-GCM on resource constrained devices with hardware software co-design approach will surely match the Security requirements for IoT enhancing the speed and storage area parameters. For prevention against replay attacks, use of different session key for encryption of plaintexts will help to guarantee confidentiality which can be done through GCM. Proposed protocol is using capability based addressing [21, 22] along with AES-GCM for access control of devices. Capability corresponds to row view of access control matrix [23].

2.6.2 Proposed Protocol

In this work, we propose on device capability based authentication and access control protocol. Novelty of this protocol is in its cryptographic capability which acts as a ticket to access other device. This capability is then encrypted using AES–GCM which strongly provides both encryption and authentication for resource constrained devices. This protocol is mutual authentication protocol and it also addresses capability based access control. Conceptually, a capability is a token, ticket, or key that gives permission to access an device. A capability is implemented as a data structure that contains items like a unique device identifier, access rights and a random number, as shown in figure 2.7. The identifier addresses or names are single to device in IoT. Any device, in this context, can be equipped

with RFID tags or sensor nodes. The access rights define the operations that can be performed on that device.

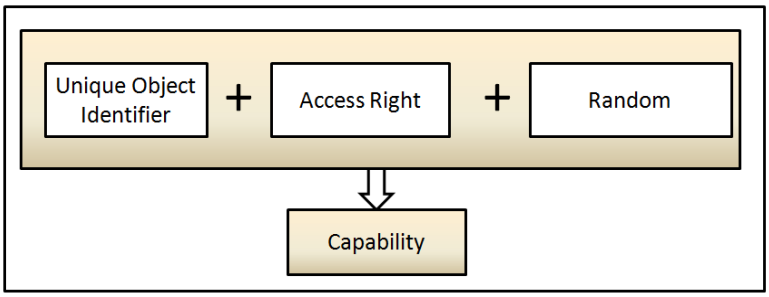

Figure 2.7: Capability structure

For simplicity, it is sufficient to examine the case where a capability describes a set of access rights for the device. Device may also contain security attributes such as access rights or other access control information. A classic capability is represented as a ticket as: (Device, Rights, Random) in which the first item is the name / id of the device, second is the set of access rights and the third is a random number to prevent forgery. Algorithm for one way hash function can be made publicly available. It should be secret keys independent because key distribution introduces other difficulties. Benefits of using one way hash function are that it is computationally infeasible to inverse hash function and, given a pair of input and matching output it is infeasible to find a second input which gets the same output. When an access request arrives together with a capability consisting of object id, the one-way function is run to check the result against the random number to detect tampering. If the capability is valid, the access is granted [23].

Table 2.2 refers to the notations used in the proposed protocol. Working of this protocol is shown in figure 2.8.

Table 2.2: Notation used

Notation	Description
ID_1	Unique device 1 identifier
$Rights_1$	Access Rights of Device 1
r	Random number generated by Device 1
ID_2	Unique device 2 identifier
$Rights_2$	Access Rights of Device 2
H()	one-way hash function
E()	Encryption using AES-GCM
K	Secret Key
RNG	Random Number Generator

There are two components of this protocol: first is the creation of capability and second component is an application of AES – GCM. Device 1 creates its capability which is a function of device id and access rights which is then encrypted and hashed along with a random number to prevent forgery. Underlying algorithm for encryption is AES-GCM. Cipher text which is created is sent to device 2. Device 2 receives the capability of device 1 in encrypted form which is decrypted using symmetric key. Tampering of received cipher text is verified using one way hash function. If the generated hash value and the received hash value do not match then it is evident that the communication has been tampered and some other device is trying to impersonate and the authentication is violated. If there is a match in generated hash value and received hash value after decryption, then device 1 is authenticated to device 2. Encryption and its hardware implementations are efficient in resource constrained devices due to features of AES-GCM. The computations overhead on device are less optimizing energy.

As it is a mutual authentication protocol, device 2 have to authenticate itself to device 1. For this, device 2 creates its capability by same method as explained above and uses the same random number sent by device 1 to prevent from replay attacks. After receiving this response at device1, it decrypts this cipher text and checks the integrity and compares the random number to ensure that this message is coming from the same device which is authenticated by device 1. After successful decryption and comparison, device 2 is authenticated to device 1 and they are free to communicate with each other over secure channel. It is very important to note that, access right has been communicated to each other securely to achieve secure access control. This protocol is challenge response type of protocol which alleviates the overhead on both the devices.

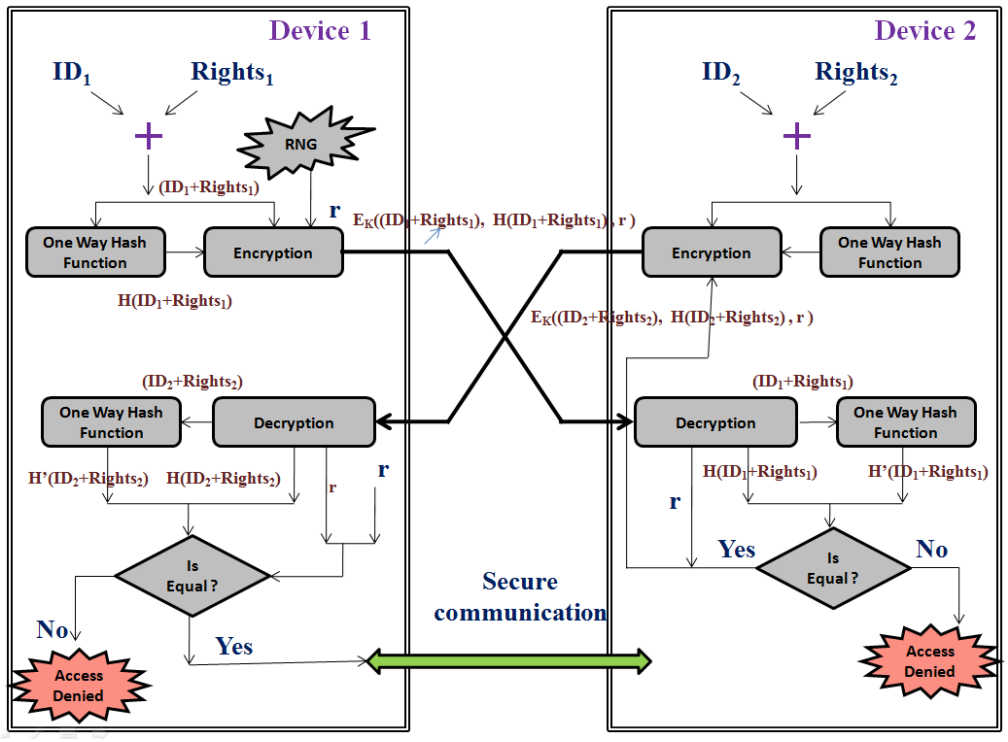

Figure 2.8 Proposed protocol

2.6.3 Evaluation of Proposed Protocol

The proposed protocol is evaluated in terms of its mutual authentication process, resistance to attack and efficiency.

- **Mutual authentication:** Only legitimate devices can generate and verify capabilities as it is based on secret key, one way hash function. As device identifiers and secret key are private and are being sent in encrypted form over communication channel, it is being prevented from forgery. AES-GCM provides encryption and authentication to capabilities and hence mutual authentication is successfully validated.
- **Replay attack resistance:** This resist-attack model is secure for replay attacks, as every challenge and response is encrypted with the random number.
- **Computational, traffic and storage cost:** The proposed protocol keeps computational costs low by requiring only four hashes to validate tampering. To guarantee that the device is legitimate, challenge and response protocol proposed here sends only three parameters. Thus the traffic cost between two devices is low. Device needs storage cost only for storing device identifier and secret key. We assume here that appropriate key management is being used.

2.7 Conclusions

Embedded security for IoT will be crucial and important with strong security mechanisms which will prevent damages and economical losses offering new business opportunities. However, sound security solutions are not attained easily. There are many challenges that should be defied. A sound solution considers the security from the beginning i.e. from design to implementation, to detect the vulnerabilities from the birth to the death of system. After discovering the sources and the reasons of vulnerabilities, safeguards should be embedded in the design methodology. A embedded security framework and architecture is dependent on precise definitions of parameters like resource constraints, the network specification (protocols, throughput, topology, services, etc...) and the system specification (protocols, device size, service which are managed, multi-rate specification, etc.). This will provide the necessary information to define the boundaries between the secure and insecure part of the system (data and hardware levels). Proper system-level study will enable the selection of the candidate solutions for the hardware and software parts. These candidates will be used, together with the specifications, as inputs for the hardware/software co-design methodology which will lead to a security framework and architecture for IoT system.

The AES-GCM protocol ensures authentication and access control by adding the capabilities as a second line of defense. It uses a secret value S, random number r, and hash function h() as both static and dynamic security guards. Only authenticated devices can recognize the right values of these numbers and access control is achieved correctly. Novelty of this protocol is in use of AES –GCM to provide both authentication and encryption with efficient low cost implementation in resource constrained devices

2.8 References

[1] Rolf H. Weber , "Internet of Things – New security and privacy challenges", Computer Law & Security Review, Volume 26, Issue 1, January 2010, 23-30.

[2] Kermani, M.M., Meng Zhang, Raghunathan A., Jha, N.K., "Emerging Frontiers in Embedded Security", 26th International Conference on VLSI Design 2013 and 12th International Conference on Embedded Systems (VLSID), 2013, vol., no., pp.203,208, 5-10 Jan. 2013.

[3] Gebotys C.H., Tiu C.C., Chen X., "A countermeasure for EM attack of a wireless PDA," International Conference on Information Technology: Coding and Computing, ITCC 2005. ,Vol. 1, 4-6 April 2005, 544-549.

[4] Hodjat A., Verbauwhede I., "High-throughput programmable crypto-coprocessor," Micro, IEEE , vol.24, no.3, pp. 34-45, May-June 2004.

[5] Tiri K. and Verbauwhede, "Design Method for Constant Power Consumption of Differential Logic Circuits", In Proceedings of the Conference on Design, Automation and Test in Europe - Volume 1 (March 07 - 11, 2005). Design, Automation, and Test in Europe. IEEE Computer Society, Washington, DC, 628-633.

[6] T. Kerins, W.P. Marnane E.M. Popovici, "An FPGA Implementation of a Flexible Secure Elliptic Curve Cryptography Processor", International Workshop on Applied Reconfigurable Computing ARC 2005, Proceedings, pp. 22-30, 2005.

[7] Murphy Gerard, Keeshan Aidan, Agarwal Rachit, Popovici Emanuel,"Hardware - Software Implementation of Public-Key Cryptography for Wireless Sensor Networks ", Irish Signals and Systems Conference, IET 2006, 28-30 June 2006, 463 – 468.

[8] Wilson P, Frey A, Mihm T, Kershaw D, Alves T., "Implementing Embedded Security on Dual-Virtual-CPU Systems" , Design & Test of Computers, IEEE Volume 24, Issue 6, Nov.-Dec. 2007

[9] RomainVaslin, Guy Gogniat, Jean-Philippe Diguet, Eduardo Wanderley, Russell Tessier, Wayne Burleson, "A security approach for off-chip memory in embedded microprocessor systems", Microprocessors and Microsystems, Volume 33, Issue 1, February 2009, 37-45

[10] Olga Gelbart, EugenLeontie, BhagirathNarahari, Rahul Simha, "A compiler-hardware approach to software protection for embedded systems", Computers and Electrical Engineering 35 (2009) 315–328, 2008 Elsevier Ltd.

[11] Fons M., Fons F., Canto, E., "Embedded security: New trends in personal recognition systems"; Microelectronics and Electronics Conference, 2007. RME. Ph.D. Research in 2-5 July 2007.

[12] Saputra H., Ozturk O., Vijaykrishnan N., Kandemir M., Brooks R.,"A data-driven approach for embedded security" ; VLSI, 2005. Proceedings of IEEE Computer Society Annual Symposium, 11-12 May 2005, pp. 104 – 109, 2005.

[13] Srivaths Ravi, AnandRaghunathan, Paul Kocher, Sunil Hattangady , "Security in embedded systems: Design challenges " ,August 2004 , Transactions on Embedded Computing Systems (TECS) , Volume 3 Issue 3 , ACM.

[14] Matthew Eby, Jan Werner, Gabor Karsai, AkosLedeczi, "Embedded systems security co-design" , April 2007, SIGBED Review , Volume 4 Issue 2 ,ACM.

[15] Parikshit Mahalle, Sachin Babar, Neeli R. Prasad and Ramjee Prasad, "Identity Management Framework towards Internet of Things (IoT): Roadmap and Key Challenges", The Third International Conference on Network Security and Applications (CNSA 2010), India, Springer Berlin Heidelberg, 2010, Volume 89, Part 2, 430-439.

[16] Sachin Babar, Parikshit Mahalle, Antonietta Stango, Neeli Prasad and Ramjee Prasad, "Proposed Security Model and Threat Taxonomy for the Internet of Things (IoT)" , The Third International Conference on Network Security and Applications (CNSA 2010), India, Springer Berlin Heidelberg, 2010, Volume 89, Part 2, 420-429.

[17] Feldhofer, Martin, Dominikus, Sandra, Wolkerstorfer, Johannes, "Strong Authentication for RFID Systems Using the AES Algorithm" , Cryptographic Hardware and Embedded Systems - CHES 2004, Lecture Notes in Computer Science 2004, Springer Berlin-Heidelberg, Volume 3156, 85-140.

[18] Hori Y., Satoh A., Sakane H., Toda K.,"Bitstream encryption and authentication with AES-GCM in dynamically reconfigurable systems," International Conference on Field Programmable Logic and Applications, FPL 2008. , vol., no., pp.23-28, 8-10 Sept. 2008

[19] Gang Zhou, Michalik H., Hinsenkamp L.,"Efficient and High-Throughput Implementations of AES-GCM on FPGAs," International Conference on Field-Programmable Technology, ICFPT 2007, vol., no., pp.185-192, 12-14 Dec. 2007.

[20] Dworkin M., "NIST Special Publication 800-38D: Recommendation for Block Cipher Modes of Operation:Galois / Counter Mode (GCM) and GMAC." , U.S. National Institute of Standards and Technology http://csrc.nist.gov/publications/nistpubs/800-38D/SP-800-38D.pdf

[21] J. B. Dennis and E. C. van Horn, “Programming Semantics for Multiprogrammed Computations”, Communications of the Association for Computing Machinery, 9(3):143–155,Mar. 1966.

[22] R. S. Fabry. “Capability-based addressing”, Communications of the Association for Computing Machinery ACM, 17(7), 1974, 403–412.

[23] Lampson, Butler W., "Protection". Proceedings of the 5th Princeton Conference on Information Sciences and Systems, 437, 1997.

[24] Li Gong, "A Secure Identity-Based Capability System", IEEE Symposium on Security and Privacy, p. 56, 1989.

3

Jamming Attack: Modelling and Evaluation

This chapter introduces the jamming attack that take place at physical layer and its classification in detail. It also provides modelling of jamming attack using activity- and sequential- modelling approach and evaluates the different jamming attack in variety of network situations. The chapter proposes new possibility of the jamming attack i.e. intelligent cluster-based jamming attack and evaluates the performance impact of cluster-based jamming attack. Lastly, the chapter discusses the requirements to design efficient defense mechanism against jamming attack.

3.1 Introduction

The research in WSN is growing in large perspective to offer the wide variety of application domains. The WSN consist of the large number of nodes, which sends the sensed information to the central base station (BS) [1]. The WSN node suffers from large energy constraint because of its limited battery power. The major requirement to achieve quality of service (QoS) in WSN is to reduce energy consumption with minimum delay and maximum throughput. These performance requirements are largely affected by security attacks, which happen at various layers of WSN.

The main objective of this chapter is to model the jamming attack [2, 3], which is one of the denials of service attack [4] which blocks the channel by introducing malicious traffic. WSN is vastly invaded by the different kinds of jamming attacks at each layer. The chapter mainly concentrates on jamming attacks, which occur at physical and medium access control (MAC) layer. Here, it is more effective and destructive because these layers are mainly responsible for allocating the resources. The different kind of active and reactive jamming attack effects on WSN constraints based behaviour, by increasing the energy consumption with increased delay and decreased throughput. These are very important performance parameter for deciding QoS of WSN. The different kinds of jamming attacks are constant jamming, deceptive jamming, random jamming, and reactive jamming. All these jamming attacks are modelled to understand the basic sequence of activities during their occurrences in the network. The author uses unified modelling language (UML) [5] based activity and sequential modelling approaches for modelling the behaviour of various jamming attacks. Activity modelling models the behaviour by considering different states and shows the various conditions, message transmission between the states. It is one of the useful ways to understand the intelligent behaviour of jamming attack. The activity modelling also gives the understanding of required security solution for reducing the effect of attack on WSN performance. Sequential modelling is one of the widely used ways to model the system using UML. It is used to illustrate the interactions between different entities of system.

The next objective of chapter is to evaluate the jamming attack and to understand the level of performance degradation due to different kind of jamming attacks. The evaluation is performed using varying time interval and number of malicious nodes in the network. The evaluation show that the reactive jamming attack is one of the unpredictable and disastrous jamming attacks as compared to other jamming attacks. The chapter also provides insight on new kind of jamming attack i.e. intelligent cluster head (CH) jamming attack. This attack initiates the attack on CH and penetrates it in the whole network. The performance evaluation of intelligent CH jamming attack shows that, the attack is more destructive than normal reactive jamming attack.

3.2 Jamming Attack classification

The jamming attack is classified as,

- Constant Jamming
- Deceptive Jamming
- Random Jamming
- Reactive Jamming

Constant jamming: The constant jamming attack jammer continually emits a radio signal, and can be implemented using either a waveform generator that continuously sends a radio signal or a normal wireless device that continuously sends out random bits to the channel without following any MAC-layer etiquette [8]. Normally, the underlying MAC protocol allows legitimate nodes to send out packets only if the channel is idle. Thus, a constant jammer can effectively prevent legitimate traffic sources from getting hold of a channel and sending packets.

Deceptive jamming: Instead of sending out random bits, in the deceptive jamming the jammer constantly injects regular packets to the channel without any gap between subsequent packet transmissions. As a result, a normal communicator will be deceived into believing there is a legitimate packet and be duped to remain in the receive state. Even if a node has packets to send, it cannot switch to the send state because a constant stream of incoming packets will be detected.

Random jamming: Instead of continuously sending out a radio signal, a random jammer alternates between sleeping and jamming. Specifically, after jamming for a while, it turns off its radio and enters a "sleeping" mode. It will resume jamming after sleeping for some time. During its jamming phase, it can behave like either a constant jammer or a deceptive jammer. This jammer model tries to take energy conservation into consideration, which is especially important for those jammers that do not have unlimited power supply.

Reactive jamming: The three models discussed above are active jammers in the sense that they try to block the channel irrespective of the traffic pattern on the channel. Active jammers are usually effective because they keep the channel busy all the time. Active methods are relatively easy to detect. An alternative approach to jamming wireless communication is to employ a reactive strategy. The reactive jammer stays quiet when the channel is idle, but starts transmitting a radio signal as soon as it senses activity on the channel. One advantage of reactive jammer is that it is harder to detect.

3.3 Modelling and Evaluation of Jamming Attack

This section models the behavior of different types of jamming attack using sequential and activity modelling approaches under unified modelling language (UML). The differences between activity modelling and sequential modelling are, (i) activity modelling gives high-level understandings of the system functionalities while sequential modelling gives low-level dynamic interaction between the objects, (ii) activity modelling describes the data flow between users and system while sequential modelling illustrates the objects involved and messages exchanged during the data transfer. Here, the modelling of jamming attack using activity- and sequential- modelling gives the complete understanding of attack behaviour with its high level data flows, objects involved, and messages exchanged during the interaction of different objects.

3.3.1 Activity Modelling of Jamming Attacks

The activity modelling explains the functional view of a system by describing or representing logical processes, or functions. Here, each logical process is represented as a sequence of

tasks and the decisions that govern when and how they are performed. Activity modelling is one of the UML representations for giving functional view of any processes or tasks [5, 8]. UML is designed to support the description of behaviours that depends upon the results of internal processes. The flow in an activity diagram is driven by the completion of an action. The activity diagram is useful tool to understand the basic flow of security attacks.

3.3.1.1 Constant Jamming

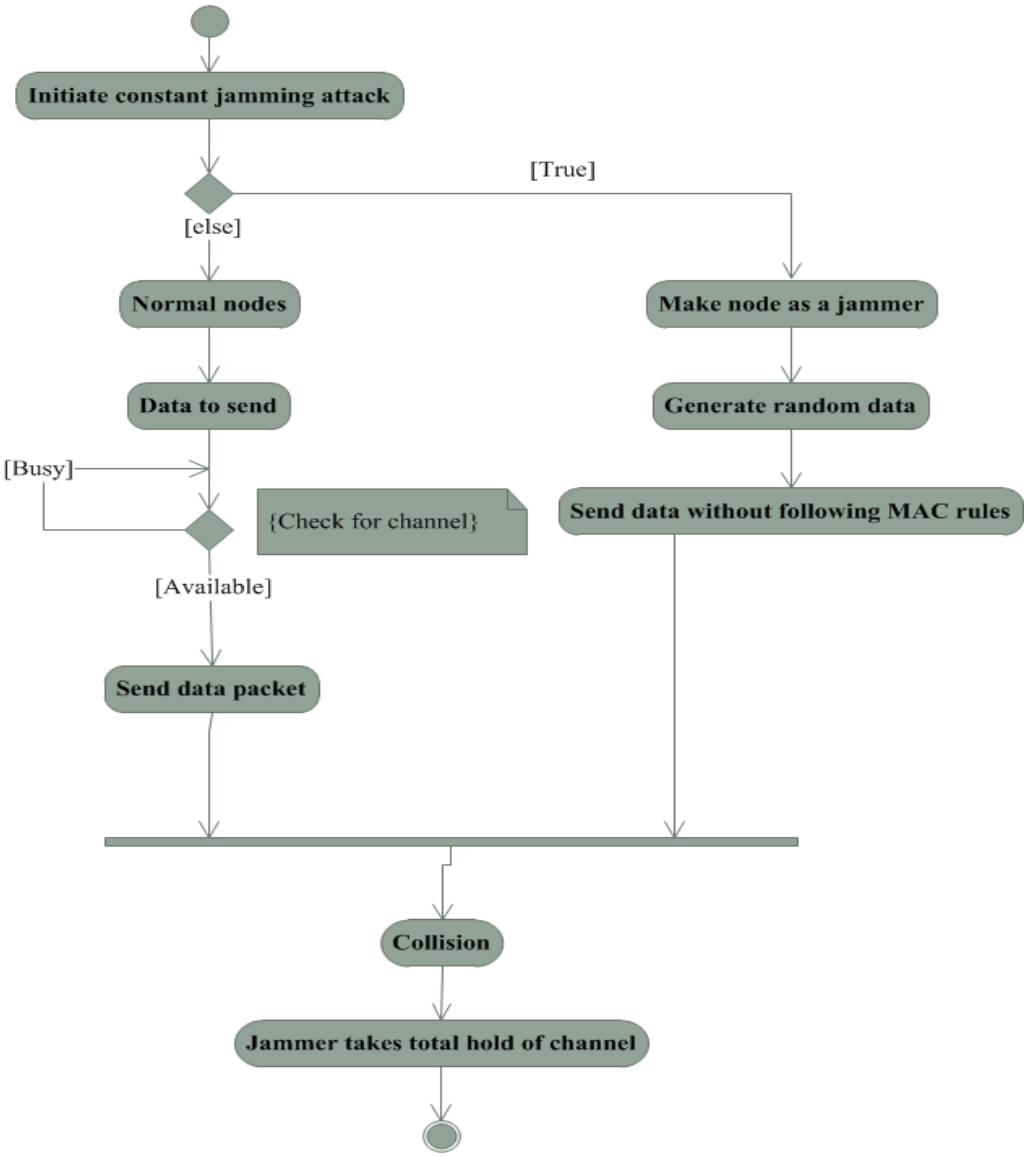

Figure 3.1: Activity modelling of constant jamming attack

Figure 3.1 shows the activity modelling of constant jamming attack. It gives insight of different activities that takes place during the execution of attack on a network. The sequences of activities are as follows,

- The attacker initiates the constant jamming attack. If attack is successful then node in a network will behave like a constant jammer and start to jam the network, otherwise node will do a regular activity.

- The normal node detects some event and tries to send the data to another node or destination. It checks for availability of channel, if channel is available then it will send data on the channel and send it towards the destination. If channel is not available then it will check for channel repeatedly after some particular interval.
- The jammer node generates the random data after some particular time interval and it will try to send the random data without following MAC rules i.e. without checking for channel.
- The random data generated from the jammer node may collide with data coming from normal node and it jams the whole traffic in the network by increasing the collision in network. The severity of constant jamming will be more if the interval between the random generations of data is too small.

3.3.1.2 Deceptive Jamming

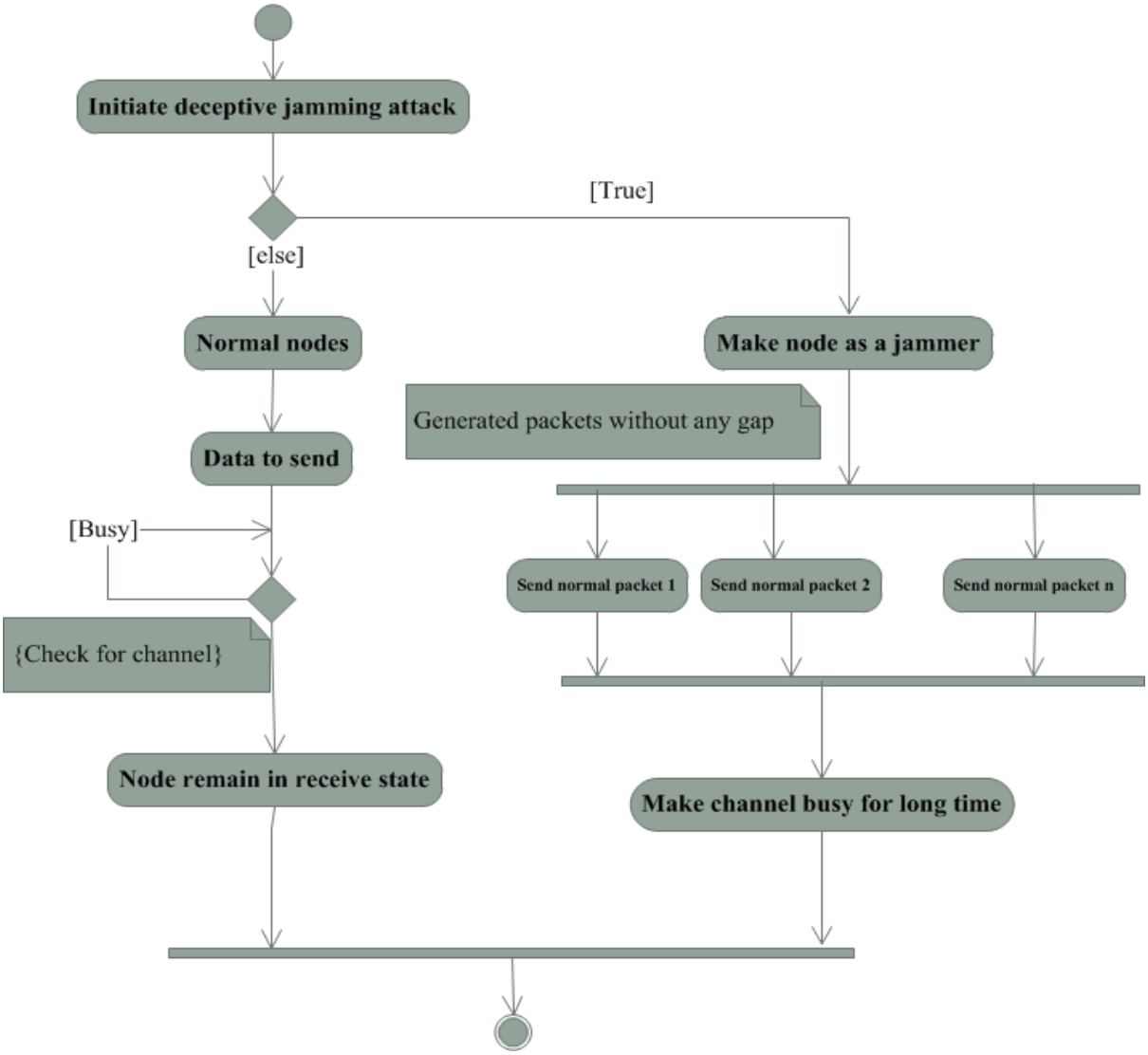

Figure 3.2: Activity modelling of deceptive jamming attack

Figure 3.2 shows the flow of activities in case of deceptive jamming attack. In case of deceptive jamming, attacker will take whole charge of channel by making the channel busy. The different activities that happen during accomplishment of attack are as follows,

- The external attacker initiates the deceptive jamming attack on node in a network. If attack is successful the normal node will act like a deceptive jammer otherwise it will behave like a normal node.
- The normal node generates the data and tries to send the data towards the destination by checking the availability of channel.
- The jammer node generates the data packets continuously without keeping any time gap between the two packets. This continuous generation of packets put the channel in busy state for long time.
- The busy state of channel because of deceptive jamming keeps other normal node to be in receiving state. This behavior of deceptive jamming increases the energy consumption, delay and decreases the total throughput of the network.

3.3.1.3 Random Jamming

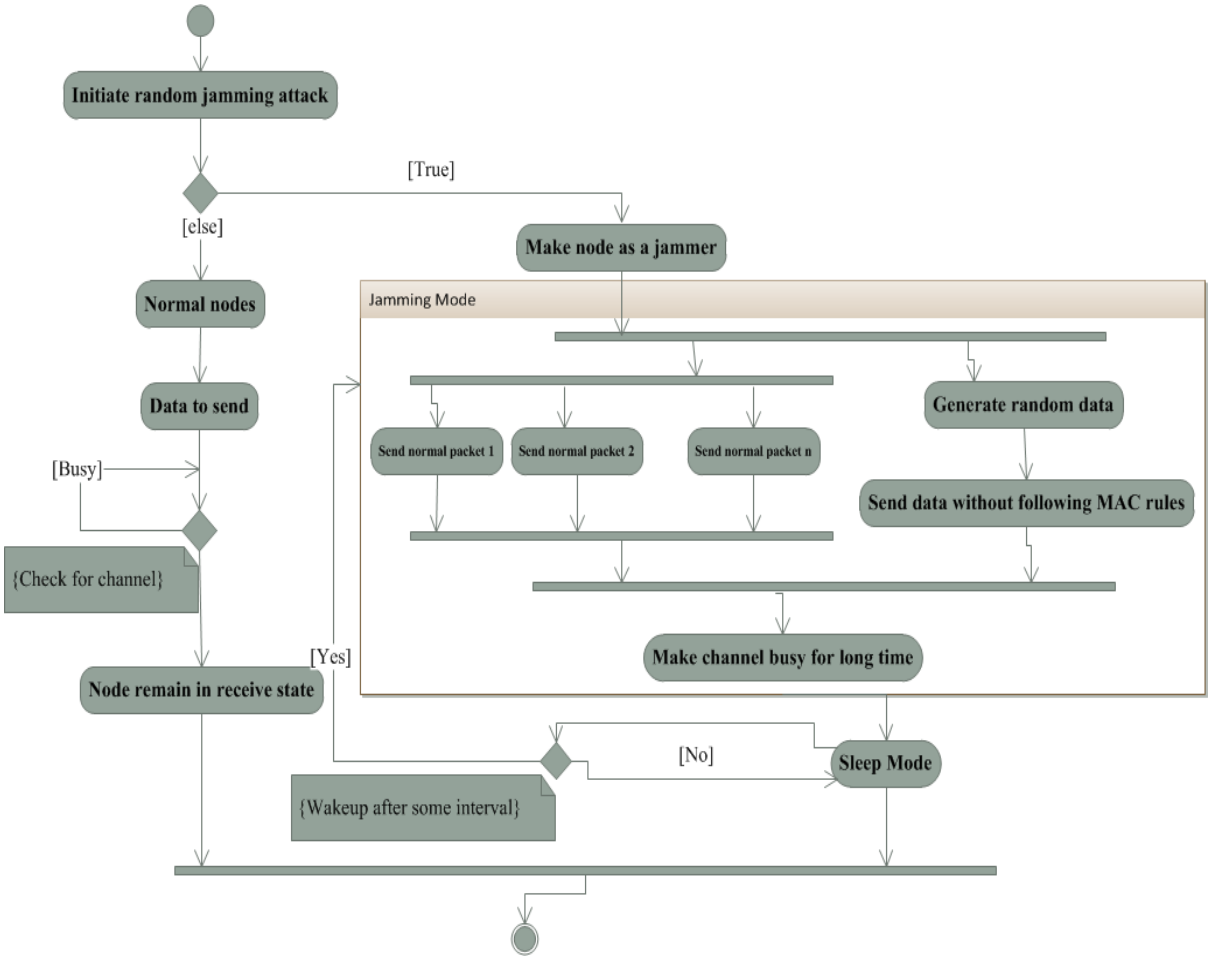

Figure 3.3: Activity modelling of random jamming attack

Figure 3.3 shows the different activities that takes place during the execution of random jamming attack. The random jamming attack is kind of intelligent attack where the jamming node thinks for saving of its own energy. Therefore, it works in two modes, jamming mode, and sleep mode. The details of execution of attack are as follows,

- If attack is successful, then the external attacker will initiate the attack by converting the normal node into jamming node.
- If channel is available, the normal node detects some event and tries to send the data packet towards another node or destination. The sender node checks for channel availability every time whenever it has data to send.
- The jammer node here works in two modes to save its energy and to last its effect for long time. In jamming mode it make channel busy either by continuously generating packet like deceptive jamming or generate random data after some specified interval without following MAC rules like constant jamming.
- The continuous block of channel by jammer node place the normal node in receive state for long time.
- The normal node changes its receiving state or can get the availability for some time whenever jammer node goes to sleep state. This behavior of attack introduces the longer amount of delay in the transmission of data from the node.

3.3.1.4 Reactive Jamming

Figure 3.4 shows the activity modelling of reactive jamming. It shows the execution steps of nodes in a network in case of reactive jamming. The steps are as follows,

- The reactive jamming attack is initiated by attacking on normal node, if it is successful then node will act like a reactive jammer, otherwise the normal node does its designated operations.
- The main feature of the attack is that it gets activated when other nodes in the network are busy to send data or if the channel is busy.
- Here, the normal node tries to send data towards the concern destination by checking the availability of channel and send the data on channel.
- The jammer node checks the status of channel. If channel is ideal it will go to quiet state where it will do nothing, else if channel is busy the jammer will activate and generate the noise packet continuously which results in collision in the network.
- The reactive jammer gets activated when the channel is busy. Therefore, it is very difficult to detect and reduce the effect of channel on performance of network.

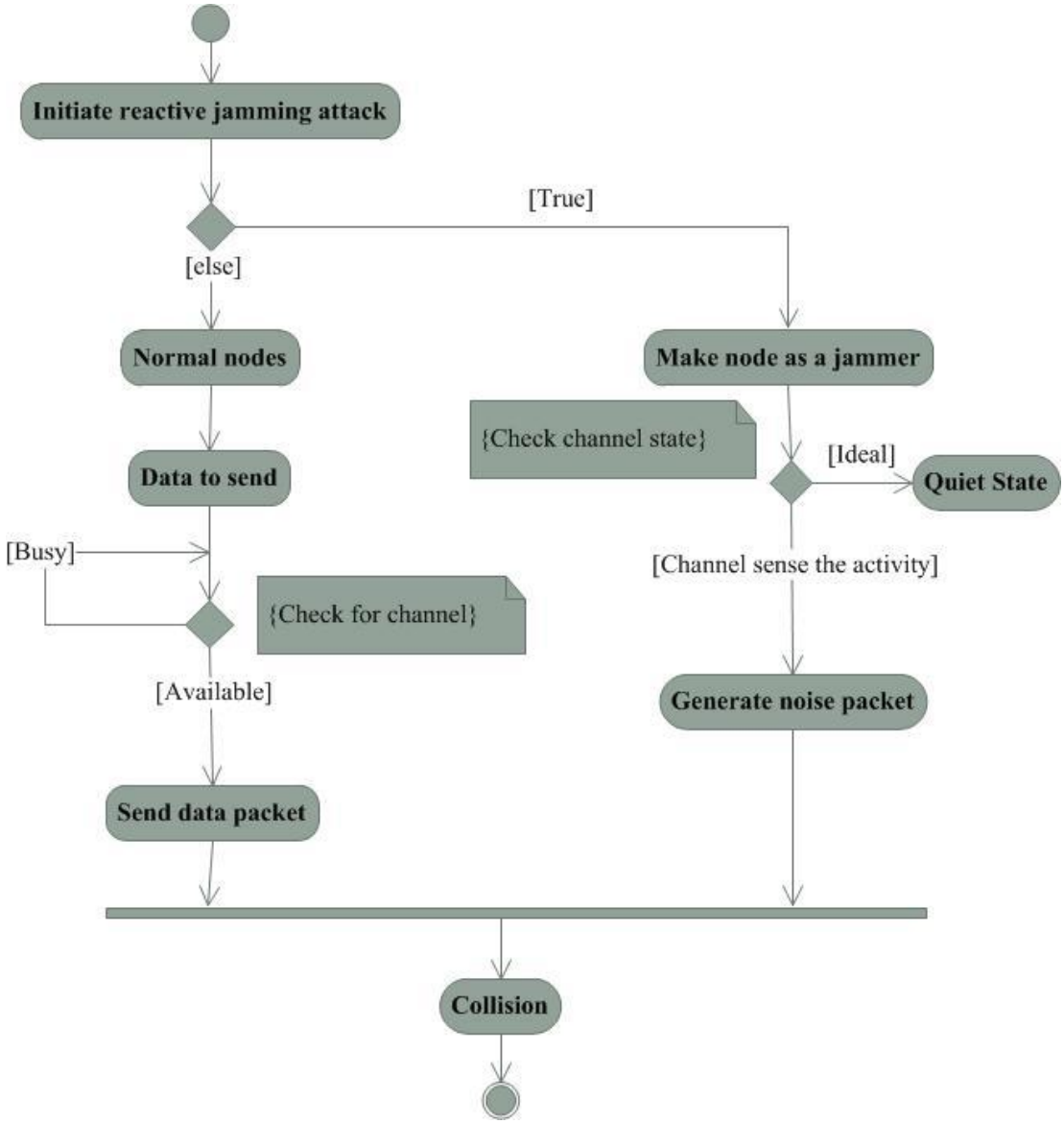

Figure 3.4: Activity modelling of reactive jamming attack

3.3.2 Sequential Modelling of Jamming Attack

The sequence diagram is used primarily to show the interactions between objects in the sequential order in which those interactions occur. The sequence diagram is also called as message sequence charts. A sequence diagram shows, as parallel vertical lines, the different processes or objects that live simultaneously, and, as horizontal arrows, the messages exchanged between them, in the order in which they occur.

It considers jamming attacker and different nodes in network as entities and interaction between them as the processes. It also considers normal behaviour of each node as, node transmit data after successful exchange of RTS and CTS. In each attack situation, external attacker initiates the attack on any of the node in the network and converts those nodes into malicious nodes, who are acting as a malicious node or jammer.

A. Constant Jamming Attack

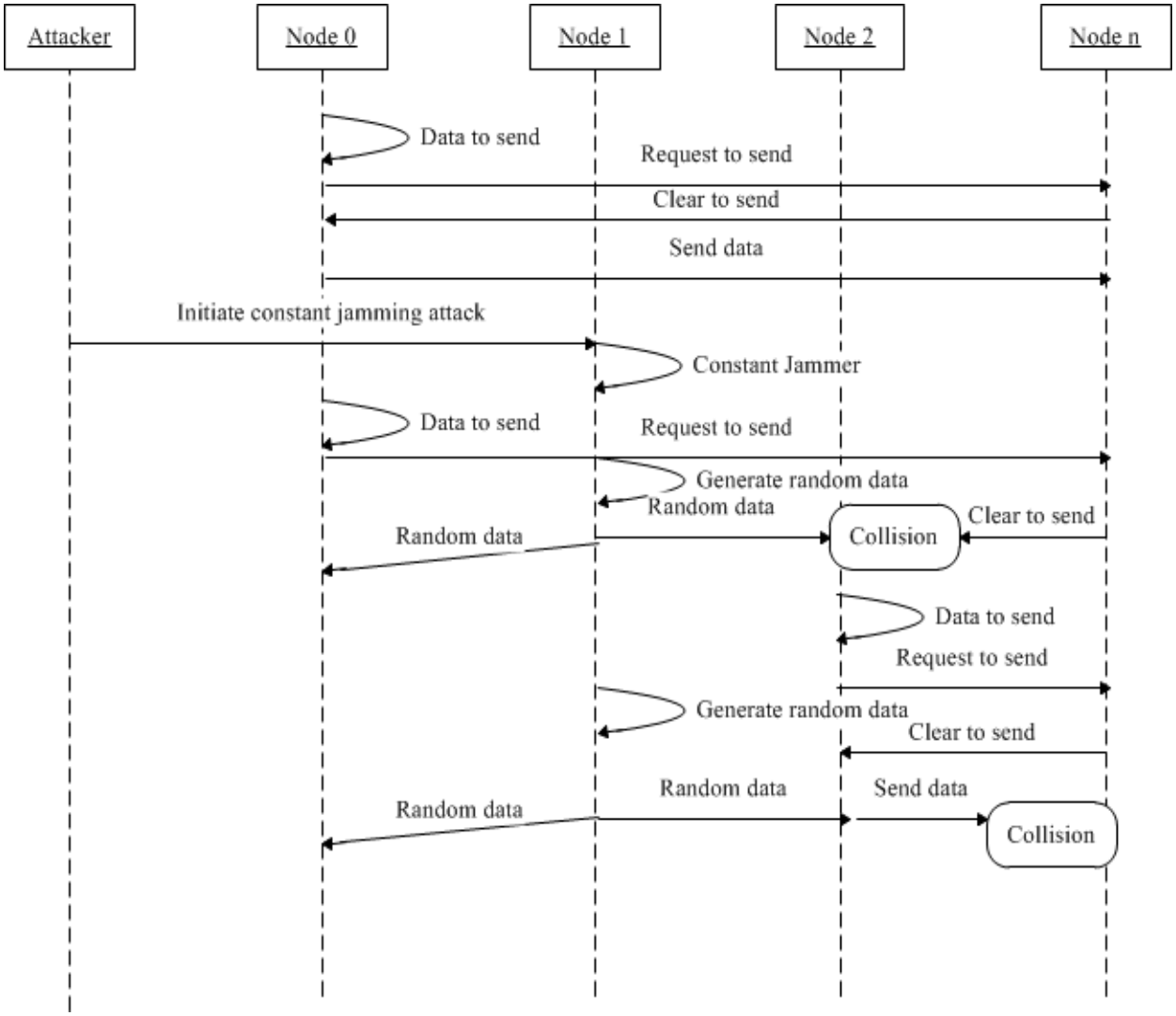

Figure 3.5: Sequential modelling of constant jamming attack

Figure 3.5 show the sequential modelling of constant jamming attack. It shows sequence of interaction between normal nodes, malicious nodes, and attacker. The sequence of activities is as follows,

- Nodes 0 have data to send, it checks the channel by transmitting request to send (RTS) packet and gets the reply as clear to send (CTS), if channel is available then it send the data towards consigned destination.
- The external attacker initiates the constant jamming attack on any of the node in network and converts that node as malicious nodes, who now act as constant jammer.
- The normal node 0 have data to send, it transmit RTS packet towards node n for checking the channel in between them. The RTS found channel idle so destination node n start to send the CTS packets towards node 0. Here, at same time if constant jammer node, node 1 generates random data, it will collide with CTS coming from node n.

- Here, constant jamming node, node 1 generated random data after some particular interval.
- The constant jammer is activated again after some interval and generates random data and transmits it in the network. During that time another node 2 send RTS, receive CTS and starts to send data but data from node 1 will collide with random data generated from constant jammer node 1.

B. Deceptive Jamming Attack

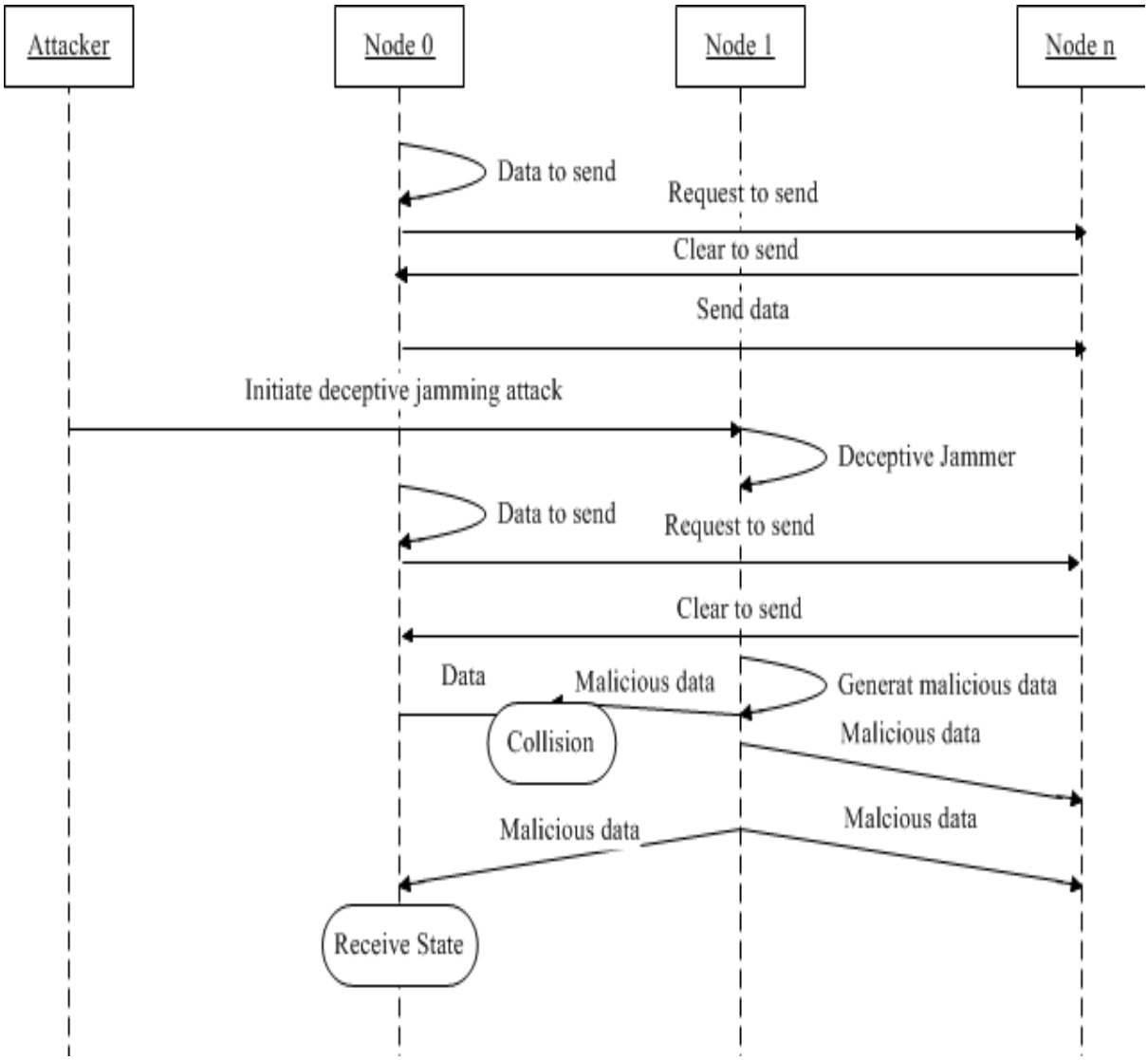

Figure 3.6: Sequential modelling of deceptive jamming attack

Figure 3.6 show the modelling of deceptive jamming attack using sequential modelling approach. Sequential modelling of deceptive jamming shows the different actions that take place on different objects (attacker and normal nodes) during execution of attack. The actions are as follows,

- Nodes 0 have data to send, it checks the channel by transmitting request to send (RTS) packet and gets the reply as clear to send (CTS), if channel is available then it send the data towards consigned destination.
- The attacker initiates the deceptive jamming attack on node 1 and converts it as deceptive jammer. The deceptive jammer generates the malicious data continuously without any difference between the two malicious data.
- Here, node 0 sends the RTS packet, receive CTS and try to send data packets towards the destination. The data packets may collide with malicious data and generates collision on channel.
- The deceptive jammer generates malicious data continuously which increase the collision in the network and may place large number of nodes in the network in receive state.

C. Random Jamming Attack

Figure 3.7 show the sequence of activities in case of random jamming attack. The different events and message passed in network are as follows,

- Nodes 0 have data to send, it checks the channel by transmitting request to send (RTS) packet and gets the reply as clear to send (CTS), if channel is available then it send the data towards consigned destination.
- The attacker initiates the random jamming attack on any node in the network. Here, it initiated the attack on node 1 and converts it as random jammer which act randomly as constant jammer or deceptive jammer.
- Figure shows that the node generates the random data after some interval and leads to collision after regular interval like a constant jammer.
- Here, the random jammer acts intelligently and tries to save its energy by going to sleep state. The jammer node goes to sleep node after jamming the network for some amount of time for saving its energy and to last the effect of attack for large time in network.
- After waking up from sleep mode it may act like a constant jammer or deceptive jammer. Here, it acts like a deceptive jammer which jams the network by increasing the collision and placing the nodes in receive state.

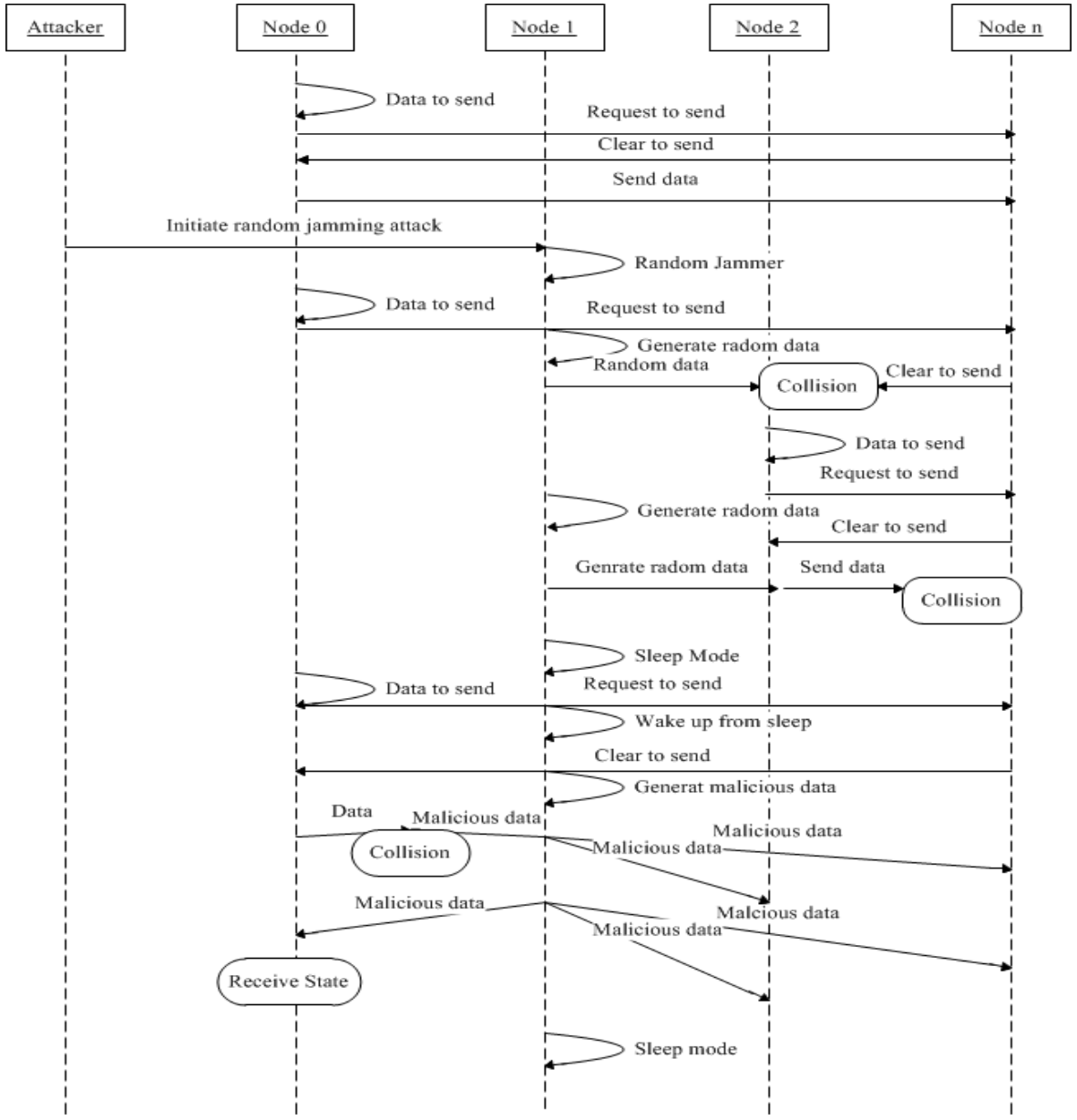

Figure 3.7: Sequential modelling of random jamming attack

D. Reactive Jamming Attack

Figure 3.8 show the sequential modelling of reactive jamming attack. The reactive jamming attack is the most intelligent jamming attack which reacts in the network by observing the events in the network. The sequences of action in the network are,

- Nodes 0 have data to send, it checks the channel by transmitting request to send (RTS) packet and gets the reply as clear to send (CTS), if channel is available then it send the data towards consigned destination.
- The attacker initiates the reactive jamming attack on node 1 and converts the normal node as reactive jammer which acts by analysing the state of the network.
- The reactive jammer goes to quiet state if there is no any event in the network.
- The normal node 0 sends RTS to node n, the reactive jammer node sense this activity on channel and sends the noise packets in the network. These noise packets may

collide with CTS packet and generate the collision in the network. The attacker gets activated here every time whenever jammer senses the activity on channel.

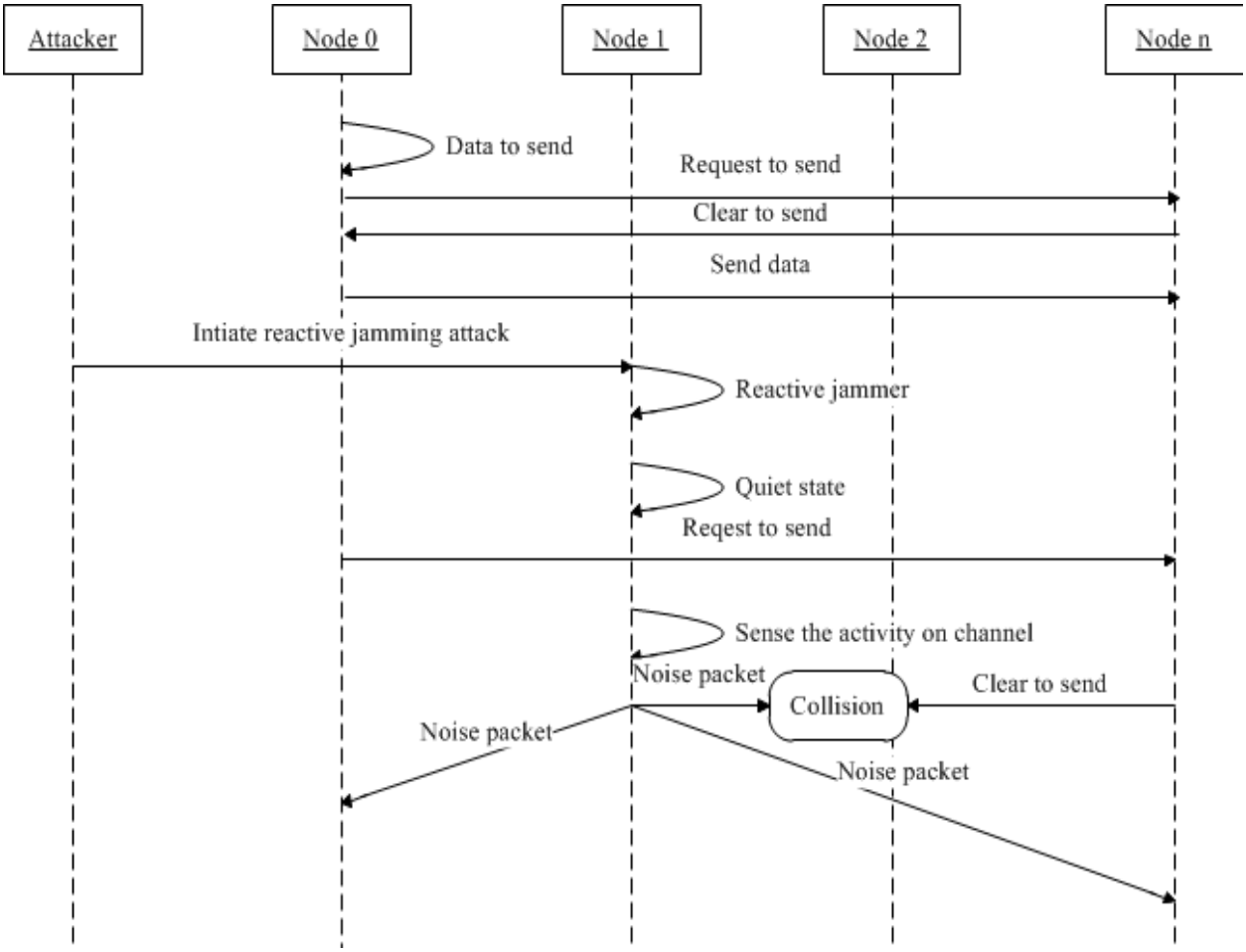

Figure 3.8: Sequential modelling of reactive jamming attack

3.3.3 Evaluation of Jamming Attacks

This section describes the evaluation of jamming attack under different network conditions. It is necessary to understand the actual working behaviour of jamming, which will be useful to develop good countermeasure on jamming. The implementation of jamming attack for evaluation is based on modelling described in the previous section. The modelling in previous section had given clear understanding of the objects involved during jamming and the interaction between them. The evaluation of jamming is performed to check the effect of jamming attack, under different traffic conditions and varying number of malicious nodes in the network. The evaluation of jamming in different traffic conditions is necessary to understand the jamming effect in varying traffic conditions. The evaluation by varying number of malicious nodes is good tool to understand the level of jamming.

3.3.3.1 Simulation Details

The implementation of all attack is performed by using discrete event simulator NS-2 (Network Simulator-2). The parameters set during simulations are shown in Table 3.1. The

idle power, receiving power, transmission power, and sleep power are considered according to IEEE 802.15.4 radio model [9].

The simulations are performed in five different conditions. The different conditions are,

- WSN without any security attack
- WSN with constant jamming attack
- WSN with deceptive jamming attack
- WSN with random jamming attack
- WSN with reactive jamming attack

The simulation of jamming attacks is done under following consideration,

- The simulation is performed by varying traffic interval, which is useful to measure the performance of attack under various traffic conditions. The traffic interval is varied from 1s to 10s. The 1s traffic interval is consider as fast traffic and 10s traffic interval is consider as slow traffic. These simulations consider number of malicious nodes in network or nodes under attack is one.
- The second set of simulation is performed by varying number of malicious nodes in the network. The number of malicious nodes in network considered is 1,2,4,8 and 16. The traffic interval considers under this simulation is 1s which is consider to be the fast traffic in network. These set of simulations will be useful to analyze the effect of attack by increasing the destructive entities in network.

Table 3.1 Simulation and node parameters

Parameter Name	**Setting Used**
Network Interface type	Wireless Physical:802.15.4
Radio Propagation Model	Two-Ray Ground
Antenna	Omni-directional antenna
Channel Type	Wireless Channel
Link Layer	Link Layer (LL)
Interface Queue	Priority Queue
Buffer size of IFq	50
MAC	802.15.4
Routing Protocol	Ad-hoc routing
Energy Model	EnergyModel
Initial Energy (initialEnergy_)	100J
Idle Power (idlePower_)	31mW
Receiving Power (rxPower_)	35mW
Transmission Power (txPower_)	31mW
Sleep Power (sleepPower_)	15μW
Number of nodes	100
Node Placement	Random

3.3.3.2 Results and Discussions

A. Performance by varying interval

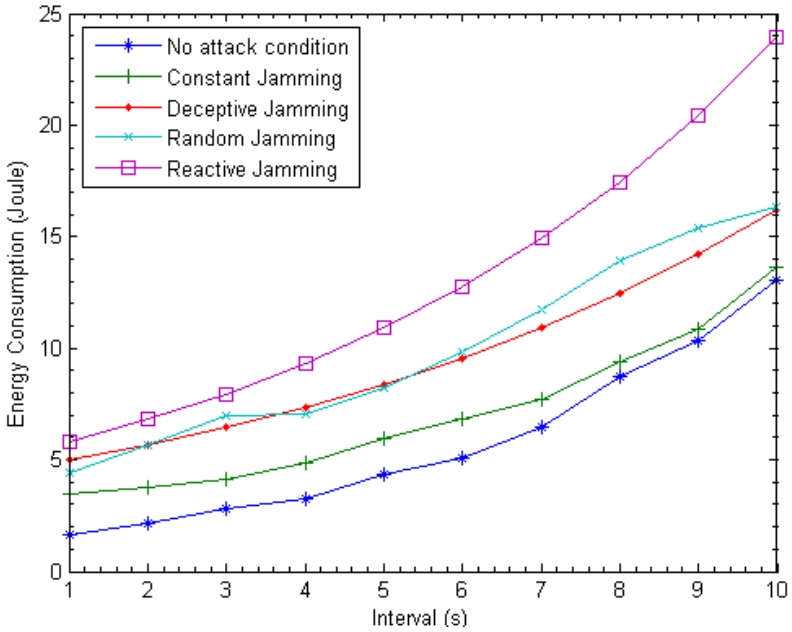

Figure 3.9: Comparative Energy Consumption Analysis of jamming attacks under varying traffic interval

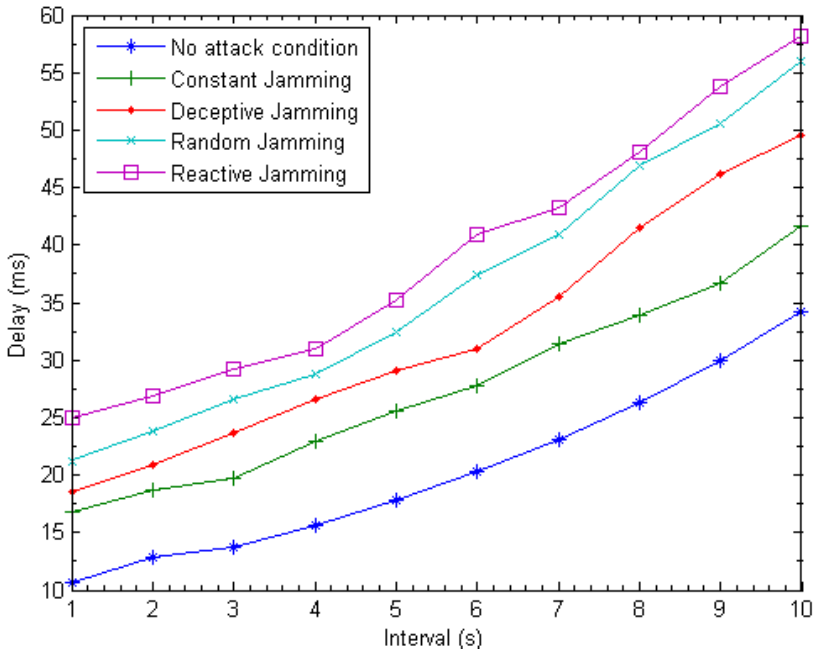

Figure 3.10: Comparative Delay Analysis of jamming attacks under varying traffic interval

Figure 3.9, 3.10, and 3.11 shows the comparative analysis of reactive jamming, random jamming, deceptive jamming, and constant jamming with no-attack condition by varying the interval in the network. The analysis is done by measuring three parameters of sensor network: energy consumption, delay, and throughput respectively as shown in figure 3.9, 3.10, and 3.11. The jamming attack reduces the performance of the WSN in larger manner. The reason of performance degradation under different types of jamming attack is as follows,

- Constant Jamming: The constant jamming attacks are initiated in the network by generating the noise packet which will be sent without following any MAC rules of the network. The figure shows that performance degradation by constant jamming attack is less than other kind of jamming attack because it jams the network after regular interval i.e. it generates the noise packets after some regular interval.
- Deceptive Jamming: The deceptive jamming jams the network by producing the noise packets continuously in the network without any time interval between the two noise packets. The main reason for showing more performance degradation than constant jamming is continuous generation of noise packets which increase the energy consumption, delay and decrease the throughput of network by producing large amount of collision in the network which jams the channel.
- Random Jamming: The random jamming randomly jams either by using constant jamming or deceptive jamming. Here, jammer node also thinks for its own energy by going to sleep for some amount of time. The performance curve of random jamming shows that its performance is varying in very random manner, sometimes it is more than deceptive jamming sometime it is less than deceptive jamming. The major reason for its random behavior is use of both kind of jamming according to situation. It is difficult to detect random attack because of its random behavior.
- Reactive Jamming: The performance graphs shows that the reactive jamming is most disastrous kind of jamming attack. Here, its performance is degrading with increase in interval. It produces the noise packets in network immediately after detecting any event on the channel. This behavior of it, corrupt or lose large number of packets in the network by introducing vast amount of collision in the network.

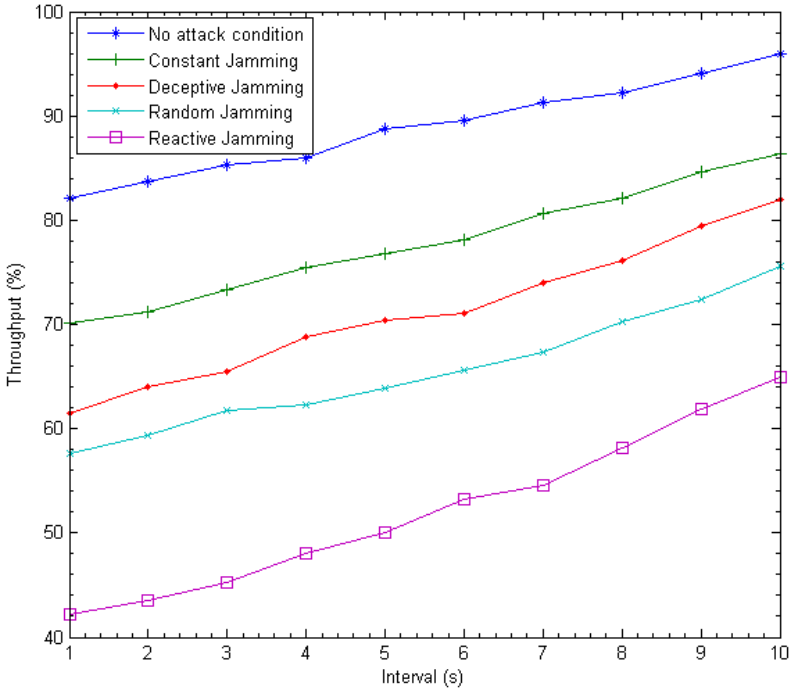

Figure 3.11: Comparative Throughput Analysis of jamming attacks under varying traffic interval

B. Performance by varying number of malicious nodes

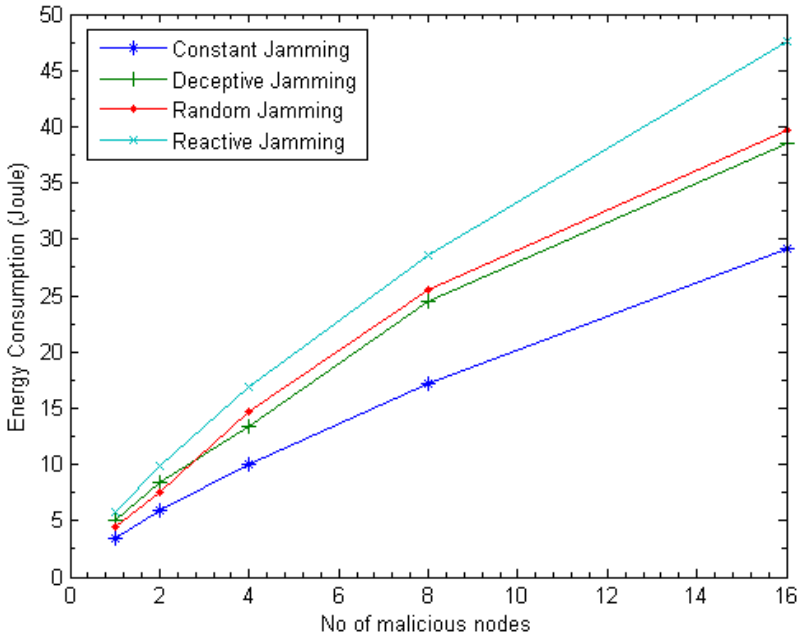

Figure 3.12: Energy consumption analysis of different jamming attacks with varying number of malicious nodes

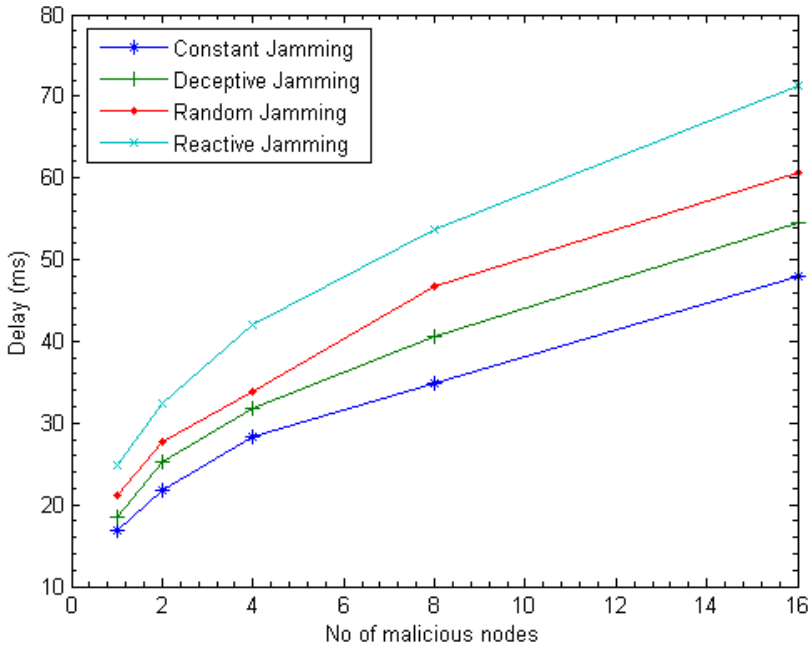

Figure 3.13: Delay analysis of different jamming attacks with varying number of malicious nodes

Figure 3.12, 3.13, and 3.14 describes the performance degradation of constant-, deceptive-, random- and reactive- jamming in terms of energy consumption, delay, and throughput by varying number of malicious nodes in the network. The graph shows that the performance degradation in network is increasing with number of malicious nodes in the network because more malicious nodes in the network generate more malicious traffic which helps in reducing the total performance of network. Figure 3.12 shows the energy consumption under different

jamming attack. Here, all attack performance reduction is increasing than one on another; only random jamming energy consumption is less than deceptive jamming under less number of nodes but it also more than deceptive- and constant- jamming with more number of nodes. The random jamming shows less energy consumption than deceptive jamming and more delay than it because it allows malicious nodes to go to sleep mode after some regular interval which helps to save energy but increase delay which is also not advantageous if number of nodes in network are more.

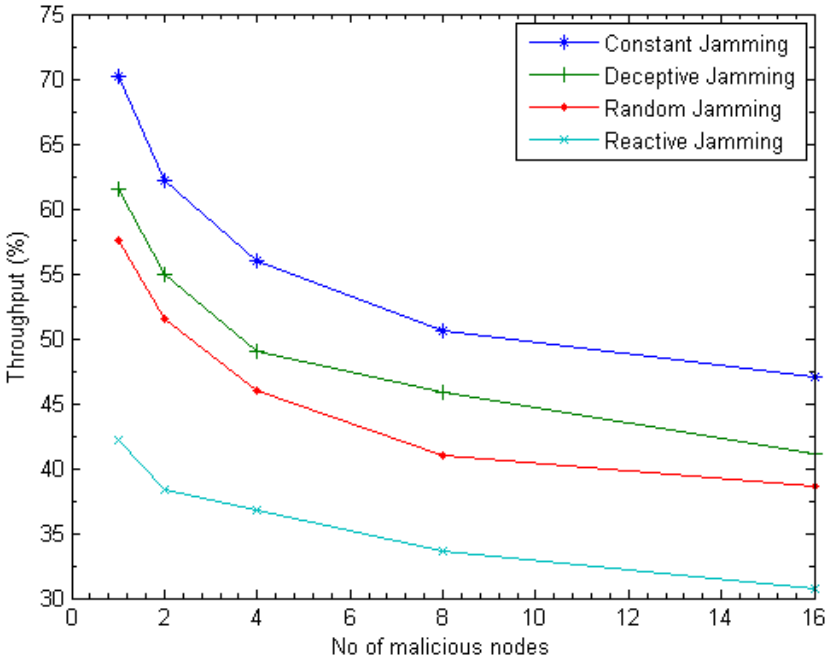

Figure 3.14: Throughput analysis of different jamming attacks with varying number of malicious nodes

3.4 Proposal of Cluster Based Jamming Attack

The previous sections of chapter describe the jamming attack, its modelling and evaluation under network situations. The jamming considered in the previous sections and in literature is mainly for flat network, where the network is not divided into the parts. This kind of network is more prone to jamming as attack penetrates in faster way and destruct the network. The other kind of network is cluster-based network, where network is divided into small parts, called as clusters. Each cluster consists of CH, other nodes in cluster communicate with CH, and CH transmits the information to BS on behalf of other nodes. Cluster-based networks are scalable, having good energy efficiency and less prone to attack, as attack penetration limits to cluster. Therefore, more IoT applications preferred to use cluster-based network [7, 10]. These growing demands of cluster-based network lead to security loopholes in the system. Here, the section gives the details of possible reactive jamming attack named “Intelligent Cluster Head Attack” in cluster-based network and its evaluation to show how it is more destructive than other kind of jamming attacks.

3.4.1 Intelligent Cluster Head Jamming Attack

The attacker consider in this attack is intelligent attacker who can differentiate between the cluster head (CH) [11] and normal node in the network and continuously taking track of cluster head traffic. The main task of CH is to aggregate the information from normal nodes in the network and send it to the base station or other in between CH. Here, intelligent jammer initiates the attack whenever it detects some event on CH i.e. whenever CH is ready to transmit some aggregated data or receive some data from normal node. Once the jammer detects the event on CH it initiates the attack on CH and makes the CH as malicious CH. The all links in the network are considered to be bidirectional. The malicious CH can generate noise packets towards the BS or other CH and also towards the normal node in that cluster. The noise packets transmitted inside the cluster jam the traffic inside the cluster i.e. it jams the intra-cluster traffic and noise packets transmitted in between the CH jams the inter-cluster traffic. This way it creates the black hole in network which starts to eat whole network by producing malicious data.

3.4.2 Sequential modelling of Intelligent Cluster-Head Jamming Attack

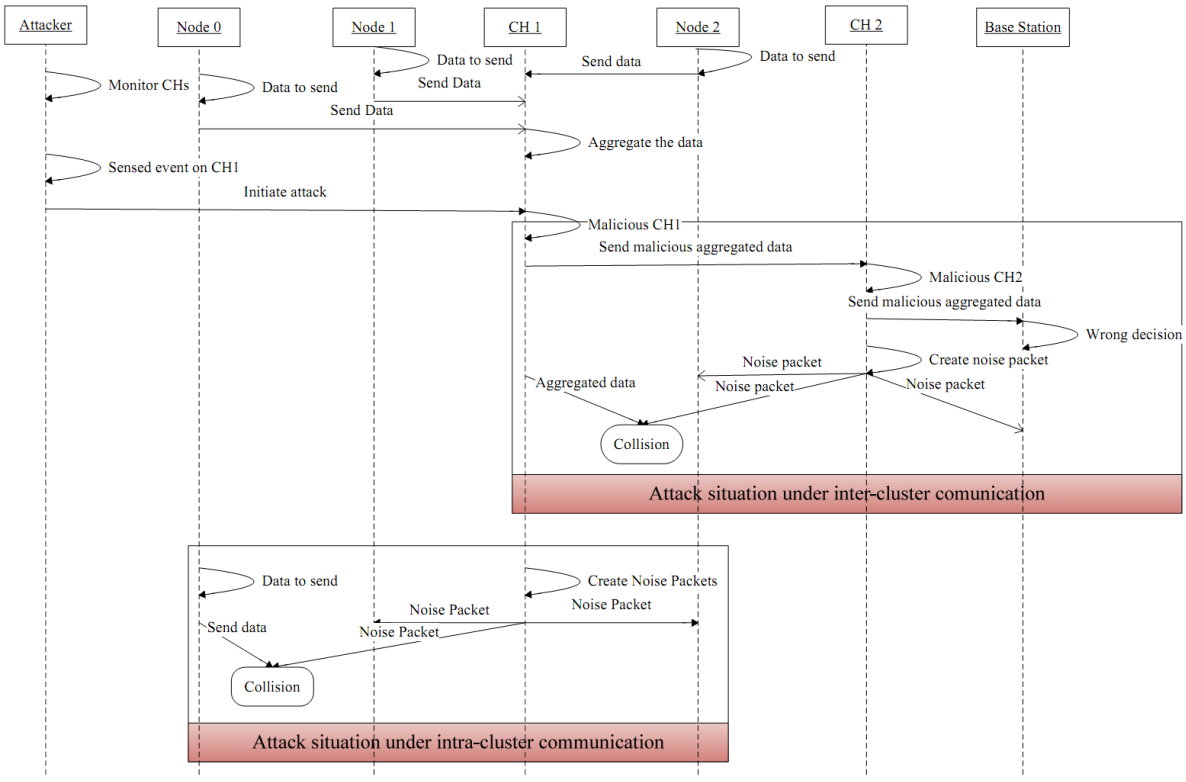

Figure 3.15: Sequential modelling of intelligent CH jamming attack

Figure 3.15 show the sequence of activities that happens during the deployment of intelligent CH jamming attack. The different activities are as follows,

- The attacker is continuously monitoring the traffic from the CH.

- Node 0, 1 and 2 have data to send and they will send it towards the CH, CH will aggregate the information and will try to send the data towards another CHs or BS.
- Whenever attacker senses the traffic on the CH it initiates the attack on CH1 and makes it malicious CH.
- Malicious CH1 send malicious aggregated data to other CHs and will try to make them malicious. This way it will make other CHs malicious by sending malicious data towards them. Therefore, whatever data will reach to the BS will be the malicious and leads to wrong decision at BS.
- The malicious CHs can also send noise packet inside and outside the cluster. The aggregated data send outside the cluster and noise packet from malicious CH may collide, that leads to inter-cluster collision. The noise packet coming inside the cluster collides with normal data send by normal node and leads to intra-cluster collision.

3.4.3 Performance Impact of Intelligent CH Jamming Attack

The simulation uses same simulation parameters as shown in Table 3.2 which was used in previous set of simulation of jamming attacks. The clustering algorithm used for formation of cluster is LEACH [12].

Table 3.2: Simulation Parameters

Parameter Name	**Setting Used**
Network Interface type	Wireless Physical:802.15.4
Radio Propagation Model	Two-Ray Ground
Antenna	Omni-directional antenna
Channel Type	Wireless Channel
Link Layer	Link Layer (LL)
Interface Queue	Priority Queue
Buffer size of IFq	50
MAC	802.15.4
Routing Protocol	Ad-hoc routing
Energy Model	EnergyModel
Initial Energy (initialEnergy_)	100J
Idle Power (idlePower_)	31mW
Receiving Power (rxPower_)	35mW
Transmission Power (txPower_)	31mW
Sleep Power (sleepPower_)	15μW
Number of nodes	100
Node Placement	Random

Figure 3.16, 3.17, and 3.18 shows the energy consumption, delay, and throughput respectively due to reactive jamming attack in cluster based network and intelligent CH jamming attack. The result of simulation shows that the energy consumption, delay, and reduction in throughput due to the intelligent CH jamming attack are more than reactive jamming attack. The main reason of reduction in performance in intelligent CH jamming

attack is its intelligent behavior. It can make the differentiation of CH and normal node, and initiate its attack on CHs which jam the inter- and intra- cluster traffic and increase the total energy consumption, delay and reduce the throughput of the network.

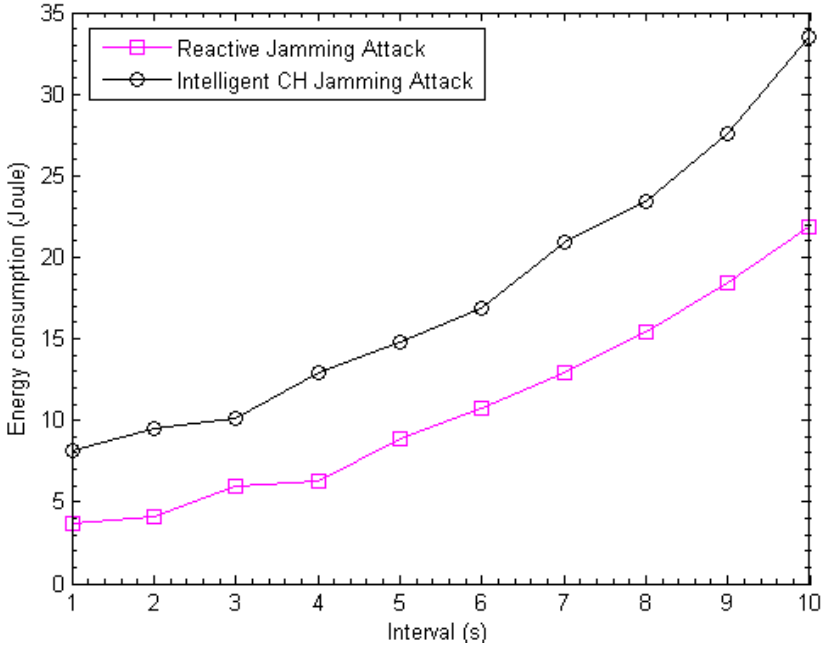

Figure 3.16: Comparative Energy consumption evaluation of reactive jamming attack with the proposed Intelligent CH jamming attack by varying the traffic interval

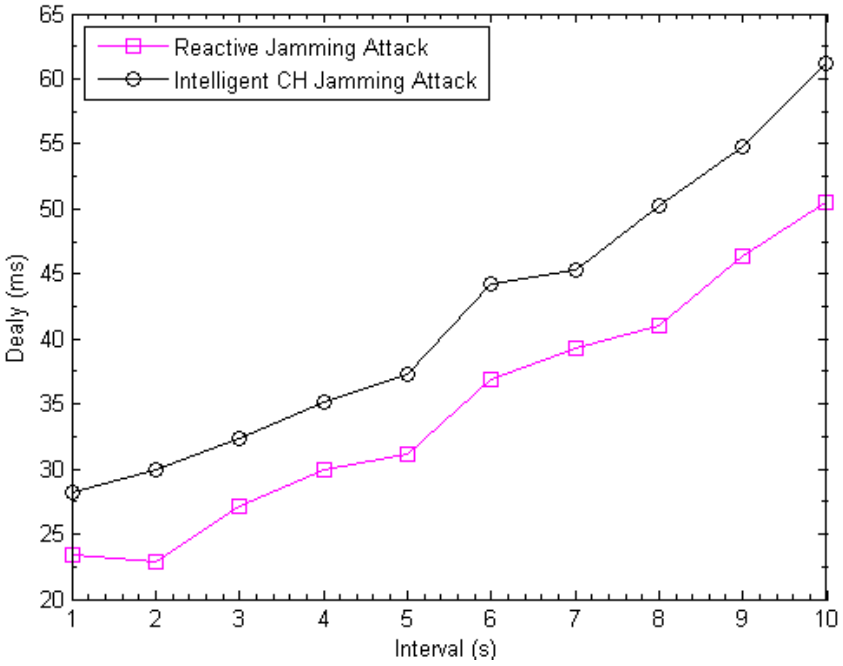

Figure 3.17: Comparative Delay evaluation of reactive jamming attack with the proposed Intelligent CH jamming attack by varying the traffic interval

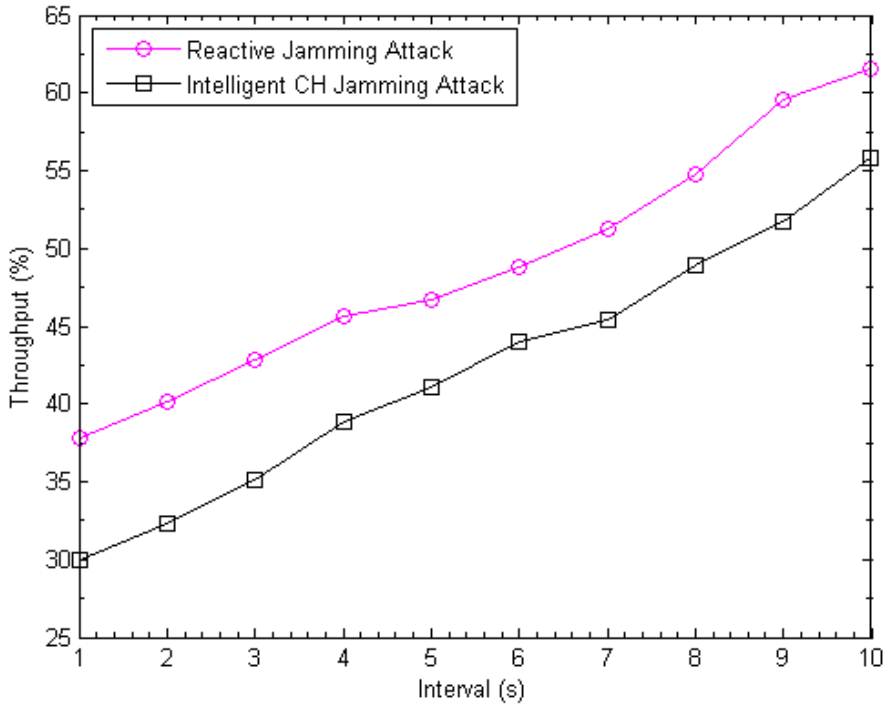

Figure 3.18: Comparative Throughput evaluation of reactive jamming attack with the proposed Intelligent CH jamming attack by varying the traffic interval

3.5 Requirements to Design Efficient Defense Mechanism against Jamming

Jamming attack can be deployed in system in many different ways and they are increasing as the WSN is getting more advanced. Therefore, to save the WSN from jamming, the defence mechanism should be developed by considering following requirements,

- Cross layer features like retransmitted RTS or DATA, failure of carrier sense, network allocator vector (NAV), etc. should be considered for detecting the attack efficiently because whenever jamming is deployed it changes the values of physical and MAC layer features.
- Nowadays most of the WSN deployments are made using cluster-based networks for improving energy efficiency and scalability. Therefore, it is necessary to develop defense mechanism by considering cluster-based networks.
- Use of threshold-based and game theoretic approach for developing efficient defense mechanism instead of traditional proactive and reactive development strategies.

3.6 Conclusions

The modelling of different jamming attack on WSN provides the functional view of sequence of activities executed during accomplishment of the jamming attack. The understanding of the activities will be useful tool to design efficient countermeasures for jamming attack. The experimental analysis of jamming attacks shows that reactive jamming is more difficult to detect than other attack because of its intelligent behavior. The behavioral modelling and analysis of jamming attack is the useful tool to understand the behavior of jamming attack and to develop the efficient defense strategy for WSN. The chapter gives the new possibility of attack in cluster-based WSN i.e. intelligent CH jamming attack and shows that this attack jam the inter- and intra- cluster traffic which is more performance intensive than jamming because of reactive jammer. The understanding of modelling of attacks and its evaluation gives the guidelines and requirements to design the efficient jamming countermeasure.

3.7 References

[1] Jennifer Yick, Biswanath Mukherjee and DipakGhosal, "Wireless Sensor Networks: A survey", Elsevier Computer Networks, Vol. 52, Issue No. 12, pp. 2292–2330, 2008.

[2] A. Mpitziopoulos, D. Gavalas, C. Konstantopoulos and G. Pantziou, "A survey on jamming attacks and countermeasures in WSNs", IEEE Communications Surveys & Tutorials, Vol. 11, Issue No.4, pp.42-56, 2009.

[3] A. R. Mahmood, H. H. Aly and M. N. El-Derini, "Defending against energy efficient link layer jamming denial of service attack in wireless sensor networks", IEEE AICCSA 27-30 December, Sharm El-Sheikh, Egypt, pp. 38-45, 2011.

[4] D. R. Raymond and S. F. Midkiff, "Denial-of-Service in Wireless Sensor Networks: Attacks and Defences", IEEE Journal on Pervasive Computing, Vol.7, Issue No.1, pp.74-81, 2008.

[5] T. Peder, "UML Bible", John Wiley & Sons, 2003.

[6] Wenyuan Xu, Ke Ma, Trappe W. andYanyong Zhang , "Jamming sensor networks: attack and defense strategies", IEEE Journal on Network, Vol.20, Issue No.3, pp. 41-47, 2006.

[7] Sachin Babar, Parikshit Mahalle, Antonietta Stango, Neeli Prasad and Ramjee Prasad, "Proposed Security Model and Threat Taxonomy for the Internet of Things (IoT)", Springer CNSA, 23- 25 July, Chennai, India, pp. 420-429, 2010.

[8] Pranav M. Pawar, Rasmus H. Nielsen, Neeli R. Prasad, Shingo Ohmori and Ramjee Prasad, "Behavioural Modelling of WSN MAC Layer Security Attacks: A Sequential UML Approch", River Publisher's Journal of Cyber Security and Mobility, Vol. 1, Issue No. 1, pp. 65-82, 2012.

[9] Derek J Corbett, Antonio G Ruzzelli, David Everitt, Gregory O'hare, "A Procedure for Benchmarking MAC Protocols used in Wireless Sensor Networks Technical Report 593", University of Sydeney, August 2006, pp. 1-28, 2006.

[10] Luigi Atzoria, Antonio Ierab, Giacomo Morabitoc, "The Internet of Things: A survey", Computer Network, Volume 54, Issue No. 15, pp. 2787–2805, 2010.

[11] Ammeer Ahmed Abbasi, Mohamed Younis, "A survey on clustering algorithms for wireless sensor network", Elsevier Computer Communication. Vol. 30, Issue No. 14-15, pp. 2826-2841, 2007.

[12] W.B. Heinzelman, A.P. Chandrakasan, H. Balakrishnan, "Application specific protocol architecture for wireless microsensor networks", IEEE Transactions on Wireless Networking, Vol. 1, Issue 4, pp. 660-670, 2002.

4

Defence Mechanism against Jamming Attack

This chapter discusses the classification of jamming countermeasures, comparison of available countermeasures and derives the open issues to develop efficient countermeasures. Threshold-based Jamming Countermeasure (TJC) with assumption made; working mechanism of algorithm and comparative simulation with result is presented. This chapter also overviews the game theory for WSN security and proposes the game formulation for jamming attack. It also proposes the jamming detection mechanism based on game theory concept. The proposed game theory-based simulation is compared with state of art solutions. The counter measure is developed for the proposed intelligent cluster-based jamming attack and it is compared with existing solutions.

4.1 Introduction

WSN is a resource constraint network, whose performance is mainly dependent on, how efficiently the resources are used [1]. Its resource constraint nature makes it more prone to different security attacks at all WSN layer. The WSN is largely affected by the jamming attack, which happens at physical and MAC layer. Jamming attack reduces the performance of constraint-based network by increasing the unnecessary use of resources. Therefore, it is necessary to save the WSN from jamming attack. The different kinds of jamming are constant jamming, deceptive jamming, random jamming, and reactive jamming [2-4]. The chapter aims to find out the individual and combine solution for different jamming attack, if all kinds of jamming exist in a network.

The chapter analyzes the different countermeasures on jamming attack. The literature survey shows that most of the solutions on jamming attack are hardware based which are quite expensive to implement and modify. The survey suggests that software based algorithm, is quite efficient and cost effective way, to stop the invasion of jamming attack. The researcher on jamming attack security did a major work for detecting the jamming attack and to reduce the effect of it on QoS of WSN by using some defensive strategies [5]. The defensive strategies can be useful to develop the efficient security model for Internet of Things (IoT) [6].

The chapter first proposes the efficient defense mechanism against jamming attack by understanding the behavior of attacks and different available countermeasures. The chapter proposes the new countermeasure against reactive jamming i.e. TJC. The TJC algorithm allows the attack into the network and starts its defensive mechanism once it detects the assaults in a network. It uses threshold based mechanism to detect the attack and to cure it. Here, every node maintains some send threshold value and it compares current transmission with threshold periodically. If it goes beyond that threshold, it understands that an attack has happened and then it applies defensive mechanism. It first detects the jamming node, then informs all neighbouring node about jammer node and change all paths coming from jammed node i.e. it will put the jammer node out of network. The chapter also simulates the TJC algorithm using Network Simulator (NS) – 2 by considering realistic conditions. The simulation results show that TJC perform in better manner in existence of reactive jamming attack. It demonstrates good performance of TJC by varying traffic interval and number of malicious nodes in network. The major advantage of TJC is that its defensive mechanism supports with increased number of jamming nodes in a network.

The second objective of the chapter is to form jamming model to understand the different jamming behavior in better way. The chapter uses the game theory for accomplishing the purpose. Game theory helps to understand the uncertainty and interdependencies in jamming attack [7]. The jamming model made considers the different player such as: constant jammer, deceptive jammer, random jammer, reactive jammer, and the monitor node. The monitor node considers two strategies continuous monitoring and periodic monitoring. Each kind of jammer behaves in different ways in different monitoring mechanism. The other important objective of chapter concern with game theory is to find the Nash equilibrium condition for

players and to propose the efficient detection mechanism against all kind of jamming. A Nash equilibrium is a set of actions of the players such that, any other action chosen by a player does not result in more favourable utility for the players. Here, Nash equilibrium is form for jamming game, where none of the player has independent motivation to change the strategy. The proposed detection mechanism uses clustering of cross layer features for efficient detection of jamming. The approach helps to easily detect the normal and abnormal behavior in game, and to inform the network to take the particular action against jamming attack. The simulation result shows that the detection mechanism has better performance (energy consumption by 25-30%, delay, and throughput by 10-15%) in different realistic situations, as compared with existing optimal strategy solution.

Chapter 3 proposed the new possibility of jamming attack in cluster based network i.e. intelligent CH jamming attack. This chapter derives the efficient defense mechanism against intelligent CH jamming attack by understanding the behavior of attacks and different available countermeasures for jamming attack. The chapter proposes the new threshold based-countermeasure against intelligent CH jamming attack. It allows the attack into the network and starts its defensive mechanism once it detects the assaults on a network. It uses threshold based mechanism to detect the attack and to cure it. Here, every node maintains some send threshold value and it compares current transmission with threshold periodically. If it goes beyond that threshold it understands that attack has happened and then it applies defensive mechanism. The mechanism maintains the threshold values at two different level one at CH level and another at base station (BS) level. It first detects the jamming node inside or outside the cluster, then informs all neighbouring node about jammer node and change all paths coming from jammed node i.e. it will put the jammer node out of clustered network. The simulation results show that proposed algorithm performs in a better manner in existence of intelligent CH jamming attack. It demonstrates good performance of algorithm by varying traffic interval and number of malicious nodes in network. The work is also verified with more realistic condition by considering random traffic interval with varying malicious nodes.

4.2 Related Works

The security countermeasures against jamming attack are classified [2] mainly into,

- Detection techniques
- Proactive countermeasures
- Reactive countermeasures
- Mobile agent-based countermeasures

Detection Technique: The purpose of detection technique is to instantly detect jamming attacks. The approaches of these category cannot cope up with jamming alone; they can significantly enhance jamming protection only when used in conjunction with other countermeasures by providing valuable data.

Proactive Countermeasures: The role of proactive countermeasures is to make a WSN immune to jamming attacks rather than reactively respond to such incidents. Proactive countermeasures can be classified in software i.e. algorithms for the detection of jamming or encryption of transmitted packets and combined software-hardware countermeasures.

Reactive Countermeasures: The main characteristic of reactive countermeasures is that they enable reaction only upon the incident of a jamming attack, sensed by the WSN nodes. Reactive countermeasures can be further classified into software and combined software – hardware.

Mobile-agent based countermeasures: This class of anti-jamming approaches enables Mobile Agents (MAs) to enhance the survivability of WSNs. The term MA refers to an autonomous program with the ability to move from host to host and act on behalf of users towards the completion of an assigned task.

Table 4.1: Survey of Jamming Attack Countermeasures

Countermeasures	Type of technique	Mechanism	Energy efficiency	Implementation Cost
The Feasibility of Launching and Detecting Jamming Attacks in WSNs [8]	Detection Technique	It detects the jamming using signal strength or location information.	Low	Low
Radio Interference Detection Protocol (RID) [9]	Detection Technique	It uses the interference calculation method and information shared by the node.	Medium	High
Energy-Efficient Link-Layer Jamming Attacks against WSN MAC Protocols [10]	Proactive Software	These techniques are mainly embedded inside the MAC to save from jamming effect. The techniques like high duty cycle, shorter data packets, encryption of link layer packet, TDMA protocol, and transmission in randomized interval are used to save from jamming.	Medium	Very Low
Defeating Energy-Efficient Jamming [11]	Proactive Software	It used frame masking, frequency hopping, and packet fragmentation with redundant encoding.	High	Medium

Hemes II nodes [12]	Proactive hardware and software	It is special kind of node which uses hybrid FHSS-DSSS technique.	Medium	High
A Jammed-Area Mapping Service for Sensor Networks [13]	Reactive Software	It detects the jamming by mapping the jam area.	Low	Medium
Channel surfing and spatial retreat [14]	Reactive hardware and software	It uses adaptive channel surfing techniques and spatial retreat mechanism.	High	High
Wormhole-Based Anti-Jamming Techniques in Sensor Networks [15]	Reactive hardware and software	It uses mechanisms like wired pair nodes, frequency hopping pairs with uncoordinated channel hopping.	Medium	High
Jamming Attack Detection and Countermeasures in WSN Using Ant System [16]	Mobile Agent	It used ant algorithm based mobility agent method.	Low	Medium
An Algorithm for Data Fusion and Jamming Avoidance on WSNs [17]	Mobile Agent	It used data fusion mechanism to reduce the effect of jamming and trying to avoid permanently.	Low	Medium
Optimal Jamming Attacks and Network Defense Policies in Wireless Sensor Networks [18]	Proactive Software	Detect the jamming by analyzing the percentage of collision and reduce the jamming effect by reducing the collision.	Low	Medium

The survey in table 4.1 shows the different countermeasures against jamming attack. The table compares all the countermeasures according to the type of technique, mechanism used, its energy efficiency, and implementation cost. The survey gives a varying concluding remark on each kind of countermeasure.

The detection techniques are less efficient according to total energy and implementation cost. Most of the detection technique cannot cope up with jamming attack individually; they require the support of some other countermeasures to work efficiently. The next kind of proactive mechanisms are better than the detection techniques by providing immunity solution to WSN against jamming attack. The proactive countermeasures are mainly

classified into proactive software countermeasures and proactive software plus hardware countermeasures. The survey shows that proactive software countermeasure techniques are more efficient than other used techniques because they use some algorithm to defense from jamming instead of allowing the jamming. The proactive countermeasures are efficient solution for active jamming attack such as constant jamming, deceptive jamming, and random jamming. The main disadvantage of proactive hardware plus software countermeasure is requirement of hardware, which increases its implementation cost.

The reactive countermeasure technique shows good performance than proactive one in case of reactive jamming attack. Reactive countermeasure allows the jamming in a network and react immediately after the detection of jamming. They are also classified into reactive software and reactive software plus hardware countermeasures. Here, also reactive software approaches are much cost efficient and energy efficient than reactive hardware plus software countermeasures. The solution mainly concentrates on the software based reactive countermeasure against reactive jamming attack.

The last kind of jamming countermeasure is mobile agent based countermeasures. It uses mobile agent who moves host to host to detect the jamming and to do the consigned task of counter-measuring against jamming attack. The major disadvantage of this technique is its increase requirement of mobile agent in network, which effects in decreasing efficiency and increase in implementation cost and complexity.

4.3 TJC: Threshold based jamming countermeasures

4.3.1 Network and Attacker Assumptions

- Network consists of *n* sensor nodes and one base station (BS).
- All nodes are connected together via bidirectional links.
- The nodes are equipped with synchronized clock, omni-directional antenna and two-ray ground propagation model. Each node is equipped with same capabilities.
- Nodes may communicate directly using single-hop communication or it may communicate using multi-hop communication.
- The nodes are distributed randomly in a network.
- Each sensor node periodically sends a message to the BS.
- The attack can be launch on any node in the network.
- The type of jamming attack assumed is reactive jamming attack, which will be activated when the jammer detects the activity on any node in the network.
- The jammer node is equipped same like a normal sensor node but with capability to generate random jamming signal (random messages).

4.3.2 Working Mechanism of TJC

This section proposes the threshold based jamming countermeasure (TJC). The key idea of algorithm is to enhance the performance of WSN in presence of reactive jamming attack and to save the WSN from harsh effects of reactive jamming. The algorithm saves the WSN by keeping some threshold at every node. The algorithm achieved it by introducing sending threshold which describe the maximum capabilities of node to send data. The detail flow of TJC algorithm is as shown in figure 4.1.

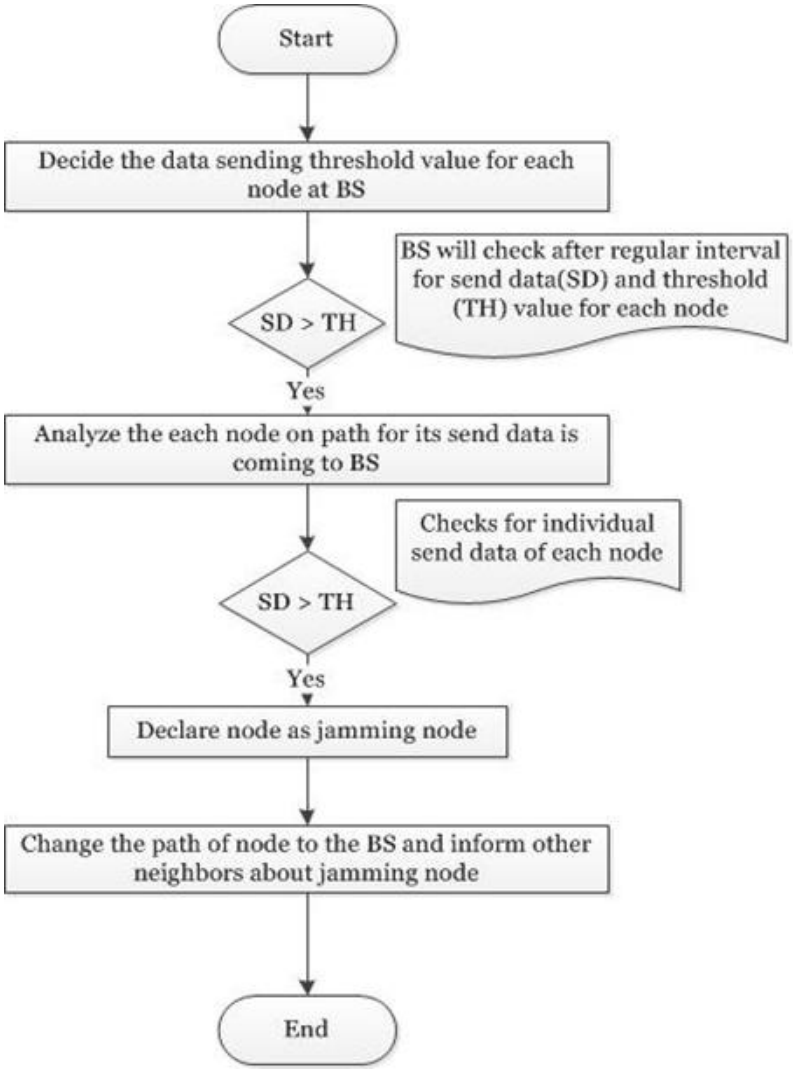

Figure 4.1: Flow of TJC algorithm

The TJC algorithm works in two phases. The first phase in the threshold based jamming countermeasure is to decide the data sending threshold value of each node. The data sending threshold value is decided at BS side. Here, the BS has capabilities to count and maintain the record of the number of times data send from each node in WSN. Each node is sending the data towards the BS after regular interval, based on amount of data received from particular node per second during normal situation; BS decides the data sending threshold value of each node. BS will maintain the number of average send coming from each node as a sending threshold value.

In the second phase, algorithm will perform the check based on sending threshold value. Here, each node maintains the three states normal state, suspicious state and attacker state. The nodes in normal state are non-attacker node, suspicious state nodes are may be an attacker, and attacker state nodes are jamming node that started to destroy the network. Initially all nodes are in normal state. The nodes are sending their information to BS either through one-hop or multi-hop way. If the BS is getting more than expected data i.e. more than consigned threshold value from the particular source node then it is changing the state of node as suspicious state. The algorithm will do the path analysis for the suspicious state node; if the suspicious source node is the direct one-hop source then detection of attacker is easy just by doing one-hop path analysis. If the suspicious node is at multi-hop distance from BS then during path analysis phase, algorithm will check for individual node on path for its number of packet transmitted per second. If the number of packets generated by the nodes is more than the average send then that node is considered to be a jammer node and algorithm will make its state as jamming state. Once the jammer node will be detected then algorithm will remove the jammer node outside the path by changing the path through jamming node and also informed to the other neighbouring node to the network that, they have jammer node in neighbour.

4.4 Simulation of TJC Algorithm and Result Discussion

4.4.1 Implementation Details

Table 4.2: Simulation and node parameters

Parameter Name	Setting Used
Network Interface type	Wireless Physical:802.15.4
Radio Propagation Model	Two-Ray Ground
Antenna	Omni-directional antenna
Channel Type	Wireless Channel
Link Layer	Link Layer (LL)
Interface Queue	Priority Queue
Buffer size of IFq	50
MAC	802.15.4
Routing Protocol	Ad-hoc routing
Energy Model	EnergyModel
Initial Energy (initialEnergy_)	100J
Idle Power (idlePower_)	31mW
Receiving Power (rxPower_)	35mW
Transmission Power (txPower_)	31mW
Sleep Power (sleepPower_)	15μW
Number of nodes	100
Node Placement	Random
Number of simulation runs	50

The implementation of all attack is performed by using discrete event simulator NS-2. The parameters set during simulations are shown in Table 4.2. The idle power, receiving power, transmission power, and sleep power are considered according to IEEE 802.15.4 radio model [19].

The simulations are performed in two different conditions. The different conditions are,

- WSN with reactive jamming attack
- WSN with reactive jamming attack with TJC countermeasure

The simulation of jamming attacks is done under following considerations,

- The simulation is performed by varying traffic interval, which is useful to measure the performance of attack and its countermeasures under various traffic conditions. The traffic interval is varied from 1s to 10s. The 1s traffic interval is consider as fast traffic and 10s traffic interval is consider as slow traffic. These simulations consider number of malicious nodes in network or nodes under attack is one.
- The second set of simulation is performed by varying number of malicious nodes in the network. The number of malicious nodes in network considered is 1,2,4,8 and 16. The traffic interval considers under this simulation is 1s which is considered to be the fast traffic in network. These set of simulations will be useful to analyse the effect of attack and its countermeasures by increasing the destructive entities in a network.
- The third set of simulation is performed by considering some realistic situations where each node is not transmitting information at same time and traffic interval consider is random traffic interval which varies in between 1s to 10s randomly.
- The last set of simulation is performed by adding random mobility to all nodes in the network. The simulation considers the random traffic interval which varies in between 1s to 10s randomly. The mobility speed consider here varies from 1km/hr to 25km/hr. This set of simulations gives the more realistic behavior of the algorithm by considering random mobility and traffic interval.

4.4.2 Result Discussions

A. Performance by varying traffic interval

Figure 4.2, 4.3, and 4.4 shows the measurement of average energy consumption, delay, and throughput by varying the traffic interval respectively. The graphs show that the proposed TJC algorithm improves the energy consumption, delay, and throughput under reactive jamming attack conditions. The algorithm detects the jamming attack by analyzing the network and reduces the effect of jamming attack by separating the jamming node from the network.

The energy consumption shown in figure 4.2 is less after applying TJC algorithm than normal reactive jamming situation. The major reason for enhancing the energy efficiency in TJC is

detection of reactive jammer and to place it out of the network. It will help to save the energy consumption that happen due to reactive jamming attack.

Figure 4.3 shows that the delay after applying TJC in a WSN is less than reactive jamming situation because TJC detects the jamming node in network and stop it by keeping it out of the network. The removal of jamming node helps to remove jam on channel, which gives the availability of channel to each node and helps in reduction of delay in case of TJC. In reactive jamming situation, which make channel busy for long time and incur a large waiting time for each node, the busy state of channel effects on to the throughput of the network, which is improved after applying TJC algorithm as shown in figure 4.4.

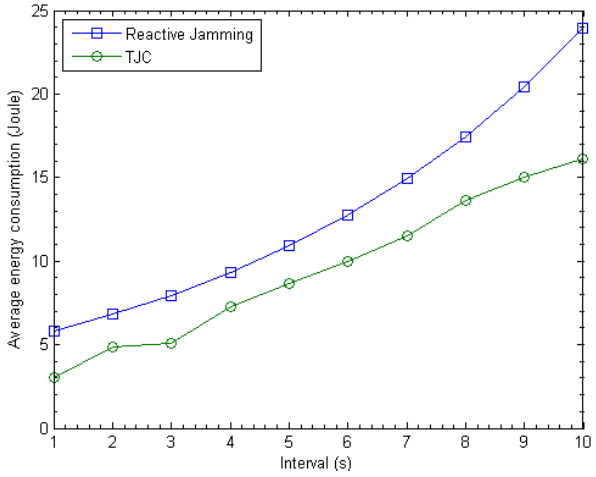

Figure 4.2: Comparative Energy Consumption Analysis of Reactive jamming and TJC under varying traffic interval

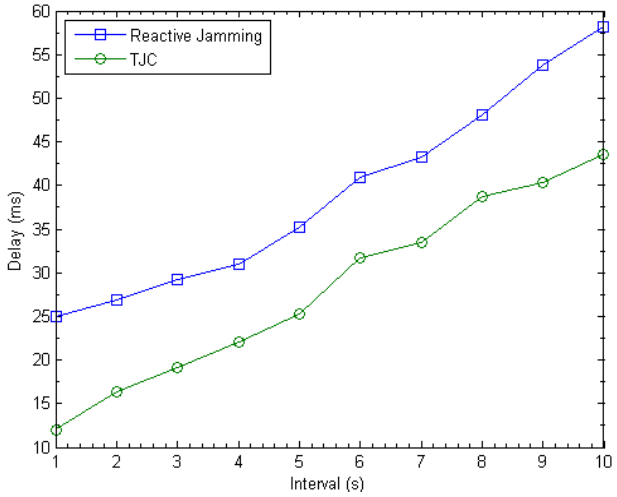

Figure 4.3: Comparative Delay Analysis of Reactive jamming and TJC under varying traffic interval

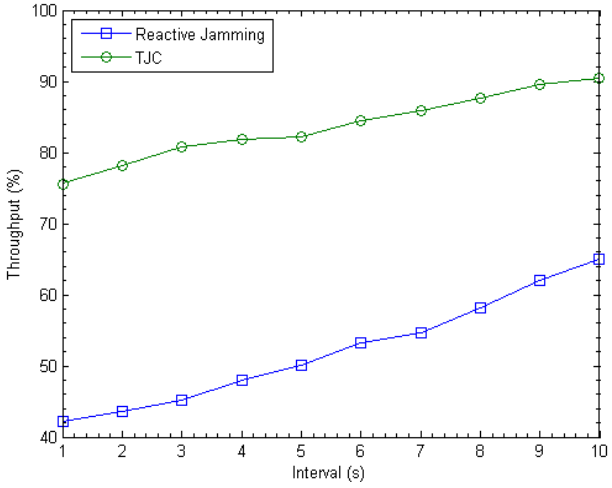

Figure 4.4: Comparative Throughput Analysis of Reactive jamming and TJC under varying traffic interval

B.Performance by varying number of malicious nodes

Figure 4.5, 4.6, and 4.7 describes the average energy consumption, delay, and throughput by changing the number of jamming nodes in the network. The number of jamming nodes in network is increasing from 1 to 16. The figures show that TJC algorithm improves performance against reactive jamming as the number of jamming nodes in network is increasing. The increasing number of jamming nodes in network gives more realistic analysis and adaptivity of TJC if amount of jamming is increasing in the network. The TJC shows efficiency by detecting the multiple jamming on the single path, which shows its perfection to cure the attack.

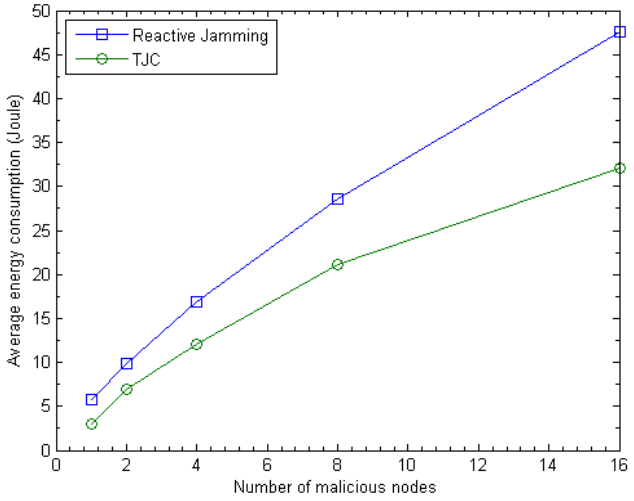

Figure 4.5: Comparative Energy consumption analysis of Reactive jamming and TJC with varying number of malicious nodes

Figure 4.5 show the average energy consumption by varying number of malicious nodes in a network, which shows TJC outperforms as number of malicious nodes is increasing. The major reason of energy saving in case of TJC is its jamming detection mechanism which helps to reduce the energy consumption due to jamming node and also helps to reduce the

energy consumption due to active state of large number of nodes in WSN without sending any data to destination. The detection mechanism of TJC also helps to reduce delay and enhance throughput as shown in figure 5.5 and 5.6. TJC reduce the delay by reducing the channel waiting time and increase throughput by giving quick channel availability to nodes in presence of reactive jamming.

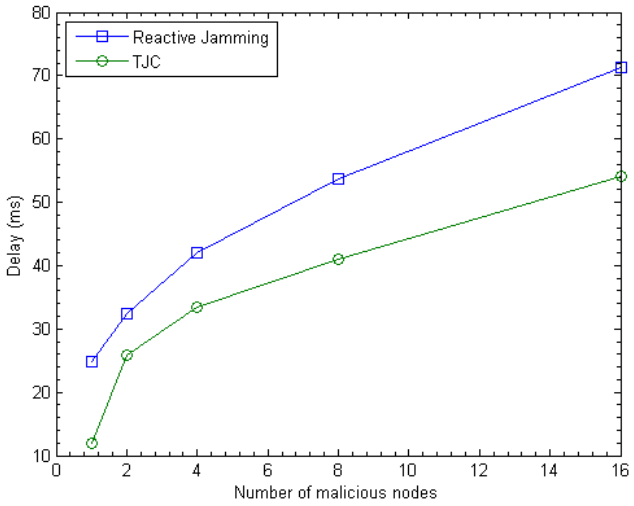

Figure 4.6: Comparative Delay analysis of Reactive jamming and TJC with varying number of malicious nodes

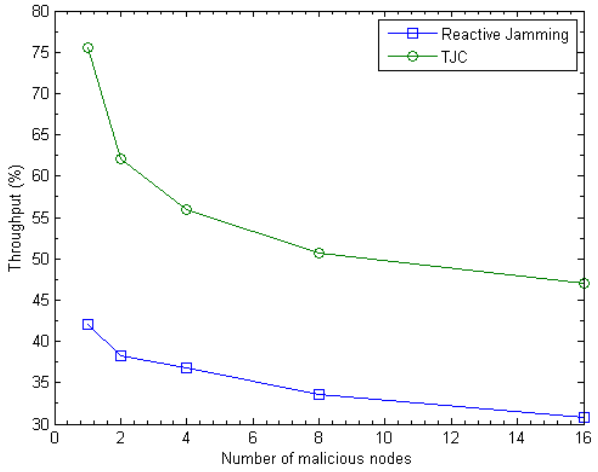

Figure 4.7: Comparative Throughput analysis of Reactive jamming and TJC with varying number of malicious nodes

C. Performance of TJC in realistic conditions

Figure 4.8, 4.9, and 4.10 shows the performance of TJC in more realistic situations such as by keeping random interval between the data packets and by transmitting data at different time instead of sending data at same time from each node. The realistic situation gives the more insight picture of performance of TJC in presence of reactive jamming attack.

Figure 4.8 show the average energy consumption of reactive jamming with and without TJC algorithm by varying number of malicious nodes. It shows that energy efficiency improves after applying TJC in realistic situations too because of technique it uses. The technique used

by TJC helps to reduce delay and enhances the throughput as shown in figure 4.9 and 4.10 respectively. The major reason of performance improvement in TJC is because of efficient channel availability than reactive jamming.

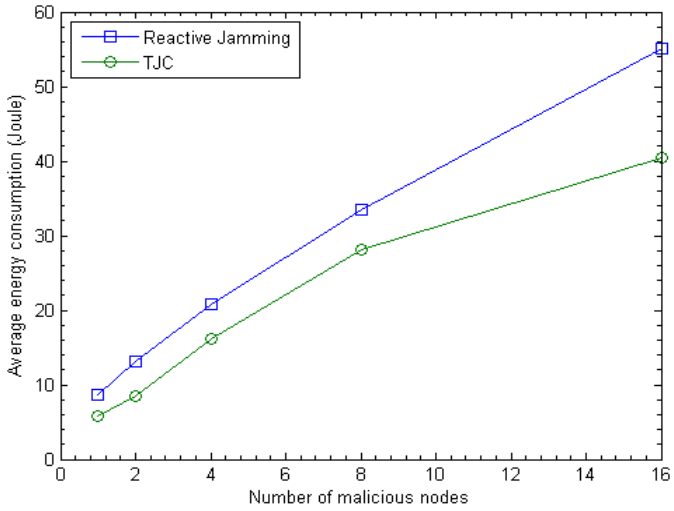

Figure 4.8: Comparative Energy consumption analysis of Reactive jamming and TJC in realistic conditions

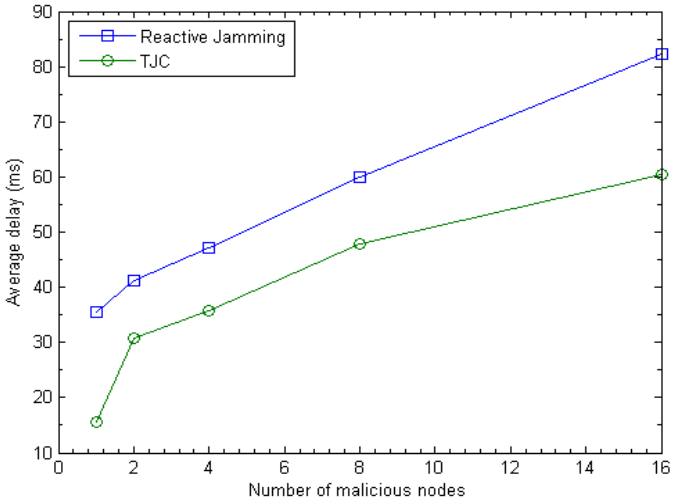

Figure 4.9: Comparative Delay analysis of Reactive jamming and TJC in realistic conditions

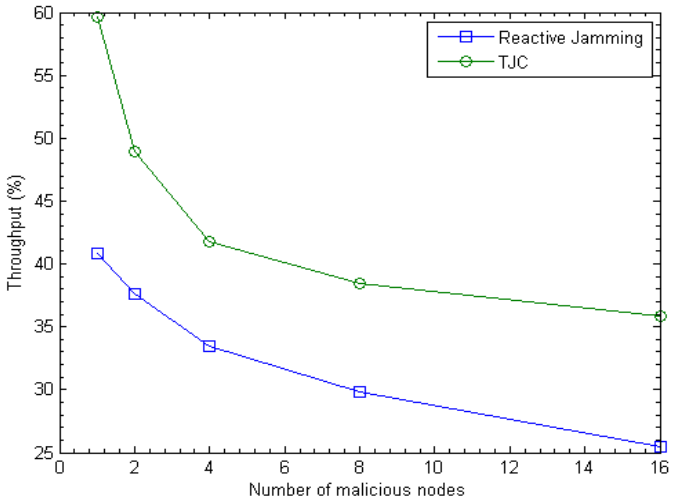

Figure 4.10: Comparative Throughput analysis of Reactive jamming and TJC in realistic conditions

D. Performance of TJC by considering mobility

Figure 4.11, 4.12, and 4.13 shows the measurement of average- energy consumption, delay, and throughput respectively by varying the number of malicious nodes in the network. The result shown gives more truthful support to the presented work because the measurement considers the random mobility among the nodes with random traffic interval. The mobility include in simulation consider the random waypoint mobility model [20]. The mobility scenario helps to check the adaptability of the concern countermeasure in presence of mobility among normal and malicious nodes.

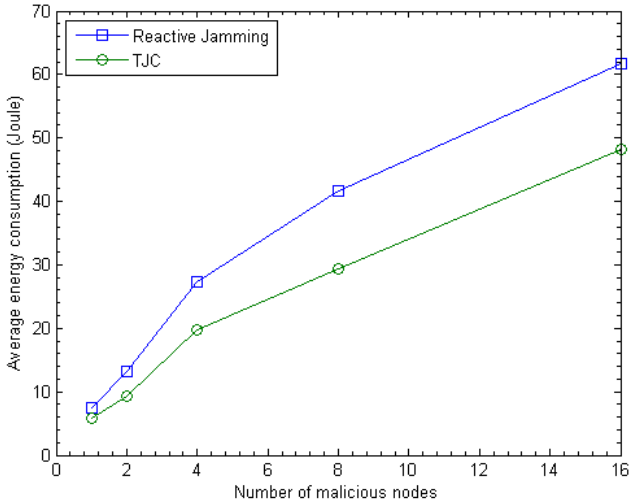

Figure 4.11: Comparative Energy consumption analysis of Reactive jamming and TJC by considering mobility

The figures shows that as the number of malicious nodes are increasing in the network average- energy consumption and delays are also increasing with it. The major reason of introducing higher energy consumption and delay is mobility. The mobility among the nodes will take more time to calculate the threshold values for each node, require more energy to scan the path and to detect the location of malicious- and neighbouring- nodes among it.

These reasons lead to increase in energy consumption and delay, they also effects on to the reduction of throughput by increasing the time of jamming detection.

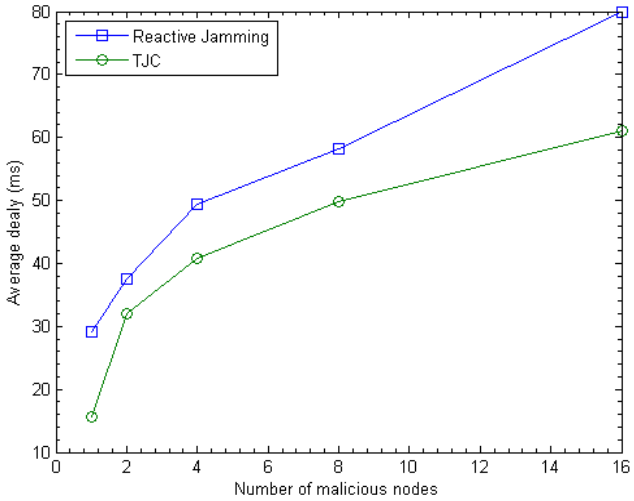

Figure 4.12: Comparative Delay analysis of Reactive jamming and TJC by considering mobility

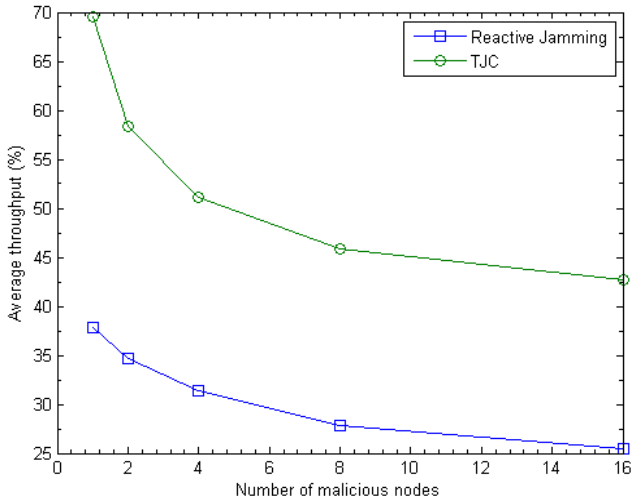

Figure 4.13: Comparative Throughput analysis of Reactive jamming and TJC by considering mobility

4.5 Game Theorotic Modelling and Defense Mechanism

4.5.1 Game Theory for Wireless Sensor Network

Game theory is a theory of decision making under conditions of uncertainty and interdependence. A game has three components: a set of players, a set of possible actions for each player, and a set of strategies. A player's strategy is a complete plan of actions to be taken when the game is actually played. Players can act selfishly to maximize their gains and hence a distributed strategy for players can provide an optimized solution to the game. In any

game, utility represents the motivation of players. A utility function, describing player's preferences for a given player assigns a number for every possible outcome of the game with the property that a higher number implies that the outcome is more preferred. The higher the number of participating nodes, the higher will be the utility [21].

A Nash equilibrium is a set of actions of the players such that, any other action chosen by a player does not result in more favourable utility for the players. The games can be classified into non-cooperative games and cooperative games. In non-cooperative games, every node act selfishly, to minimize their individual utility in a distributed decision-making environment. This is in contrast to cooperative games where nodes agree on pre-mediated strategies to maximize their payoffs [21].

In WSNs involving non-cooperative energy-efficiency games, nodes can act selfishly to conserve their power by refusing to participate as relays in multi-hop networks. In doing so, a node conserves its power; however the nodes involved in transmission and reception of the message have already used a fraction of their power and decreased their lifetime. The utility function for the nodes is the savings in battery power achieved by not cooperating in packet forwarding of other nodes. Another utility function [22] is the mapping of number of sensor nodes participating in a sensory computation to a number. Such selfish nodes can be encouraged to participate in communication by offering incentives. Incentives for the case of wireless sensor networks could be tokens, in the form of reputation or monetary benefits.

4.5.2 Game Theory for WSN Security

In security-oriented games malicious nodes within the network might launch an active attack on other nodes in the WSN, where the objective of the malicious nodes is to disrupt network operation without consideration for their own lifetime. Another category of attacks are passive attacks, where malicious nodes prevent broadcast messages and other service-availability related messages from reaching other nodes in the network [23]. Game theory offers ways to formulate problems posed by selfish and/or malicious nodes; it can serve as a favourable tool for analysis of WSNs, wherein, optimizing energy consumption in various node activities and enabling secure network operation can be modelled as games with nodes as the players.

Game theory offers models to capture the interaction between players, in this case, nodes, by modelling the players as components of social networks, where players can act in ways that would maximize their own utility, which does not always lead to favourable outcomes for the game. While game theory still lets players choose the best available action, it provides a situation where other player's utilities are also maximized. Possible techniques to relate game theory to jamming attacks so as to find out a secure defensive mechanism for jamming attacks:

- Pruning Technique
- Nash equilibrium technique
- Bayesian theoretic technique
- Repetitive Theoretic technique

Table 4.3: Various securities related game theoretic approaches [7, 23, 24]

Types of attack	Defense Strategy	Ideal Strategy	Payoff Function
External intruder: Attacks most vulnerable node in the network	IDS protects clusters of nodes from the intruder	IDS protects the same cluster which the intruder attacks	Function of utility, cost of defending/ protecting a cluster
External intruder: Injects malicious packet in the network	Service provider tries to detect malicious packets by sampling network flows at various links	Sampling strategy should be greater than the maximum flow of packets	Function of the probability of detecting a malicious packet
External attacker: Causes nodes to turn malicious by causing them to prevent broadcast messages from reaching other nodes	A certain subset of nodes, unknown to attackers sends acknowledgement to the base station for the broadcast messages	Detect attacked nodes so that attacker payoff goes to zero	Attacker payoff is proportional to the number of nodes deprived of the broadcast messages
Internal: Malicious nodes do not forward incoming packets	Introduce reputation ratings for collaboration between nodes	Catch nodes in the process of being malicious, i.e. while dropping packets	Function of a discount factor times the previous payoff
Internal: Malicious nodes in mobile WSNs do not forward incoming packets	Maintain good cooperation, reputation and quality of security ratings at each node	Nodes cooperate only if there has been a good history of cooperation, reputation and quality of security, otherwise they defect	Function of the distance between nodes, number of packets forwarded and received, quality of service of traffic as a % of exposed traffic when security is compromised

4.5.3 Game role definition in different jamming attacks

This section explains the game role definition of different jamming attacks. This game role definition will be helpful to define the detail game model for each attack. Table 4.4, 4.5, 4.6, 4.7 shows the game role definitions for constant jamming, deceptive jamming, random jamming and reactive jamming respectively.

Table 4.4: Game role definition of constant jamming

Player	**Wireless Node**	**Constant Jammer**
Strategy	Sense the channel and send data packet	Send random data packet after some regular interval without following communication rules.
Reward	Successful packet delivery with less collision in network. No jamming in network.	Introducing the collision in network, increasing the energy consumption, reducing the throughput of network
Cost	Energy required for sensing the channel and sending the packet. Required extra energy for retransmission if packet is loss or collide because of jamming.	Consume energy to create noise packet in regular interval.

Table 4.5: Game role definition of deceptive jamming

Player	**Wireless Node**	**Deceptive Jammer**
Strategy	Sense the channel and send data packet	Send regular packets continuously without checking availability of channel.
Reward	Successful packet delivery with less collision in network. No jamming in network.	Jam the network by making channel busy for long time, place most of the node in network in receive mode
Cost	Energy required for sensing the channel and sending the packet. Required extra energy for retransmission if packet is loss or collide because of jamming.	Consumes high amount of energy for producing packets continuously.

Table 4.6: Game role definition of random jamming

Player	**Wireless Node**	**Random Jammer**
Strategy	Sense the channel and send data packet	Send regular packets continuously without checking availability of channel or send random data packet after some regular interval without following communication rules. Goes to sleep mode to save the energy.
Reward	Successful packet delivery with less collision in network. No jamming in network.	Place most of the node in network in receive mode, Introducing the collision in network, increasing the energy consumption, reducing the throughput of network
Cost	Energy required for sensing the channel and sending the packet. Required extra energy for retransmission if packet is loss or collide because of jamming.	Consumes energy intelligently by placing node in sleep mode.

Table 4.7: Game role definition of reactive jamming

Player	**Wireless Node**	**Reactive Jammer**
Strategy	Sense the channel and send data packet	Generate noise packet only when sense the activity on channel otherwise put himself in quiet state.
Reward	Successful packet delivery with less collision in network. No jamming in network.	Introduce collision and increase the energy consumption in network.
Cost	Energy required for sensing the channel and sending the packet. Required extra energy for retransmission if packet is loss or collide because of jamming.	Consumes energy only when transmitting packets.

4.5.4 Jamming Game Formulation

Jamming can be formulated as a game between two players - the jammer and the communicator (transmitter-receiver pair) with different objectives. Jammers are players who prevent and deny wireless channel access to regular users by jamming their communication. Communicator nodes are players whose objective is to utilize the wireless channel effectively to increase their overall throughput. Here the game can also be model as the game between jammer and monitor node where the monitor nodes are players responsible for detecting the jamming attack.

The jamming attack and detection is model as a two-player, non-cooperative, and zero sum game. The player set as J = {J1, J2}, where J1 is the monitor node and J2 is the jammer. The nodes can choose to continuously monitor (Mc) the wireless channel or perform periodic monitoring (Mp) for a predefined time interval. The tradeoffs in the proposed jamming game are: continuous monitoring can detect jamming but results in high energy consumption; periodic monitoring consumes less energy, but with the potential risk of missing the attack.

The jammer can have multiple attack strategies. The game adopts following jamming strategies constant jammer, deceptive jammer, random jammer and reactive jammer.

Let us denote CJ, DJ, RJ and ReJ represent constant jamming, deceptive jamming, random jamming and reactive jamming actions. The action set for the monitoring node consists of two strategies with different monitoring durations (Mc, Mp), while jammer strategies are CJ, DJ, RJ and ReJ. Let us denote the strategy set as S = S1 × S2, where S1= {Mc, Mp} for Player 1 and S2= {CJ, DJ, RJ, ReJ} for Player 2.

The utility function represents the objective of the player. For monitoring nodes, two possible utility functions can be considered – detection rate and false positive rate. The utility functions indicate the efficiency of the monitoring node in terms of number of attacks successfully detected and the number of falsely classified attacks. The objective of jammer node in the network is to prevent transmissions in the channel. Hence it launches a denial of service attack aimed at reducing the throughput of the network. From the jammer's point of view, this is equivalent to its attack success. The utility function for the jammer can hence be

defined as the success in attack expansion. The utility function is denoted as {U} = {U1, U2}, where U1 =detection rate and U2 = attack gain.

Consider the above notations for strategic game form,

- G_d is the gain of detecting the attack.
- t be the time for periodic monitoring.
- A_D is the attack duration.
- P_c and P_p be the cost or payoff for attack detection using continuous and periodic monitoring.
- G_a is the attacker gain for successfully launching an attack.
- P_{cj}, P_{dj} and P_{rej} are the payoffs or costs of attacking for the constant-, deceptive- and reactive- jammers.
- T_s is sleep time for jammer node.
- T_i is the interval for generating jamming packet.

Table 4.8: Strategies in game

	Continuous Monitor	Periodic Monitor
CJ	$T_i (G_a - P_{cj})$, $T_i (G_d - P_c)$	$T_i(G_a - P_{cj})$, $tT_i (G_d - P_p)$
DJ	$(G_a - P_{dj})$, $(G_d - P_c)$	$(G_a - P_{dj})$, $t(G_d - P_p)$
RJ	$T_i (G_a - P_{cj})$, $T_i (G_d - P_c)$, $(G_a - P_{dj})$, $(G_d - P_c)$	$T_i(G_a - P_{cj})$, $tT_i (G_d - P_p)$, $(G_a - P_{dj})$, $t(G_d - P_p)$
ReJ	$A_D(G_a - P_{rej})$, $(G_d - P_c)$	$A_D(tG_a - P_{rej})$, $t(A_D G_d - P_p)$

Table 4.8 shows the strategies of game for continuous monitoring and periodic monitoring. The table considers four different players in game, constant jammer (CJ), deceptive jammer (DJ), random jammer (RJ) and reactive jammer (ReJ). Each player has different strategies for continuous and periodic monitoring. The strategies are explained as follows,

- Constant Jammer: The constant jammer sends the random packets after some particular fixed interval T_i. During continuous monitoring strategies will be, it gain G_a if attack is successfully launched and for launching the attack it has to pay P_{cj} cost i.e. $G_a - P_{cj}$. Another strategy will work if attack is detected, during that for detecting the attack it has to gain G_d and for it the payoff will be P_c i.e. $G_d - P_c$. Both of these strategies will work after fixed interval T_i.
  During periodic monitoring, the first strategy will be same like the continuous monitoring. The second strategy will be, for obtaining gain G_d for detection the node has to pay P_p i.e. $G_d - P_p$ and it will happen periodically with some fixed interval tT_i.
- Deceptive Jammer: The deceptive jammer sends the packet continuously without checking for channel. During continuous monitoring DJ can gain the G_a and G_d alternatively by paying the cost P_{dj} or P_c respectively. In periodic monitoring the DJ can gain G_a by paying cost P_{dj}, while it can gain G_d by paying P_p after some period t.

- Random Jammer: The random jammer uses the combine strategies of constant jammer and deceptive jammer. The different strategies of random jammer are as shown in table 4.8.
- Reactive Jammer: The reactive jammer reacts only when it senses any event on the channel, we can say the duration in which reactive jammer reacts or attack as attack duration A_D. In continuous monitoring reactive jammer can achieve the gain G_a by paying cost P_{rej} during attack duration A_D, otherwise it gain G_d by paying cost of continuous monitoring P_c. During periodic monitoring the ReJ can gain G_a with period t by paying P_{rej} in every attack duration A_D, else it can gain G_d in every A_D by paying P_p after period t.

4.5.5 Equilibrium Conditions

This section investigates the Nash Equilibrium for the jamming game where none of the player has independent motivation to change the strategy. In jamming game each player is trying to maximize their payoff utilities. The payoff can be maximized by using mixed strategies which is probability distribution over set of strategies. Consider, m be the probability of continuous monitoring the channel and (1-m) be the probability of using periodic monitoring. For calculating the equilibrium conditions consider that if interval for jamming in case of constant and random jamming is too small, which is almost equal to the continuous jamming i.e. deceptive jamming. Therefore, j be the probability to jam channel constantly, deceptively or randomly and (1-j) be the probability to jam the channel reactively. Hence, the Nash Equilibrium condition (m*, j*) for the game will be,

$$m^* = \frac{tG_a - P_{rej}}{G_a(1-t)}$$

$$j^* = \frac{A_D G_d - P_p}{G_d(1 - A_D)}$$

Here, m* and j* are proportional to the attack cost and detection cost respectively. The equilibrium point shows that the monitoring probability using continuous strategy is reliant on the attack gain. When the channel sense the large number of events then frequency of reactive jamming is more frequent and the cost of attacking is close to constant jamming. Therefore, the best response for the monitor node is to choose continuous monitoring strategy. The equilibrium probability of jammer is proportional to monitor's detection gain. When the monitor has a high detection rate the probability of jammer using continuous monitoring decreases. Therefore the jammer's equilibrium strategy is dependent on the cost of periodic monitor. When the monitor deploys periodic monitoring frequently, the best response for the jammer is to constantly jam the channel to increase its attack success.

4.5.6 Detection Mechanism for Jamming Attack

The detection mechanism for the monitor node considers the clustering approach. The clustering is one of the efficient and suitable solutions to detect the intruder in real time detection mechanism. It takes an unsupervised learning approach and do not require the prior knowledge of concern entities and its instances.

The jamming attack can be efficiently detected by understanding the cross-layer features. Here, the clustering algorithm is used to analyze the cross-layer features. The different cross layer feature consider for the efficient detection are retransmitted RTS or DATA, failure of carrier sense and network allocator vector (NAV). These different features values decide the level of jamming in the network. During most of jamming attack the node want to do communication, get channel busy in that case failure of carries sense is important feature to be consider. In some case of jamming the values of RTS and data determines the level of contention. NAV is important indicator for the occupancy of channel. Clustering will be used to monitor the decision when attack takes place in the network.

Here, consider that there are two clusters: one is normal and another is abnormal. The object near to the normal cluster is considered as normal and the object near to abnormal cluster will be abnormal. This means that, there is use of two different set of features one in normal set of features and another is abnormal set of features.

Consider a sampling interval of pt seconds. If the monitor node is on during the sampling period, the action of attack is determined when the feature set is classified as abnormal, and no attack is determined if the feature set is classified as normal. It is evident that accurate detection of the attack is dependent on the relation between sampling period and the current monitoring strategy. The choice of pt determines the efficiency of the detection mechanism. For a monitoring duration of t seconds, clustering analysis performs better with more cross layer feature samples. If the sampling period pt<<t, the number of feature samples collected may not be sufficient for accurate detection. With a large pt value it obtain more samples, and hence a higher detection rate.

Since multiple features are observed over a period of time, it can also reduce the number of outliers significantly thus reducing false positives. On the other hand, it must ensure that pt does not exceed the monitoring duration time. Also, there is a significant correlation between attack duration and monitoring interval. If the monitor is on during the sampling interval and there is no attack, clustering utilizes the observed cross-layer features and precisely identifies the set as normal. However in case of a smart jammer initiating reactive jamming attack, if the monitor is not activated during the attack, it misses the attack. Hence, it observes that the overall detection gain is contingent on the feature sampling rate, attack duration and monitoring duration.

4.5.7 Implementation Details and Results

A. Implementation Details

The implementation is performed by using discrete event simulator NS-2 (Network Simulator-2) [25]. The parameters set during simulations are shown in Table 4.9. The idle power, receiving power, transmission power, and sleep power are considered according to IEEE 802.15.4 radio model [19].

The simulations are performed in three different conditions. The different conditions are,

- WSN without any security attack

- WSN with game theory based detection mechanism
- WSN with optimal detection strategy [18]

The implementation of game theory based detection mechanism and optimal strategy based detection is performed by considering all attack conditions.

The simulation of jamming attacks is done under following consideration,

- The simulation is performed by varying traffic interval, which is useful to measure the performance of attack and its countermeasures under various traffic conditions. The traffic interval is varied from 1s to 10s. The 1s traffic interval is consider as fast traffic and 10s traffic interval is consider as slow traffic.
- The second set of simulation is performed by varying number of malicious nodes in the network which shows the realistic performance of network when network consist of one or more than one jamming attacker. The number of malicious nodes in network considered is 1,2,4,8 and 16. The traffic interval considered under this simulation is 1s which is consider being the fast traffic in network. These set of simulations will be useful to analyze the effect of attack and its countermeasures by increasing the destructive entities in network.

Table 4.9: Simulation and node parameters

Parameter Name	**Setting Used**
Network Interface type	Wireless Physical:802.15.4
Radio Propagation Model	Two-Ray Ground
Antenna	Omni-directional antenna
Channel Type	Wireless Channel
Link Layer	Link Layer (LL)
Interface Queue	Priority Queue
Buffer size of IFq	50
MAC	802.15.4
Routing Protocol	Ad-hoc routing
Energy Model	EnergyModel
Initial Energy (initialEnergy_)	100J
Idle Power (idlePower_)	31mW
Receiving Power (rxPower_)	35mW
Transmission Power (txPower_)	31mW
Sleep Power (sleepPower_)	15μW
Number of nodes	100
Node Placement	Random

B. Results and Discussion

-Measurement by varying interval

Figure 4.14, 4.15, and 4.16 shows the comparative evaluation of no attack condition, game theory solution, and optimal detection strategy by varying the traffic interval. The comparative evaluation considers the measurement of three parameters average- energy consumption, delay, and throughput.

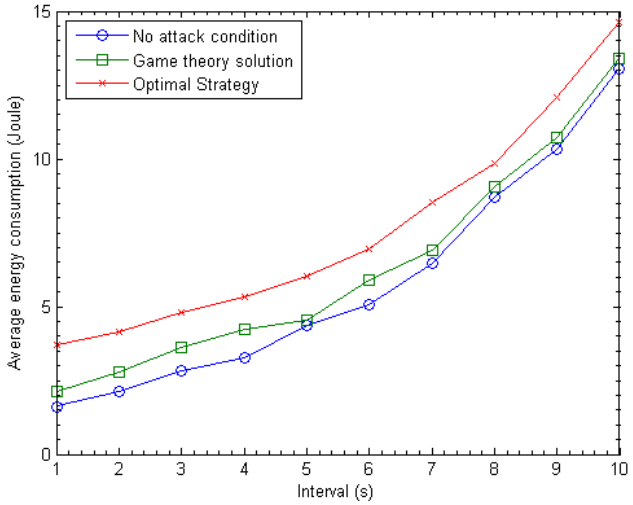

Figure 4.14:Comparative Energy Consumption Analysis of No Attack condition, Game theory solution and Optimal strategy under varying traffic interval

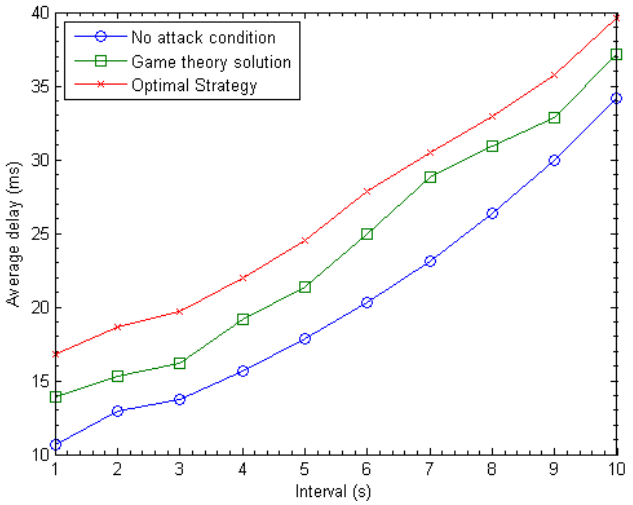

Figure 4.15: Comparative Delay Analysis of No Attack condition, Game theory solution and Optimal strategy under varying traffic interval

Figure 4.14 shows the average comparative energies of three different conditions. It gives the conclusion that the game theory solution reduces energy consumption in presence of attacks than optimal strategy based solution. The main reason for showing efficiency in terms of energy is its game theory based cross layer based detection mechanism which helps to detect jamming earlier and reduce the losses. Another benefit of game theoretic solution on optimal strategy based solution is it tries to achieve the equilibrium conditions, which helps to

maintain synergy among the involved nodes. This synergy helps to improve the energy consumption.

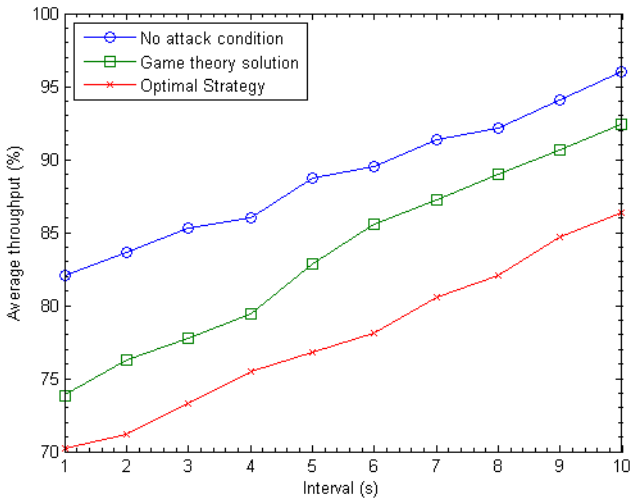

Figure 4.16: Comparative Throughput Analysis of No Attack condition, Game theory solution and Optimal strategy under varying traffic interval

Figure 4.15 and 4.16 gives the comparative average- delay and throughput respectively. The game theoretic solution shows the reduced average delay and increased throughput over another compared solution. The game theoretic solution reduces the chances of false detection by using multiple different strategies and tries to achieve equilibrium over multiple moves. The reduced chances of false detection help to improve the average- delay and throughput of the game theoretic solution. The use of multiple features to detect the attack is also useful to reduce the false detection and benefited by reducing delay and increased throughput.

-Measurement by varying number of malicious nodes

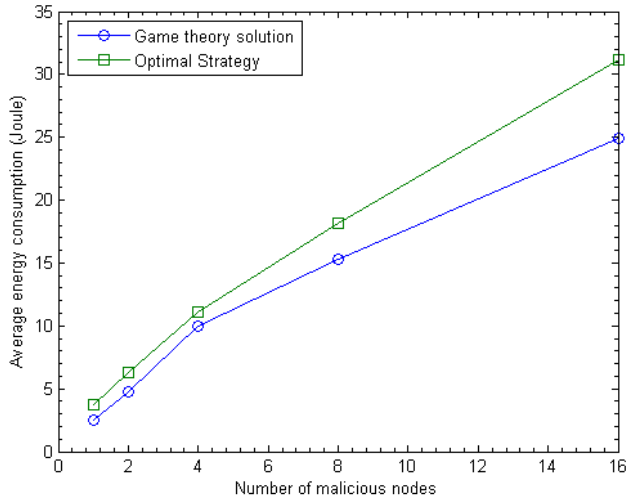

Figure 4.17: Comparative Energy consumption analysis of Game theory solution and Optimal strategy with varying number of malicious nodes

Figure 4.17, 4.18, and 4.19 shows the average- energy consumption, delay, and throughput of the game theory solution and optimal strategy by changing the number of malicious nodes in a network. The variation in number of malicious nodes shows the more accurate situation in network where network consist of more than one malicious nodes and it keeps on increasing as attack penetrate in network. The scenario is made more realistic by introducing the different jamming behavior for each node which select randomly whenever jamming is activated.

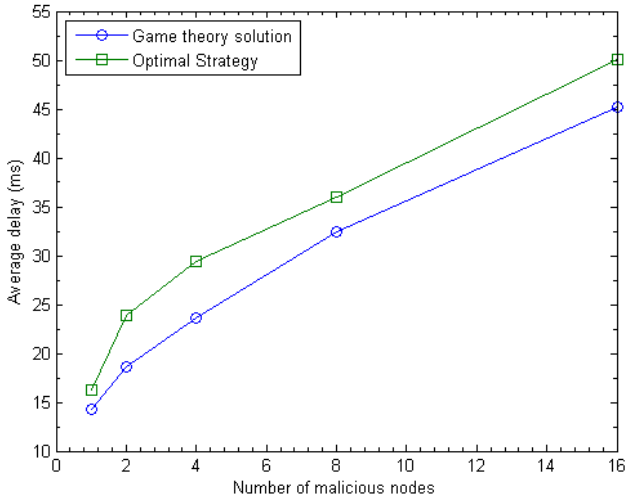

Figure 4.18:Comparative Delay analysis of Game theory solution and Optimal strategy with varying number of malicious nodes

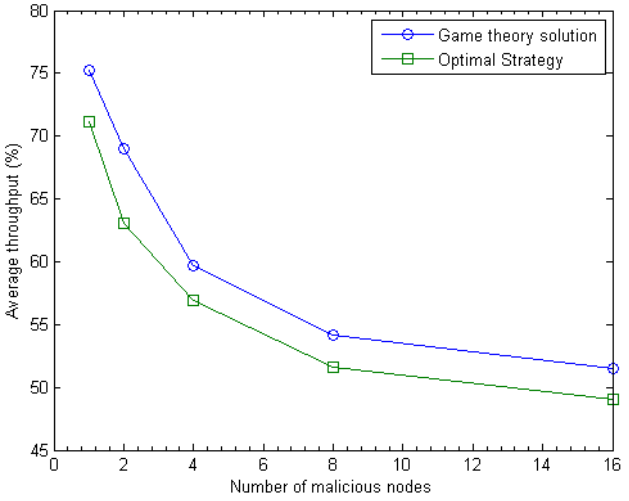

Figure 4.19: Comparative Throughput analysis of Game theory solution and Optimal strategy with varying number of malicious nodes

The graph concludes that the game theory solution shows better results in all three conditions over the optimal strategy by considering more realism in simulation. The major reasons for improving performance are use of cross layer features, reduction in possibility of false detection and cooperation among nodes to achieve equilibrium condition during game play.

The performance of optimal strategy is reduced by introducing the tradeoff between severity of attack and possibility that it can be detectable and the considered optimal model is also not good when the amount of attack keep on increasing in network.

4.6 Defense against Cluster based Jamming

4.6.1 Defense Mechanism

The countermeasure for cluster head jamming attack can be developed by extending TJC mechanism where BS maintains the data sending threshold value of each node [26]. If the BS gets more than expected data which is higher than consigned threshold value from the particular source node then it is making that node as suspicious node. The algorithm analyses the path if the source node is not the direct one-hop source.

The TJC algorithm is extended by distributing the responsibility of the BS among different CH. The threshold values are maintained at two different levels. Level 1 will be at CH side and level 2 will be at BS side.

- Level 1: The level 1 is useful to control the jamming inside the cluster i.e. it is useful for detecting the intra-cluster jamming. Here, the CH maintains the data sending threshold value for each node inside the cluster. If the CH is getting more than expected packets then it will not aggregate the information and it will perform the path analysis inside the cluster, find out the malicious jamming node in the cluster and reroute all the paths going via jamming node.
- Level 2: The level 2 is useful to control the jamming in between the CH i.e. for controlling inter-cluster jamming. Here, the BS maintains the aggregated data sending threshold value for each CH. If the BS gets more than expected data from the CHs then it will declare that path as suspicious path and do the path analysis for detecting the malicious CH in the network. If the malicious CH is being detected, then that will be declared as jamming node and rerouting will be done for transmitting information in-between the CHs and BS. The re-election of CH will be done for getting new CH in cluster.

Figure 4.20 explain the flow of proposed countermeasure for intelligent CH-jamming attack. The algorithm start with the decision of threshold values TH. It decides the threshold values-at CH for level 1 and at BS for level 2. Level 1 is for controlling intra-cluster communication and level 2 is for controlling inter-cluster communication. The CH and BS will check after regular intervals for send data (SD) and TH values for each node. If SD is greater than TH either during inter- or intra- cluster communication, then CH or BS analyze the each node on path for its SD value coming to CH or BS. The path analysis will be done by comparing individual SD of each node with TH. If any node on path having SD greater than TH, then algorithm declares that node as jamming node, either inside or outside the cluster.

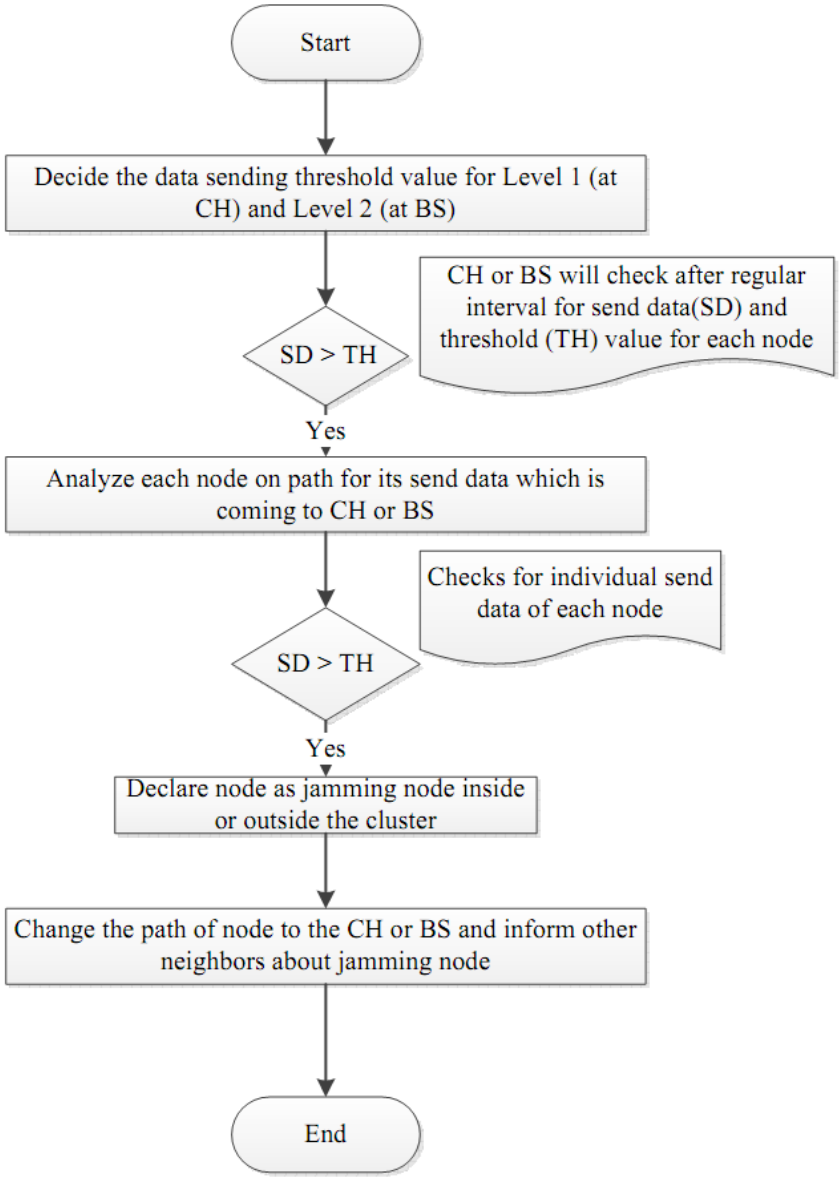

Figure 4.20: Flowchart of proposed countermeasure

4.6.2 Comparative Simulation and Discussion

4.6.2.1 Simulation Details

The implementation is performed by using discrete event simulator NS-2 (Network Simulator-2). The parameters set during simulations are shown in Table 4.10. The idle power, receiving power, transmission power, and sleep power are considered according to IEEE 802.15.4 radio model [19].

The simulations are performed in four different conditions. The different conditions are,

1. WSN with intelligent CH jamming attack
2. WSN with countermeasure for CH jamming attack
3. WSN with TJC countermeasure

4. WSN with optimal strategy based countermeasure

The simulations are performed by considering following scenarios,

- The simulation is performed by varying traffic interval, which is useful to measure the performance of attack and its countermeasures under various traffic conditions. The traffic interval is varied from 1s to 10s. The 1s traffic interval is considered as fast traffic and 10s traffic interval is considered as slow traffic.
- The second set of simulation is performed by varying number of malicious nodes. The number of malicious nodes in network considered is 1,2,4,8 and 16. The traffic interval considered under this simulation is 1s which is consider being the fast traffic in a network. These set of simulations will be useful to analyze the effect of attack and its countermeasures by increasing the destructive entities in network.
- The third set of simulation is performed by considering some realistic situation where each node is not transmitting information at same time and traffic interval considered is random traffic interval which varies in between 1s to 10s randomly.

Table 4.10: Simulation and node parameters

Parameter Name	**Setting Used**
Network Interface type	Wireless Physical:802.15.4
Radio Propagation Model	Two-Ray Ground
Antenna	Omni-directional antenna
Channel Type	Wireless Channel
Link Layer	Link Layer (LL)
Interface Queue	Priority Queue
Buffer size of IFq	50
MAC	802.15.4
Routing Protocol	Ad-hoc routing
Energy Model	EnergyModel
Initial Energy (initialEnergy_)	100J
Idle Power (idlePower_)	31mW
Receiving Power (rxPower_)	35mW
Transmission Power (txPower_)	31mW
Sleep Power (sleepPower_)	15μW
Number of nodes	100
Node Placement	Random

4.6.2.2 Results and Discussions

A. Measurement by varying interval

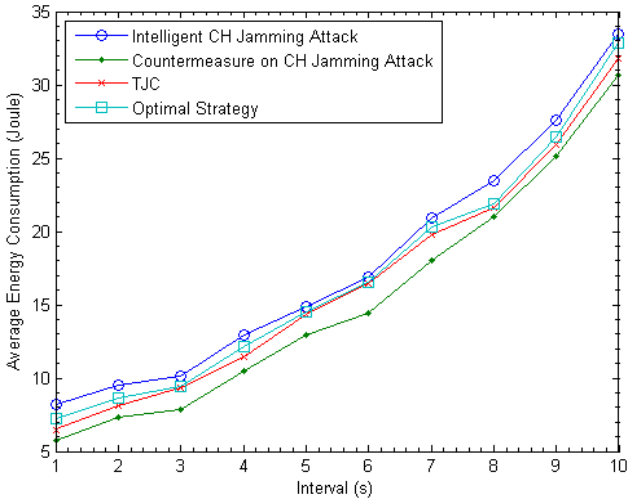

Figure 4.21: Comparative Energy Consumption Analysis of Intelligent CH jamming Attack, Countermeasure on CH Jamming Attack, TJC and Optimal strategy under varying traffic interval

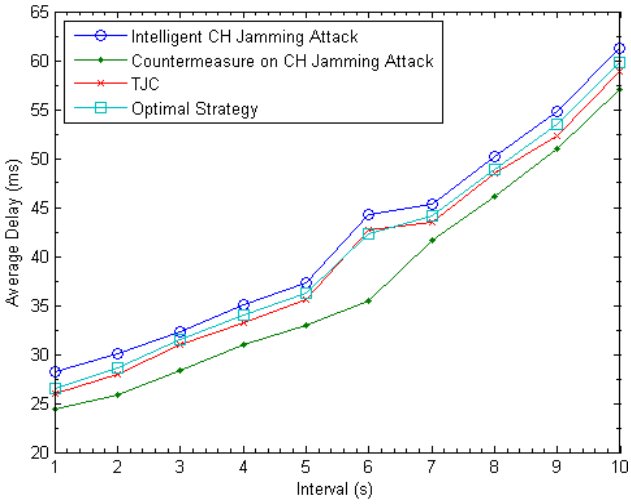

Figure 4.22: Comparative Delay Analysis of Intelligent CH jamming Attack, Countermeasure on CH Jamming Attack, TJC and Optimal strategy under varying traffic interval

Figure 4.21, 4.22, and 4.23 shows the energy consumption, delay, and throughput respectively for intelligent CH jamming attack, its countermeasure, TJC algorithm and optimal game theoretic strategy. The clustering algorithm used for formation of cluster is LEACH [27, 28]. All three results show the performance enhancement after applying the countermeasure against the intelligent CH jamming attack. The main reason for performance enhancement is that the mechanism detects inter- and intra- cluster jamming and avoid the jamming by removing the jamming node or by rerouting the network without considering the jamming node. The proposed countermeasure is also compared with TJC algorithm which is the countermeasure for reactive jamming attack; it shows lower performance than proposed one. Here, TJC is implemented to detect intelligent CH jamming attack, which shows lower

performance than proposed countermeasure. The TJC algorithm is insufficient to detect the intelligent CH jamming attack which takes place in and out of the cluster. The comparison of proposed algorithm is also performed with optimal game theoretic strategy. The optimal game theoretic strategy is insufficient to efficiently detect CH jamming attack. This technique involves large number of calculation which increase the overheads and increase the implementation cost.

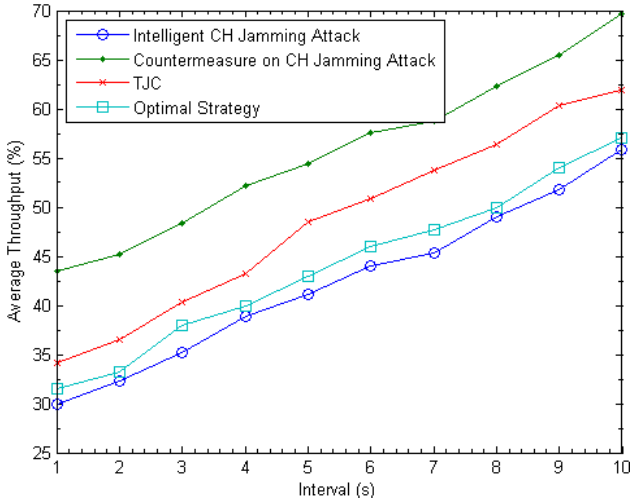

Figure 4.23 : Comparative Throughput analysis of Intelligent CH jamming Attack, Countermeasure on CH Jamming Attack, TJC and Optimal strategy under varying traffic interval

B. Measurement by varying number of malicious nodes

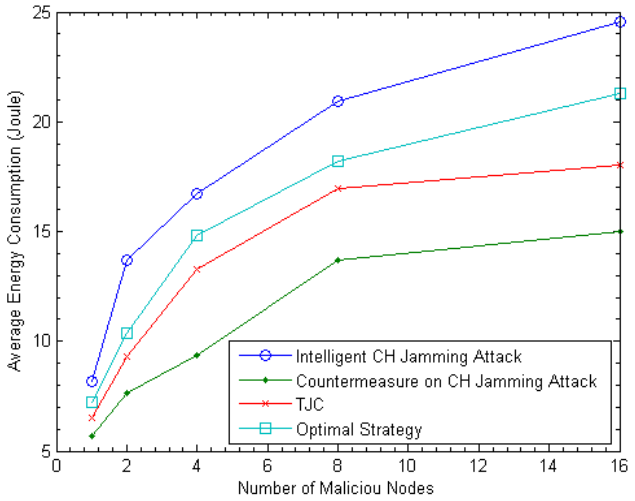

Figure 4.24:Comparative Energy consumption analysis of Intelligent CH jamming Attack, Countermeasure on CH Jamming Attack, TJC and Optimal strategy with varying number of malicious nodes

Figure 4.24, 4.25, and 4.26 describes the average energy consumption, delay, and throughput by changing the number of malicious nodes in the network. The number of malicious nodes are increasing from 1 to 16. The figures shows that proposed countermeasure on CH jamming attack is showing improvement against intelligent CH jamming attack, existing countermeasure TJC and optimal strategy. The increasing number of malicious nodes in a

network gives more realistic analysis and adaptivity of proposed countermeasure, if amount of intelligent CH jamming is increasing in the network. The proposed countermeasure shows efficiency by detecting the multiple intelligent CH jamming attack on the single path, which shows its perfection to cure the attack.

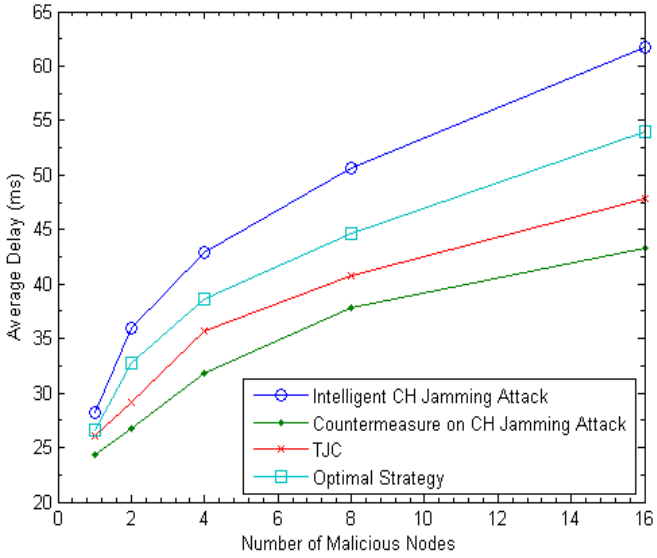

Figure 4.25:Comparative Delay analysis of Intelligent CH jamming Attack, Countermeasure on CH Jamming Attack, TJC and Optimal strategy with varying number of malicious nodes

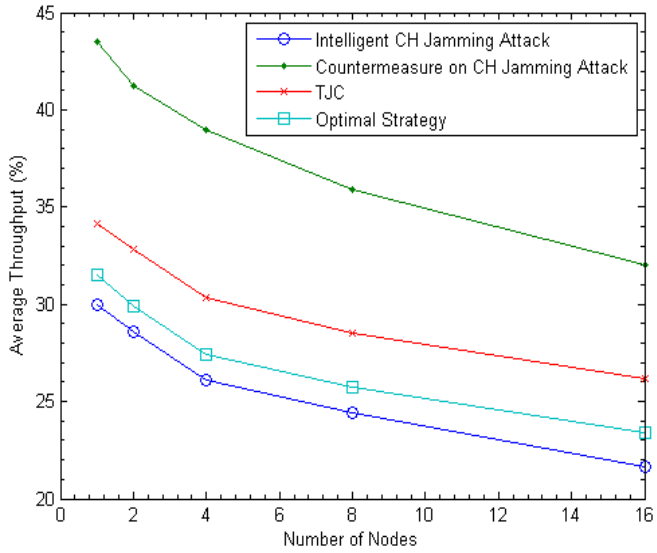

Figure 4.26:Comparative Throughput analysis of Intelligent CH jamming Attack, Countermeasure on CH Jamming Attack, TJC and Optimal strategy with varying number of malicious nodes

Figure 4.24 shows the average energy consumption by varying number of malicious nodes in a network, which shows the proposed countermeasure outperforms over other approches. The major reason of energy saving in proposed countermeasure is its intelligent CH jamming detection mechanism. It helps to reduce the energy consumption due to jamming node or CH and also helps to reduce the energy consumption due to active state of large number of nodes in WSN without sending any data to destination. The detection mechanism of proposed

countermeasure also helps to reduce delay and enhance throughput as shown in figure 4.25 and 4.26. The proposed countermeasure reduce the delay by reducing the channel waiting time and increase throughput by giving quick channel availability to nodes in presence of intelligent CH jamming attack.

C. Measurement using random traffic interval

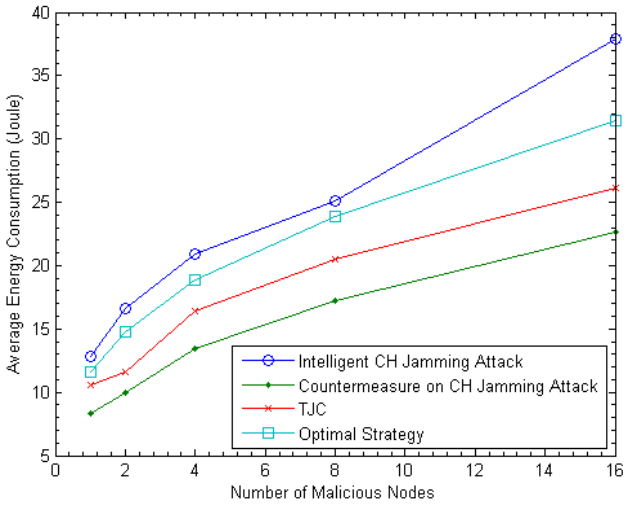

Figure 4.27:Comparative Energy consumption analysis of Intelligent CH jamming Attack, Countermeasure on CH Jamming Attack, TJC and Optimal strategy in realistic conditions

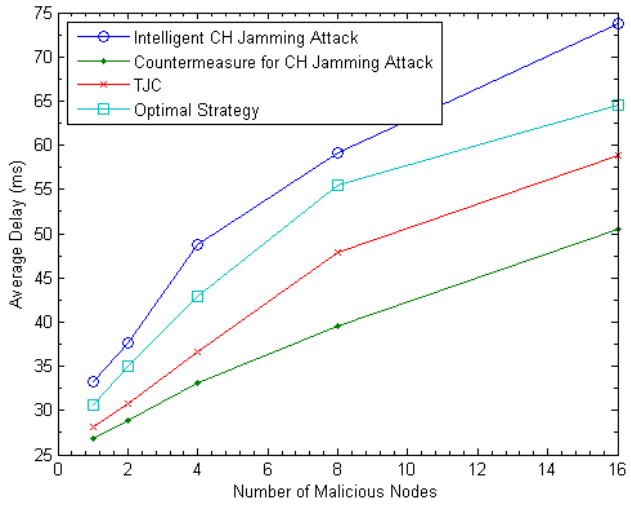

Figure 4.278:Comparative Delay analysis of Intelligent CH jamming Attack, Countermeasure on CH Jamming Attack, TJC and Optimal strategy in realistic conditions

Figure 4.27, 4.28 and 4.29 shows the performance of proposed countermeasure in more realistic situations such as by keeping random interval between the data packets and by transmitting information or data at different time instead of sending data at same time from each node. The realistic situation gives the more insight picture of performance of proposed countermeasure in presence of intelligent CH jamming attack.

Figure 4.27 shows the average energy consumption of intelligent CH jamming attack with proposed countermeasure, TJC, optimal strategy countermeasure and without any

countermeasure by varying number of malicious nodes. It shows that energy efficiency of proposed countermeasure improves in realistic situations too because of technique used. The technique used by proposed countermeasure helps to reduce delay and enhances the throughput as shown in figure 4.28 and 4.29 respectively. The major reason for performance improvement in proposed countermeasure is because of efficient channel availability than others.

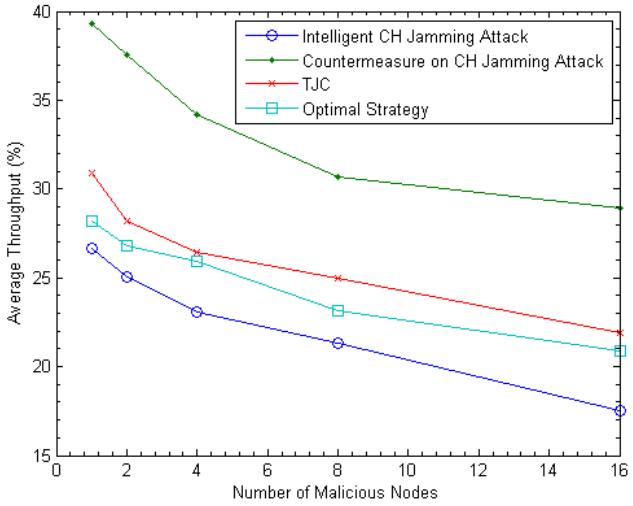

Figure 4.29: Comparative Throughput analysis of Intelligent CH jamming Attack, Countermeasure on CH Jamming Attack, TJC and Optimal strategy in realistic conditions

4.7 Conclusions

The chapter proposes the different countermeasures to save from jamming attack. The first proposed countermeasure TJC, which shows good performance against reactive jamming attack with varying traffic interval and number of malicious nodes in a network. The proposed TJC algorithm is also tested by considering more realistic conditions where each node is not transmitting in particular time interval but nodes are transmitting at different time instance. The results in different conditions show that TJC is good solution against reactive jamming attack. The simulation of algorithm by considering mobility shows TJC adaptability with changing position of nodes in the network.

The security threats because of jamming attack are increasing in large way and it is necessary to understand the conduct of different jamming attack in better manner. The second part of chapter gives the modelling of the jamming attack using game theory which explains the detailed moves in all kinds of jamming attack in continuous and periodic monitor states. The author also finds the Nash equilibrium condition and detection mechanism for jamming attack. The detection mechanism shows better performance in terms of energy consumption (25-30%), delay, and throughput (10-15%) than existing optimal game theoretic strategy.

The security threats of jamming attack are increasing and they appear in a network in different ways. Chapter 3 gives the brief idea of new jamming attack situation i.e. intelligent CH jamming attack which can takes place in cluster-based network. Chapter 4 proposes

countermeasure on intelligent CH jamming which shows good performance against proposed attack with varying traffic interval and number of malicious nodes in the network. The proposed countermeasure also shows good performance with more realistic situation such as random traffic interval with number of malicious nodes in network. The proposed countermeasure gives 15-20% improvement than state-of-art countermeasures.

4.8 References

[1] Jennifer Yick, Biswanath Mukherjee, DipakGhosal, "Wireless Sensor Networks: A survey", Elsevier Computer Networks, Vol. 52, Issue No. 12, pp. 2292–2330, 2008.

[2] Aristides Mpitziopoulos, DamianosGavalas, CharalamposKonstantopoulos, and GrammatiPantziou, "A Survey on Jamming Attacks and Countermeasures in WSNs", IEEE Communications Surveys & Tutorials, Vol. 11, Issue No. 4, pp. 42-56, 2009.

[3] Raymond D. R., Midkiff S. F., "Denial-of-Service in Wireless Sensor Networks: Attacks and Defenses", IEEE Pervasive Computing, Vol. 7, Issue No. 1, pp. 74-81, 2008.

[4] Wenyuan Xu, Ke Ma, Trappe W. and Yanyong Zhang, "Jamming sensor networks: attack and defense strategies", IEEE Journal on Network, Vol.20, Issue No.3, pp. 41-47, 2006.

[5] Wenyuan Xu, Ke Ma, Trappe W. andYanyongZhang , "Jamming sensor networks: attack and defense strategies", IEEE Journal on Network, Vol.20, Issue No.3, pp. 41-47, 2006.

[6] Sachin Babar, ParikshitMahalle, AntoniettaStango, Neeli Prasad and Ramjee Prasad, "Proposed Security Model and Threat Taxonomy for the Internet of Things (IoT)", Springer CNSA, 23- 25 July, Chennai, India, pp. 420-429, 2010.

[7] Renita Machado, SirinTekinay, "A survey of game-theoretic approaches in wireless sensor networks", Elsevier, Computer Networks, Vol 52, Issue 16, pp. 3047-3061, 2008.

[8] Wenyuan Xu, Wade Trappe, Yanyong Zhang, and Timothy Wood, "The feasibility of launching and detecting jamming attacks in wireless networks", ACM MobiHoc, 25-28 May, Urbana Champaign, IL, USA, pp. 46-57, 2005.

[9] G. Zhou, T. He, J. A. Stankovic and T. Abdelzaher , "RID: radio interference detection in wireless sensor networks", IEEE INFOCOM, 13-17 March, Miami, FL, USA, pp. 891- 901, 2005.

[10] Y. Law, L. van Hoesel, J. Doumen, P. Hartel, and P. Havinga, "Energy-Efficient Link-Layer Jamming Attacks against Wireless Sensor Network MAC Protocols", ACM Transaction on Sensor Network, Vol. 5, Issue No. 1, pp. 6.1-6.38, 2009.

[11] A. D. Wood, J. A. Stankovic and Gang Zhou, "DEEJAM: Defeating Energy-Efficient Jamming in IEEE 802.15.4-based Wireless Networks", IEEE SECON, 18-21 June, San Diego, CA, USA, pp.60-69, 2007.

[12] A. Mpitziopoulos, D. Gavalas, G. Pantziou and C. Konstantopoulos, "Defending Wireless Sensor Networks from Jamming Attacks", IEEE PIMRC, Athens, Greece, 3-7 September, pp. 1-5, 2007.

[13] A. D. Wood, J. A. Stankovic and S. H. Son, "JAM: a jammed-area mapping service for sensor networks", IEEE RTSS, 3-5 December, Cancun, Mexico, pp. 286-297, 2003.

[14] W. Xu, T. Wood, W. Trappe, and Y. Zhang., "Channel surfing and spatialretreats: defenses against wireless denial of service", ACM workshop on Wireless security, 26 September – 1 October, NY, USA, pp. 80-89, 2004.

[15] M. Cagalj, S. Capkun and J. P. Hubaux, "Wormhole-Based Antijamming Techniques in Sensor Networks", IEEE Transactions on Mobile Computing, Vol. 6, Issue No.1, pp.100-114, 2007.

[16] Rajani Muraleedharan and Lisa Osadciw, "Jamming Attack Detection and Countermeasures in Wireless Sensor Network Using Ant System", SPIE, 12 March, Orlando, FL, pp.1-5, 2006.

[17] A. Mpitziopoulos, D. Gavalas, C. Konstantopoulos and G. Pantziou, "JAID: An Algorithm for Data Fusion and Jamming Avoidance on Distributed Sensor Networks", Elsevier Journal of Pervasive and Mobile Computing, Vol. 5, Issue No. 2, pp. 135-147, 2006.

[18] Mingyan Li, Koutsopoulos I. and Poovendran R., "Optimal Jamming Attack Strategies and Network Defense Policies in Wireless Sensor Networks", IEEE Transactions on Mobile Computing, Vol. 9, Issue No.8, pp.1119-1133, 2010.

[19] Derek J Corbett, Antonio G Ruzzelli, David Everitt and Gregory O'hare, "A Procedure for Benchmarking MAC Protocols used in Wireless Sensor Networks", Technical Report 593, August, School of IT, University of Sydney, pp. 1-28, 2006.

[20] Bettstetter C., Resta G., Santi P., "The Node Distribution of the Random Waypoint Mobility Model for Wireless Ad hoc Networks" , IEEE Transaction on Mobile Computing, Vol. 2, Issue 3, 2003, 257-269.

[21] A.B. MacKenzie, L.A. DaSilva, "Game Theory for Wireless Engineers(Synthesis Lectures on Communications)", Morgan & Claypool Publishers, 2006.

[22] J. Byers, G. Nasser, "Utility-based decision-making in wireless sensor Networks", First ACM International Symposium on Mobile ad hoc networking and Computing, Poster Session, pp. 143–144, 2000.

[23] A. Agah, M. Asadi, S.K. Das, "Prevention of DoS attacks in sensor networks using repeated game theory", International Conference on Wireless Networks, pp. 1-5, 2006.

[24] IoannaKantzavelou, SokratisKatsikas, "A game-based intrusion detection mechanism to confront internal attackers", Elsevier Computers & Security, Vol 29, Issue 8, pp. 859-874, 2010.

[25] Network Simulator – 2, www.isi.edu/nsnam/ns/

[26] Sachin D. Babar, Neeli R. Prasad, Ramjee Prasad, "Activity Modelling and Countermeasures on Jamming Attack", Journal of Cyber Security and Mobility, Vol. 2, Issue no. 2, pp. 1-22, 2013.

[27] Ammeer Ahmed Abbasi, Mohamed Younis, "A survey on clustering algorithms for wireless sensor network" Elsevier Computer Communication. Vol. 30, Issue No. 14-15, pp. 2826-2841, 2007

[28] W.B. Heinzelman, A.P. Chandrakasan, H. Balakrishnan, "Application specific protocol architecture for wireless microsensor networks", IEEE Transactions on Wireless Networking, Vol. 1, Issue 4, pp. 660-670, 2002.

5

Secure Key Management

The goal of this chapter is to illustrate the related work in the area of key management and compare the different key management algorithm according to the technique used. This chapter proposes the efficient secure key management technique and discusses the system model, proposed key management scheme, and its performance evaluation. Security evaluation and performance analysis of the proposed scheme shows that overall performance of the proposed scheme improves as compared to the state of the art.

5.1 Introduction

Wireless Sensor Network (WSN) is a network of small sensor nodes, which communicate with each other using radio. Nowadays WSN has been used in many different real time and mission critical applications. The use of WSN in mission critical application produces the new requirement to the WSN application. These requirements are security and mobility. The security is important in WSN to save it from malicious attack and mobility is necessary to increase the area of network reachability. It is difficult and challenging to address these two issues together. The objective of this chapter is to address WSN security in mobile scenarios [1,2].

The WSN security is more complex and constrained as compared with traditional security mechanisms. The major research on WSN concentrated on the cryptographic solution for the security. The cryptographic solutions are mainly concentrating on to the key management issues. Different types of key management algorithms are proposed in the literature by considering different network management and sharing of key among the different nodes. Large numbers of key management algorithms are develop by considering flat network and no mobility in the network. The important contribution of this work is key management in cluster-based mobile environment. The cluster-based networks are efficient in terms of scalability and energy efficiency. These kind of network arrangement helps to improve management of keys and reduce the fast penetration of security attack in the network. The sensors have limited battery, therefore any key management algorithm for sensor should have minimum amount of computation and message transmission. The cluster-based mechanism also helps to improve it.

The chapter proposes the new key management algorithm by considering above stated challenges. The challenges are addressed by the new scheme called, Cluster-based Mobile Key Management Scheme (CMKMS). The scheme is based on below two WSN case studies or scenarios,

- In the first case, consider a cluster-based mobile sensor network. Here, cluster head (CH) is assume static and aggregating information from all other nodes in cluster. The other nodes in cluster are mobile nodes and may move from one cluster to other cluster. The work considers that CH is acting as a key manger (KM) who manages the keys of all nodes inside the cluster. The challenge consider in this scenario is whenever a node is changing a position and moving from one CH to other i.e. from his home CH (HCH) to foreign CH (FCH).
- The second case also considers a cluster-based mobile sensor network. Here, assume that CH and nodes both are mobile i.e. KM and nodes both are mobile. The proposal here is to transfer the key management responsibility to other node in cluster i.e. to make new KM or CH in network. The work will assume that whenever the CH or KM is coming near to the boundary of cluster it transfer the key management responsibilities to other CH by running CH selection algorithm.

The proposed scheme satisfy the first case by considering two different private keys for each node, one is its home key which is permanent and another is foreign key which many change when node is moving from one cluster to other. The second case is satisfied by transferring the key manager responsibilities to other node and informs other nodes about new key manager. Here, algorithm considers the two phases, first one is setup-phase, which helps to establish cluster and distribute the keys in a network. The second phase is controlling the maintenance of keys during node mobility.

The chapter simulates the proposed algorithm using Network Simulator-2 (NS-2) and compares its performance with state-of-art key management solution Energy-Efficient Distributed Deterministic Key Management for Wireless Sensor Networks (EDDK). The results are major in terms of computational overheads, energy consumption, and delay required for managing and maintaining the keys. The CMKMS algorithm outperforms over EDDK, in static and mobile scenarios.

5.2 Related Works

Key management can be defined as a set of techniques and procedures that support the establishment and maintenance of keying relationships between authorized parties. The key management technique for a secure application must minimally incorporate authenticity, confidentiality, integrity, scalability, and flexibility [1]. The different key management schemes are majorly classify as [2],

- Network-wide key
- Full pairwise
- Probabilistic
- Matrix-based
- Polynomial-based
- Combinatorial design
- Deployment knowledge

Network-wide key: The most straightforward key distribution possible is to have a single master key, which is loaded into all sensors. Such simplicity results in a high level of efficiency and flexibility, requiring minimal memory for the storage of keys no matter the size of the network. By loading the master key in new nodes, the scheme also allows the introduction of any number of sensors after the initial deployment. Furthermore, since all nodes certainly share the same master key, this scheme provides perfect key connectivity.

Full pairwise: In this case, each of the n nodes in the network receives n-1 pairwise keys to communicate with every other node. This approach assures a high security level, providing features such as node-to-node authentication and perfect resilience, which thwarts node replication attacks. It also makes the revocation of individual sensor nodes easier: even without the intervention of a secure base station, the nodes on the network may identify malicious IDs and revoke the corresponding pairwise keys.

Probabilistic: In probabilistic schemes, each node receives a group of keys, the so-called key chain, whose size is normally much lower than the size of the network itself. The reasoning

behind this strategy is to provide a good key connectivity and, at the same time, avoid both the memory overhead involved in the Full Pairwise scheme and the low security level offered by a single master key.

Matrix-based: Matrix-based scheme allows the creation of pairwise keys. It makes node authentication and revocation functionalities easier. Additionally, the scheme provides perfect key connectivity and its resilience.

Polynomial-based: The scheme is non-interactive; it does not add communication overhead to the key establishment process. Thus, the main constraints in this solution are the memory required for storing polynomial shares and the processing power needed for its operations.

Combinatorial design: Here, the keys those are preloaded into each node are carefully selected in a deterministic and optimized manner. These strategies are more adequate for adoption in dense networks since the key connectivity achieved by them depends on the proximity of the nodes.

Deployment knowledge: These schemes are deployment specific. They show higher flexibility and efficiency because they are built by considering specific deployment scenario.

Table 5.1: Comparison of key management schemes

Scheme	Scalability (S)	Node Authentication (A)	Deployment Knowledge(D)
BROSK [3]	High	No	No
LKMS [4]	Moderate	No	No
Full Pairwise [5]	Low	Yes	No
Q-Composite [6]	Moderate	No	No
Multipath key reinforcement [6]	Moderate	No	No
Pairwise key establishment [7]	Moderate	No	No
RGM [8]	Moderate	No	No
Blom's Scheme [9]	Moderate	Yes	No
Multiple space key [10]	Low	Yes	No
Grid-based [11]	Moderate	Yes	No
DMBS [12]	Moderate	No	No
GQ Design [13]	Moderate	No	No
Group-based deployment [1]	High	No	Yes
Closet Pairwise keys [13]	High	Yes	Yes
HGKM [14]	Moderate	Yes	Yes
Matrical Closet Pairwise [15]	Moderate	Yes	Yes
EDDK[18]	High	Yes	Yes

5.3 CMKMS: Cluster-based Mobile Key Management Scheme

5.3.1 System Model and Notation used

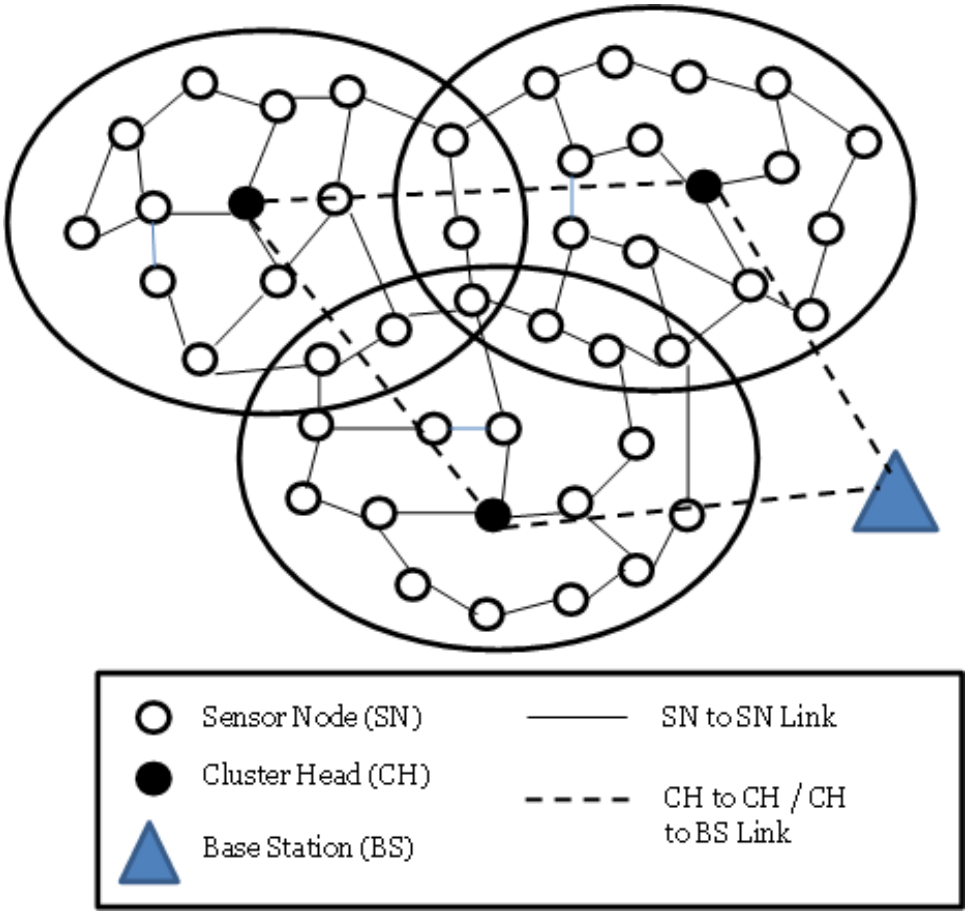

Figure 5.1: System model for key management

Figure 5.1 shows the considered system model for designing efficient key management scheme for wireless sensor network. The system model shows the network which is divided into number of clusters. The cluster is used to improve the scalability and energy efficiency of system. Each cluster consists of cluster head (CH) which aggregates the information from all sensor nodes (SN) in the cluster and transfers the aggregated information to the other CH or to the BS. The communication in between the SN to SN is intra-cluster communication which takes place via SN to SN link. The transmission in between the CH to CH or CH to BS is inter-cluster communication which takes place via CH to CH link or CH to BS link. The key management algorithm here considers the CH as key manger. The work make an assumption that SN can move from one position to another but CH and BS are fixed at one position.

The notations used for explaining the algorithm are as follows,

- N_i : Node ID
- C_i : Cluster ID
- K_h : Home key

- K_f : Foreign key
- N_c: Average number of nodes per cluster
- n_c : Number of cluster in the network
- l : Average number of cluster neighbour
- MAC_k : Hash function

5.3.2 Working Mechanism

The working of algorithm is dividing into two parts, which are as follows,

- Setup phase: It establish the cluster in a network and setup the cluster keys in a network.
- Key maintenance: The key maintenance is responsible for maintaining and managing the key during node mobility.

A. Setup Phase

Setup phase consist of two parts, organizing network into clusters and setting up a cluster keys for each cluster. It is responsible for establishing secure link between clusters to make the whole network connect securely. Here, consider that each *SN* is assigned a unique *ID* that identifies them distinctly in a network. The algorithm considers that each node maintains the two keys, key K_h i.e. home key and another K_f i.e. foreign key. K_h is used to do communication inside its own cluster and K_f is used to perform communication with foreign CH or nodes during node mobility. These keys will be used for secure information exchange in between the nodes.

In first part (as shown in Figure 5.2), after deployment, each node waits a random time before broadcasting the following HELLO message to declare its decision to become a cluster head $E_k\{N_i|K_h,N_i|K_f,MAC_k(N_i|K_h),MAC_k(N_i|K_f)\}$.Upon receiving a HELLO message, if the node has decided its role, it rejects all messages to avoid becoming cluster head and member at the same time. If the node has not decided yet, it responses only the HELLO, cancels timer, send ACK back and joins the cluster of the node that sent the message. The ACK message contains its id encrypted with key K_h and K_f. Then node set $C_i=N_i$ and set K_h and K_f as cluster keys. The cluster head construct the polynomial by using,$h(x) = f(x) + (K_{hc}^{i} \oplus K_{fc}^{i})$.

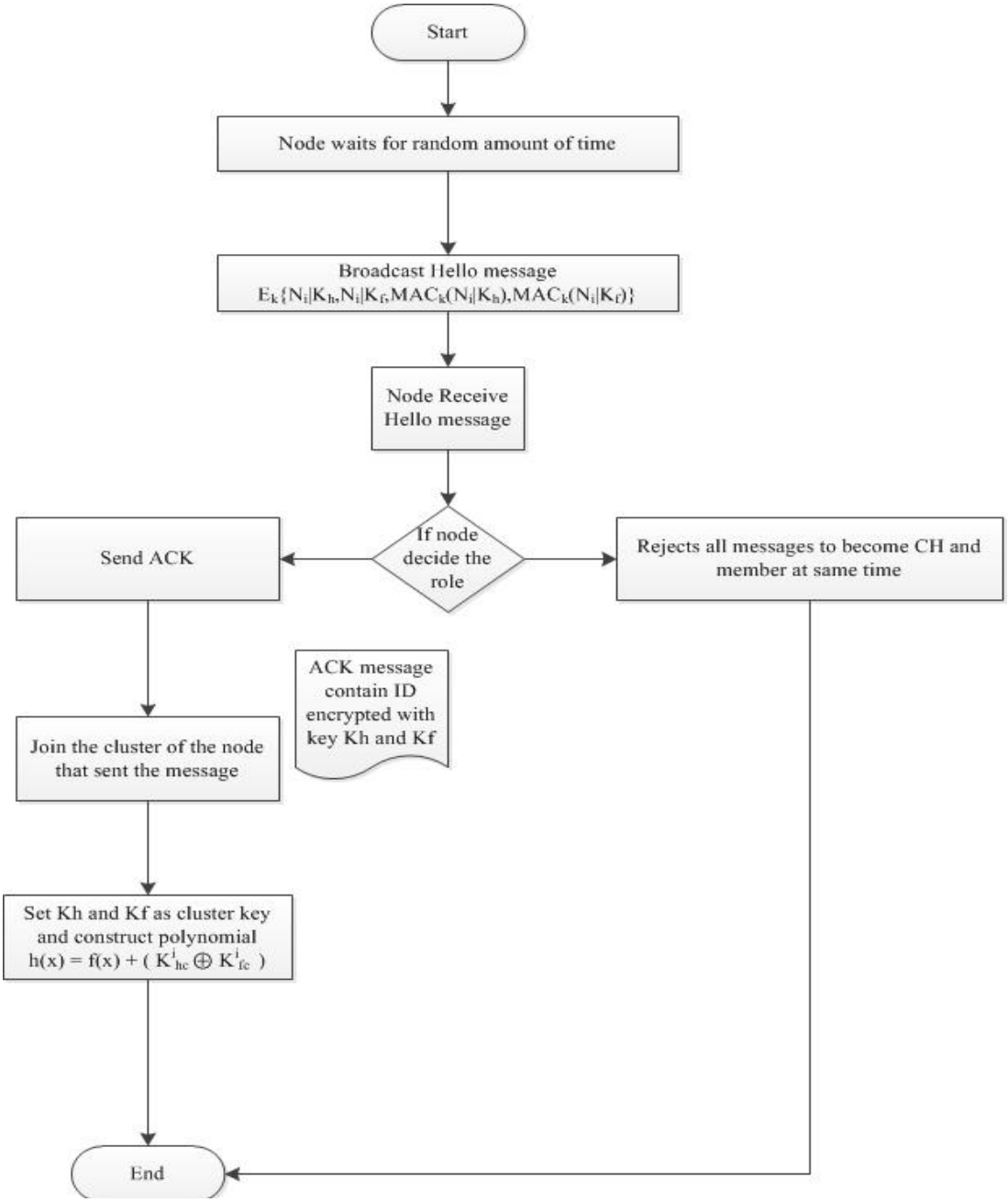

Figure 5.2:Flow chart for Key Management Setup phase part 1

The aim of the second part (shown in Figure 5.3) is to make the whole network connect securely. In the algorithm, nodes store cluster key of other neighbour clusters in the form of foreign key K_f. Once a node is compromised, it's all neighbour clusters must be evicted from network. In order to solve this problem, the algorithm generate a unique pairwise key for each neighbour nodes pair. For example, node 1 and node 2 are neighbour nodes pair in different cluster, they can establish pairwise key. The pairwise key is generated as follows, node 1 and node 2 exchange their foreign keys encrypted by home key: $E_k\{C_i|K_h|K_f, MAC_k(C_i|K_h|K_f)\}$. Nodes in the same cluster will ignore the message, while any nodes from neighbouring clusters store $\{C_i|K_h|K_f\}$. Then nodes located in two different clusters can compute their pairwise key, such as node 1 and node 2, they compute their pairwise key by

computing: $Khf_{Ci,Cj}^{1,2} = Khf_{Ci}^{1} \oplus Khf_{Cj}^{2}$. Then neighbour clusters can establish secure links in a network.

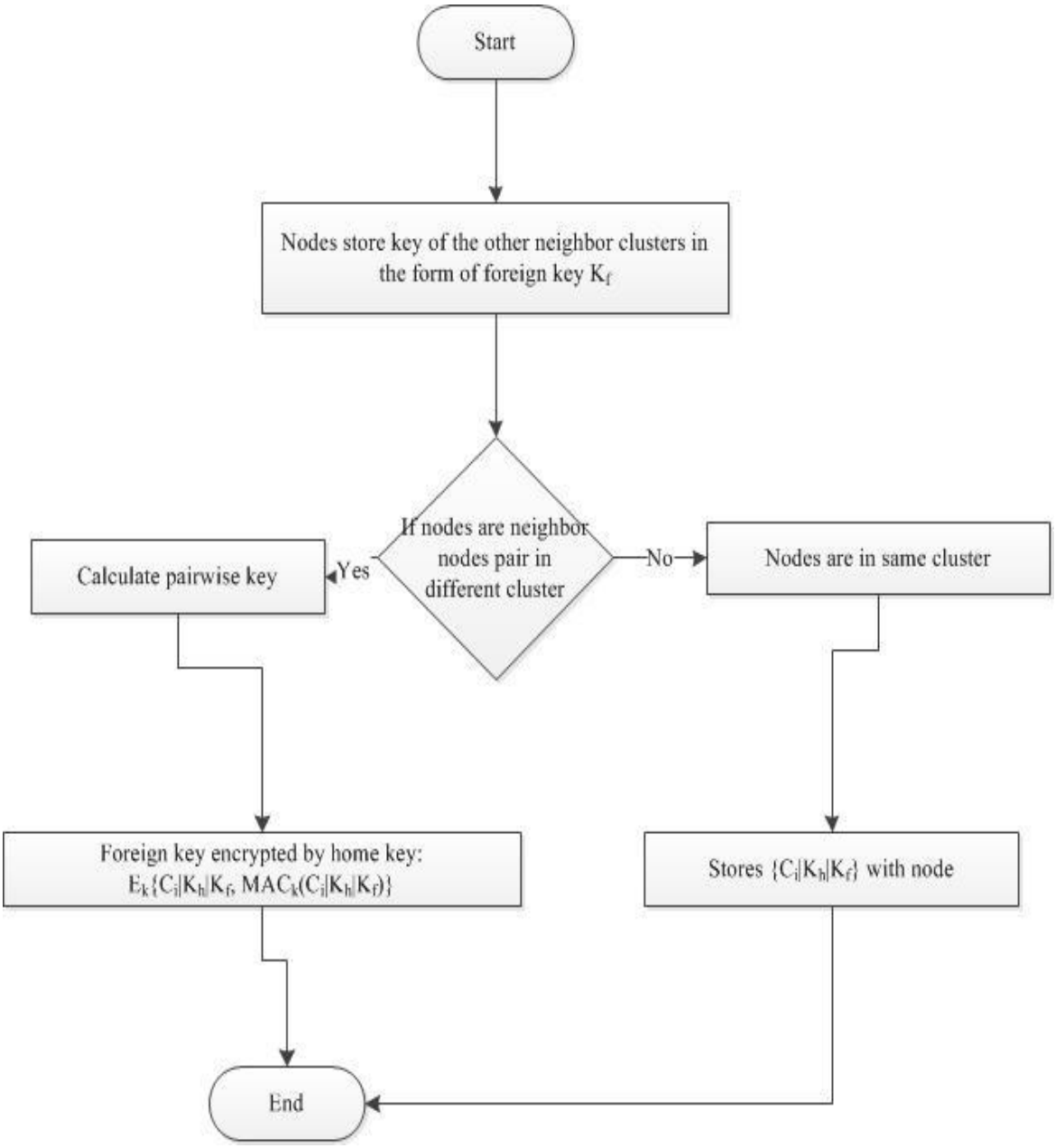

Figure 5.3: Flowchart for Key Management Setup phase part 2

B. Key Maintenance

The key maintenance phase tries to maintain the keys in following different situations,

- Case 1: When new node join the cluster
- Case 2: When any node move from one cluster to the other cluster

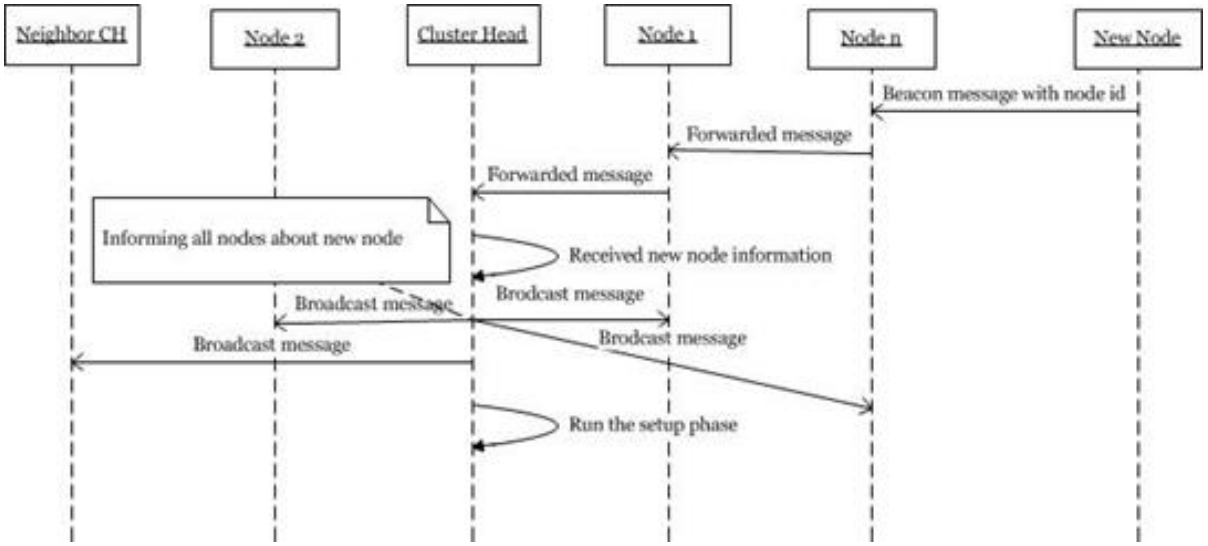

Figure 5.4: Key Maintenance Case 1 sequence diagram

As shown in figure 5.4, when new members supposed to join the cluster, it will beacons the message with its id. The beacon message is received by some neighbouring nodes and forwards it to the CH, or it may also receive it by CH directly. CH broadcast this message to other member and to other cluster heads. When new member join the cluster it will get the home key and foreign key by running the setup phase.

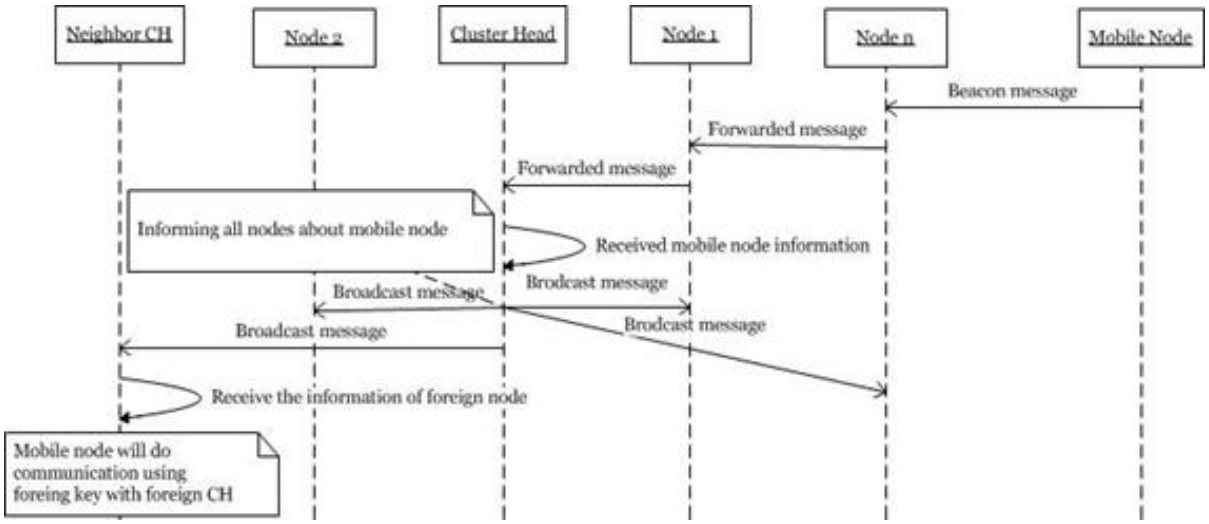

Figure 5.5: Key Maintenance Case 2 sequence diagram

As shown in figure 5.5, when a node wants to move from cluster it sends beacon message to CH that it wants to move from one cluster to other. CH updates the information to its members and to the neighbouring CH that node is moving from my territory to another one. Therefore the neighbouring cluster is getting understanding that the particular node will communicate with us using other cluster foreign key.

5.4 Simulation and Comparative Evaluation

5.4.1 Simulation Details

Table 5.2: Simulation and node parameters

Parameter Name	Setting Used
Network Interface type	Wireless Physical:802.15.4
Radio Propagation Model	Two-Ray Ground
Antenna	Omni-directional antenna
Channel Type	Wireless Channel
Link Layer	Link Layer (LL)
Interface Queue	Priority Queue
Buffer size of IFq	50
MAC	802.15.4
Routing Protocol	Ad-hoc routing
Energy Model	EnergyModel
Initial Energy (initialEnergy_)	100J
Idle Power (idlePower_)	31mW
Receiving Power (rxPower_)	35mW
Transmission Power (txPower_)	31mW
Sleep Power (sleepPower_)	15μW
Number of nodes	Varying from 25 to 250
Node Placement	Random

The implementation is performing by using discrete event simulator NS-2 (Network Simulator-2) [16]. The parameters set during simulations are as shown in Table 5.2. The idle power, receiving power, transmission power, and sleep power are as consider according to IEEE 802.15.4 radio model [17]. The implementation uses RC5 (with 12 rounds) as the block cipher to implement the encryption/decryption algorithm. The simulations also used MAC with RC5 to provide the pseudo random functions that is use to derive the individual keys as well as the pairwise keys. The performance of the proposed algorithm is compare with state of art algorithm Energy-Efficient Distributed Deterministic Key Management for Wireless Sensor Networks (EDDK) [18]. The simulation of EDDK and CMKMS are perform by considering same simulation and node parameters.

The simulations are performed in following different conditions. The different conditions are,

- WSN without any mobility
- WSN with random mobility speed and random number of mobile nodes.

The performance of the proposed mechanism is measure by using three parameters computational overheads, average energy consumption, and average delay.

Computational overheads: The computational overheads are mainly major by considering the work of setup phase/initialization phase. The computation overhead for each node includes the encryption and authentication of the local broadcast message, the verification and decryption of the received messages from neighbours, and the computation of the pseudorandom function.

Average Energy Consumption: The average energy consumption is majorly the average of energy spend during the different phases. It considers the average of energy spend by all nodes in the network. The energy consumption is directly proportional to computational overheads, if computational overhead will increase its effect on increasing energy consumption.

Average Delay: The average delay is concern with computational overheads. The failure and maintenance of key increase the delay of each node in system. The average delay considered is the average of total delay of each node in network.

5.4.2 Results and Comparative Evaluation

A. Results without WSN Mobility

In case of key management algorithms, large numbers of overheads are incurring during network initialization phase. The network initialization includes the encryption and authentication of the local broadcast messages, the verification, and decryption of the received message from neighbours and calculation of functions. The overheads are majorly measure in terms of computation overheads, which shows number of packets transmitted for initialization, average- energy, and delay incurred for it.

Figure 5.6, 5.7, and 5.8 show the computation overhead in terms of packets transmission, average energy consumption in joules and average delay in millisecond respectively. These three results are measure by varying the number of nodes in the network from 25 to 250. The result shows that proposed scheme CMKMS shows fewer overheads than EDDK. The major reason of the lower performance of EDDK than CMKMS is, EDDK considers local cluster key and pairwise keys for each node while CMKMS consider local, and foreign keys for each cluster nodes and pairwise keys only for the common nodes in between the clusters. The overheads incurred for establishing local cluster key and pairwise key is more than establishing keys in CMKMS algorithm. Another reason of improved performance of CMKMS over EDDK is that CMKMS uses less complex function than EDDK.

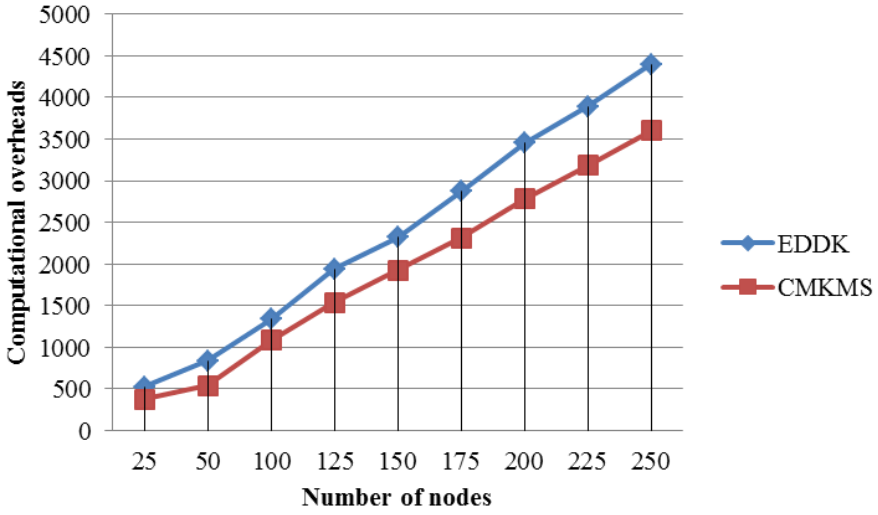

Figure 5.6: Comparative Key management computational overheads of EDDK & CMKMS under varying number of nodes without mobility

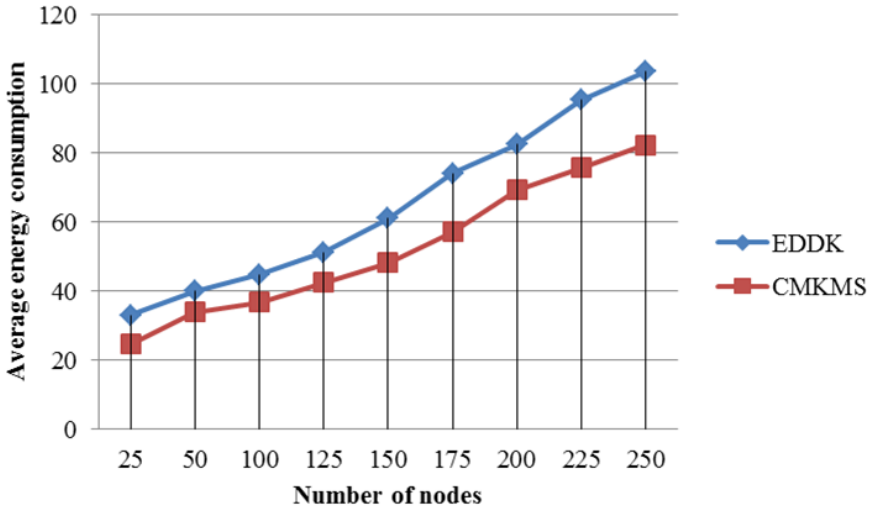

Figure 5.7: Comparative Key management average energy consumption performance of EDDK & CMKMS under varying number of nodes without mobility

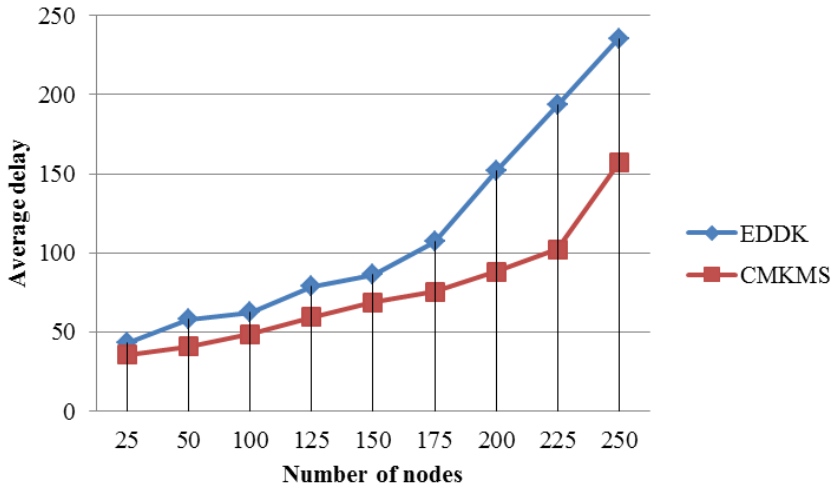

Figure 5.8: Comparative Key management average delay performance of EDDK & CMKMS under varying number of nodes without mobility

B. Results with Random WSN Mobility

Figures 5.9, 5.10, and 5.11 shows the performance of EDDK and CMKMS in realistic situations. The simulation considers that the nodes in the network are mobile which moves from one position to another expect the CH. The above results of computational overheads, average energy consumption, and average delay show that performance overheads in case of mobility are more than the results without mobility. The main reason of increasing performance overhead is when node goes mobile, it changes its neighbourhood. The change in neighbourhood directly effects on calculation of pairwise- and individual keys. Here, the performance of EDDK is lower than CMKMS. The major reason of lower performance of EDDK is that, it calculates the pairwise keys and change in neighbourhood effect on calculation of pairwise keys, which may give wrong instance of pairwise keys and need the recalculation of pairwise keys.

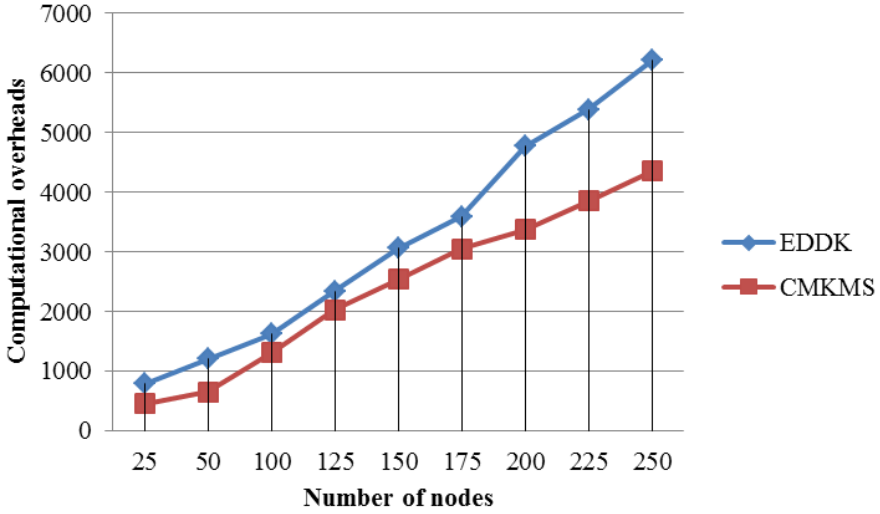

Figure 5.9: Comparative Key management computational overheads of EDDK & CMKMS under varying number of nodes with mobility

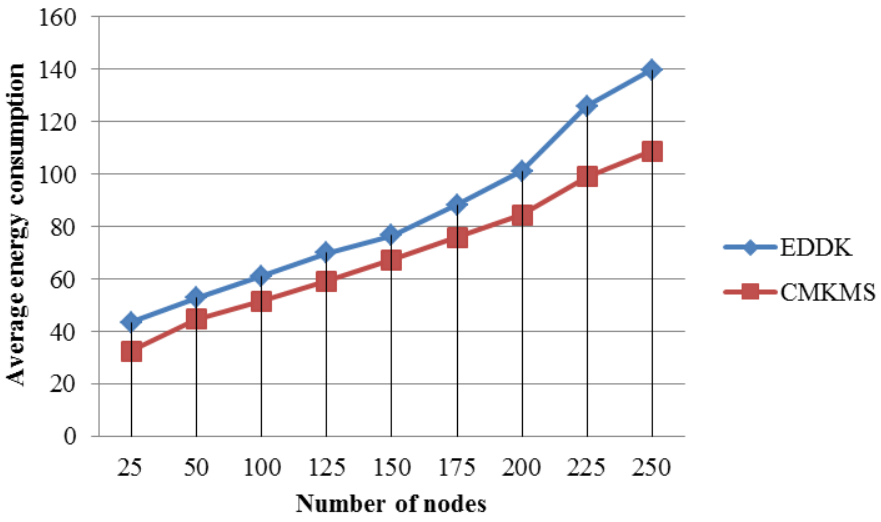

Figure 5.10: Comparative Key management average energy consumption performance of EDDK & CMKMS under varying number of nodes with mobility

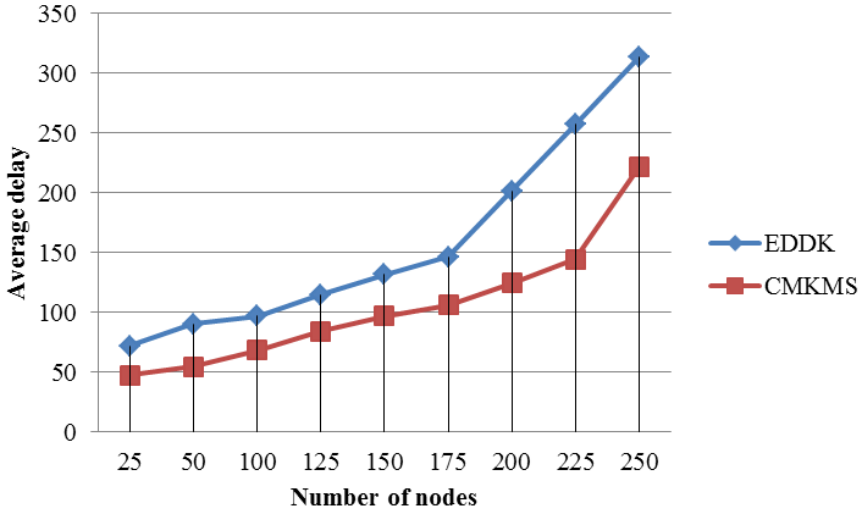

Figure 5.11: Comparative Key management average delay performance of EDDK & CMKMS under varying number of nodes with mobility

C. Results with Mobile CH

Figure 5.12, 5.13, and 5.15 shows the performance of EDDK and CMKMS in case of more proficient scenario where CHs are also mobile likes other nodes. The above results of computational overheads, average energy consumption, and average delay show that performance overheads in case of mobile CH are more than previous results. The main reasons of increasing performance overhead when CH goes mobile are: (i) it changes its neighbourhood and the change in neighbourhood effect on calculation of keys, pairwise- and individual keys and (ii) it also reflects in re-election of CHs. Here, the performance of EDDK is lower than CMKMS. The major reason of lower performance of EDDK is that, it calculates the pairwise keys and change in neighbourhood effect on calculation of pairwise keys, which may give wrong instance of pairwise keys and need the recalculation of pairwise keys.

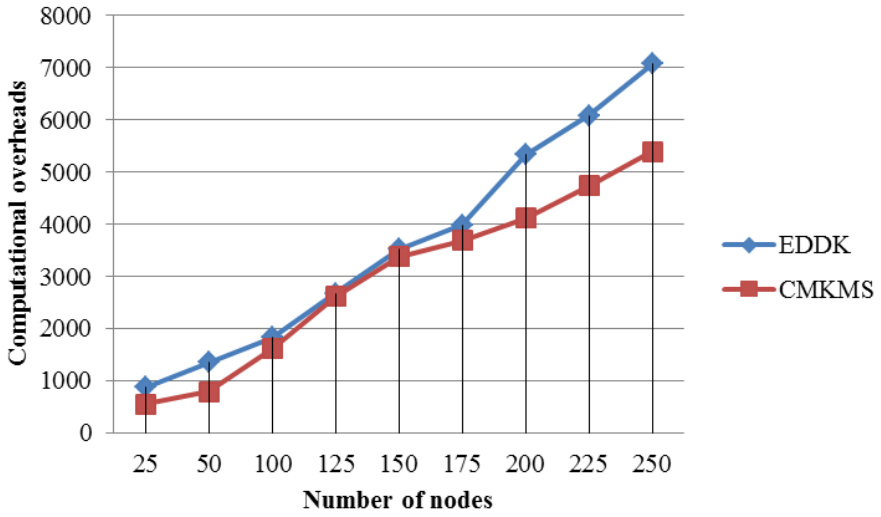

Figure 5.12: Comparative Key management computational overheads of EDDK & CMKMS under varying number of nodes and mobile CH

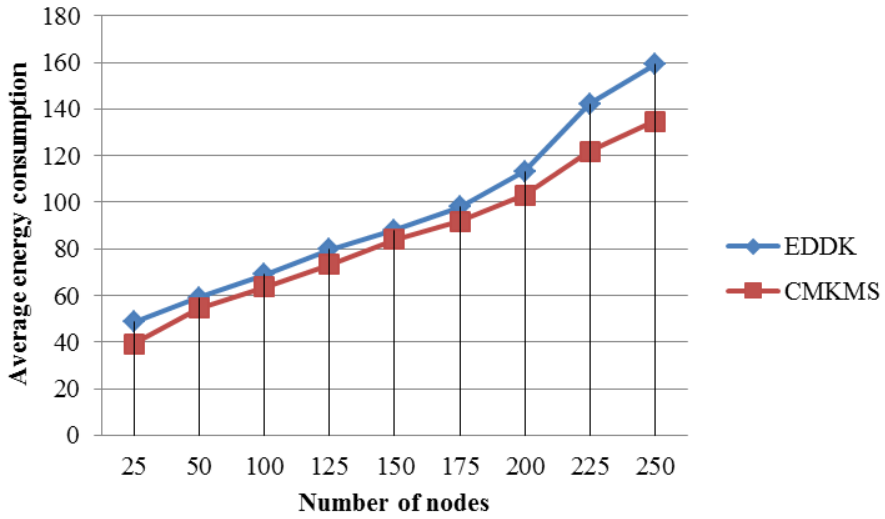

Figure 5.13: Comparative Key management average energy consumption performance of EDDK & CMKMS under varying number of nodes and mobile CH

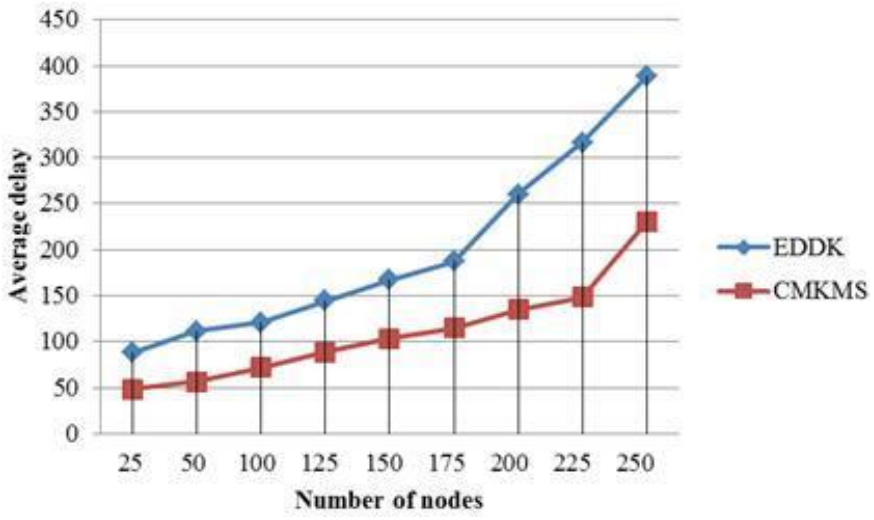

Figure 5.14: Comparative Key management average delay performance of EDDK & CMKMS under varying number of nodes and mobile CH

5.5 Conclusions

The growing demands of Wireless Sensor Networks (WSNs) in variety of real time and mission-critical applications, increases the challenges in terms of energy efficiency, security and mobility. The security is important to avoid malicious attacks and improve the energy efficiency, while mobility helps to improve the reachability of network.

The chapter addresses these two issues by proposing new Cluster-based Mobile Key Management Scheme (CMKMS). The CMKMS algorithm focused on the management and maintenance of keys under cluster-based mobile WSN network. The scheme consider two phases, first for key maintenance which establish the two private keys, home key for own cluster and foreign key when node moves from one cluster to another. The second phase maintain the keys when cluster head (CH) moves from one cluster to another. The proposed algorithm improves the efficiency of key management algorithm in terms of security, mobility, energy efficiency, and scalability of network. The simulation of scheme in different realistic situation shows that proposed solution shows less computational overheads, energy consumption and delay as compared with state-of-art solution.

5.6 References

[1] Yang Xiao, Venkata Krishna Rayi, Bo Sun, Xiaojiang Du, Fei Hu, Michael Galloway, “A survey of key management schemes in wireless sensor networks”, Computer Communication, Vol. 30, 2007, pp. 2314-2341.

[2] Junqi Zhang, Vijay Varadharajan, “Wireless sensor network key management survey and taxonomy”, Journal of Network and Computer Applications, Vol. 33, 2010, pp. 63–75.

[3] B. Lai, S. Kim, I. Verbauwhede, "Scalable session key construction protocol for wireless sensor networks", IEEE LARTES, IEEE Computer Society, Washington, DC, USA, 2002, pp. 1-7.

[4] B. Dutertre, S. Cheung, J. Levy, "Lightweight key management in wireless sensor networks by leveraging initial trust", Technical Report SRI-SDL-04-02, System Design Laboratory, SRI International, April 2004.

[5] H. Chan, V. Gligor, A. Perrig, G. Muralidharan, "On the distribution and revocation of cryptographic keys in sensor networks", IEEE Transactions on Dependable and Secure Computing, Vol. 2, Issue. 3, 2005, pp. 233-247.

[6] H. Chan, A. Perrig, D. Song, "Random key pre-distribution schemes for sensor networks", IEEE Symposium on Security and Privacy (SP'03), IEEE Computer Society, Washington, DC, USA, 2003, pp. 197–213.

[7] S. Zhu, S. Xu, S. Setia, S. Jajodia, "Establishing pairwise keys for secure communication in ad hoc networks: a probabilistic approach", IEEE ICNP'03, Washington, DC, USA, 2003, pp. 326–335.

[8] M. Ergun, A. Levi, E. Savas, "A resilient key pre-distribution scheme for multiphase wireless sensor networks", IEEE ISCIS'09, Washington, DC, USA, 2009, pp.375–380.

[9] R. Blom, "An optimal class of symmetric key generation systems", EUROCRYPT 84 Workshop on Advances in Cryptology: Theory and Application of Cryptographic Techniques, Springer, New York, NY, USA, pp. 335–338, 1985.

[10] W. Du, J. Deng, Y. Han, P. Varshney, J. Katz, A. Khalili, "A pairwise key pre-distribution scheme for wireless sensor networks", ACM CCS'03, New York, NY, USA, 2003, pp. 42–51.

[11] D. Liu, P. Ning, "Establishing pairwise keys in distributed sensor networks", ACM CCS'03, New York, NY, USA, 2003, pp. 52–61.

[12] W. Du, J. Deng, Y. Han, S. Chen, P. Varshney, "A key management scheme for wireless sensor networks using deployment knowledge", IEEE INFOCOM'04, Los Alamitos, CA, USA, 2004, pp. 586–597.

[13] D. Liu, P. Ning, "Improving key pre-distribution with deployment knowledge in static sensor networks", ACM Transactions on Sensors and Networks, Vol. 1, Issue. 2, 2005, pp. 204–239.

[14] N. Canh, Y.-K. Lee, S. Lee, "HGKM: a group-based key management scheme for sensor networks using deployment knowledge", IEEE CNSR'08, Los Alamitos, CA, USA, 2008, pp. 544–551.

[15] Z. Yu, Y. Guan, "A key management scheme using deployment knowledge for wireless sensor networks", IEEE Transactions on Parallel Distribution and Systems Vol. 19, Issue. 10, 2008, pp. 1411–1425.

[16] Network Simulator – 2, www.isi.edu/nsnam/ns/

[17] Derek J Corbett, Antonio G Ruzzelli, David Everitt, Gregory O'hare, "A Procedure for Benchmarking MAC Protocols used in Wireless Sensor Networks Technical Report 593", University of Sydeney, August 2006, pp. 1-28.

[18] Xing Zhang, Jingsha He, QianWei, "EDDK: Energy-Efficient Distributed Deterministic Key Management for Wireless Sensor Networks", EURASIP Journal on Wireless Communications and Networking, Volume 2011, pp. 1-11.

6

Conclusions and Future Work

This chapter concludes the thesis and proposes the future work, which can be researched and build based on the ideas proposed. This thesis addresses the security issues in the IoT and proposes an embedded security framework for IoT. The thesis have given major contribution in embedded IoT security framework, AES-GCM based embedded security protocol, taxonomy of different IoT security attacks, modelling and analysis of different types of jamming attacks, development of countermeasures on jamming attack, explore the possibilities of new kind of jamming attacks and development of countermeasures on it, and development of new lightweight key management technique by considering mobile scenarios of wireless sensor network. The novel methods together with implementation and simulation results are presented in this thesis. Throughout the thesis, either the proof of concept, simulation results and the implementation results are presented to validate the finding.

6.1 Summary of contributions

This chapter gives the summary of the thesis contributions with concluding remark on each contribution. Then the future work for each of the milestone that can be built upon this thesis is presented. The thesis mainly addressed the issues in embedded security by considering the IoT scenario and developed the mechanism to save IoT from jamming attack. The main three challenges consider in thesis are designing efficient IoT security framework, security solution on jamming attack, and key management in WSN-IoT. The thesis have given major contribution in embedded IoT security framework, AES-GCM-based embedded security protocol, taxonomy of different IoT security attacks, modelling and analysis of different types of jamming attacks, development of countermeasures on jamming attack, explore the possibilities of new kind of jamming attacks and development of countermeasures on it, and development of new optimized key management technique by considering mobile scenarios of wireless sensor network.

In the first chapter, thesis describes the security as the main pillar in IoT pillars. The thesis described the importance of IoT security by considering different real time example such as virtual shopping scenario for IoT. The requirements of IoT security are understood by using given example and derived the different objectives of the IoT in concern with security. The survey of IoT security framework had given the high level security requirements for IoT, as user identification, tamper resistant, secure software execution, secure content, secure network access, availability, secure data communications, identity management and secure storage. The chapter surveys the different security attacks on IoT such as physical attacks, side channel attacks, cryptanalysis attacks and software attacks and network attacks. All considered attack reduced the performance of IoT in major amount. The thesis considered the jamming attack, which is one of the denial of service attack, it harm the network in large amount by taking total control of the network. The chapter describes the motivation and problem statement of the thesis by understanding the different IoT scenarios, security frameworks, and security attacks. The chapter gives insight on the methodology used for completing the research, which helps to understand the flow of research and different development stages of research. The chapter also describes the novelty and contribution of research in figure 1.8 of Chapter 1, which helps to understand the evolution of research and problem addressed.

The security frameworks play a major role in performance improvement of the IoT. The chapter 2 addressed it by considering the embedded security in IoT, which consist of three approaches software only approach, hardware only approach and hybrid approach. The thesis has given the functionality comparison in between different types of embedded security approaches by considering countermeasures against attack and optimization of the basic security functions. The comparison helps to understand the different embedded security issues in IoT and help to enhance the embedded security by proposing the embedded security framework and architecture in figure 2.3 of chapter 2. The security consideration for IoT security evolved into AES-GCM-based embedded security protocol. The protocol consists of capability structure, which is combination of unique object identifier, access right, and randomization. The protocol is evaluated in terms of mutual authentication, replay attack resistance, and computation, traffic, and storage cost. AES-GCM provides both efficient authentication and encryption with efficient low cost implementation in resource-constrained devices.

The thesis considers jamming attack as major attack on WSN. The Chapter 3 survey the different jamming attacks and modelled them using activity and sequential modelling technique. The activity and sequential modelling of jamming attack gives the insights of the working of attack, which will be an efficient tool to develop the defensive mechanism against jamming attack. The evaluation of jamming attack describes in chapter conclude that the reactive jamming attack is one of the most disastrous jamming attack. The growing deployment of cluster-based network has given major possibility of attack in WSN-IoT. The chapter proposes the new possibility of jamming attack i.e. intelligent CH attack, which attack on CH and increase the possibility of hazards in the network. The intelligent CH jamming attack is compared with reactive jamming attack, which shows that intelligent CH jamming attacks are more destructive than reactive jamming attack. The modelling and evaluation of jamming attack gives the requirements to design efficient defense mechanism against jamming. The requirements considers the cross-layer features for efficient detection of attack, cluster-based network and use of threshold-based and game theoretic approach for developing efficient mechanism.

The fourth chapter majorly describes the classification of jamming countermeasures and compared different jamming countermeasures by considering type of technique, mechanism used, energy efficiency, and implementation cost. The comparative discussion gives the major advantages and disadvantages of existing approaches, which gives insight to develop new jamming countermeasure. The chapter made the three major contributions first one is TJC algorithm, second is game theory-based approach for jamming detection, and last is countermeasure on CH jamming attack. The TJC-based algorithm is based on send threshold of each node. The simulation of algorithm shows that TJC algorithm shows better performance against reactive jamming attack. The TJC algorithm also shows good performance in presence of increased number of jamming nodes in a network. The disadvantage of algorithm is that, it increase the overheads by maintain send threshold on each node. The game theory-based countermeasure, counteract to all kind of jamming attack in WSN. It considers the cross layer approach to detect wrong moves during the jamming game. The proposed game theory-based approach shows scalable performance in different realistic situations as compared with state-of-art solutions. The last contribution of chapter is countermeasure against cluster-based jamming, which is developed by extending TJC countermeasure for cluster-based network. It also helps to maintain safe situation in network form inter- and intra- cluster attacks.

The last chapter addressed the key management issue in WSN-IoT by considering mobility scenario. The major contribution of chapter is Cluster-based Mobile Key Management Scheme (CMKMS) for efficiently managing the keys under cluster-based mobile WSN network. The scheme consider two phases, first for key maintenance which establish the two private keys, home key for own cluster and foreign key when node moves from one cluster to another. The second phase maintain the keys when cluster head (CH) moves from one cluster to another. The proposed algorithm improves the efficiency of key management algorithm in terms of security, mobility, energy efficiency, and scalability of network. The simulation of scheme in different realistic situation shows that proposed solution shows less computational overheads, energy consumption, and delay as compared with state-of-art solution.

Hence, the thesis proposes the new architecture for IoT security and supporting defensive mechanism against jamming attack on IoT. The proposed solutions enable to enhance the secure and reliable applicability of IoT in increased application domain.

6.2 Future Work

Every research is complete and incomplete on its own sense of understanding. Therefore, there is always a scope to improve it and enhance it for better applicability. The address research problem on IoT security will be enhance in following ways,

- The research proposed the embedded security framework and architecture; this architecture will be enhance to improve the efficiency of embedded security by considering lightweight cryptography, physical security for trusted platforms, standardized the security protocols, secure operating system and secure storage.

- The IoT will also be improving in better manner by considering authorization, trust, and privacy at same time. It will directly effect on high level of interconnections between things and services.

- The thesis approaches to security and privacy during communication in IoT. The security and privacy will be also address during naming and addressing of IoT devices. Device discovery and network discovery of IoT devices will be made more secure by considering trust and reputation for its working mechanism.

- The thesis concentrate on modelling and development of countermeasure by considering jamming attack at physical layer and MAC layer, but it can be extend by considering combine effect of jamming on all layers of IoT protocol stack.

- The countermeasure was majorly developed by considering only the jamming attack. The work will be extend by considering combine effect of other IoT attacks such as physical attack, side channel attacks, cryptanalysis attack, software attack and network attack.

- The solution developed in thesis considered one or two cross layer features for effectively detecting the attack. The better solution will be developed by considering multi-cross layer features for jamming countermeasures.

- The future work for CMKMS is to exploit the key management algorithm according to specific attack such as jamming attack. The work can also be extended by considering the different kind of message patterns in the network.

Appendix A

List of Publications

My publications and Contributions are:

A. Journal Publications

1. **Sachin D. Babar**, Neeli R. Prasad, Ramjee Prasad, "**Activity Modelling and Countermeasures on Jamming Attack**", Journal of Cyber Security and Mobility, Vol. 2, Issue no. 2, pp. 1-27, April 2013**.**

2. **Sachin D. Babar**, Neeli R. Prasad, Ramjee Prasad, "**CMKMS: Cluster based Mobile Key Management Scheme for Wireless Sensor Network**", International Journal of Pervasive computing and Communications (IJPCC) : Special Issue on Adaptive Security for IoT, Vol. 10, Issue 2, pp-196-211, April 2014.

3. **Sachin Babar**, Parikshit N Mahalle, Neeli R. Prasad and Ramjee Prasad, "**A Hash Key-based Key Management Mechanism for Cluster-based Wireless Sensor Network**", Journal of Information Security and Applications, Elsevier editorial system. (Submitted)

B. Conference Publications

1. **Sachin Babar**, Parikshit N. Mahalle, Antonietta Stango, Neeli R Prasad and Ramjee Prasad, **"Proposed Security Model and Threat Taxonomy for the Internet of Things (IoT),"** In proceedings of 3rd International Conference CNSA 2010, Book titled: Recent Trends in Network Security and Applications - Communications in Computer and Information Science, Springer Berlin Heidelberg, pp. 420 - 429 Volume: 89. Chennai – India, July 23-25, 2010.

2. **Sachin Babar**, Antonietta Stango, Neeli Prasad, Jaydip Sen and Ramjee Prasad, "**Proposed Embedded Security Framework for Internet of Things (IoT)**" , In proceedings of 2nd International Conference on Wireless Communications, Vehicular Technology, Information Theory and Aerospace & Electronic Systems, Wireless VITAE 2011, vol., no., pp.1-5, Feb. 28, 2011 - March 3, 2011.

3. **Sachin Babar**, Parikshit N Mahalle, Neeli R. Prasad and Ramjee Prasad, **"Proposed on Device Capability based Authentication using AES-GCM for Internet of Things (IoT),"** In proceedings of 3rd International ICST Conference on Security and Privacy in Mobile Information and Communication Systems (Mobisec 2011), Aalborg – Denmark, May 17-19, 2011.

4. **Sachin D. Babar**, Neeli R. Prasad, Ramjee Prasad, "**Jamming Attack: Behavioral Modelling and Analysis**", In proceedings of the 3rd International Conference on Wireless Communications, Vehicular Technology, Information Theory and Aerospace & Electronic Systems, Wireless VITAE 2013, Princeton, New Jersey, USA, June 24-26, 2013.

5. **Sachin D. Babar**, Neeli R. Prasad, Ramjee Prasad, "**Proposed Game Theoretic Modelling of Jamming Attack and Attack Detection Mechanism**" , In proceedings of the 16th International Symposium on Wireless Personal Multimedia Communications, WPMC 2013, Atlantic City, New Jersey, USA, June 24 - 27, 2013.

6. **Sachin D. Babar**, Neeli R. Prasad, Ramjee Prasad, "**Countermeasure for Intelligent Cluster-head Jamming Attack in Wireless Sensor Network**" , In the proceedings of the International Conference on Privacy and Security in Mobile Systems, PRISMS 2013, Atlantic City, New Jersey, USA, June 24 - 27, 2013.

C. Other Publications

1. Parikshit N. Mahalle, **Sachin Babar**, Neeli R Prasad and Ramjee Prasad, "**Identity Management Framework towards Internet of Things (IoT): Roadmap and Key Challenges**" In proceedings of 3rd International Conference CNSA 2010, Book titled: Recent Trends in Network Security and Applications - Communications in Computer and Information Science, Springer Berlin Heidelberg, pp. 430 - 439 Volume: 89. Chennai – India, July 23-25, 2010.

Appendix B

Publications toward Chapters

Sr. No.	Publications	Chapters				
		Chapter 1	Chapter 2	Chapter 3	Chapter 4	Chapter 5
1	Proposed Security Model and Threat Taxonomy for the Internet of Things (IoT)	√				
2	Proposed Embedded Security Framework for Internet of Things (IoT)		√			
3	Proposed on Device Capability based Authentication using AES-GCM for Internet of Things (IoT)		√			
4	Activity Modelling and Countermeasures on Jamming Attack			√	√	
5	Jamming Attack: Behavioral Modelling and Analysis			√	√	
6	Proposed Game Theoretic Modelling of Jamming Attack and Attack Detection Mechanism			√	√	
7	Countermeasure for Intelligent Cluster-head Jamming Attack in Wireless Sensor Network				√	
8	CMKMS: Cluster based Mobile Key Management Scheme for Wireless Sensor Network					√
9	A Hash Key-based Key Management Mechanism for Cluster-based Wireless Sensor Network					√
10	Identity Management Framework towards Internet of Things (IoT): Roadmap and Key Challenges	√				

Appendix C

Short CV

Sachin D. Babar is ISTE Life Member. He is graduated in Computer Engineering from Pune University, Maharashtra, India in 2002 and received Master in Computer Engineering from Pune University, Maharashtra, India in 2006. From 2002 to 2003, he was working as lecturer in D.Y. Patil College of Engineering, Pune, India. From 2003 to 2004, he was working as lecturer in Bharati Vidyapeeth College of Engineering, Pune, India. From 2005 to 2006, he was working as lecturer in Rajarshi Shahu College of Engineering, Pune, India. From July 2006, he has been working as an Assistant Professor in Department of Information Technology, STES's Sinhgad Institute of Technology, Lonavala, India. Currently he is pursuing his Ph.D. in Wireless Communication at Center for TeleInFrastruktur (CTIF), Aalborg University, Denmark. He has published 20 papers at national and international level. He has authored two books on subjects like Software Engineering and Analysis of Algorithm & Design. He has received the Cambridge International Certificate for Teachers and Trainers at Professional level under MISSION10X Program. He is IBM DB2 certified professional. His research interests are Data Structures, Algorithms, Theory of Computer Science, IoT and Security.