Unmanned Vehicle Based Security System for Wireless Sensor Networks

Thesis submitted to

COCHIN UNIVERSITY OF SCIENCE AND TECHNOLOGY

in partial fulfilment of the requirements

for the award of the degree of

DOCTOR OF PHILOSOPHY

under the Faculty of Technology by

ARUN MADHU

Register No:4134

Under the guidance of

Dr. A. Sreekumar

Department of Computer Applications

Cochin University of Science and Technology

Kochi - 682 022, Kerala, India.

February 2017

Unmanned Vehicle Based Security System for Wireless Sensor Networks

Ph.D. thesis

Author:
ARUN MADHU
Research Scholar
Department of Computer Applications
Cochin University of Science and Technology
Kochi - 682 022, Kerala, India.
Email: arunmadhu99@yahoo.com

Research Advisor:
Dr. A. Sreekumar
Associate Professor
Department of Computer Applications
Cochin University of Science and Technology
Kochi - 682 022, Kerala, India.
Email: askcusat@gmail.com

Department of Computer Applications
Cochin University of Science and Technology
Kochi - 682 022, Kerala, India.

February 2017

To My Dear Teachers

&

Loving Family

Dr. A. Sreekumar
Associate Professor
Department of Computer Applications
Cochin University of Science and Technology
Kochi - 682 022, India.

28^{th} February 2017

Certificate

Certified that the work presented in this thesis entitled "Unmanned Vehicle Based Security System for Wireless Sensor Networks" is based on the authentic record of research carried out by Shri.ARUN MADHU under my guidance in the Department of Computer Applications, Cochin University of Science and Technology, Kochi-682 022 and has not been included in any other thesis submitted for the award of any degree.

A. Sreekumar
(Supervising Guide)

Phone : +91 484 2577602 +91 484 2556057 Email: askcusat@gmail.com

Dr. A. Sreekumar
Associate Professor
Department of Computer Applications
Cochin University of Science and Technology
Kochi - 682 022, India.

28^{th} February 2017

Certificate

Certified that the work presented in this thesis entitled "Unmanned Vehicle Based Security System for Wireless Sensor Networks" submitted to Cochin University of Science and Technology by Sri. ARUN MADHU for the award of degree of Doctor of Philosophy under the faculty of Technology, contains all the relevant corrections and modifications suggested by the audience during the pre-synopsis seminar and recommended by the Doctoral Committee.

A. Sreekumar
(Supervising Guide)

Phone : +91 484 2577602 +91 484 2556057 Email: askcusat@gmail.com

Declaration

I hereby declare that the work presented in this thesis entitled "Unmanned Vehicle Based Security System for Wireless Sensor Networks" is based on the original research work carried out by me under the supervision and guidance of Dr. A. Sreekumar, Associate Professor, Department of Computer Applications, Cochin University of Science and Technology, Kochi–682 022 and has not been included in any other thesis submitted previously for the award of any degree.

ARUN MADHU

Kochi– 682 022
28^{th} February 2017

Acknowledgement

First and foremost I thank the Almighty God without whom I would not have completed my work. It was indeed the power and blessings of God which has motivated me to work constantly and overcome the difficulties during the course of my research period.

I take immense pleasure in expressing gratitude to my supervising guide and mentor, Dr. A Sreekumar, Professor, Dept. of Computer Applications, Cochin University of Science and Technology for his sustained enthusiasm, creative suggestions, motivation and exemplary guidance throughout the course of my doctoral research. I am deeply grateful to him for providing me necessary facilities and excellent supervision to complete this work. His passion for work has exceptionally inspired and enriched my growth as a student and a researcher. His patience and tolerance throughout the span of my doctoral work had been remarkable. I extend my profound gratitude for his constant encouragement and timely advice.

I offer my sincere gratitude to Dr. Kannan B, HoD, Department of Computer Applications, Cochin University of Science and Technology for his help and encouragement during the period of my research. I sincerely thank Dr. K. V. Pramod, Professor, Dept. of Computer Applications, Cochin University of Science and Technology for his continuous motivation and support for my work. I would like to acknowledge the Faculty members, Office staff and lab staff of the Department of Computer Applications, Cochin University of Science and Technology who have helped me during the various stages of my research. The Research scholars in the Department of Computer Applications deserve special mention. I thank from the bottom of my heart my co-scholars Binu V.P, Dr.Remya A.R, Dr.Bino Sebastian V, Dr.Ramkumar R, Jino P.J,

Malathi S, Santosh Kumar M.B, Jasir M.P, Vijith T.K, Vinu V.S, Jestin Joy, Sukrith B, Shyam Sundar, and Sunil Kumar R for making me more passionate towards research activities.

The personality who invoked my research attitude is Dr. K P Soman, HoD, Computational Engineering and Networking, Amrita Vishwa Vidyapeetham, Coimbatore. He gave a different dimension to visualise the mathematical problems and provided immense support by providing the infrastructure and wireless sensor network hardware for my research work. I take this opportunity to extend my profound thankfulness for his constant encouragement and timely advice. I consider Amrita Vishwa Vidyapeetham Coimbatore campus as a perfect place which has imbibed new ideas in me.

I was doing my research as part-time and I was badly in need of support from my Employers and Supervisors. I extend my thanks to Tata Consultancy Services for granting permission for my research work. I will always be grateful to the college authorities, St Joseph's College of Engineering and Technology, Pala for their whole hearted support and co-operation . I am greatly indebted to Ms. Smitha Jacob, HoD, St Joseph's College of Engineering and Technology for giving me freedom and support during the completion stages of my research. I always got constant support and encouragement from my colleagues. A few of them deserve special mention namely Jacob P Cherian, Dr.Jubilant J Kizhakkethottam, Deepu Job, Sarju S, Smija Das, Suma R, Mereen Thomas, Sreejith V, Sreesh P R, Paul Jose, Darsan Lal, Riji N Das, and Keerthi A S Pillai.

I would like to acknowledge my friends for their constant motivation and support. I would like to specially mention Jino Sebastian, Dr.Abi P Mathew, Baiju Jacob, Babu Sankar, Jacob P Cherian, Sarath Mangalat, Praveena Das, Dr. Anand Kumar,

Vikram Raja, and Hari Krishnan for their endless support during this time period.

I am blessed with a wonderful family without whose support I would not have achieved anything in life. I am always thankful to my parents Late M. Madhusudanan and K. Viayakumari for their blessings and proper guidance in shaping my professional career and life. My younger brother Dr. Kiran Madhu, has always been supportive and his matured decisions have helped me a lot during my research period to a great extent. I also thank my in-laws for the encouragement they have extended throughout my research period. My better half Indu V.S, has always been with me and the extreme confidence that she has shown in my abilities has encouraged me to do my level best in the research work. Her wholehearted support and presence has always been my strength in the smooth journey of my life. Last but not the least, my daughter Thanmaya Arun, who joined me in the last lap of research period and whose smile has always been an energy to go ahead with confidence.

Arun Madhu

Preface

Nations are spending massive amounts for monitoring the security of sensitive areas. Human monitoring is expensive and vulnerable due to human errors. The wireless sensor networks are used as an alternative to monitor the sensitive areas. The major drawback of the wireless sensor network monitoring is the unattended nature of the network which makes the system susceptible to attacks by conflicting troupes. The limited battery capacity and processing power makes it difficult to implement complex cryptographic solutions to protect the network. The compromised or intruder motes will be used as a launcher of various kinds of attacks in network. In these networks, a large number of sensor nodes are deployed to monitor a vast field, where the operational conditions are most often harsh or even hostile. Since these networks are usually deployed in remote places and left unattended, they should be equipped with security mechanisms to defend against attacks such as node capture, physical tampering, eavesdropping, denial of service, etc.,

An intrusion detection system monitors a host or network for suspicious activity patterns outside normal and expected behaviour.

It is based on the assumption that there exists a noticeable difference in the behaviour of an intruder and legitimate user in the network such that an IDS can match those pre-programmed or possible learned rules. The current intrusion detection systems used to protect the network rely on neighbour information to take a decision. There is no mechanism to cross check the validity of event reported by the network mote. The proposed solution is to use a tiny vehicle navigating in the network controlled by base station for analysing the traffic and mote behaviours to detect the attacker mote and protect network. The vehicle can also act as a watch dog to protect the network from unexpected attacks.

The costly sensors and camera can be attached to vehicle for military monitoring. The GPS unit and calibrated clock unit can be used for Localization and Time synchronization of the network nodes. The proposed system brings together a general architecture called Guarding Architecture for Unattended Deployment Applications (GARUDA) for Wireless sensor network Security. A cluster-based approach is used to classify the network nodes based on functionality and priority. A key Pre-distribution technique is used to protect the key-management schema. The modified Localized Encryption and Authentication Protocol are used for hierarchical key management. The Rivest Cipher 5 (RC5) algorithm is used for encryption of sensitive data.

The proposed system has an unmanned vehicle with sensors to protect the network from attacks and report the malicious activities to the base station. Implementation is done using a tiny car controlled by micaz mote inside the vehicle and nesC is used for programming. Vehicle can navigate and monitor the entire network by using GPS location information. The system was tested by using micaz motes programmed to monitor an event and the intruder

motes were programmed to deploy the different types of attacks. The result shows a sharp increase in the detection of attacker motes and finding the false alerts. The mobile nature and periodic maintenance of vehicle makes it adaptable to the real time challenges.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

This chapter throws light on Wireless Sensor Networks and performs a thorough survey on the various issues which prevail in wireless sensor network security and defensive mechanisms.

The heart of a Wireless sensor networks (WSNs) is the numerous tiny devices which are equipped with sensing, processing, and communication capability. These wireless sensor networks act as watchful eyes to monitor the activities of the real world environment. The pivotal applications of WSNs include critical military surveillance, environmental monitoring and industrial environment safety. In these networks, a large number of sensor nodes are deployed to monitor a vast field, where the operational conditions are most often harsh or even hostile. Nevertheless, the nodes in WSNs suffer from serious resource constraints due to their scanty processing power, limited battery and restricted memory. These networks are usually deployed in secluded places and may be left unattended. Hence, they should be provisioned with immaculate security mechanisms to safeguard them against various attacks like node capture, physical tampering, eavesdropping and denial of

service. Unfortunately,the traditional security mechanisms with high overhead are not feasible for the resource drained sensor nodes.

Over the years many eminent researchers have worked on wireless sensor network security and has come up with a handful of solutions [10]. In addition to the traditional security issues such as secure routing and secure data aggregation, security mechanisms deployed in WSNs should inculcate collaboration among the nodes. This is due to the decentralized nature of the network and the lack of a proper infrastructure. As the wireless sensor nodes are physically left insecure and unattended, they are prone to outside attacks.

The first section of this chapter provides an overview of WSN. The second Section lists out the major constraints of WSNs. The third Section supplies an overview of security requirements for WSNs. The fourth section lists out the various possible attacks and the corresponding countermeasures required in WSN. A holistic view of the security issue is presented in the final section. These issues are classified primarily into six categories: cryptography, key management, secure routing, secure data aggregation, intrusion detection and trust management.

1.1 Wireless Sensor Network

A wireless sensor network is a collection of nodes organized into a co-operative network. The wireless sensor networks have begun to be replaced at varied sectors of life. WSN technology is exciting with unlimited potential for numerous areas of application encompassing environmental, medical, military, transportation, entertainment, crisis management, homeland defence and smart spaces. Each node in a WSN comprises of a processor, multiple types of memory, RF

transceiver and a power source. It can accommodate various sensors and actuators. The communication among nodes is wireless and often they self-organize after being deployed in an ad-hoc fashion. Such systems can help rejig the way we live and work.

1.1.1 Routing

Multihop routing is a critical service required for WSN. Internet and Mobile Ad hoc Network (MANET) routing techniques when effectuated, may not adapt to the complex requirements of the WSN. Internet routing assumes the availability of highly reliable wired connections where packet errors are minimal; while this is not true in a WSN. Most of the MANET routing solutions depend on symmetric links between neighbours; which is a falsity as far as the WSNs are concerned. This contrariety has necessitated the invention and deployment of brand new solutions. The routing mechanism of a WSN typically begins with the process of neighbour discovery. Nodes send rounds of messages and build a local neighbour table. This table includes the minimum information pertaining to each neighbour's ID and location. The nodes must be self aware of their geographic location prior to neighbour discovery. Information such as remaining energy of node, delay via the node, and an estimate of link quality is also maintained in the table.

Once the table is created, the messages are directed from a source location to a destination location based on the geographic coordinates. A typical routing algorithm that works in this manner is the Geographic Forwarding (GF). The GF works as follows : each node is aware of its location and the destination address. Using Euclidean distance calculation, each node identifies the neighbour node which can most effectively act as the next hop in the

transmission path. The message is then forwarded to this hop. In variants of GF, a node could also take into account delays, reliability of the link and remaining energy. Another important routing paradigm for WSN is directed diffusion. This solution integrates routing, queries and data aggregation. In this technique, a query is disseminated indicating an interest in data from remote nodes. A befitting node responds with an attribute-value pair. The attribute-value pair is drawn towards the requester based on gradients, which are set up and updated during query dissemination and response. Along the path from the source to the destination, data can be aggregated to reduce communication costs. Data may also traverse over multiple paths, intensifying the robustness of routing.

1.1.2 Node Localization

Node localization is the task of determining the geographical location of each node in the system. Localization is one of the rudimentary assignments of a wireless sensor network. It is also considered to be a laborious task. Various issues such as the cost of extra localization hardware, degree of location accuracy required, mode of setting up, line of sight requirements, Dimensionality of the problem etc., Some of the issues are not a cause of concern as there are ways to solve them. For eg, if cost and form factor are not a major concern and accuracy of a few meters is acceptable, then for outdoor systems, equipping each node with GPS is a simple solution. If the system is manually deployed one node at a time, then a simple GPS node can designate the location of each node.

Most of the solutions for localization in WSN are either range-based or range-free. Range-based schemes use various

techniques to first determine the distance between nodes and then compute the location using Euclidean formula. To determine the distance, supplementary hardware is usually required, e.g: hardware to detect the time difference in the arrival of sound and radio waves. This difference can be converted to a distance measurement. In range-free schemes distance is not determined directly, howbeit the hop counts are considered. Once the hop counts are determined, the distance between nodes are reckoned using the average distance per hop metric, and then geometric principles are used to compute location. Range-free solutions are not as accurate as range based solutions and often require more number of messages. However, they do not require extra hardware on every node.

1.1.3 Clock Synchronization

Clock synchronization is of vital importance in a wireless sensor network. The clock corresponding to each node should read the same local time. Since clocks drift over time, they must be periodically re-synchronized and in some instances where high accuracy is required, it is necessary for nodes to account for clock drift between synchronization periods. Clock synchronization is of essential importance because of a variety of reasons. It is often necessary to know when and where an event has occurred in a WSN. Clocks are also used for handling many system and application level tasks. For example, sleep/wake-up schedules, some of the localization algorithms and sensor fusion depend on the synchronization of clocks. Application tasks such as tracking and computing of velocity are also dependent on synchronized clocks. The clock synchronization protocols that have been developed for WSN are Reference Broadcast Synchronization (RBS), Flooding Time

Synchronization Protocol (FTSP) and Time Synchronization Protocol for Sensor networks (TPSN).

In RBS, a reference time message is broadcast to the neighbour nodes. The receiver nodes make a note of the time of receiving the broadcast message. The Nodes exchange their recorded time values and synchronize their clocks accordingly. This protocol suffers no transmitter side non-determinism since timestamps are only on the receiver side. Accuracy is around 30 microseconds for 1 hop. In TPSN, a spanning tree is created for the entire network. This solution assumes that all links in the spanning tree are symmetric. Pair wise synchronization is performed along the edges of the tree starting at the root. Since there is no broadcasting as in RBS, TPSN is expensive. A key attribute of this protocol is that the timestamps are attached to the outgoing messages in the MAC layer thereby reducing non-determinism. Accuracy is in the range of 17 microseconds. In FTSP, there are radio-layer timestamps, skew compensation with linear regression, and periodic flooding to make the protocol robust to failures and topology changes. Both transmission and reception of messages are time stamped in the radio layer and differences are used to compute and adjust clock offsets.

1.1.4 Power Management

Many devices such as Mica2 and Micaz that are used in WSN run on two AA batteries. Depending on the activity level of a node, its lifetime may only be a few days, if no power management schemes are used. Since most of the system requires much enhanced lifetime, significant research has been undertaken to increase its lifetime without compromising the functional requirements. At the hardware level it is possible to add solar cells or scavenge energy from motion

or wind. Due to technical advancements the battery quality has improved to a great extent. If the form factor is not to be a cause of concern, more batteries can be incorporated into the node structure. Great advancements have been seen w.r.t low power circuits and micro controllers. Most hardware platforms allow multiple power saving states (off, idle, on) for each component of the device (each sensor, the radio, the micro controller). This ensures that only the required components are active at any instant.

At the software level power management solutions are targeted at

(i) minimizing communication since transmitting and listening for messages is energy expensive

(ii) creating sleep/wake-up schedules for nodes or specific node components.

Minimizing the number of messages is a cross-cutting problem. For example, with a good MAC protocol there are fewer collisions and retries. With a good routing scheme, short paths and congestion avoidance or minimization can be achieved and this in turn minimizes the number of messages forwarded.

1.2 Constraints of WSNs

A wireless sensor network consists of a large number of sensor nodes which are inherently resource-constrained. These nodes have limited processing capability, very low storage capacity, and constrained communication bandwidth. These limitations are due to the limited energy and physical size of the sensor nodes. Due to these constraints, it is a demanding process to directly employ the conventional security mechanisms in WSNs. Some of the major constraints of a WSN are listed below.

1.2.1 Energy Constraints

Energy is the prime cause of concern for a WSN because of its limited battery power. Due to the left alone nature of the WSN, its battery cannot be replaced effectively. Energy consumption in sensor nodes can be categorized mainly into three parts: (i) energy for the sensor transducer, (ii) energy for communication among sensor nodes, and (iii) energy for microprocessor computation. Each bit transmitted in WSNs consumes about as much power as executing about 800 to 1000 instructions. Thus, communication is costlier than computation in WSNs. Any message expansion caused by security mechanisms occur at a significant cost. Further, higher security levels in WSNs usually correspond to more energy consumption for cryptographic functions. While considering security as prime objective, the communication and computation overhead will substantially reduce the life time of the WSN.

1.2.2 Memory Limitations

A sensor is a tiny device with a limited amount of memory and storage space. Memory units of a sensor node usually include flash memory and RAM. Flash memory is used for storing downloaded application code and RAM is used for storing application programs, sensor data and intermediate results of computations. Usually sufficient space is not available to run complicated algorithms after loading the OS and application code. TinyOS consumes about 4 KB bytes of instructions,leaving only 4500 bytes for security and applications. A common sensor type micaz mote has a 128 KB flash memory and 4 KB RAM for storing application program and data corresponding to security protocols. The limited storage capacity is a

bottleneck while considering the complex cryptographic algorithms required for implementing security.

1.2.3 Unreliable Communication

Unreliable communication is another major threat to sensor security. Normally the packet-based routing of sensor networks is based on connectionless protocols and thus inherently unreliable. Packets may get damaged due to channel errors or may get dropped at highly congested nodes. Furthermore, the unreliable wireless communication channels may also lead to damaged or corrupted packets. Higher error rate also mandates robust error handling schemes to be implemented leading to higher overhead. In certain situations, even if the channel is reliable, the communication may not be reliable due to the broadcast nature of wireless communication, as the packets may collide in transit and may need a retransmission.

1.2.4 Higher Latency in Communication

In a WSN, multi-hop routing, network congestion and processing in the intermediate nodes may lead to higher latency in packet transmission. This makes synchronization very difficult to achieve. The synchronization issues may sometimes be very critical as far as the security is concerned, as some security mechanisms may rely on critical event reports and cryptographic key distribution.

1.2.5 Unattended Operation of Networks

In most cases, the nodes in a WSN are deployed in remote regions and are left unattended. The likelihood that a sensor node encounters a

physical attack in such an environment is therefore, very high. Remote management of a WSN makes it virtually impossible to detect physical tampering. This makes security in WSNs a particularly difficult task.

1.3 Wireless Sensor Network Security Requirements

A WSN is a special type of network. It shares some commonalities with a typical computer network, but also exhibits many characteristics which are unique to it. The security services in a WSN should protect the information communicated over the network and the resources from attacks and misbehaviour of nodes. The most important security requirements in WSN are listed below.

1.3.1 Data Confidentiality

Data confidentiality is the vital issue in network security. Every network which focus on security should take into consideration, the aspect of data confidentiality. In sensor networks, the confidentiality relates to the following:

- A sensor network should not allow sensor readings to pass on to its neighbours. Especially in a military application, the data stored in the sensor node may be highly sensitive.

- In many applications, nodes communicate highly sensitive data, e.g., key distribution. Therefore, it is highly important to build a secure channel in a wireless sensor network.

- Public sensor information such as sensor identities and public keys, should also be encrypted to some extent to protect themselves against traffic analysis attacks. The standard approach for maintaining the secrecy of data is to encrypt the data with a secret key which the intended receiver will possess and hence achieve confidentiality.

1.3.2 Data Integrity

With the implementation of confidentiality, an adversary may be unable to steal information. However, it doesn't mean that the data is safe. The adversary can change the data, so as to send the sensor network into disarray. For example, a malicious node may add some fragments with the message sent or manipulate the data within a packet. This new packet can be sent to the intended recipient. Damage or data loss can even occur without the presence of a malicious node due to the harsh communication environment. Thus, data integrity ensures that any receiver can rely on the data received.

1.3.3 Availability

Extra costs are incurred whilst the traditional encryption algorithms are implemented on to the wireless sensor network scenario. Some approaches choose to modify the code, or make use of code reusability, while other approaches try to make use of additional communication to achieve the same goal. Yet another set of methodologies force strict limitations on the data access, or propose an unsuitable scheme (such as a central point scheme) in order to simplify the algorithm. But all these approaches weaken the availability of a sensor and sensor network for the following reasons:

- Additional communication consumes more energy. As communication increases, so does the chance of incurring a communication conflict.

- Additional computation consumes considerable amount of energy. If all the energy is drained, the data will no longer be available.

- A single point failure will be introduced if the central point scheme is used. This is a threat to network availability. The requirement of security not only affects the operation of the network, but is also vital in maintaining the availability of the entire network.

1.3.4 Data Freshness

WSNs also need to ensure the freshness of each message. Informally, data freshness suggests that the data is recent, and it ensures that no old messages have been replayed. This requirement is important when there are shared key strategies employed in the design. Typically, shared keys need to be replaced over time. However, it takes time for new shared keys to be propagated to the entire network. In this case, it is easy for the adversary to use a replay attack. Also, it is effortless to disrupt the normal work of the sensor, if the sensor is unaware of the new key change time. To solve this problem a nonce, or another time-related counter can be appended to the packet which in turn ensures data freshness.

1.3.5 Self-organization

Each node in a WSN should be self-organizing and self-restoring. This feature of a WSN also pose a great challenge to the security. The dynamic nature of a WSN makes it impossible to deploy any previously installed shared key mechanisms among the nodes and the base station . A number of key distribution schemes have been proposed in the context of symmetric encryption. However, for application of public key cryptographic techniques an efficient mechanism for key distribution is very much essential. It is desirable that the nodes in a WSN self-organize among themselves not only for multiple hop routing but also to carry out key management and developing trust relations.

1.3.6 Secure Localization

In many situations, it becomes necessary to accurately and automatically locate each sensor node in a WSN. For example, a WSN designed to locate faults would require accurate locations of sensor nodes identifying the faults. If the location information is not secured properly, a potential adversary can easily manipulate and provide false location information by reporting false signal strength, replaying messages etc.,

1.3.7 Time Synchronization

Time synchronization can be defined as the characteristic which ensures that the clock of each node in a WSN reads the same time within epsilon and remain that way. Since clocks drift over time, they must be periodically re-synchronized and in some instances where high accuracy is required, it is even important for nodes to

account for clock drift between synchronization periods. Most of the applications of sensor networks require time synchronization. Any security mechanism for WSN should also be time-synchronized. A collaborative WSN may require synchronization among a group of sensors.

1.3.8 Authentication

It ensures that the communicating node is the one that it claims to be. An adversary cannot only modify data packets but also change a packet stream by injecting fabricated packets. It is, therefore, essential for a receiver to have a mechanism to verify whether the received packets have indeed come from the actual sender node. In case of communication between two nodes, data authentication can be achieved through a message authentication code (MAC) computed from the shared secret key among the nodes. A number of authentication schemes for WSNs have been proposed by researchers.

1.4 Security Vulnerabilities in WSNs

Wireless Sensor Networks are vulnerable to various types of attacks. These attacks are mainly of three types.

- Attacks on secrecy and authentication: Standard cryptographic techniques can protect the secrecy and authenticity of communication channels from outsider attacks such as eavesdropping, packet replay attacks, and modification or spoofing of packets.

- Attacks on network availability: Attacks on availability of WSN are often referred to as denial-of-service (DoS) attacks. This affects the normal working of the services provided by the WSN.

- Stealthy attack against service integrity: In a stealthy attack, the goal of the attacker is to make the network accept a false data value. For example, an attacker compromises a sensor node and injects a false data value through the sensor node. In these attacks, keeping the sensor network available for its intended use is essential.

1.4.1 Denial of Service (DoS) Attacks

The DoS attack refers to an intruder attempt to disrupt, subvert, or destroy a network. Wood and Stankovic have defined a DoS attack as an event that diminishes or attempts to reduce a network's capacity to perform its expected function. There are several standard techniques existing in the literature to cope with some of the more common denial of service attacks. Most of the defence mechanisms require high computational overhead and hence not suitable for resource constrained WSNs. Some of the important types of DoS attacks in WSNs are discussed below.

Physical Layer Attacks

The physical layer is responsible for frequency selection, carrier frequency generation, signal detection, modulation, and data encryption [1]. As with any radio-based medium, the possibility of jamming is inherent. In addition, nodes in WSNs may be deployed in hostile or insecure environments where an attacker has physical

access to the sensor node. The two types of attacks in physical layer are jamming and tampering.

Jamming is a type of attack which interferes with the radio frequencies range which is the normal operational frequency range of a sensor node. A jamming source may be powerful enough to disrupt the entire network. Even with less powerful jamming sources, an adversary can potentially disrupt communication in the entire network by strategically distributing the jamming sources. Even an intermittent jamming may prove detrimental as the message communication in a WSN may be extremely time-sensitive.

Tampering is a physical attack which causes irreversible damage to the nodes. Sensor networks typically operate in outdoor environments. Due to the unattended and distributed nature, the nodes in a WSN are highly susceptible to physical attacks. The adversary can extract cryptographic keys from the captured node, tamper with its circuitry, modify the program codes or even replace it with a malicious sensor [20]. Sensor nodes such as MICA2 motes can be compromised in less than one minute time .

Link Layer Attacks

The link layer is responsible for multiplexing of data-streams, data frame detection, medium access control, and error control [1]. Attacks at this layer include purposefully created collisions, resource exhaustion, and unfairness in allocation. A collision occurs when two nodes attempt to transmit on the same frequency simultaneously. When packets collide, they are discarded and need to be retransmitted. An adversary may strategically cause collisions in specific packets such as acknowledgement control messages. A possible result of such collisions is the costly exponential back-off.

The adversary may simply violate the communication protocol and continuously transmit messages in an attempt to generate collisions. Repeated collisions can also be carried out by an attacker to cause resource exhaustion. Unfairness is a weak form of DoS attack . An attacker may cause unfairness by intermittently using the above link layer attacks. In this case, the adversary causes degradation of real-time applications running on other nodes by intermittently disrupting their frame transmissions.

Network Layer Attacks

The network layer of WSNs is vulnerable to the different types of attacks such as: (i) spoofed routing information , (ii) selective packet forwarding, (iii) sinkhole, (iv) Sybil, (v) wormhole, (vi) hello flood, (vii) acknowledgement spoofing etc., These attacks are described briefly in the section below.

Spoofed routing information is an attack against a routing protocol to target the routing information in the network. An attacker may spoof, alter, or replay routing information to interrupt traffic in the network [29]. These interruptions include creation of routing loops, attracting or repelling network traffic from selected nodes, extending or shortening source routes, generating fake error messages, causing network partitioning, and increasing end-to-end latency.

Selective forwarding means an attacker may compromise a node in such a way that it selectively forwards some messages and drops the other ones. This is very difficult to detect since it randomly drops the packets [29].

Sinkhole Attack is an attack when an attacker makes a compromised node look more attractive to its neighbours by forging the routing information [29]. This results in the neighbour node

choosing the compromised node as the next-hop node to route their data through. This type of attack makes selective forwarding very simple as all traffic from a large area in the network would flow through the compromised node.

Sybil Attack is an attack where one node exhibits more than one identity in a network. It was originally described as an attack intended to defeat the objective of redundancy mechanisms in distributed data storage systems in peer-to- peer networks . In addition to defeating distributed data storage systems, the Sybil attack is also effective against routing algorithms, data aggregation, voting, fair resource allocation, and foiling misbehaviour detection. Regardless of the target (voting, routing, aggregation), the Sybil algorithm functions similarly. All the techniques involve utilizing multiple identities. For instance, in a sensor network voting scheme, the Sybil attack might utilize multiple identities to generate additional votes. Similarly, to attack the routing protocol, the Sybil attack would rely on a malicious node taking up the identity of multiple nodes, and thus routing multiple paths through a single malicious node.

Wormhole Attack is a low latency link between two portions of a network over which an attacker replays network messages [29]. This link may be established in two ways. The first method is where a single node forwards messages between two adjacent but otherwise non neighbouring nodes. Secondly, by a pair of nodes in different parts of the network which can communicate with each other. The latter case is closely related to sinkhole attack as an attacking node near the base station can provide a one-hop link to that base station via the other attacking node at a distant part of the network.

Hello flood attacker may use a high-powered transmitter to deceive a large number of nodes and make them believe that they are

within its neighbourhood [29]. Most of the protocols that use Hello packets make the assumption that receiving such a packet implies that the sender is within the radio range of the receiver. Subsequently, the attacker node falsely broadcasts a shorter route to the base station, and all the nodes which receive the Hello packets, attempt to transmit data to the attacker node. However, these nodes are out of the radio range of the attacker.

Acknowledgement Spoofing is an attack where the attacker node may overhear packet transmissions from its neighbouring nodes and spoof the acknowledgements, thereby providing false information to the nodes [29]. In this way, the attacker is able to disseminate wrong information about the status of the nodes.

Transport Layer Attacks

The attacks that can be launched on the transport layer in a WSN are flooding attack and de-synchronization attack.

Flooding Attack come into picture when a protocol is required to maintain state at either end of a connection. It becomes vulnerable to memory exhaustion through flooding [29]. An attacker may repeatedly make new connection request until the resources required by each connection are exhausted or reach a maximum limit. In either case, further legitimate requests will be ignored.

De-synchronization Attack refer to the disruption of an existing connection [29]. An attacker may, for example, repeatedly spoof messages to an end host causing the host to request the retransmission of missed frames. If timed correctly, an attacker may degrade or even prevent the ability of the end hosts to successfully exchange data causing them to waste energy attempting to recover

from errors which never exist in reality. The possible DoS attacks are listed in Table 1.

Layer	*Attacks*
Physical	1. Jamming 2. Tampering
Link	1. Collision 2.Exhaustion
	3.Unfairness
Network	1. Spoofed routing information
	2. Selective forwarding 3. Sinkhole
	4. Sybil 5. Wormhole
	6.Hello Flood 7. ACK flooding
Transport	1. Flooding
	2. De-synchronization

Table 1.1: Layer Wise DoS Attacks on WSNs

1.4.2 Attacks on Secrecy and Authentication

This type of attack concentrates on highly secure data transmission to get the valid information and tries to make a false authentication. Following is a list of attacks belonging to this category:

Node Replication Attack

An attacker attempts to add a node to an existing WSN by replicating (i.e. copying) the node identifier of an already existing node in the network. A Replicated node joined in the network in this manner can potentially cause severe disruption in message communication by corrupting and forwarding the packets through wrong routes. This

may also lead to network partitioning, communication of false sensor readings. In addition, if the attacker gains physical access to the entire network, it is possible for an attacker to copy the cryptographic keys and use these keys for message communication from the replicated node. The attacker can also place the replicated node in strategic locations in the network so that he could easily manipulate a specific segment of the network, possibly causing a network partitioning.

Attacks on Privacy

Since WSNs are capable of automatic data collection through efficient and strategic deployment of sensors, these networks are also vulnerable to potential abuse of the vast data sources. Privacy preservation of sensitive data in a WSN is particularly a difficult challenge [18]. Moreover, an adversary may gather seemingly innocuous data to derive sensitive information if he knows how to aggregate the data collected from multiple sensor nodes. The privacy preservation in WSNs is even more challenging since these networks make large volumes of information easily available through remote access mechanisms. Since the adversary need not be physically present to carry out the surveillance, the information gathering process can be done anonymously with a very low risk. In addition, remote access allows a single adversary to monitor multiple sites simultaneously. Following are some of the common attacks on sensor data privacy.

Eavesdropping and Passive monitoring is the most common and easiest form of attack on data privacy. If the messages are not protected by cryptographic mechanisms, the adversary could easily understand the content. Packets containing control information in a WSN convey more information than accessible through the location

server.Eavesdropping on these messages prove more effective for an adversary.

Traffic Analysis Attack is to analyse the traffic flow to get vital information. In order to make an effective attack on privacy, eavesdropping should be combined with traffic analysis. Through an effective analysis of traffic, an adversary can identify some sensor nodes with special roles and activities in a WSN. For example, a sudden increase in message communication between certain nodes signifies that those nodes have some specific activities and events to monitor. Deng et al have demonstrated two types of attacks that can identify the base station in a WSN without even underrating the contents of the packets being analyzed in traffic analysis [12].

Camouflage Attack is an adversary which may compromise a sensor node in a WSN and later on use that node to masquerade a normal node in the network. This camouflaged node may advertise false routing information and attract packets from other nodes for further forwarding. After the packets start arriving at the compromised node, it starts forwarding them to strategic nodes where privacy analysis on the packets may be carried out systematically. It may be noted from the above discussion that WSNs are vulnerable to a number of attacks at all layers of the TCP/IP protocol stack. There may be other types of attacks possible which are not yet identified. Securing a WSN against all these attacks may prove quite a cumbersome task.

1.5 Security Mechanisms for WSNs

Defence mechanism for combating various types of attacks on WSNs discussed in this section. Initially different cryptographic mechanisms

for WSNs are discussed, which include public key cryptography and symmetric key cryptographic techniques. A number of key management protocols for WSNs are discussed in the following section. Various methods of defending against DoS attacks, secure broadcasting mechanisms and various secure routing mechanisms are also detailed. In addition, various mechanisms for defending the Sybil attack, node replication attack, traffic analysis attacks, and attacks on sensor privacy are also presented. Finally, intrusion detection mechanisms for WSNs, secure data aggregation mechanisms and various trust management schemes for WSN security are deliberated upon.

1.5.1 Cryptography in WSNs

Selecting the most appropriate cryptographic method is vital in WSNs as all security services are ensured by cryptography. Cryptographic methods used in WSNs should meet the constraints of sensor nodes and be evaluated by code size, data size, processing time, and power consumption. In this section, the focus is on the selection of cryptographic algorithm in WSNs. First, the public key cryptography scheme is discussed, followed by symmetric key cryptography

Public key cryptography in WSNs

The WSN's limitations such as code size, data size, processing time, and power consumption makes it undesirable for public key algorithm techniques, such as Diffie-Hellman [36] or RSA signatures. Public key algorithms such as RSA are computationally intensive and usually execute thousands or even millions of multiplication instructions to perform a single-security operation. Further, a microprocessor's public

key algorithm efficiency is primarily determined by the number of clock cycles required to perform a multiplication instruction [10]. Brown et al found that public key algorithms such as RSA usually require on the order of tens of seconds and up to minutes to perform encryption and decryption operations in resource-constrained wireless devices, which exposes a vulnerability to DoS attacks. On the other hand, Carman et al found that it usually takes a microprocessor, thousands of nano-joules to perform a simple multiplication function with a 128-bit result [10]. On the contrary, symmetric key cryptographic algorithms and hash functions consume much less computational energy than public key algorithms [10].

Two of the major techniques used to implement public-key cryptographic systems are RSA and elliptic curve cryptography (ECC). These are too complex for use in wireless sensor networks. Recently, however, several groups have successfully implemented public-key cryptography (to varying degrees) in wireless sensor networks. Both RSA and elliptic curve cryptography are possible using 8-bit CPUs with ECC, demonstrating a performance advantage over RSA. The elliptic curve cryptography shows promise over RSA due to its higher efficiency compared to the private-key operations of RSA.

Symmetric key Cryptography in WSNs

Since most of the public key cryptographic mechanisms are computationally intensive, most of the research studies for WSNs focus on the use of symmetric key cryptographic techniques. Symmetric key cryptographic mechanisms use a single shared key between the two communicating hosts which is used for both encryption and decryption. However, a major challenge for

deployment of symmetric key cryptography is on how to securely distribute the shared key between the two communicating hosts. This is a non-trivial problem since pre-distributing the key may not be always feasible.

Recent studies on public key cryptography have demonstrated that public key operations may be practical in sensor networks. However, private key operations are still too expensive in terms of computation and energy cost of implementation in a sensor node. The application of private key operations to sensor nodes needs to be studied further. Symmetric key cryptography is superior to public key cryptography in terms of speed and low energy cost. However, the key distribution schemes based on symmetric key cryptography are not perfect. Efficient and flexible key distribution schemes need to be designed. It is also likely that more powerful motes need to be designed in order to support the increasing requirements on computation and communication in sensor nodes.

1.5.2 Key Management Protocols

Key management is yet another area actively studied upon by researchers. Key management is a core mechanism which ensures security in network services and applications in WSNs. The goal of key management is to establish the keys among the nodes in a secure and reliable manner. In addition, the key management scheme must support node addition and revocation in the network. Since the nodes in a WSN have computational and power constraints, the key management protocols for these networks must be extremely light-weight. Most of the key management protocols for WSNs are based on symmetric key cryptography because public key cryptographic techniques are in general computationally intensive. In

this section, a brief overview on some of the most important key management protocols is given.

Key Management Based on Network Structure

Depending on the underlying network structure, the key management protocols in WSNs may be classified as centralized or distributed. In a centralized key management scheme, there is only one entity that control the generation, re-generation, and distribution of keys. This entity is called key distribution centre (KDC). The main drawback of this scheme is its single point of failure. If the central controller fails, the entire network fails and the system is prone to security threats. Lack of scalability is another issue. Moreover, it does not provide data authentication. In the distributed key management protocols, different controllers are used to manage key-related activities. These protocols do not have the vulnerability of having a single point of failure and they allow better scalability.

Key Management on Probability of Key Sharing

The key management protocols for WSNs may be classified on the probability of key sharing between a pair of sensor nodes. Depending on this probability, the key management schemes may be either deterministic or probabilistic.

Deterministic key distribution schemes use a specific set of predetermined keys for encryption. The localized encryption and authentication protocol (LEAP) proposed by Zhu et al is a key management protocol for WSNs based on symmetric key algorithms. It uses different keying mechanisms for different packets depending on their security requirements. Four types of keys are established for each node: (i) an individual key shared with the base station

(pre-distributed), (ii) a group of key shared by all the nodes in the network (pre-distributed), (iii) pair-wise key shared with immediate neighbour nodes, and (iv) a cluster key shared with multiple neighbour nodes. Lai et al have proposed a broadcast session key (BROSK) negotiation protocol. BROSK assumes that a master key is shared by all the nodes in the network. To establish a session key with its neighbor node B, a sensor node A broadcasts a key negotiation message and both arrive at a shared session key.

Probabilistic key distribution schemes propose a random key pre-distribution scheme for WSNs that relies on probabilistic key sharing among nodes of a random graph. The mechanism has three phases: **key pre-distribution, shared key discovery, and path key establishment**.

In the key pre-distribution phase, each sensor is equipped with a key ring stored in its memory. The key ring consists of k keys which are randomly drawn from a large pool of P keys. The association information of the key identifiers in the key ring and sensor identifier is also stored at the base station. Each sensor node shares a pair-wise key with the base station. In the shared key discovery phase, each sensor discovers its neighbours with whom it shares the keys. The authors have suggested two methods for this purpose. The simplest method is for each node to broadcast a list of identifiers of the keys in their key rings in plain text, allowing neighbouring nodes to check whether they share a key. However, the adversary may observe the key-sharing patterns among sensors in this way.

The second method uses the challenge-response technique to hide key-sharing patterns among nodes from an adversary. Finally, in the path key establishment phase, a path key is assigned for those sensor nodes within the communication range which do not share a key, but connected by two or more links at the end of the second phase. If

a node is compromised, the base station can send a message to all other sensors to revoke the compromised node's key ring. Re-keying follows the same procedure as revocation. The messages from the base station are signed by the pair-wise key shared by the base station and sensor nodes, thus ensuring that no adversary can forge a station. The probability of an intruder attacking a compromised node is given by the following equation:

$$probability = \frac{k}{P}$$

1.5.3 Current Security Issues

The major security issues on a WSN are listed below. The current approaches need to consider these issues.

Memory

High security and lower overhead are two objectives that a key management protocol need to achieve. Strong security protocols usually require large amounts of memory cost, high-speed processors and large power consumption. However, they cannot be easily supported due to the constraints on hardware resources of the sensor platform. The ways to reduce memory cost without compromising on desired security levels is a key task.

End-to-end Security

The merit of symmetric key cryptography is its computational efficiency. End-to-end communication at the transport layer is very common in many WSN applications. For example, to reduce unnecessary traffic, a fusion node can aggregate reports from many

source nodes and forward a final report to the sink node. In hostile environments, however, any node can be compromised. If one of the intermediate nodes along a route is compromised, the message delivered along the route can be exposed or modified by the compromised node. Employing end-to-end security can effectively prevent message tampering by any malicious intermediate node.

Efficient Symmetric Key Algorithms

There is still a demand for the development of efficient symmetric key algorithms because encryption and authentication based on symmetric keys are frequent in the security operations of sensor nodes. For example, in the link layer security protocol TinySec, each packet must be authenticated.Encryption can be triggered if critical packets are transmitted. Therefore, fast and cost-efficient symmetric key algorithms need to be developed.

Key Update and Revocation

Once a key has been established between two nodes, the key can act as a master key and can be used to derive different sub-keys for many purposes. If each key is used for a long time, it may be exposed due to cryptanalysis over the cipher intercepted by adversaries. To protect the master key and sub-keys from cryptanalysis, it is wise to update keys periodically. The period of update, however, is difficult to choose. The cryptanalytic capability of adversary is unknown,hence it is very difficult to estimate how long it takes for the adversary to expose a key by cryptanalysis. If the key update period is too long, the corresponding key may also be exposed. If it is too short, frequent updates can incur large overhead.

Node Compromise

Node compromise is the most detrimental attack on sensor networks. This is due to the fact that compromised nodes have all the authentic key details, which can result in severe damage to WSN applications and cannot be easily detected. Most of the current security protocols attempt to defend the impact of node compromise through careful protocol design such that the impact of node compromise can be restricted to a small area. However, a hardware approach is more promising. With advances in hardware design and manufacturing techniques, much stronger, tamper-resistant, and cheaper devices can be installed on the sensor platform.

1.5.4 Defence Against DoS Attacks

Various types of **DoS** attacks in WSNs have been discussed in Section 1.3.1 . In this section, defence mechanisms for each of these attacks are presented in detail.

Defence in the Physical Layer

Jamming attack may be defended by employing variations of spread-spectrum communication such as frequency hopping and code spreading . Frequency-hopping spread spectrum (FHSS) is a method of transmitting signals by rapidly switching a carrier among many frequency channels using a pseudo-random sequence known to both the transmitter and the receiver. However, it requires greater design complexity and energy and thus not very suitable for WSNs. In general, to maintain low cost and low power requirements, sensor devices are limited to single-frequency use and are therefore highly susceptible to jamming attacks. One approach for tolerance against

jamming attack in a WSN is to identify the jammed part of the network and effectively avoid it by routing around. Wood and Stankovic have proposed an approach where the nodes along the perimeter of a jammed region report their status to the neighbors and collectively the affected region is identified and packets are routed around it.

Defence in the Link Layer

A typical defence against **Collision** attack is the use of error-correcting codes. Most codes work best with low levels of collisions such as those caused by environmental or probabilistic errors. However, these codes also add additional processing and communication overhead. A possible solution for energy exhaustion attack is to apply a rate limiting MAC admission control. This would allow the network to ignore the requests which intend to exhaust the energy reserves of a node. An alternate technique is to use time-division multiplexing where each node is allotted a time slot in which it can transmit. This eliminates the need for arbitration of each frame and can solve the indefinite postponement problem in a back-off algorithm. The effect of unfairness caused by an attacker by launching a link layer attack can be lessened. This is achieved by the use of small frames which reduces the amount of time an attacker gets at his disposal to capture the communication channel.

Defence in the Network Layer

A countermeasure against **Spoofing** attack is to append a message authentication code (MAC) after the message. By adding a MAC to the message, the receivers can verify whether the messages have been spoofed or altered. To defend against replayed information, counters or

time-stamps may be introduced in the messages. A possible defence against selective forwarding attack is to use multiple paths to send data [29]. A second defence mechanism is to detect the malicious node or assume it has failed and seek an alternative route.

Hu et al have proposed a novel and generic mechanism called packet leashes for detecting and defending against **Wormhole** attacks. In a wormhole attack, a malicious node eavesdrops on a series of packets, then tunnels them through a path in the network, and replays them. This is done in order to make a false implication of the distance between the two colliding nodes. It is also used, more generally, to disrupt the routing protocol by misleading the neighbour discovery process [29]. Wang and Bhargava have used a visualization approach to detect wormholes in a WSN. In the mechanism proposed by the authors, a distance estimation is made between all the sensor nodes in a neighbourhood. Using multi-dimensional scaling, a virtual layout of the network is computed, and a surface smoothing strategy is used to adjust the round-off errors. Finally, the shape of the resulting virtual network is analysed. If any wormhole exists, the shape of the network will bend and curve towards the wormhole, otherwise the network will appear flat.

To defend against **Flooding** DoS attack at the transport layer, Aura et al have proposed a mechanism using client puzzles [3]. The central idea is for each connecting client to demonstrate its commitment to the connection by solving a puzzle. As an attacker in most likelihood, does not have infinite resource, will not be able to create new connections fast enough to cause resource starvation on the serving node.

A possible defence against **De-synchronization** attack on the transport layer is to enforce a mandatory requirement of authentication of all packets communicated between the nodes. If the

authentication mechanism is secure, an attacker will be unable to send any spoofed messages to any destination node.

Defence Against Attacks on Routing Protocols

Many routing protocols have been proposed for WSNs. These protocols can be divided into three broad categories according to the network structure: (i) flat-based routing, (ii) hierarchical-based routing, and (iii) location-based routing. In flat-based routing, all nodes are typically assigned equal roles or functionality. In hierarchical-based routing, nodes play different roles in the network. In location-based routing, sensor node positions are used to route data in the network. One common location-based routing protocol is GPSR. It allows nodes to send packets to a region rather than a particular node. The goal of a secure routing protocol for a WSN is to ensure the integrity, authentication, and availability of messages.

SPINS includes two building blocks: (i) secure network encryption protocol (SNEP) and (ii) micro version of timed efficient streaming loss-tolerant authentication protocol μTESLA. SNEP provides data confidentiality, two-party data authentication, and data freshness for peer-to-peer communication (node to base station) and μ TESLA ensures authentication. SPINS assume that each node is pre-distributed with a master key K which is shared with the base station at its time of creation. All the other keys, including a key Kencr for encryption, a key Kmac for MAC generation, and a key Krand for random number generation are derived from the master key using a one-way string function. SPINS uses RC5 protocol for confidentiality.

SNEP provides the following properties, 1. Semantic security: the counter value is incremented after each message and thus the

same message is encrypted differently each time. 2. Data authentication: a receiver can be assured that the message is originated from the claimed sender if the MAC verification produces positive results. 3. Replay protection: the counter value in the MAC prevents replaying old messages by an adversary. 4. Weak freshness: Weak freshness provides partial message ordering and carries no delay information. 5. Low communication overhead: the counter state is kept at each endpoint and need not be send in each message [19].

μTESLA (the micro version of the Timed, Efficient, Streaming, Loss-tolerant Authentication protocol) and its extensions [32] [33] have been proposed to provide broadcast authentication for sensor networks. μTESLA is broadcast authentication protocol which was proposed by Perrig et al for the SPINS protocol [30]. μTESLA introduces asymmetry through a delayed disclosure of symmetric keys resulting in an efficient broadcast authentication scheme. For the operation, it requires the base station and the sensor nodes to be loosely synchronized. In addition, each node must know an upper bound on the maximum synchronization error.

Defence Against the Sybil Attack

Any defence mechanism against the Sybil attack must ensure that a framework must be in place to validate that a sole identity is held by a given physical node [42]. Newsome et al primarily describe direct validation techniques, including a radio resource test. In the radio test, a node assigns each of its neighbours a different channel and listens to each of them. If the node detects a transmission on the channel, it is assumed that the node transmitting on the channel is a physical node. Similarly, if the node does not detect a transmission on the specified channel, it assumes that the identity assigned to the channel is not a

physical identity. Another technique to defend the Sybil attack is to use random key pre-distribution techniques.

In random key pre-distribution, a random set of keys or key-related information are assigned to each sensor nodes, so that in the key set-up phase, each node can discover or compute the common keys it share with its neighbours. The common keys are used as shared secret session keys to ensure node-to-node secrecy. Newsome et al have proposed that the identity of each node is associated with the keys assigned to the node [42]. With a limited set of captured keys, there is little probability that an arbitrarily generated identity will work.

Detection of Node Replication Attack

Parno, Perrig and Gligor have proposed a mechanism for distributed detection of node replication attacks in WSNs. The two algorithms for node replication detection are: (i) randomized multicast and (ii) line-selected multicast. The randomized multicast algorithm distributes location information of a node to randomly-selected witnesses, exploiting birthday paradox to detect replicated nodes. The line-selected multicast uses the network topology to detect replication as discussed below. The randomized broadcast has evolved from traditional node-to-node broadcasting. In traditional node-to-node broadcasting, each node in the network uses an authenticated broadcast message to flood the network with its location information.

Defence Against Traffic Analysis Attack

Deng, Han and Mishra have proposed a mechanism for defending traffic analysis attack in WSN [12]. The authors have argued that since the base station is a central point of failure, once the location

of the base station is discovered, an adversary can disable or destroy the base station, thereby rendering the data-gathering functionality of the entire WSN ineffective. The mechanism proposed by the authors prevent rate monitoring attack and time correlation attack. The methodology involves four techniques. First, a multiple parent routing scheme is introduced which allows a sensor node to forward a packet to one of its multiple parents. This makes the patterns less pronounced in terms of routing packets towards the base station. Second, a controlled random walk is introduced into the multi-hop path traversed by a packet through the WSN towards the base station. This distributes packet traffic, thereby rendering the rate monitoring attack less effective. Third, random fake paths are introduced to confuse an adversary from tracking a packet as it moves towards a base station. This mitigates the effectiveness of time correlation attacks. Finally, multiple, random areas of high communication activities are created to deceive an adversary as to the true location of the base station, which further increases the difficulty of rate monitoring attacks. The combination of these four strategies make the proposed mechanism extremely robust to any traffic analysis attack.

Defence Against Attacks on Sensor Privacy

A number of mechanisms have been proposed for protecting information privacy in WSNs. They are listed below.

Anonymity mechanisms : Precise location information enable accurate identification of a user. This is a serious threat to privacy. One way to handle this problem is to make data source anonymous. An anonymity mechanism depersonalizes the data before it is released from the source. Grusterand and Grunwald have presented

an analysis on the feasibility of anonymizing location information in location-based services in an automotive telematics environment [17]. Beresford and Stajano have proposed anonymity techniques for an indoor location system based on the Active Bat. Since ensuring total anonymity is almost an impossible proposition, in almost all practical scenarios, a trade off is to be made between anonymity and disclosure of public information in most of the privacy protection mechanisms.

Policy-based approaches : In policy-based defence mechanisms, the access control decisions and authentication techniques are made on the basis of a specified set of privacy policies. Molnar and Wagner have presented the concept of private authentication and demonstrated its application in radio frequency identification (RFID) domain [40]. Duri et al have proposed a policy-based framework for protecting sensor information, where a computer inside a car acts as a trusted agent for location privacy [13].

Information flooding Xi, Schwiebert and Shi have described a successful attack on the flooding-based phantom routing. The authors have also proposed greedy random walk (GROW) protocol, a two-way random walk, i.e., from both source and sink, to reduce the chance of eavesdropper collecting the location information. In the proposed mechanism, the sink first initiates an N-hop random walk, and the source then initiates an M-hop random walk. Once the source packet reaches an intersection of these two paths, it is forwarded through the path created by the sink. Local broadcasting is used to detect when the two paths intersect. In order to minimize the chance of backtracking along the random walk, the nodes are stored in a bloom filter as the walk progresses. At each stage, the intermediate nodes are checked against the bloom filter to ensure that the backtracking is minimized.

Intrusion Detection

An intrusion detection system (IDS) monitors a host or network for suspicious activity patterns outside normal and expected behaviour . It is based on the assumption that there exists a noticeable difference in the behaviour of an intruder and legitimate user in the network such that an IDS can match those pre-programmed or possible learned rules. Based on the analysis model used for analysing the audit data to detect intrusions, intrusion detection systems are usually classified into two types: (i) Rule-based intrusion detection systems and (ii) Anomaly-based intrusion detection systems . Rule-based intrusion detection systems are used to detect known patterns of intrusions The anomaly-based systems are used to detect new or unknown intrusions. Rule-based IDS has a low false-alarm rate compared to an anomaly-based system, and an anomaly-based IDS has a high intrusion detection rate in comparison to a rule-based system.

1.6 Motivation, Objective and Scope

1.6.1 Motivation

The previous section gave an overview of the common type of attacks and defence mechanism against these attacks. This section describes the motivation behind selecting the wireless sensor network security as my research problem. WSN is resource constrained, distributed wireless network used for monitoring applications. After deployment, the nodes self-configure to form the network. The WSN is used in many low cost unattended applications like Monitoring, Home and hospital automation and Mission critical applications. The

major concern apart from the power and computational limitations of the WSN is the security of data and authenticity of communication. The Wireless sensor nodes can be deployed to monitor the area and periodically report the sensor readings. The opponents may try to attack the unattended wireless sensor network deployed by home side to find the secret information and tarnish the application. The major attacks include

- **Jamming attacks** to disturb the wireless communication
- **Privacy attacks** by listening to sensitive packets.
- **Denial of service attacks** like packet drop, flooding packets and injecting packets by an attacker mote or compromised mote.
- **Traffic analysis attacks** to locate the base station.

There are various approaches to make the application more secure, but most of them failed to address limitations like power and battery charge for wireless sensor network. The prime motivation is to develop a user friendly solution to the problem without adding much overhead to the existing system.

The WSN motes are deployed in an ad hoc manner for the purpose of monitoring. It is very difficult to secure WSN due to the unattended nature. The unmanned vehicle can navigate the network to change the static behaviour of the network. The vehicle mote can collect information from each nook and corner in the network, preventing various types of attacks. Military relevance is a major application of wireless sensor network. The motes are deployed in the area which is controlled by a base station. The vehicle controlled by Wireless sensor network is equipped with sensors and cameras. This can be used as a monitoring tool to track enemy movements. The

vehicle and base station can exchange messages between them in the following way: the vehicle can send data signals to the base station and the base station can in turn send control messages back to the vehicle node. The military movements can be coordinated by the data received from the driver motes. The enemy's movement information can help in planning the next step. On receiving sensitive data indicating an intrusion, the vehicle can be transported to that location for detailed analysis.

1.6.2 Objective

The conventional cryptographic approaches involve high computational cost and powerful processing unit for the implementation. But the sensor nodes have limited power, processing capacity and memory which makes it difficult to implement the conventional cryptographic solutions. For mitigating these issues, a potential security solution is proposed to secure a wireless network. This is a low cost solution which can secure the wireless network from different type of attacks. The key motivation is to build a secure architecture which completely implements clustering, key-management, encryption and intrusion detection to safeguard the wireless sensor network. Though implemented on wireless sensor networks, it is applicable to Internet of Things with limited processing power and memory. The aim of this research is to ensure network security without additional computational and communication overhead. The unmanned vehicle navigation is incorporated with security architecture to resolve the unattended and resource constrained nature of the network. The research work is divided into three major sections.

- Implementing tiny Vehicle navigation controlled by WSN

- Implementing an Intrusion detection system using vehicle navigation in WSN
- Implementing a Guarding Architecture for WSN security

The remaining chapters explain in detail on how the objectives are achieved. An application level example scenario is also presented along-with.

1.6.3 Scope

The wireless sensor network is an emerging technology. The work done has resulted in a Vehicle navigation system controlled by wireless sensor network. The main objective was to strengthen the security of the wireless sensor network and to protect it from various type of attacks. The general approach Guarding Architecture for Unattended Deployment Applications (GARUDA) can be customized based on hardware and application. It can also act as an additional security layer along with existing intrusion detection system to improve the performance of the system. The system has various applications such as military, monitoring, maintenance and fire detection. The vehicle is equipped with GPS which helps find the exact location. The implementation was done on a small car with five functionalities. Based on the application, the capabilities of the vehicle and the mote programming will change. The future scope of this work is to implement the system a real world application scenario. The GARUDA system provides a guarding architecture against the common security issues in Ad hoc Networks. The vehicle node provides a mobile nature to the guarding architecture. The vehicle can be used to address other issues such as localization and

time synchronization along with providing overall security to the network.

Chapter 2

Wireless Sensor Network Controlled Vehicle Navigation System and It's Applications

2.1 Overview

Wireless sensor networks (WSN), frequently referred to as wireless sensor and actuator networks (WSAN), are distributed spatially and are autonomous sensors which monitor physical or environmental conditions including temperature, pressure, sound, etc., and cooperatively pass their data along the network to a main location[1, 65]. The evolution of wireless sensor networks has been marked as a remarkable feat in the golden age of computer advancement. A wide variety of sensors are used in WSN based on

the application. For example, infrared sensors are used to detect events like human motion and thermistor sensor is used to determine the temperature [45]. These sensor nodes are equipped with a radio to communicate with each other and to send data to a central computer where the data can be parsed and viewed.

This Chapter revolves around the central theme: the Vehicle Navigation system which is controlled by a wireless sensor network. The vehicle is capable of motion along a geographic region under surveillance. This introduces an element of dynamism to the otherwise static nature of the wireless sensor network. The wireless sensor network mote placed inside the vehicle is connected to the vehicle through an interface card. The mote is programmed to control all the vehicle movements using a hardware interface unit. The crossbow mote acts as a driver of the vehicle which is controlled by base station commands and the messages from the surrounding nodes. The vehicle navigates with the help of Location information and the readings of neighbour mote sensor. The selection of a Localization algorithm is based on the application for which the vehicle is being utilized. This chapter sheds light on how the vehicle navigation system can be used for military purpose. The implementation is done with the help of a toy car with five controls under the control of a Micaz mote. The mote programming is done using nesC language. The mote controlling the vehicle has three different inputs at each stage. Initially the data from the sensors is connected to the mote followed by the data from surrounding motes and finally the data from the base station. The mote takes into consideration the three different inputs and makes an appropriate decision at each stage.

The vehicle navigation inside the wireless sensor network is based on the location information of the vehicle. The importance of

localization is to relate the vehicle location with a local map or global map [38]. Various localization techniques can be used to determine the position of motes deployed [28]. If GPS is used, the global position can be obtained, with this position information the next movement of the vehicle can be determined [61]. An alternative is to consider the relative position based on the location of the surrounding motes. Initially, the WSN nodes self-configure to form a network and computes the relative position. The mote which is placed inside the vehicle utilizes the location information from the surrounding nodes and decides the next movement of the vehicle.

An important task of the vehicle mote is to make the vehicle navigate by using Location information or by the communication with Neighbour motes. If the mote knows the local map, it can freely navigate the vehicle to the destination. Another type of control is based on the neighbour mote's communication in which the driver mote decides the movement based on the information it acquires from the surrounding nodes or the base station [2]. With the help of the coordinate system,the insider mote decides the movement of the vehicle. At first, the X direction gap will be moved and followed by the Y direction gap. The GPS location or relative position is used to find the location of the vehicle. Based on the location, the node takes the next action.

2.2 Application

The Vehicle Controlled by Wireless Sensor Network has various applications. Here two major applications are considered. The two applications are Military application and Monitoring and Maintenance application.

2.2.1 Military Application

Military is a major application area of wireless sensor network. The motes are deployed in the area and a base station control the motes [64]. The vehicle controlled by Wireless sensor network can be used as a monitoring tool to monitor the enemy movements. The vehicle can be used to detect the movement by attaching sensors and a camera to it [60]. The vehicle senses the data and sends it to the Base station. The Base station gives direction to the Vehicle mote. In addition to this, the vehicle mote also receives the surrounding mote data to get information about enemy movements. The vehicle has direct communication with the base station [43]. It can be used to monitor movements on both sides. The military movements can be very well coordinated by obtaining the information from driver mote. The enemy's movement information helps plan the next step. If the sensors send sensitive data about intrusion, vehicle can be moved to that location for a detailed analysis.

Security is a serious concern in a Military environment [47]. It is very difficult to find whether a mote is compromised or not. Compromised nodes has to be found out by cross checking the readings sent by the node with the actual readings from the vehicle mote. If a node is detected to be compromised, the vehicle can destroy that node. The vehicle can self explode if it is captured by an enemy. As the number of vehicles is increased, the security is improved to a greater extent. If there are a number of vehicles, then the total area is to be divided into various clusters and each cluster is assigned to each vehicle. These vehicles can be used to detect mines and bombs. The vehicles can also be utilized in battle field surveillance and detection of Nuclear, Biological and Chemical attacks.

2.2.2 Monitoring and Maintenance

The vehicle controlled by Wireless sensor can be used for monitoring and maintenance of the WSN. It can be used to update the clock and location of the motes. The vehicle can move around the network to collect data from all the motes and the consolidated data is sent to the base station [31]. The vehicle can take readings from isolated nodes. After covering the entire area, the vehicle returns to the base station for the purpose of maintenance. The vehicle can be used to remove a dead mote or a node with low battery power. The replaced node will also be in the same location. The vehicle can be used to deploy mote in an area where no motes are present. If the sensors are mobile, we can use our maintainer to keep track of location and topology. If the vehicle is under an invasive attack, it will self destroy.

2.3 Vehicle Navigation System Components

The vehicle navigation system consists of a tiny Vehicle, Driver mote, Base station mote, and Surrounding motes. Each of the components are explained in the section below.

2.3.1 Vehicle

The vehicle is the main part of the system. The selection of the vehicle will depend up on the application for which it is used. For military application, a vehicle like a small tank can be used. For monitoring and maintenance application, a small utility vehicle can be used. The vehicle motion is controlled by a mote placed inside the vehicle [25]. The vehicle and the mote is connected through an interface card.

2.3.2 Driver Mote

This is a wireless sensor network mote placed inside the vehicle. The mote is connected to the vehicle with the help of an interface card. The mote is connected with sensors based on the application [51]. The mote is connected with a GPS device to obtain the location. The mote is able to receive messages from the base station mote and the surrounding motes. The vehicle motion is controlled by the driver mote. The decision of the driver mote depends upon the Base station and the surrounding nodes. Driver mote is programmed in such a way to produce the output signals that depends on the input of base station nodes, surrounding nodes and the sensors connected to the motes. The output signal is passed on to the vehicle through an interface card. Corresponding to each signal from the mote, a vehicle operation is mapped.

The driver node gets the exact location of the vehicle from GPS connected with the mote [27]. In the absence of GPS, driver mote uses the relative position. The Vehicle movement to destination is based on the program running in the driver mote. The mote is responsible for controlling the vehicle navigation. Apart from navigation, driver mote can control the functionalities like deploying the motes and monitoring the battle field.

2.3.3 Base Station Mote

Base station mote has powerful features compared to the other motes. These include powerful batteries, powerful communication system and high computational capacity. Base station node controls the entire network. Base station mote has the global data collected from all the nodes [62]. Based on the data collected from the entire network, the mote controls the driver mote. The driver mote is connected with

the base station mote directly or indirectly. If the base station mote gets information from a mote about a malicious activity, it passes the information to the driver node. The driver node will move the vehicle to the location and performs a detailed analysis of the situation and take the necessary action.

2.3.4 Surrounding Motes

These are the motes surrounding the vehicle and the driver mote. The motes are connected with sensors and the sensors collect the data and sensitive information, which is passed to the driver mote. The driver mote analyses the situation and before taking a step, it analyses the data of the surrounding nodes and takes an appropriate decision. The surrounding nodes use the sensors to get the information and analyse the data to find relevant information. If it has already obtained relevant information it may be passed to the driver mote. In the absence of a GPS, the surrounding motes provide relative location information to the driver mote.

2.4 Implementation Details

In this section, the software and hardware implementation of the vehicle control system is discussed. The software parts are mainly coded in NesC and is implemented in Tiny Os[16]. The hardware part deals with the vehicle component, Motes, Sensors and the interface part.

2.4.1 Hardware Implementation

In the vehicle part, a toy car with five functions is used. The car is controlled by a mote through an interface card. The interface card and the mote are placed inside the car. A sensor board is placed inside the vehicle in order to sense the changes in the vehicle surroundings[56].

2.4.1.1 Car

A tiny Car is used as the vehicle component in the vehicle navigation system[49]. The car is equipped with five functions: Forward, Reverse, Left, Right, and Stop. The car components are explained in detail in the sections below

2.4.1.1.1 Forward and Reverse Motor The CD motor is used as forward and reverse motor. This motor is connected with the back wheels and as the battery power is given, the motor and the back wheels run forward resulting in the forward movement of the car. As the polarity of the battery connected with the motor is reversed, the motor runs in the opposite direction. The wheels also run in opposite direction. Thus the car moves in the reverse direction.

2.4.1.1.2 Stepper Motor The stepper motor is used for turning purpose. As the front wheels need to be turned, they are connected with the stepper motor.

2.4.1.1.3 Transmitter Circuit The core of the unit is the CAR transmitter circuit. The mote is connected to this circuit. The mote interface is done through the MDA320CA board. The mote generates the output based on its program. Through MDA320CA interface board, the signals are passed to the transmitter circuit. The primary

use of the transmitter circuit is to convert the signals from the mote to control the signals for the car. For this purpose, IC TX-2B is used.[39]. TX-2B is designed with five functionalities. Figure 2.1 shows the TX-2B circuit diagram. The TX-2B/RX-2B is a pair of CMOS LSIs and is designed for remote controlled car applications. The TX-2B/RX-2B has five control keys for controlling the motions (i.e., forward, backward, rightward, leftward and the stop function) of the remote controlled car.

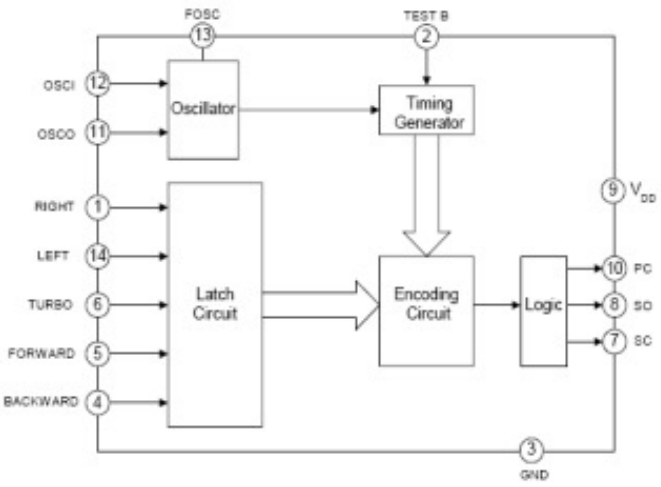

Figure 2.1: TX-2B IC[39]

Figure 2.2 shows the circuit diagram for the transmitter part of the Car. The TX-2B is connected to the output of the mote. Based on the output of the mote, the corresponding switch is enabled in the intermediate circuit. Once the switch is enabled, the respective function is initiated and the transmitter generates signals suitable for

each function. These signals are converted as radio signals which are then sent to the receiving part.

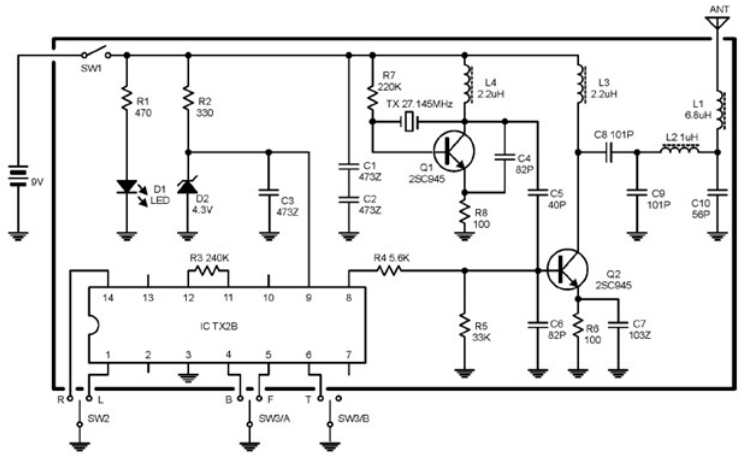

Figure 2.2: Transmitter Circuit

2.4.1.1.4 Receiver Circuit In the receiver unit, the signals are received and are passed on to the receiver. The receiver acts according to the corresponding actions. Figure 2.3 shows the circuit diagram of the receiver. At the receiver side, an RX-2B IC is used as the receiver IC. The received signals are then passed on to the IC. Corresponding to each signal, there is a function associated with RX-2B[39]. This IC is connected to two motors. As shown in the circuit diagram, the forward and reverse pins are connected to the CD motor. The stepper motor is in turn connected to the right and left pin. Based on the signal from the transmitter, the appropriate pin is made active. The forward pin runs CD motor in the forward direction and the vehicle moves in the forward direction. The reverse pin runs CD motor in the

reverse direction and the vehicle moves in the reverse direction. The left pin turns the stepper motor in left direction. The right pin turns the stepper motor in right direction. Only a single pin is active at any instant of time.

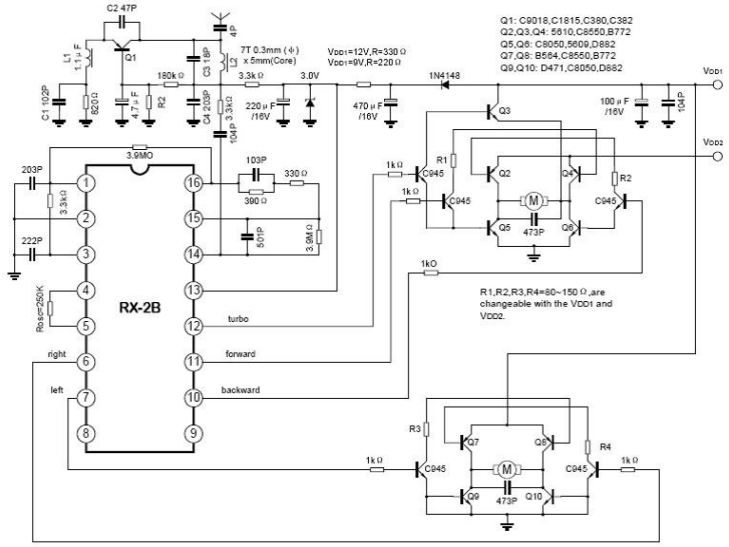

Figure 2.3: Receiver Circuit

2.4.1.2 Motes

The motes used for the implementation is MPR2400 (Micaz). This mote is used as both the driver mote and surrounding mote.

2.4.1.2.1 MPR2400 (Micaz) The Micaz is the mote from Crossbow Technology. The MPR2400 (2400 MHz to 2483.5 MHz band) uses the Chipcon CC2420, IEEE 802.15.4 compliant, ZigBee

ready radio frequency transceiver integrated with an Atmega128L micro-controller[26]. Micaz uses 51 pin I/O connector, and serial flash memory. All MICA2 application software and sensor boards are compatible with the MPR2400.

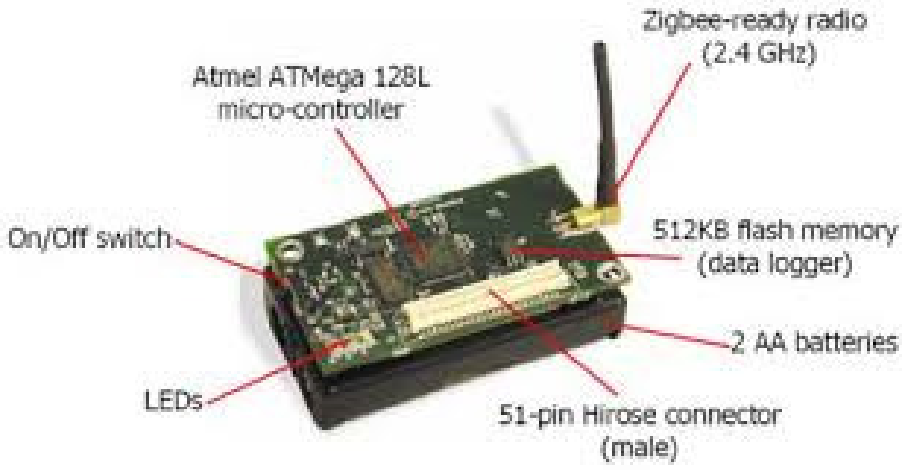

Figure 2.4: MPR2400 (MICAz)[26]

The parts of the MPR2400 mote are given below.

1. Atmel Atmega processor which is connected with flash memory.
2. A radio transceiver in order to send/receive the message.
3. An antenna to propagate the message.
4. 51 pin expansion connector to connect the external devices like Sensor and Interface card.
5. Three Led's are used to display the states.
6. Atmel Atmega battery is used as power source.
7. 512 KB flash memory
8. On/Off switch

2.4.1.3 Interface Card

The data acquisition board is used as the interface card between the Mote and vehicle transmitter unit. The MDA320CA is used as the interface card.

2.4.1.3.1 MDA320CA The MDA320CA is a high-performance data acquisition board with up to 8 channels containing 16-bit analog input. It combines a reduced feature set with the same versatile functionality found in Crossbow's popular MDA300 data acquisition board. This board is designed for use in cost-sensitive applications requiring precise data collection and analysis. With improved micro-terminal connections, the MDA320CA offers users a rapid and convenient interface to a wide variety of discrete external sensing devices. Data logging and display is supported via Crossbow's MoteView user interface. Crossbow's MoteView software is designed to be the primary interface between a user and a deployed network of wireless sensors. MoteView provides an intuitive user interface to database management along with sensor data visualization and analysis tools. Sensor data can be logged to a database residing on a host PC, or to a database running autonomously on a Stargate gateway.

2.4.1.4 Sensor Board

The MTS series of sensor boards and MDA series of sensor/data acquisition boards are designed to interface with Crossbow's MICA, MICA2, and MICA2DOT family of wireless Motes[21]. The sensor board used in this work are MTS101CA and MTS320 CA.

Figure 2.5: MDA320CA

2.4.1.4.1 MTS101CA The MTS101CA series sensor boards have a precision thermistor, a light sensor and a general prototyping area. The prototyping area supports connection to five channels of the Mote's analog to digital converter (ADC3-7) and the I2C digital communications bus.

2.4.1.4.2 MTS320CA The MTS320CA is flexible sensor board with a variety of sensing modalities. The sensing capabilities include light, temperature, microphone, buzzer, accelerometer and magnetometer.

2.4.2 Software Implementation

TinyOS has been used as the Operating System for Implementation purpose. TinyOS is an open-source event-driven "real-time" operating system designed by U.C. Berkeley. TinyOS is designed for

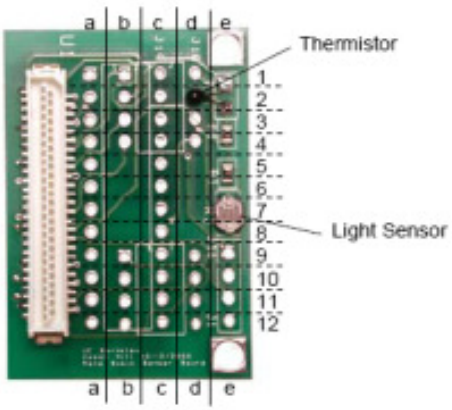

Figure 2.6: MTS101CA

Figure 2.7: MTS320CA

use in low-power/limited resource applications which utilize wireless embedded sensor networks [4]. The programming of mote is done in a C-based language, known as nesC [16]. NesC program in the driver mote attends the messages from the base station and surrounding motes and decide the next movement or action. The driver mote is programmed to navigate the vehicle to a particular location. The surrounding motes are programmed to report the incidents in its premises. The base station is programmed to analyse the sensor data and give commands to the driver mote.

2.4.2.1 TinyOS

TinyOS is an open-source event-driven real-time operating system designed by U.C. Berkeley. TinyOS is designed for use in low-power/limited resource applications which utilize wireless embedded sensor networks [4]. TinyOS is a component based operating system which minimizes the code size and power consumption. Components which are not used are not included in the compiled program. Also, the components are initially turned off which assist in reducing the power consumption. TinyOS can reside on a multiple platform, each of which supports different sensor boards. All Micaz applications which use the standard means of communication utilize the Micaz radio stack. The standard means of communication is defined as direct or indirect use of the genericComm or genericCommPromiscuous component.

Wireless sensor networks show their potential when they exist in an ad-hoc network environment. An ad-hoc wireless sensor network is a self-forming autonomous network of sensors which allows nodes to be beyond direct, single hop, communication distance from the base station. Nodes can easily be moved, removed, or added to the

network with minimal impact. With ad-hoc networks, the potential applications for wireless sensor networks grow substantially. Unfortunately, ad-hoc network operation in a power and memory constrained environment creates a limitation on throughput from any given node. The risk of flooding a network becomes higher, and in order to address this problem throughput must be limited by reducing the number of messages sent by any given node. TinyOS provides an ad-hoc routing component known as Multi-hop routing. The Multi-hop routing scheme organizes all the nodes, within communication range, into a routing tree. The root node in the routing tree is the base station node. Other nodes are placed in the tree based on their proximity to the root node and the quality of their link with the other nodes. Multi-hop routing scheme is a collection based routing scheme which has both pros and cons. The Multi-hop routing module can easily be integrated into any application and, as its name implies, provides for a much more expansive range for data collection.

There are three methods for power management: the user application directly handles power management, the user application enables power management and controls when the mote can enter the power saving mode, the application enables power management and lets the operating system control power management. In the first method the user application need to determine the sleep level to enter and the time to enter the sleep mode. Directly handling the power management necessitates the user application to have an intrinsic knowledge of the micro controller and have a direct control on the hardware. The second method utilizes a component of the operating system to handle power management decisions, but requires the user application to issue the sleep command. Although the second method abstracts some of the hardware aspects such as

device registers and the devices, it still requires the user application to initiate the sleep command. This method still circumvents one of the components of an operating system, namely scheduling. The third method not only utilizes a component provided by the operating system, but also allows the scheduler to initiate the sleep mode when there are no pending events or tasks. TinyOS provides a power management component PowerManagement and the HPLPowerManagementM. The power management component is disabled by default; this is no indication that the micro-controller is not allowed to sleep. The mote may enter the default sleep level, IDLE, when the sleep command is issued by the scheduler.

2.4.2.2 NesC Program

The TinyOS system, libraries, and applications are written in nesC, a new language for programming structured component-based applications. The nesC language is primarily intended for embedded systems such as sensor networks. NesC has a C-like syntax, but supports the TinyOS concurrency model, as well as mechanisms for structuring, naming, and linking together software components into robust network embedded systems [16]. The principal goal is to allow application designers to build components that can be easily composed into complete, concurrent systems, and yet perform extensive checking at compile time.

NesC provides support for tasks, events, and commands, as well as standard C. Tasks are typically posted in response to an event, and cannot pre-empt one another, but can be pre-empted by other events. Events are run in response to a hardware interrupt or signalled by a component. Unlike tasks, events can pre-empt one another. Commands are called via other components, and run in the

current execution thread. Two execution threads exist, the task execution thread and the hardware event handler execution thread. NesC checks for potential data races which result from this concurrency model. NesC provides for interfaces and components. Interfaces provide the only means of communication between components. Interfaces consist of commands and events. A component can be either a configuration or a configuration and a module. Configurations "wire up" components and, if it exists, the main code module create a new component or application. Modules consist of code which is executed in response to an event. The main application typically consists of a configuration and a module. A component can provide and use multiple interfaces. Modules provide application code, implementing one or more interfaces. Configurations are used to assemble other components together, connecting interfaces used by components to interfaces provided by others. This is called wiring. Every nesC application is described by a top-level configuration that wires together the components inside. NesC uses the file name extension ".nc" for all source files, interfaces, modules, and configurations.

TinyOS executes only one program consisting of selected system components and custom components needed for a single application. There are two threads of execution: tasks and hardware event handlers. Tasks are functions whose execution is deferred. Once scheduled, they run to completion and do not pre-empt one another. Hardware event handlers are executed in response to a hardware interrupt and also runs to completion, but may pre-empt the execution of a task or other hardware event handlers. Commands and events that are executed as part of a hardware event handler must be declared with the async keyword. Tasks and hardware event handlers may be pre-empted by other asynchronous code, nesC

programs are susceptible to certain race conditions. Races are avoided either by accessing shared data exclusively within tasks, or by having all accesses within atomic statements. The nesC compiler reports potential data races to the programmer at compile-time. It is possible that the compiler may report a false positive. In this case a variable can be declared with the no-race keyword. The no-race keyword should be used with extreme caution.

Interfaces are bidirectional: they specify a set of functions to be implemented by the interface provider (commands) and a set to be implemented by the interface user (events). This allows a single interface to represent a complex interaction between components. This is critical because all lengthy commands in TinyOS (e.g. send packet) are non-blocking; their completion is signalled through an event (send done). By specifying interfaces, a component cannot call the send command unless it provides an implementation of the sendDone event. Typically commands call downwards, i.e., from application components to those closer to the hardware, while events call upwards. Certain primitive events are bound to hardware interrupts.

Components are statically linked to each other via their interfaces. This increases the run time efficiency, encourages robust design, and allows for better static analysis of programs. NesC is designed under the expectation that the code will be generated by whole-program compilers. This allows for better code generation and analysis. An example of this is nesC's compile-time data race detector. The concurrency model of nesC is based on run-to-completion tasks, and interrupt handlers which may interrupt tasks and each other. The nesC compiler signals the potential data races caused by the interrupt handlers.

2.5 Security In Military Application using Unmanned Vehicle

Wireless sensor networks help in military operations by delivering critical information rapidly and dependably to the right individual or organization at the right time, thereby significantly improving the efficiency of combat operations. Wireless Sensor Network is widely used in Military Applications like Tracking the enemy movements and force protection. Security is the major concern in this application and is very difficult to achieve due to the unattended nature, limited memory and limited power of network. An unmanned vehicle is incorporated in the military application to make the system more secure and improve the life time and connectivity of the network. The vehicle is equipped with a mote, and controlled by commands from a Base Station and data from neighbouring nodes.

This section describes the implementation of the secure unmanned Vehicle Navigation system controlled by wireless sensor network for military application. A cluster based approach is used to prevent the various types of attacks in a military field. An armed and sealed mote is proposed to prevent the physical attacks. In order to avoid a single key compromise lead to the entire network compromise, a modified version of LEAP for key management is proposed. The vehicle can directly send the messages to the nearest sink node. The vehicle is utilized for making the application more secure and all the costly devices needed for the application can be incorporated with the vehicle. This section demonstrates how the unmanned vehicle navigation system is capable to be used in Military applications. The section briefly explains how the vehicle is used to detect the intruders and diminish the various security threats in a military field.

The Wireless Sensor Network is used for monitoring and tracking application in military field. Security is a major concern as far as a military scenario is concerned. In the first section applications of WSN in military field is described. The second section gives an overview about the different types of attacks possible in a military network. The third section deals with how the unmanned vehicle is able to protect the network from common types of attacks in military application. In the fourth section the security architecture to protect the military application is discussed. Various techniques are used to mislead the attacker from getting the information. The system can be used for monitoring and protecting the military forces from intruders.A solution has been proposed against the major types of attacks in a military field. The unmanned vehicle can be used to check the reliability and integrity of the network.

2.5.1 Military Field Monitoring using WSN

Military security is a major application area of wireless sensor network. The wireless sensor networks deployed in military field can monitor the enemy movements and coordinate the activities of the army. Fig 2.8 illustrates the motes being deployed in the area to be monitored and a base station which is used to collect information from various motes. The motes are connected with sensors to sense the environment for detection of enemy movements and to coordinate the military activity. The motes connected with sensors looks for particular events and give periodical messages to the base station. In case of suspicious activities, the motes immediately send messages to the base station. The base station receives the information from various motes and take the necessary action. The actions include: informing the command in charge for that region,

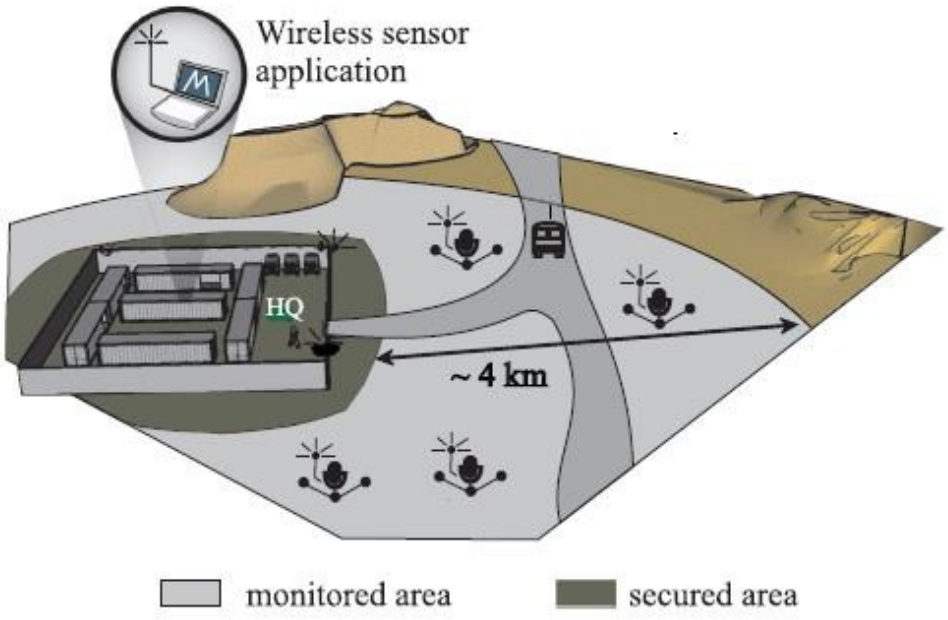

Figure 2.8: Wireless Sensor network based military monitoring [47]

give messages to motes surrounding the area. The base station is set up in a safe area and motes are deployed in an area to be monitored surrounding the base station.

Security is a major concern in military WSN applications. Due to limitations like limited power and low computation, conventional ways like cryptography failed in providing security to the network. The left-alone nature of the network makes it difficult to protect the network from various attacks. The enemy movement information helps to plan the next step. The motes are connected with various sensors to sense the physical values. Based on the sensed data, the mote can

be programmed to report vital information to the base station. Base station analyses the data and takes corresponding action to secure the military network.

2.5.2 Security Issues in Military Field

Security is a major concern in the military field because if the intruder or compromised node gets the secret information it may be passed on to the enemy. The various types of attacks against the military application are listed below.

2.5.2.1 Denial of Service Attack

A very common attack on Military wireless sensor networks is simply to jam a node or set of nodes by the intruder node. Jamming, in this case, is simply the transmission of a radio signal that interferes with the radio frequencies being used by the sensor network. Another form of attack is when a malicious node continuously transmit messages in an attempt to cause collisions. The intruder node can also drop some messages. The above attack will lead to retransmission of packets.

2.5.2.2 Sybil Attack

The Sybil attack is done by an intruder mote or device when it takes on multiple identities. It is originally described as an attack which is able to defeat the redundancy mechanisms of distributed data storage systems in peer-to-peer networks. In military application, a single malicious node can send data about an imaginary event multiple times as different entities. This creates a trust that the event has actually occurred. This node can compromise the entire network and is very difficult to be identified.

2.5.2.3 Traffic Analysis Attack

Wireless sensor networks in military application are typically composed of many low-power sensors communicating with a few relatively robust and powerful base stations. If the enemy get information about the base station they can simply disable it and the entire network becomes useless. A rate monitoring attack simply makes use of the idea that the nodes closest to the base station tend to forward more packets than those farther away from the base station. An attacker only needs to monitor which nodes are sending packets and follow those nodes that are sending more packets. In a time correlation attack, an adversary simply generates events and monitors to whom a node sends its packets.

2.5.2.4 Node Replication Attack

An attacker seeks to add a node to an existing sensor network by replicating the node ID of an existing sensor node. This node can get the cryptographic keys and secret messages passing through the network. It can drop the packet and disconnect a section of the network from the whole network. If the enemy can introduce more number of such nodes, they can control the whole network.

2.5.2.5 Attacks Against Privacy

Monitor and Eavesdropping is the most obvious attack to privacy. By listening to the data, the adversary could easily discover the communication contents. The traffic conveys the control information about the sensor network configuration, which may contain location information of the nodes in the network. If an adversary node gets this information it can use the information to estimate the position

of critical areas. Through the analysis on the traffic, some sensors with special roles or activities can be effectively identified. Adversaries can insert their node or compromise the nodes to hide the sensor network to get the secret information like a spy in military application.

2.5.2.6 Physical Attack

Military sensor networks typically operate in hostile outdoor environments. In such environments, the small form factor of the sensors, coupled with the unattended and distributed nature of their deployment make them highly susceptible to physical attacks, i.e., threats due to physical node destruction. Unlike many other attacks mentioned above, physical attacks destroy sensors permanently, so the loss is irreversible. For instance, attackers can extract cryptographic secrets, tamper with the associated circuitry, modify programming in the sensors, or replace them with malicious sensors under the control of the attacker.

2.5.3 Unmanned Vehicle to Protect The Network

The unmanned vehicle controlled by WSN can be utilized for Military application. This vehicle can navigate through the network and monitor the network security. Since the vehicle is tiny it is difficult to be noticed by the enemy units. The vehicle is controlled by a mote inside it and if a particular event is reported from an area, the vehicle can go and check whether the event has occurred. The vehicle node can be recharged at the base station. The maintenance of the vehicle can be done periodically. Other facilities like camera and costly sensors may be added to the vehicle. The importance of localization is to relate the vehicle location with local map or global

map. If GPS is used, the global position can be obtained and the next movement of the vehicle can be decided. In the absence of a GPS, the driver mote inside the vehicle can send request information to the surrounding motes. The motes can immediately reply with the next movement information. Figure 2.9 shows the Unmanned vehicle to protect the battle field.

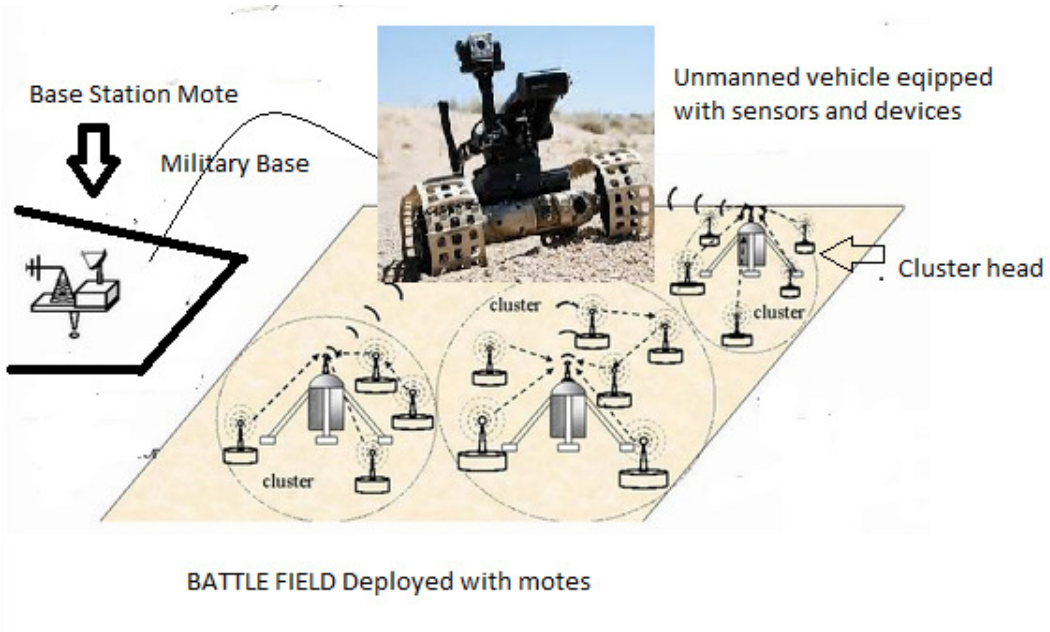

Figure 2.9: Unmanned vehicle to protect the battle field

The vehicle design in military field should be in such a way to reach the entire battle field and be small enough to nullify the presence. The driver mote placed within the vehicle consider the messages from both the surrounding motes and the base station and take appropriate actions. It controls the devices and sensors

connected with the vehicle and can send the valuable information to the base station. The driver mote has the recharge felicity for battery and high computational power like the base station. It collects the sensitive data and checks the credibility of data. If the data is of utmost importance, it encrypts the message using modern encryption techniques and directly sends them to the base station. The driver node gets the exact location of the vehicle from the GPS connected with the mote. It moves to the next destination based on the program running in the driver mote. The mote is controlling the vehicle navigation. Apart from navigation, driver mote has other controls like deploying the motes and maintenance of the WSN network.

2.5.4 Security Architecture

The security architecture proposed in this section is to secure the wireless sensor network in military applications. The wireless sensor network is divided into clusters. Each cluster contains a set of motes in a particular area. The group elects a cluster head. The cluster head aggregates the messages from cluster members and the cluster head selection is based on the remaining energy of the mote. The key management scheme adopted is LEAP [6]. Out of the four sets of keys available in LEAP, only two keys been considered:Individual Key and Cluster Key [10] [14]. The individual keys are distributed to all the nodes before deploying and with the help of these nodes, encrypted messages are sent to the base station and the Vehicle driver mote. Only the base station and Vehicle mote can decrypt and read the messages. The cluster nodes in a cluster share a cluster key within that cluster. The nodes sense the environment and the sensor readings are encrypted by the cluster key and sent to the cluster

head. The cluster head aggregates the messages and encrypt with individual key and send it to the base station through the network. The base station eventually decrypt the messages from cluster head and take appropriate actions. This is based on a threshold value and if the readings from the sensor is greater than the threshold, the base station judge it as an intrusion or possibility of an attack. In certain situations, the base station may take a decision only on the basis of a cluster head information and this may lead to more issues. Sometimes, a compromised node may send the false readings as a cluster head and can mislead the base station from the actual event. This is a serious issue and the unmanned vehicle in the network can solve this problem.

The vehicle connected with sensors and powerful devices like camera and GPS can help the base station to take critical decisions. Let us consider the case that infrared sensors are attached to all the motes to detect the movements of enemies in the area. These nodes send the messages to the cluster head and cluster head reports this to the base station. The base station can cross check whether the event has actually occurred or not by moving the unmanned vehicle to the region. The vehicle move to the area using GPS or surrounding sensor location. The driver mote navigate the vehicle to that particular location where the event is actually reported. The driver mote will collect the data with the help of sensors from the area and immediately send the exact values to the base station. The base station compares the value and if found same it take the appropriate action like informing the military commander or activating the mines etc., The vehicle can be used to further examine the situation by using advanced devices like camera or other sensing devices. The communication between base station and vehicle driver mote is direct and conventional encryption techniques can be used

between them as both the nodes are powerful and have unlimited battery life. The vehicle movement can also be used to recharge the battery in the vehicle.

2.6 Summary

The vehicle controlled by wireless sensor network has been implemented. The vehicle moved under the control of the driver node which is attached with the vehicle. The driver node is controlled by the sensor readings from the sensors attached, surrounding node's reading and by the control of Base station nodes. The proposed system implemented has achieved 90 percentage accuracy on the location where the vehicle was planned to move. The wireless sensor network worked as a unit for the successful navigation of the vehicle. The overall performance of the vehicle navigation system is found to be satisfactory.The Vehicle navigation system controlled by the driver node can be implemented using GPS or GPS free localization algorithms. The comparison between these techniques are given in Table 2.1. The decision of which algorithm to be used is taken based on the application in which the vehicle navigation algorithm being used.

The comparative study based on the accuracy of localization is done based on different algorithms. The GPS based algorithms are more accurate. An anchor free localization is used in the experimental study. The graph shown in fig 2.10 depicts Accuracy comparison based on connectivity of the nodes. As the connectivity of the nodes increases, the percentage location error decreases. So as the connectivity increases, the location tends to be more accurate.

Parameters	Vehicle Navigation Using GPS	GPS Free Vehicle Navigation
GPS Connectivity	Not usable in absence of GPS	Can be used in all locations
Cost	Costlier	Less Costlier
Accuracy	More Accurate	Less Accurate
Sensor Reading	No Reading from Sensors	Readings used in applications
Node Density	No influence	Directly proportional to accuracy

Table 2.1: Comparison between Vehicle Navigation Techniques

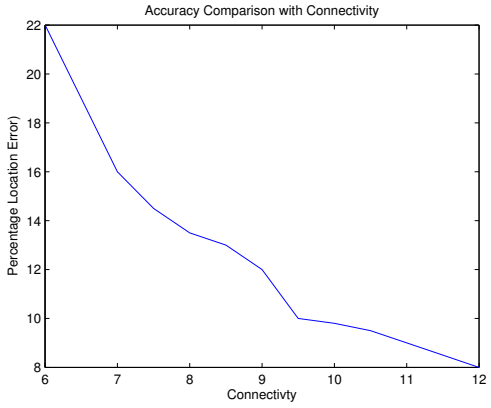

Figure 2.10: Accuracy Comparison with Connectivity

Chapter 3

Unmanned Tiny Vehicle Based Intrusion Detection System for Wireless Sensor Networks

3.1 Overview

Security is a major concern due to the unattended nature and broadcast communication of packets. Compromised or intruder motes can massively launch a massacre of attacks in the network. This chapter proposes a novel security solution in which a tiny vehicle controlled by base station navigates through the network with the aid of a Global Positioning System. The vehicle analyses the network traffic and act as a jury to mote behaviour. It can predominantly identify fraudulent nodes and helps to secure the

system. The vehicle uses a tiny car which is controlled by Micaz mote and programmed using nesC. The system was exposed to various types of attacks, and the results obtained is one step ahead of the existing systems. The performance as evaluated, shows a steep increase in the overall performance with a better accuracy. The system has brilliantly identified attacks with minimum average time for detection than other approaches.

The wireless sensor network consists of low cost motes with sensors with watchful eyes to detect and report events. The nodes, with its low cost implementation and architectural flexibility has found itself in a popular position. The major concern lingering in the mind of one is the limited security and authenticity of packets passing through the network [58]. The conventional mechanisms used for ensuring data security are not applicable to Wireless sensor networks due to the limited processing power and battery life . WSN's are commonly exposed to active and passive attacks. Most of the active attacks are launched by an attacker mote or compromised mote in the network. The solitary nature of the network makes it extremely difficult to detect the presence of intruder nodes [48, 58].

3.1.1 Motivation

The possible malignant behaviour and the lack of a trusted monitoring system in wireless sensor networks brained up the idea to propose a flawless effective system. The current approaches for intrusion detection make use of a dedicated set of nodes, termed the watch dogs which monitors the wireless environment. However, the watch dogs themselves can be a victim of a vicious attack. The focus is on building a much reliable and secure monitoring system without the burden of complex computational tasks. The proposed system

contains an unmanned tiny vehicle which is controlled by a base station. Vehicle navigation eradicates the concept of unattended nature of the network. The vehicle can be maintained periodically to ensure the unlimited life time and power supply for the vehicle. The vehicle mote serves the purpose of watch dog to detect the malicious activities. The vehicle can work independently or collaborate with the cluster head to detect the pernicious activities and report it to the base station. The vehicle mote enjoys the privilege of having a strong security mechanism and powerful communication system.The Vehicle can act as an additional security system along with the existing security solutions to safeguard the network.

3.1.2 Related Work

There are various prevailing techniques which protect the network from numerous kinds of attacks. The limitations of WSNs hold back the conventional intrusion detection mechanisms. Brutch and Ko have discussed various types of possible attacks against WSNs and presented three different architectures for intrusion detection namely standalone architecture, distributed and cooperative architecture and hierarchical architecture [7]. Zhang and Lee proposed a Rule-based intrusion detection system to detect known patterns of intrusions [66]. Khanna and Liu has discussed Genetic algorithm for decision making in intrusion detection [50]. Mamun and Kabir proposed a hierarchical design for intrusion detection for monitoring and making decisions [37]. Sa M and Nayak introduced an external trusted agent for intrusion detection [55]. The idea of the proposed solution is absorbed from the hierarchical approach using hybrid intrusion detection system for clustered wireless sensor network proposed by Sedjelmaci and Feham [57]. The aerial vehicle navigation was

introduced for data collection and data aggregation in wireless sensor network. Jiafu and Hui proposed a test platform for vehicle navigation controlled by wireless sensor network [63]. Arun Madhu and Sreekumar proposed tiny vehicle navigation controlled by a driver mote and a vehicle which can be used to secure the military application [34] [35].

3.1.3 Challenges

The performance of an intrusion detection system depends on the intricacy of detection algorithm and the number of messages exchanged between the motes. The limited battery power of the motes may drain out the entire system. To tackle the problem,the approach is to use a cluster based hierarchical model for intrusion detection along with an unmanned tiny vehicle. The vehicle plays a pivotal role in detecting intrusion by analysing the network traffic and cross checking the data . The prime challenge is to detect the intrusion by distributing duties to base station, vehicle mote, cluster head and cluster members. The basic version of the intrusion detection system works fine in the absence of a vehicle and the proposed system act as a guarding shield to the system.

3.2 Security Issues in Wireless Sensor Networks

The attacks on a WSN can be broadly classified as Physical Attacks, Denial of Service attacks, Routing Attacks and Mote Replication Attacks. Physical attacks may tamper the mote and robs the programming and implementation details. This piece of information

is handy for the attacker who may plan a future attack and re-program the mote. The denial of service attack intends to disturb the services provided by the application. This includes jamming of the transmitting signal, dropping packets and continuously transmitting messages in an attempt to generate collision. Routing attacks advertise false routing information, delayed transmission of packets. In Mote Replication attack,an unlawful mote may replicate the ID of an existing sensor mote and access critical network knowledge to compromise other motes [58] [48]. The challenge of an intrusion detection system is to thwart all possible attacks and develop a sublime security layer.

3.3 System Architecture

The proposed system contain wireless sensor motes which detect an event and delineates it to the base station. As described previously,it contains an unmanned vehicle which voyages the network. The vehicle unit is controlled by a special wireless sensor network mote called Driver mote. The base station is directly connected with the driver mote to facilitate the exchange of packets without intermediary links. For every application, the wireless sensor motes were stored with global key, secret key and unique ID before deployment. Once the motes are deployed, they identify the immediate neighbours and establishes a pair wise key using global key and erases global key as a defence to avert the succeeding attacks. Once the network is configured, all the motes undergo cluster head election based on distributed random selection. The cluster member motes chose the pre-elected cluster heads who act as a prime authority in managing a cluster of motes. The life

expectancy of the network is ensured by the periodic re-election process of cluster heads.

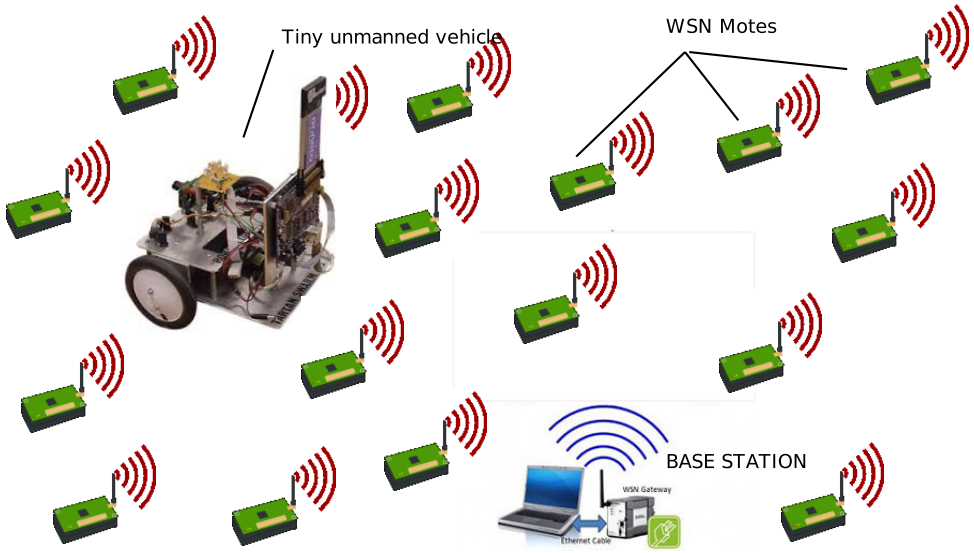

Figure 3.1: Tiny vehicle navigation in WSN

The proposed system uses a hierarchical distributed detection using Naive Bayesian classifier for intrusion detection [57]. Every mote has two modules for intrusion detection, Misuse detection module and Anomaly detection module. The Misuse detection module, uses rule based anomalies method to compare the given data with a predefined set of rules.Anomaly detection module is triggered only when anomalies are detected by the misuse detection module. The Naive Bayesian Classifier aid the anomaly detection module [57]. The proposed intrusion detection system uses a hierarchical model for anomaly detection.The aberrations in the network is perceived by traffic analysis and data monitoring.The actors in the network are hierarchically classified into four; Base

station mote, Driver mote, Cluster head mote and Cluster member motes. The actors try to detect the spurious activities and join force to perform intrusion detection and recovery.

3.3.1 Cluster Member Roles

Each cluster member need to store the location information from the driver motes and the shared secret key corresponding to its immediate neighbours. The cluster member will analyse the shared packets using stored information. They share the information about the dubious motes with cluster head or driver mote.

3.3.2 Cluster Head Roles

The Cluster head will analyse the packets within the cluster. The cluster head contains data about all the cluster motes. The cluster head analyses the packets received from the cluster members. These along with the anomalies reported by the cluster members are useful for identification of intrusion detection. If the cluster head finds a creepy match with any of the predefined anomaly detection events, it alarms the cluster motes and the base station. If it fails to take a decision then the relevant information will be forwarded to the base station or driver mote.

3.3.3 Driver Mote Roles

The driver mote has a Global Positioning System (GPS) unit attached to determine the location information and share it with the motes in that locality. The driver mote analyses all the broadcast packets in its area and monitors the behaviour of the nearby motes. Using

high power radio communication, the Driver mote establishes a one to one communication with the base station. In collaboration with the motes, the Driver mote can pinpoint the attacker nodes in a region. The driver mote can dare the attacker mote to prove its authenticity without sharing the secret information. The vehicle node reports the malicious node information to the base station.

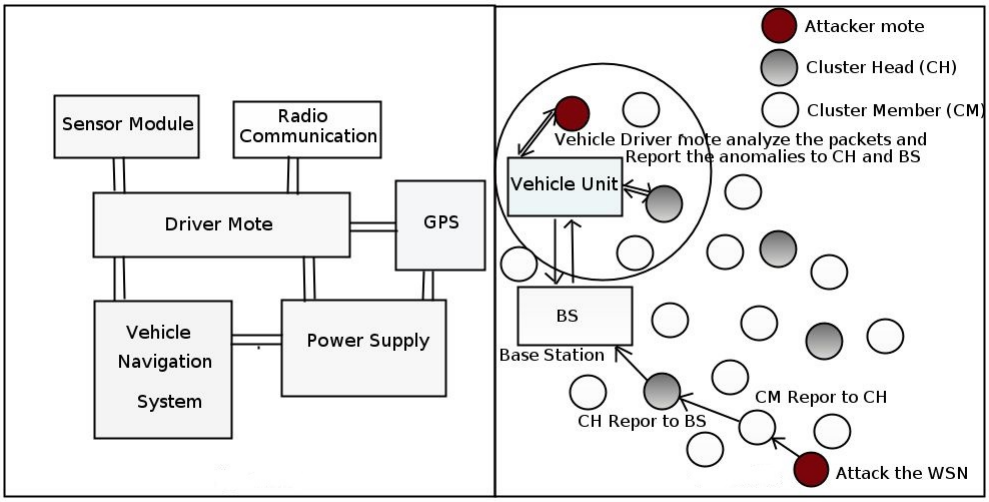

Figure 3.2: System Architecture: a. Tiny Vehicle Architecture b. Monitoring and Reporting Anomalies

3.3.4 Base Station Roles

The base station broadcasts the details of the malicious node which was identified by the cluster head or the vehicle mote. The base station is also the repository to handle the numerous anomalies that may be reported by the other motes. The vehicle movement is under the control of base station.All the computational overhead associated

with finding the malicious node is performed by the base station. The base station holds the secret key information corresponding to mote id. The figure 3.2.b shows the overall working of the system. As per the architecture, the Base station enjoys the highest priority.It also provides secret information to vehicle node which is used to cross check the validity of information shared by other member nodes.

3.3.5 Tiny Vehicle Architecture

The Tiny Vehicle Architecture embodies a Wireless sensor mote which is placed inside the vehicle. The motes, attached with a sensor unit comprising of highly priced sensors are in complete charge of monitoring the area. As shown in figure 3.2.a high power radio communication unit is attached with the vehicle. The mote connected to the vehicle navigation system can control Forward, Reverse, Stop and Turn operations of the vehicle. The geographic location is obtained from the GPS unit. The radio communication system links the driver mote to the surrounding motes. Depending on the geographic location and the kind of application, the choice of vehicle is made. The notion of an unassailable vehicle system is achieved using a rigid vehicle body and a self defense system, which eliminates all possible threats to the loss of data and information within [34] [35].

3.4 Naive Bayesian Classifier for Intrusion Detection

It is an anomaly detection framework for WSN using agent-based learning and distributed data mining technique. First, the network is structured into two-tier hierarchical topology, with different capabilities for sensors at each tier. According to these capabilities, nodes are divided into two types: Forwarding nodes; for activity sensing and data forwarding to higher-tier nodes, and Cluster heads; responsible for collecting and processing data from lower-tier.

In this framework, sensor nodes sense the action and then report to their corresponding cluster head to be processed. Then cluster heads send sensed data file to base station. The data collected at cluster heads may contain erroneous or wrong information (anomaly), so before sending the data file to base station, cluster heads need to detect anomalies and remove them.

The distributed detection process is accomplished through the agent residing between base station and each cluster head. This agent performs detection using two modules as discussed below and shown in Figure 3.3.

3.4.1 Misuse Detection Module

This module compares the given data with a predefined set of rules using rule-based method. The rules will be defined to detect the anomalies by processing the data based on application. If the desired anomaly has occurred based on the rules defined then the Anomaly detection module will be activated.

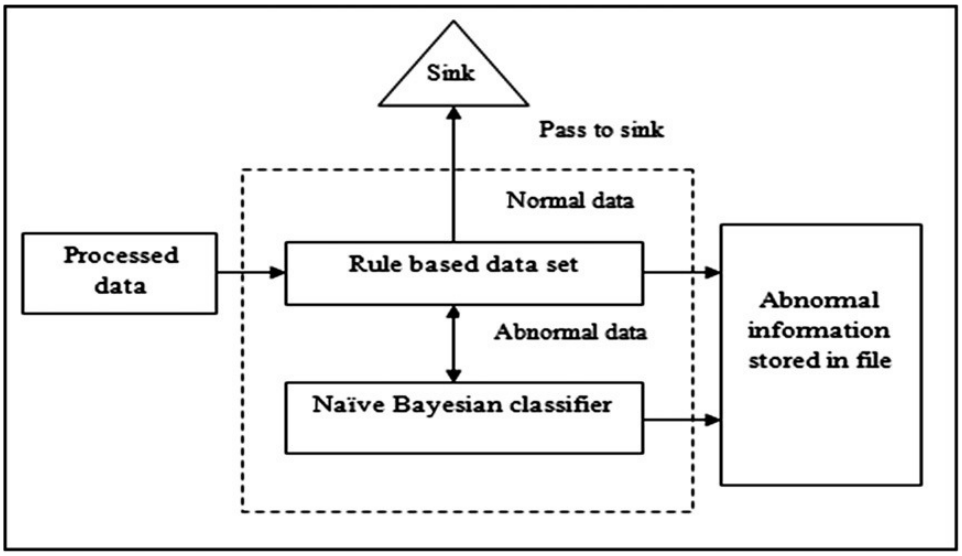

Figure 3.3: Internal agent architecture in Naive Bayesian classifier based IDS

3.4.2 Anomaly Detection Module

It is activated only if anomalies are detected by previous module to further detect using Naive Bayesian classifier. The algorithm explaining the working of anomaly detection is shown in Figure 3.4.

3.5 Anomaly Detection Algorithm

The distributed Anomaly detection algorithm makes use of the information available from all the agents. The details of the malicious nodes are shared with trusted agents,once the anomalies are found. In the absence of a vehicle, the nodes operate based on the general algorithm. The presence of a vehicle in the radio

Input:
Processed n data set files
Output:
Percentage of abnormal data stored in a file

Step 1. Read processed data set file for each cluster head;
Step 2. Call naïve Bayesian classifier program for training the classifier for abnormal event detection, and store that information into a test file;
Step 3. Test this file with the classifier model and write them to an output file
Step 4. Calculate the percentage of abnormal data;

Figure 3.4: Naive Bayesian classifier algorithm

communication region allows the intrusion detection system to respond to the vehicular queries.The algorithm requests vehicular service in the case of complex attacks.The overall working of the general IDS algorithm is explained in Table 3.2. The intrusion detection algorithm in driver mote is explained in Table 3.3.

3.5.1 General Intrusion Detection Algorithm

This algorithm runs in all the networking nodes. It works as a normal algorithm in the absence of a vehicle unit. The Base station generates Global key and shares it with all the member nodes before deployment. Base station also generates the secret key for each individual member nodes and assign the unique ID to the members. After deployment the Cluster members self configure to form the

Abbreviation	Expansion
CH	Cluster Head
BS	Base Station
CM	Cluster Member
DM	Driver Mote
Mid	Mote Unique ID
Kg	Global Key
Ks	Secret Key
Kp	Pairwise Key
DMp	Driver mote position
Mp	Mote position

Table 3.1: Algorithm Notations

network. They generate private shared key with immediate neighbours and store that key for future communication. After this, cluster head election will happen and cluster head will be periodically changed. Cluster member analyses each packet and if some anomalies are found, like unexpected packets, key miss match, new ID for a packet, it will report these to the cluster head. The cluster head will analyse the situation based on the data from other cluster members. If some anomaly is found, it will report this to the base station. The base station will cross check and if the attacker node is found, it will inform all the network nodes.

3.5.2 Driver Mote Intrusion Detection Algorithm

This algorithm runs in the driver node inside the vehicle. The Base station generates Global key and share it with driver nodes. Base station also generates the secret key for driver mote and assigns the unique ID to driver mote. The vehicle has a GPS unit to determine

General Intrusion Detection
Step 1. BS assigns Mid, Ks and kg to CM
Step 2. Deploys Motes to form Network
Step 3. CH selection and assigns CM
Step 4. CM generates Kp with neighbours using kg
Step 5. CM analyse the packets for anomalies
Step 6. CM report anomalies to CH
Step 7. CH cross check the anomalies reported
Step 8. CH reports anomalies to BS
Step 9. BS blacklist Attacker after detailed analysis

Table 3.2: General Anomaly Detection Algorithm

the position and assign it to the cluster members. The cluster member position will be refined during the motion of the vehicle inside the network. The vehicle mote will monitor the network traffic and do complex analysis to find the intrusion. It will also challenge the network nodes to prove its identity and cross check with base station. The base station can control or command the vehicle to do specific tasks which helps to detect the attacker node. It can also even attack the attacker mote by draining attackers battery power or physically destroy the attacker mote by special purpose vehicle.

3.6 Hardware Implementation

The Crossbow MPR2400 (MICAz) motes were used as the cluster member, driver mote and cluster head. A micaz mote attached with a high power laptop acted as the base station. The crossbow MTS101CA and MTS320 CA Sensor boards were used for implementation. The MTS101CA series sensor boards have a

Driver mote Intrusion Detection
Step 1. Assigns Mid, Ks and kg to DM
Step 2. Vehicle navigates to assign Mp
Step 3. Refine Mp based on distance
Step 4. Monitor the traffic to find attacker
Step 5. Challenge the suspected nodes
Step 6. Report malicious node info to BS, CH
Step 7. BS navigate the DM to check events
Step 8. DM directly sends packet to BS
Step 9. Attacks the malicious nodes

Table 3.3: Anomaly Detection Algorithm with vehicle

precision thermistor, a light sensor/ photocell, and general prototyping area. The sensing capabilities of MTS320 CA include light, temperature, microphone, buzzer, accelerometer and magnetometer [21] [4]. The base station contains a laptop and a micaz mote connected with the laptop using crossbow MIB520CB interface board. All the secret information and data sent by various nodes are stored in the laptop which acts as the base station. Complex analysis algorithms were implemented in the laptop for analysing the data sent by the network nodes. The next section describes the implementation details of Vehicle node, the main hardware addition compared to the convectional wireless sensor network.

3.6.1 Vehicle and Driver Mote

A tiny car was used as vehicle component for the implementation. The micaz mote was used as the driver mote inside the vehicle. The driver mote was connected with the MTS320CA sensor board to

sense the surrounding environments. The Forward and Reverse Motors were connected with back wheels for forward and reverse motion and Stepper Motor was connected with the front wheels for turning purpose. The driver mote was connected with the transmitter control unit of the vehicle by using a MDA320CA interface card to send the control messages to navigate the vehicle. An electronic circuit was developed to convert the signals from the mote to control the movements of the car. This circuit has switches controlled by driver mote output. Based on the function the output will be generated by the driver mote and the corresponding switch will be enabled. The switches connect the power supply with motors which controls the movement of the vehicle. The driver mote is attached with a GPS module using MDA320CA interface card and high range radio communication module is attached with the driver mote.

3.7 Software Implementation

The Micaz motes used for implementation runs on TinyOS 2.1 operating system. All the motes were coded in nesC language. The motes were programmed to report the events. The encryption algorithm RC5 was implemented in nesC. The driver mote was purely coded by using nesC for vehicle navigation and intrusion detection. The intrusion detection module was written along with normal code in all the micaz motes by an event attached along with an intrusion detection. The base station mote connected to a laptop creates the secret informations and manages data sent by the network nodes. The highly sensitive pieces of information are encrypted by using secret key and only base station can decrypt

these messages. The priorities are assigned in driver mote to choose between the instructions from base station and surrounding motes.

3.8 Summary

The unattended and distributed operation has become slightly controlled and strictly monitored by the mobile driver node. It provides a secure environment for wireless network applications and the major results are listed below. The wireless network nodes were programmed to introduce various types of attacks. The system was able to detect the intruder motes by checking the unique Id and location information. The various types of attacks and how the system was able to prevent those attacks were listed in the table 3.4.

The algorithms used in this chapter are anomaly detection algorithms. The general algorithm for intrusion detection works without the presence of a vehicle. The anomaly detection algorithm is complex compared to the general algorithm. All the processes in the general algorithm need to be present in the vehicle based algorithm. In addition, vehicle specific steps are also present in the algorithm. The additional data feed from driver mote need to be considered for anomaly detection in the Anomaly detection algorithm (with vehicle). Hence the Anomaly detection algorithm with vehicle is accurate with additional complexity and load to the central processing system at base station.

The Intrusion Detection System was able to provide a top-notch security mechanism without the burden of complex computational tasks. The movement of the vehicle through the system has vanquished the threats posed by the solitary nature of the network. The proposed system was able to perform splendidly in identifying

Attack	Prevention Method
Node replication	Cross check unique Id, pairwise key and assigned location
Packet Drop	Cluster head and Driver mote monitoring
False Messages	Driver mote can move to that location and cross check the information
Compromised clusters	The vehicle can travel and driver mote can report the malicious information
Jamming communication	The driver mote can directly send information to the base station using wi-fi
Traffic analysis	The driver mote can inject packets to confuse the attacker
Denial of service	The driver mote can monitor and report the anomalies
Sink hole	Based on the location information provided by driver mote

Table 3.4: Intrusion Detection Defence against attacks

and eliminating all possible kinds of threats which can be posed by an intruder. The vehicle mote and the driver mote certify the verity of the reported events. The Driver mote provides a classy solution to the localization and time synchronization of the entire network. The system can also be used as a trusted third party for key distribution. The attacks against the vehicle must be addressed in order to ensure the overall system security. The time required for Intrusion detection is substantially reduced. The system is a sure shot solution for the wireless sensor network threats, bettering the security ambiance of the system.

Chapter 4

Guarding Architecture for Unattended Deployment Applications of Ad Hoc Networks : GARUDA

4.1 Overview

Nations are spending massive amount for monitoring the security of sensitive areas. Human monitoring is expensive and vulnerable due to human errors. The Ad Hoc networks are used as an alternative to monitor the sensitive areas. The major drawback of the Ad Hoc network monitoring is the unattended nature of the network which makes the system susceptible to attacks by conflicting troupes. The limited battery capacity and processing power makes it difficult to implement complex cryptographic solutions to protect the network.

4.1.1 Methods/Statistical analysis

The proposed system brings together a general architecture called Guarding Architecture for Unattended Deployment Applications (GARUDA) for Ad Hoc network security. A cluster-based approach is used to classify the network nodes based on functionality and priority. A key pre-distribution technique is used to protect the key-management schema. The modified Localized Encryption and Authentication Protocol are used for hierarchical key management. The Rivest Cipher 5 (RC5) algorithm is used for encryption of sensitive data. The system has an unmanned vehicle with sensors to protect the network from attacks and report the malicious activities to the base station.

4.1.2 Findings

The architecture was successfully implemented in wireless sensor network, set up by micaz motes. The RC5 algorithm was programmed using nesC language in micaz mote for encrypting the sensitive data. The highly confidential data can be directly sent to the base station from the vehicle unit. The absence of key exchange and presence of individual key makes the system sheltered. The GARUDA architecture can be used in any resource constraint monitoring application of unattended nature to make it secure. The vehicle unit can resolve network problems like time synchronization and localization in Ad hoc Networks.

4.2 Introduction

Ad Hoc network is resource constrained, distributed wireless network used for monitoring applications. After deployment, the network nodes self-configure to form the network. The wireless sensor network (WSN) is a type of ad-hoc network used in many low cost unattended applications like Monitoring, Home and hospital automation and Mission critical applications. The major concern apart from the power and computational limitations of the WSN is the security of data and authenticity of communication [8, 15]. Wireless sensor nodes can be deployed to monitor the area and periodically report the sensor readings. The opponents may try to attack the unattended wireless sensor network deployed by home side to find the secret information and tarnish the application. The major attacks include

a. Jamming attacks to disturb the wireless communication.

b. Privacy attacks by listening to sensitive packets.

c. Denial of service attacks like packet drop, flooding packets and injecting packets.

d. Traffic analysis attached to locate the base station.

4.2.1 Motivation

The conventional cryptographic approaches involve high computational cost and powerful processing unit for the implementation. But the sensor nodes have limited power, processing capacity and memory which make it difficult to implement the conventional cryptographic solutions. For mitigating these issues, a potential security solution is proposed to secure a wireless network. This is a low cost solution which can secure the wireless network

from different types of attacks. The prime motivation is to build a secure architecture that completely implements clustering, key-management, encryption and intrusion detection to safeguard the wireless sensor network. Though implemented on wireless sensor networks, it is applicable to Internet of Things applications with limited processing power and memory. The aim is to ensure the network security without additional computational and communication overhead. The unmanned vehicle navigation is incorporated with security architecture to resolve the unattended and resource constraint nature of the network. The GARUDA is trying to resolve these issues,

a. Privacy of the network data and communication.
b. Reliability and authenticity of information.
c. Intruder node detection and removal.
d. Network maintenance without additional overhead.

4.2.2 Related Works

The wireless sensor network security is a difficult scenario to tackle due to limited resources, unattended operation and unreliable communication channel. The first step is the key distribution, i.e., to share the secret key among the nodes. Various key distribution schemes are available to ensure secure key distribution among the nodes. The key pre distribution schema Localized Encryption and Authentication Protocol (LEAP) is widely used in sensor networks [11, 14, 23, 44]. Clustering techniques were used to group the sensor nodes to form clusters and the cluster head can communicate with other cluster heads and the base station. This provides an efficient security mechanism. The Low Energy Adaptive Clustering Hierarchy (LEACH) is used for the cluster based approach [41]. A survey shows

that the LEACH has the highest throughput and packet delivery ratio compared to AODV, DSDV and OLSR [54]. Due to the limited processing power, the Symmetric encryption techniques like DES and RC5 were introduced in wireless sensor network [14, 23, 44]. The light weighted versions of public key cryptographic system such as RSA and ECC is implemented in wireless sensor network [11, 44]. Unmanned vehicle navigation was introduced in various applications to change the solitary nature of the network [2, 34, 63]. The tiny vehicle navigation was introduced for securing sensor network applications in military field [35].

4.3 GARUDA System Overview

The proposed system uses a cluster based approach for data collection and management. An optimized version of Low Energy Adaptive Clustering Hierarchy (LEACH) is used for clustering. The cluster head aggregates the data among cluster nodes and manages the cluster. The cluster head is re-elected periodically [41, 54]. Localized Encryption and Authentication Protocol (LEAP) is modified and used for hierarchical key management [5, 67]. The need for key exchange is evaded by storing unique ID, Individual key and Global Key which is stored in network nodes before their deployment. The Global key is used for pairwise key generation and cluster key formation. The Global key is obliterated after key formation. The data is encrypted based on the previous agreement and nature of the information. The RC5 symmetric encryption technique used to encrypt the data [52, 53]. GARUDA consists of an additional unmanned vehicle unit along with the conventional system. The vehicle unit is controlled by a wireless sensor network

mote known as the driver mote. The vehicle can navigate inside the network region without any human interaction. The vehicle can be equipped with sensors or other costly devices for monitoring the region. The vehicle can cross check the validity of information shared by network nodes and monitor the network traffic to detect the anomalies. The vehicle mote has direct high power communication with the base station to send the sensitive information without any hindrance. It can also interact with network motes to challenge the nodes to prove their identity without sharing the secret information.

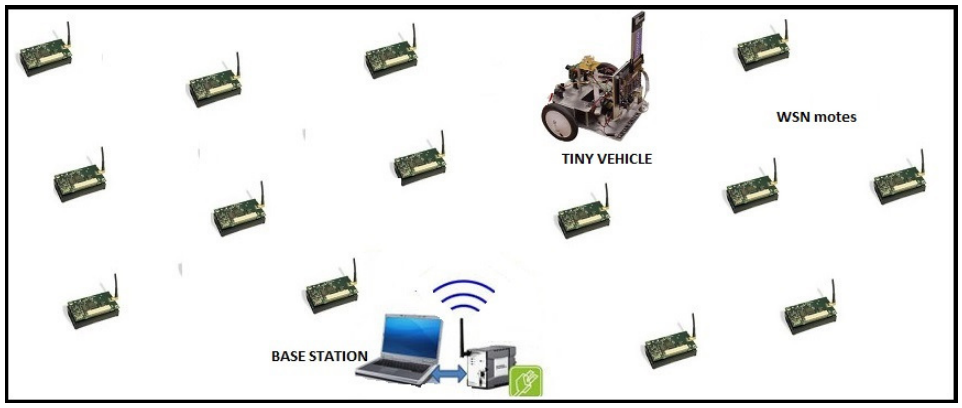

Figure 4.1: GARUDA overview

4.4 Clustering in GARUDA

The clustering approach is used in WSN to ensure that the failure due to key or information loss would not affect the entire network. There are four different classes of nodes in the GARUDA architecture. They are: Cluster Heads (CH), Cluster Members (CM), Vehicle Mote (VM) and Base Station (BS). Base station is the most powerful unit

that stores the data sent by network nodes and analyses it to make a decision. The Low Energy Adaptive Clustering (LEACH) protocol is used for cluster formation [41, 54]. LEACH is dynamic because the cluster heads are assigned on a rotation based policy, which will expand the life time of network. The LEACH network has two phases: the Set-up phase and the Steady-state. In Set-Up phase the cluster-heads are chosen and the nearby nodes are assigned to the cluster head to form the cluster. In Steady-State the cluster-head is maintained and it collects data from cluster members. The aggregated data is sent to the Base station. Initially, when clusters are created, each node decides whether or not to become a cluster-head for the current round. This decision is based on the suggested percentage of cluster heads for the network (determined a priori) and the number of times the node had been a cluster-head so far. This decision is made by the node n choosing a random number between 0 and 1. If the number is less than a threshold , the node becomes a cluster-head for the current round.

$$ThresholdValue = T(n) = \frac{P}{(1-P)rMod(P^{-1})}$$

for all n elements of G

$$ThresholdValue = T(n) = 0$$

for all n not element of G

G = Set of nodes that weren't Cluster Heads in previous rounds

P = Cluster Head Probability (Based on Number of Clusters Heads required)

r = Current Round

After the election, the Cluster heads will send advertisement packets to the nearby nodes. The nodes will accept the request of the nearest Cluster Head based on the signal strength of the advertisement message or the distance based on number of hops.

4.4.1 LEAP key management

The localized encryption and authentication protocol (LEAP) proposed by Zhu et al is a key management protocol for WSNs based on symmetric key algorithms [44] . It uses different keying mechanisms for different packets depending on their security requirements. Four types of keys are established for each node: (i) an individual key shared with the base station (pre-distributed), (ii) a group of keys shared by all the nodes in the network (pre-distributed), (iii) pair-wise key shared with immediate neighbour nodes, and (iv) a cluster key shared with multiple neighbour nodes. The pair-wise keys shared with immediate neighbour nodes are used to protect peer-to-peer communication and the cluster key is used for local broadcast [14, 23].

It is assumed that the time required to attack a node is greater than the network establishment time, during which a node can detect all its intermediate neighbours. A common initial key is loaded into each node before deployment. Each node derives a master key which depends on the common key and its unique identifier. Nodes then exchange Hello messages, which are authenticated by the receivers (since the common key ad identifier are known, the master key of the neighbour can be computed). The nodes then compute a shared key based on their master keys. The common key is erased in all nodes after the establishment, and by assumption, no node has been compromised up to this point. Since no adversary can get the common key, it is impossible to inject false data or decrypt the earlier exchange messages. Also, no node can later forge the master key of any other node. In this way, pair-wise shared keys are established between all immediate neighbours. The cluster key is established by a node after the pair-wise key

establishment. A node generates a cluster key and sends it encrypted to each neighbour with its pair-wise shared key. The group key can be pre-loaded, but should be updated once any compromised node is detected. This could be done, in a naive way, the base station sending the new group key to each node using its individual key, or a hop-by- hop basis using cluster keys. Other sophisticated algorithms have been proposed for the same. Further, the authors have proposed methods for establishing shared keys between multi-hop neighbours.

4.5 Key Management in GARUDA

Key Management in GARUDA uses a key pre distribution scheme and a key agreement scheme. As Key is the vital information, the loss of key information messages will affect the entire network. The concept of LEAP key management schema is used for implementation with few modifications [5, 67]. The different keys maintained in the network are,

1. Individual key, between a WSN mote and base station.

2. Global Key, known to all members before deployment.

3. Establishing Pairwise Key between Immediate Neighbours and Vehicle mote.

4. After Cluster Formation, Cluster Key is shared to cluster members by cluster head. The sequential steps in establishing the keys in GARUDA architecture is explained in this session. The Base station generates and stores Unique Identity Value (UID), Individual Key (IK) and a 256 bit Global Key (GK) for all the nodes in the network. This UID, IK, GK value will be stored in all the nodes before deployment. After deployment, the node needs to select its pairwise key and share it with its neighbours. Instead of sharing the key as such, a 32 bit key is selected from 256 bit global key. The left

index and right index is shared with the neighbours. The node will send the initial hello packets containing node <UID, Left Limit, Right Limit> encrypted by using initial 32 bits of GK to immediate neighbours. The immediate neighbours will receive the hello packet and decrypt it using initial 32 bits of GK to get the UID, Left Limit and Right Limit values. The Left Limit and the Right Limit indicate the pairwise key index and it should satisfy the below condition.

0 <= Left Limit < Right Limit =< 256

Right Limit - left Limit = 32

By using these delimiters, extract a 32 bit key from 256 bit global key. This key is stored in all the immediate neighbours of the sending node corresponding to node's UID. This is the 32 bit key used to securely communicate with that node. The neighbours will encrypt the sensitive information by using this key. All the network nodes will establish a pairwise key to its neighbours without key exchange. After establishing the pairwise key the global key will be erased and the chosen left limit and right limit will be stored in a specific node. The key information stored in a particular node with UID=155 is as shown in Table 4.1 considering it has three immediate neighbours 156, 157, 158.

After Cluster formation, the Cluster head selects a random 32 bit cluster key and share this cluster key to the entire cluster members by encrypting it using a pairwise key. The cluster members will use the cluster key for encrypting sensitive information to the cluster head and cluster head will aggregate the information and send to the base station after encrypting using Individual key. The vehicle node has a stored global key and when a node responds to a hello request from the vehicle, it shares <UID, Left Limit, Right Limit> to the vehicle and the vehicle driver mote gets the pairwise key of the mote to communicate with the mote.

KEY TYPE	Stored Key and UID	Other Information
Individual key (32 bit)	<155, IK155>	Assigned by Base station and stored before deployment
Global Key (256 bit)	<GK>	Assigned by BS and removes after pairwise key establishment.
155's Pairwise key (32 bit)	<155, Left Limit, Right Limit, PK155>	<UID, Left Limit, Right Limit> Shared to all neighbours.
156's Pairwise key	<156, PK156>	Selects from GK Using Left limit and Right Limit shared by 156
157's Pairwise key	<157, PK157>	Selects from GK Using Left limit and Right Limit shared by 157
158's Pairwise key	<158, PK158>	Selects from GK Using Left limit and Right Limit shared by 158

Table 4.1: Key Information stored in Node 155 after pairwise key establishment

4.6 GARUDA Encryption Algorithm

The Encryption algorithm used in GARUDA is Rivest Cipher 5 (RC5). It is a Symmetric block cipher algorithm suitable for wireless sensor network. It can vary in block size, key size and number of rounds based on the security requirements. The algorithm has a Key expansion phase, encryption algorithm and decryption algorithm. The key expansion routine expands the user's secret key K to fill the expanded key array S. The initial values used in GARUDA are, word size(w) = 32, Rounds(r) = 12, Key size(b) = 16, size of table(t) = 16 and number of words in key(c) =4.

Pw = P32 = odd ((2.71-2)232) = B7E15163

Qw = Q32 = odd ((1.618-1)232) = 9E3779b9

In order to convert the Secret Key from Bytes to Words Copy the secret key K[0...b-1] into an array L[0...c-1], any unfilled byte positions of L are zeroed.

Pseudo Code for Initializing the Expanded Array S	Pseudo Code for Mixing in the Secret Key	Encryption Pseudo Code	Decryption Pseudo Code

S[0] = P_{32};
for i = 1 to 15
{
S[i] = S[i-1] + Q_{32};
}
	i = j = 0;
	a = b = 0;
	for k = 1 to 3 * max(t, c)
	{
	a = S[i] = (S[i] + a + b)
	<<< 3;
	b = L[i] = (L[j] + a + b)
	<<< (a + b);
	i = (i + 1) mod (t);
	j = (j + 1) mod (c);
	}
		A = A + S[0];
		B = B + S[1];
		for i = 1 to r
		{
		A = ((A Xor B) <<< B)
		+ S[2 * i]
		B = ((B Xor A) <<< A)
		+ S[2 * i + 1]
		}
			for i = r downto 1
			{
			B = ((B -S[2 * i + 1] >>>
			A) Xor A;
			A = ((A-S[2 * i] >>> B)
			Xor B;
			B = B -S[1];
			A = A -S[0];
			}

Figure 4.2: Pseudo Code for RC5 Key expansion, Encryption and Decryption[52] [53].

4.7 Implementation Details

In this section, the implementation details of GARUDA security solution are discussed. The major hardware component is the crossbow MICAz motes MPR2400 used as the cluster members and cluster head. The Tinyos2.1 Operating system and nesC is used as the programming language. Before deployment the nesC code is deployed in MICAz mote after establishing connection with the base station. A laptop can act as a base station to generate a unique key and unique identifier value for each node. In the NesC code, the individual key, Global key and unique ID is hard coded before

deployment. The pairwise key establishment, clustering and encryption techniques are coded in nesC in all the MICAz motes. In addition, a vehicle component with attached MICAz mote and costly sensors like GPS can navigate throughout the network. The implementation is done by using a toy car having five functions. The vehicle navigation system is connected through an MDA 320 CA interface card with MICAz mote placed inside the vehicle and it acts as the driver mote to control the forward, backward, left, right and stop motion of the vehicle. The driver mote is connected with various sensors through MTS320CA sensor board. The GPS unit is also connected with driver mote using interface card.

4.8 GARUDA Protection Against Attacks

The GARUDA has significantly improved the overall security of the network by improving the probability of detecting an attack. The unattended and distributed operation has become slightly controlled and strictly monitored by the mobile driver node. It provides a secure environment for wireless network applications and this section discusses how the new system can prevent the most common attacks in military wireless network.

4.8.1 Protection Against Physical Attacks

The major Physical attacks are tampering the nodes, reverse engineering to get the vital information, changing the legitimate node to attacker and introducing a new attacker node. The direct physical attacks on sensor nodes and vehicle can be prevented by

providing a self-defence system for the vehicle as well as the nodes. In case of an attack, the vehicle can move away from the attacker using sensor information. The vehicle can even surprise the attacker by sound or an activity. If the attacker captures the vehicle, it can act as suicidal bomb to prevent it from losing the most valuable information. This will prevent the attacker from using the vehicle against the network. The encoding schema and keys shared among vehicle and normal nodes are different to make it more secure. The nodes can prevent the physical attacks by small earthing or vibration in case of tampering attempt and diffuse the entire data in case an attacker opens a sealed node.

4.8.2 Protection Against Denial of Service Attacks

The major denial of service attacks is physically jamming the network with same communication frequency wave or an attacker node is introduced in the network. The attacker node can introduce Hello flood or Replay attack and the attacker can drop the packets or modify the data contents by sending false information. The jamming attacks can be prevented by using the vehicle node to send information directly to base station using another radio frequency or Wi-Fi communication. The intruder sending false information can be cross checked by driver node after reaching the region. By constantly monitoring the network traffic, the delayed replay and Hello flood attacks can be detected and prevented. The attacker dropping the packets can also be tracked and located by using the vehicle. Since the vehicle can travel to the interrupted region, the driver mote can directly send the information to the base station.

4.8.3 Protection Against Privacy Attacks

Monitoring and Eavesdropping are the most obvious attacks against privacy of data. The RC5 encryption technique is used to make it difficult for the attacker. The Vehicle along with the driver node can act as a trusted third party to store and exchange the session key related information to the motes in the network. The asymmetric encryption can be used between driver mote and base station since both are having high computational power compared to the surrounding nodes. The surrounding nodes can send the encrypted information by using the secret key to the driver mote and the driver mote can send it directly to the base station by using asymmetric encryption techniques. This makes the system less vulnerable for the attackers.

4.8.4 Protection Against Traffic Analysis Attacks

The main aim of these types of attacks is to find the location of the base station and behaviour of the network at times of a false attack. The common counter measures include random forwarding of packets and false forwarding of packets in order to confuse the attacker. The mobile driver mote can confuse the attacker by sending random messages.

4.8.5 Protection Against Node Replication Attacks

An attacker seeks to add a node to an existing sensor network by replicating the node ID of an existing sensor node. This type of attacks can be prevented by adding location information to the message. The

Parameters	GARUDA	Existing System's
Security	Highly secure against all sort of attacks and security is scalable.	The single system which can prevent all sorts of attacks are not present.
Connectivity	The system can ensure connectivity to the entire region.	If the node battery life is over the connectivity will be lost.
Cost	Costlier	Less Costlier
Accuracy	More Accurate since it provides additional security	Less Accurate than GARUDA.
Costlier Sensors	Can be incorporated with GARUDA	Can't be incorporated with Existing system
Localization	Can be used to assign location information	Can't be used to assign location information
Time Synchronization.	Can be used to ensure Time Synchronization	Can't be used to solve Time Synchronization

Table 4.2: Advantages of GARUDA over Existing Systems

vehicle with GPS can assign the location information to all legitimate nodes and the vehicle can verify whether the surrounding nodes are keeping the actual location information.

4.9 Summary

The proposed GARUDA system provides a guarding architecture against the common security issues in Ad hoc Networks. The proposed system is implemented in wireless sensor network to check the feasibility of the system. The key pre distribution and hierarchical key management makes it difficult for the attacker to plan the attack. The symmetric encryption technique RC5 was

successfully programmed in nesC. The vehicle node gives a mobile nature to the guarding architecture. The advantages of GARUDA over existing systems are listed in Table 4.2. The vehicle can be used to address other issues such as localization and time synchronization along with providing security to the network.

Chapter 5

Performance Analysis of the GARUDA Using NS2

5.1 Overview

In the previous chapter a general architecture called Guarding Architecture for Unattended Deployment Applications for Ad Hoc network (GARUDA) was introduced. This chapter analyses some features of GARUDA using NS2 simulator. The key features of the GARUDA are listed below.

- A cluster-based approach used to classify the network nodes which focus on functionality and priority.
- The modified Localized Encryption and Authentication Protocol are used for hierarchical key management.
- The Rivest Cipher 5 (RC5) algorithm is used for encryption of sensitive data.

- The system has an unmanned vehicle with sensors to protect the network from attacks and report the malicious activities to the base station.

The architecture has been successfully implemented in wireless sensor network, set up by micaz motes. The GARUDA architecture can be used in any resource constraint monitoring application of abandoned nature to make it secure. The vehicle unit can resolve network problems like time synchronization and localization in Ad Hoc Networks.

Even though the GARUDA algorithm is implemented in micaz motes, due to the limited number of micaz motes, the performance analysis is not done using micaz motes. This chapter primarily aims to complete the task of simulating the step by step processes involved in GARUDA architecture. Network Simulator 2 (NS2) is used for simulating the GARUDA algorithm. The first section of the chapter gives an introduction to network simulator 2. The second section explains the wireless sensor network simulation using NS2. The third section explains step by step execution of the GARUDA algorithm using NS2 simulator. The GARUDA clustering algorithm is explained in detail in this section. The GARUDA vehicle navigation is explained in the last section.

5.2 Introduction to NS2 Simulator

Network Simulator (Version 2), fondly known as NS2, is simply an event driven Open source simulation tool that has proved useful in studying the dynamic nature of communication networks. The code can be written in such a way that any event can be triggered at any particular time. The nodes can be created, the data transfer between

the nodes and the attacks can be demonstrated. The network simulator can be incorporated for a wide variety of applications, protocols (such as TCP and UDP) and many network parameters. It runs on multiple platforms like UNIX, Mac and Windows. The NS2 tool allows for the design of a Wireless Sensor Network and allows for the establishment of connection among the various sensor nodes.

Figure 5.1 shows the basic architecture of NS2 Simulator [59]. NS2 consists of two key languages: C++ and Object-oriented Tool Command Language (OTcl). The C++ defines the internal mechanism (i.e., a back end) of the simulation objects and OTcl sets up simulation by assembling and configuring the objects as well as scheduling discrete events (i.e., a front end). The C++ and the OTcl are linked together using TclCL. NS2 provides users with executable command ns which takes the input argument, the name of a Tcl simulation scripting file. In most cases, a simulation trace file is created. The trace file is used to plot the graph or to create a Network animator (Nam) file. Nam is a Tcl/TK based animation tool for viewing network simulation traces and real world packet traces. It is mainly intended as a companion animator to the ns simulator.

There are many advantages of a NS2 Simulator:

1. It is freely available on-line.

2. It is cheaper than most simulators.

3. Any complex network can be simulated and used for testing.

4. The results can be obtained easily in the form of graph or in a network animator.

5. It supports a wide variety of applications and protocols like TCP , UDP etc.,

6. It can run on a variety of platforms like Windows, Linux, Unix etc.,

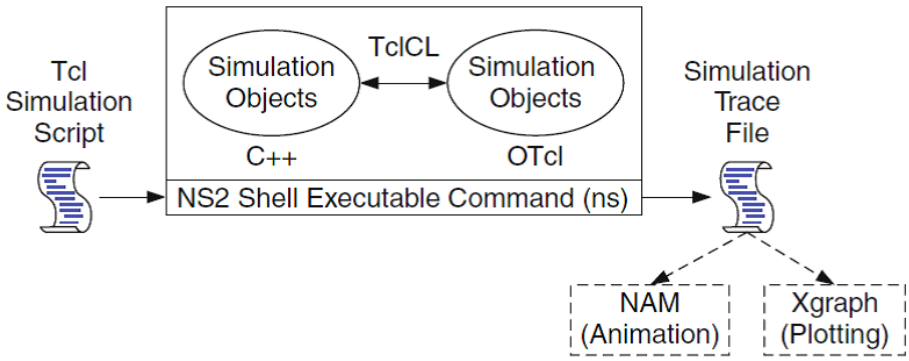

Figure 5.1: Basic Architecture of NS2 Simulator[59]

5.3 Wireless Sensor Network Simulation Using NS2

Network Simulator 2 can be used as the laying foundation for a wide variety of networks. However, this work is limited to the simulation of NS2 for a wireless sensor network[22]. The NS2 provides various features that differentiate a wireless sensor network simulation from other adhoc routing techniques and wired simulations. The key features are explained using a sample simulation. The key features of wireless sensor networks can be classified into Transmission, Energy Distribution and Spatial Distribution.

5.3.1 Transmission

The WSN nodes are spatially separated and there is no direct link between the nodes. The Transmission medium is wireless and radio propagation at a particular frequency has been used. The transmission is radio-propagation and the Wireless network standard

for communication is IEEE 802.15.4. The example given below shows how to define these properties in NS2.

set val(chan) Channel/WirelessChannel; ChannelType

set val(prop) Propagation/TwoRayGround; radio-propagation

set val(netif) Phy/WirelessPhy/802.15.4

set val(mac) Mac/802.15.4

The wireless sensor nodes have many unique properties which makes it dissimilar from normal nodes. They possess a radio transmitter to send the packets and radio receiver to receive the packets. They are featured with an omni directional antenna. One of the important properties of Antenna is the 'antenna height' which is defined in NS2. The antenna model and height settings are defined as shown below:

set val(ant) Antenna/OmniAntenna; antenna model

Antenna/OmniAntenna set X 0

Antenna/OmniAntenna set Y 0

Antenna/OmniAntenna set Z 1.5

Antenna/OmniAntenna set Gt 1.0

Antenna/OmniAntenna set Gr 1.0

The wireless nodes transmit radio signals at a particular frequency and bandwidth. In NS2 bandwidth, frequency and loss in signal strength due to transmission has to be defined. Based on these parameters, the capacity of the channel is determined.

Phy/WirelessPhy set freq 2.4e+9 ; The working band is 2.4GHz

Phy/WirelessPhy set L 0.5 ;Define the system loss in TwoRayGround

Phy/WirelessPhy set bandwidth 28.8*10e3 ;28.8 kbps

5.3.2 Energy Distribution

Energy distribution is an important aspect of wireless sensor network. Compared to the conventional network with unlimited power, the WSN nodes have limited power with a battery unit as the energy source. Once the battery power is drained, the node can be considered as dead. Four parameters are associated with energy.

1. Initial energy: It is the initial battery energy associated with a node immediately after deployment. This parameter directly depends on the initial battery power. The mote utilizes the battery energy for all the activities. The major activities include sensing the real world phenomenon by sensors and sending and receiving radio signals.

2. Transmitted power or Tx power : This is the power consumption associated with transmission of radio signal by radio transmitter. The radio signal encoded with data is to be transmitted from source to destination. The usual transmission is broadcast and the intended receiver will accept the signal.

3. Receive power RX : This is the power consumption associated with receiving of radio signal by radio receiver. In order to receive the signal, the receiver should be in passive listening mode and once the signal is available it is changed to active listening mode. The Rx power is greater than the Tx power.

4. Sense Power: This is the power consumption associated with sensor unit by sensing a real world phenomenon. The WSN mote is in active state while sensing through a sensor and there are events associated with the data value.

The NS2 simulation allows to specify all the energy related variables in the simulation. The first step is to simulate the energy of a signal which decreases with increase in distance. The signal strength at various distance will be specified in NS2.

set dist(5m) 7.69113e-06
set dist(9m) 2.37381e-06
set dist(10m) 1.92278e-06
set dist(11m) 1.58908e-06
set dist(15m) 8.54570e-07
set dist(16m) 7.51087e-07
set dist(20m) 4.80696e-07
set dist(25m) 3.07645e-07
set dist(30m) 2.13643e-07
set dist(35m) 1.56962e-07
set dist(40m) 1.20174e-07

We have to define the remaining energy parameters to simulate the wireless sensor network. The initial Energy, Tx Power and Rx Power are defined in the below example.

-energyModel "EnergyModel"
-initialEnergy 100
-rxPower 0.3
-txPower 0.3

5.3.3 Spatial Distribution

Spatial distribution is an indication of how the nodes are spatially distributed and the traffic among different nodes are indicated in spatial distribution. Wireless sensor nodes are event driven nodes and they have a specific time interval to send the packets to the base station. The packets may be directly routing to base station if the base station is in the range. Alternatively, other nodes in range can also forward the packets to the base station. In NS2, a Constant Bit Rate (CBR) traffic for the nodes has to be defined. The CBR traffic contains 7 parameters for defining a single packet flow between two

nodes. They are Source, Destination, Time interval, Start time, Stop time, Packet Size and Stage. The activities shall be grouped to occur in one time interval to stages, so that it is easy to manage the occurrences as different stages. The stage indicates a particular position during the random motion of the nodes. It also indicates the topology of the nodes and distance to the base station.

The code illustrated below demonstrates a sample wireless sensor network traffic defined by using the CBR. There are a total of four nodes. Node 3 is the base station. Node 0, 1 and 2 are the sensor nodes which send packets to the base station. In the example illustrated below, destination is the base station node, i.e., node 3 and all the other three nodes send packets according to the time interval shown below and size of the packets. The Routing protocol used is Ad hoc On Demand Distance Vector (AODV) Routing.

(src, dest, interval, start, stop, packetsize, stage)

The first sensor activity

cbrtraffic 0 3 0.2 0.1 0.9 5 1;

cbrtraffic 0 3 0.2 1.0 1.9 8 2;

cbrtraffic 0 3 0.2 2.0 3.9 10 3;

The second sensor activity

cbrtraffic 1 3 0.2 0.2 0.9 1 1;

cbrtraffic 1 3 0.2 1 1.9 3 2;

cbrtraffic 1 3 0.2 2 2.9 5 3;

The third sensor activity

cbrtraffic 2 3 0.2 2 2.9 1 1;

cbrtraffic 2 3 0.2 3 3.9 3 2;

cbrtraffic 2 3 0.2 4 4.9 5 3;

cbrtraffic 2 3 0.2 5 5.9 8 4;

The parameters related to wireless sensor network in the TCL file are defined. After running the simulation TCL file, the nam file and

trace file is obtained. The Network animator (NAM) takes the .nam file and produces the animation. A sample animation is given in the figure 5.2.

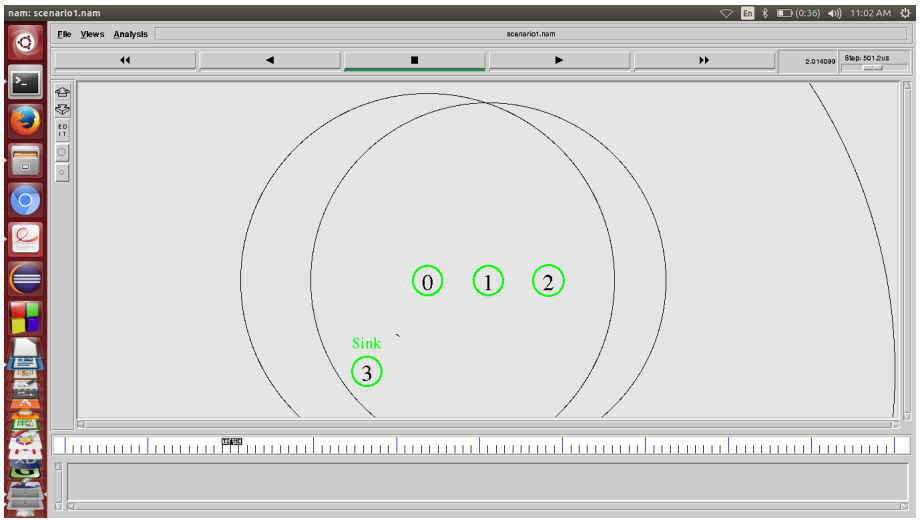

Figure 5.2: A sample NS2 Simulation

5.4 GARUDA Simulation using NS2

The GARUDA architecture can be generalized for ad hoc networks. The NS2 Simulation of GARUDA focus on those aspects which were not possible to be tested in hardware implementation due to limited number of motes. The first aspect is Clustering in GARUDA. An optimized version of Low Energy Adaptive Clustering Hierarchy (LEACH) had been used for clustering. The cluster head aggregates the data among the cluster nodes and manages the cluster. The cluster head is re-elected periodically. The Clustering is simulated

using NS2 simulation and is explained in detail. The second aspect is the simulation of vehicle navigation in wireless sensor network. GARUDA consists of an additional unmanned vehicle unit along with the conventional system. The vehicle unit is controlled by a wireless sensor network mote known as the driver mote. The vehicle can navigate inside the network without any human intervention. The vehicle navigation inside a wireless sensor network region has different modes of operation. These different modes are simulated by using NS2.

5.4.1 NS2 Simulation of Clustering in GARUDA

The clustering approach is used in WSN to ensure that the failure due to key or information loss would not affect the entire network. There are four different classes of nodes in the GARUDA architecture. They are: Cluster Heads (CH), Cluster Members (CM), Vehicle Mote (VM) and Base Station (BS). Base station is the most powerful unit that stores the data sent by network nodes. Apart from storing, it also analyses the data to take appropriate decisions. In NS2, a topology had been selected and the nodes were deployed. The figure 5.3 shows the topology of nodes. 20 nodes were selected and distributed in a random fashion. The Node 0 is assigned as the base station and node 1 is assigned as a vehicle node. These two nodes do not participate in the clustering while the other nodes take part in clustering.

LEACH protocol is used for cluster formation [41, 54]. LEACH is dynamic because the cluster heads are assigned on a rotation based policy, which improves the overall life time of the network. The LEACH network has two phases: the Set-up phase and the Steady-state. In Set-Up phase, the cluster-heads are chosen and the nearby nodes are assigned to the cluster head to form a cluster. In

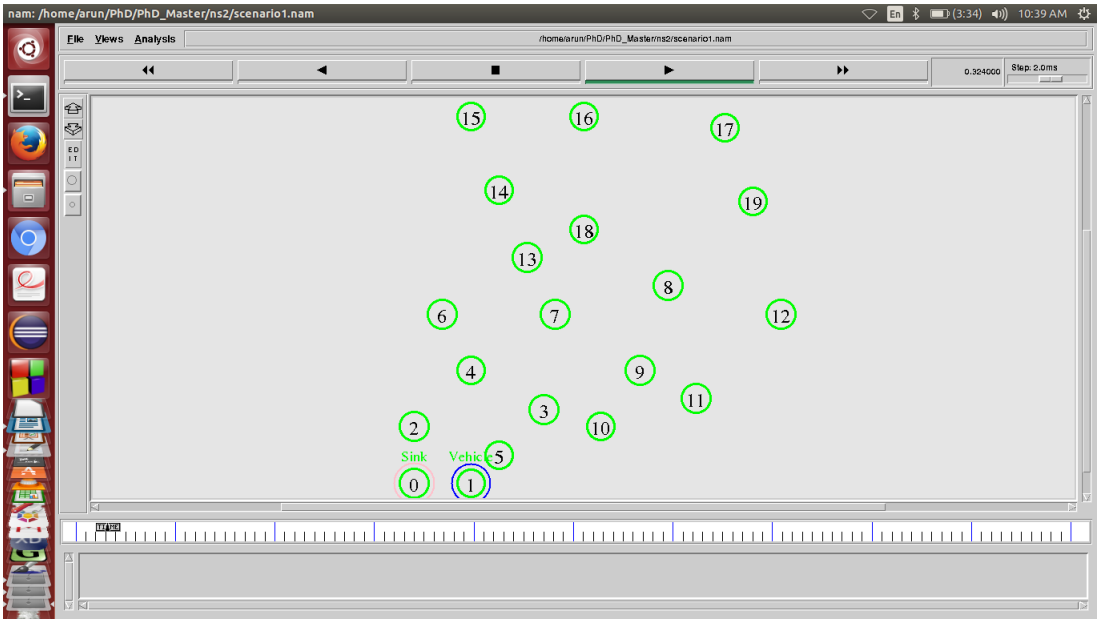

Figure 5.3: Topology of nodes

Steady-State, the cluster-head is maintained and it collects data from cluster members. The aggregated data is sent to the Base station. Initially, when clusters are created, each node decides whether or not to become a cluster-head for the current round. This decision is based on the suggested percentage of cluster heads for the network (determined a priori) and the number of times the node has been a cluster-head so far. This decision is made by the nodes by choosing a random number between 0 and 1. If the number is less than a threshold value, the node becomes a cluster-head for the current round.

$$ThresholdValue = T(n) = \frac{P}{(1-P)rMod(P^{-1})}$$

for all n element of G

$$ThresholdValue = T(n) = 0$$

for all n not element of G

G = Set of nodes that weren't Cluster Heads in previous rounds

P = Cluster Head Probability (Based on Number of Clusters Heads required)

r = Current Round

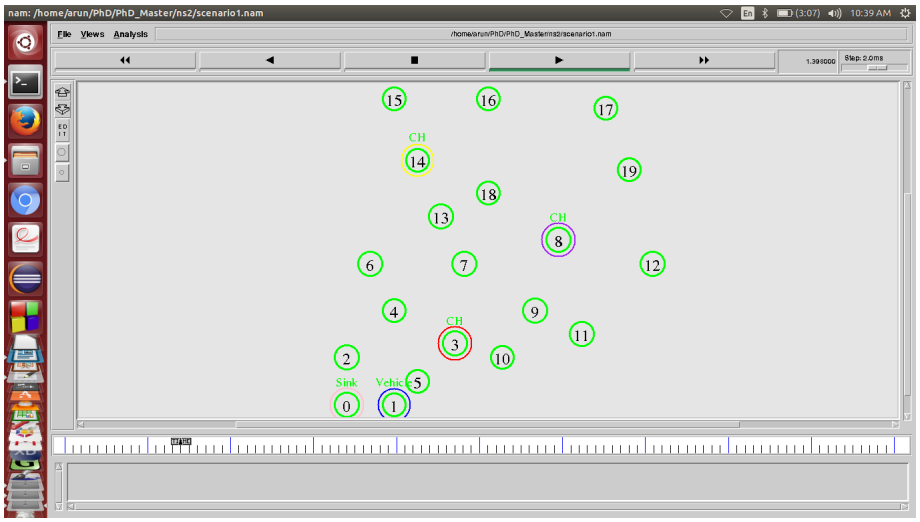

Figure 5.4: Initial Cluster Head election

As already explained, the cluster head selection for round 1 is the initial step. Figure 5.4 shows the simulation after Cluster Head election. The highlighted nodes become the cluster heads and they form the clusters. The nodes 3,8 and 14 become the cluster heads for round 1.

After the election, the Cluster heads send advertisement packets to the nearby nodes. The nodes accept the request of the nearest Cluster

Head based on the signal strength of the advertisement message or the distance based on the number of hops. All the nodes are grouped into clusters. Figure 5.5 shows the cluster formed after the set-up phase.

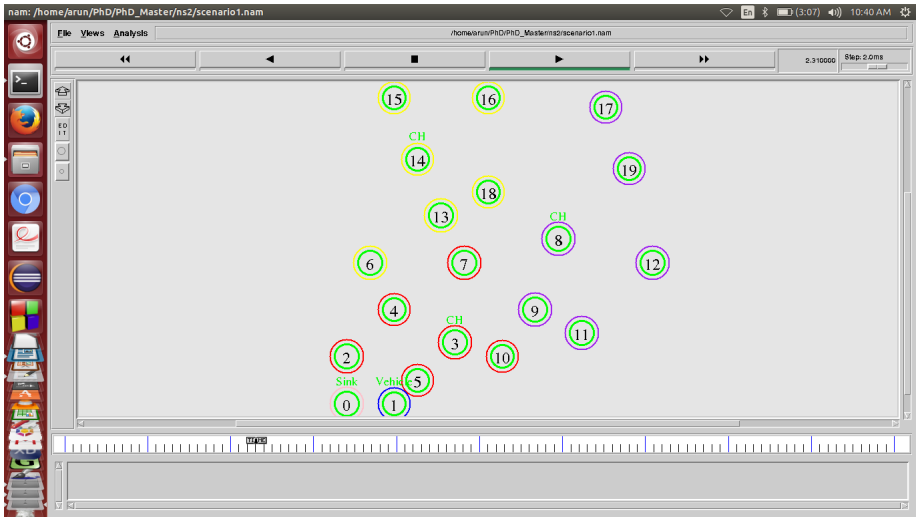

Figure 5.5: Cluster formed after Steady phase

During Steady phase, the cluster head aggregates the data from cluster members and reports it to the base station. The cluster head is in charge of the cluster. After a particular time interval, the cluster head needs to be re-elected in order to prevent the battery drain of-the cluster head. All the nodes except the current cluster heads participate in the election and based on the above algorithm elects the cluster head. Nodes 4, 11 and 16 are elected as cluster heads as shown in Figure 5.6.

As in the previous rounds, the other nodes accept the request of the nearest Cluster Head based on the signal strength of the advertisement message. Figure 5.7 Cluster formation after round 2. This process will continue until all the nodes become cluster heads.

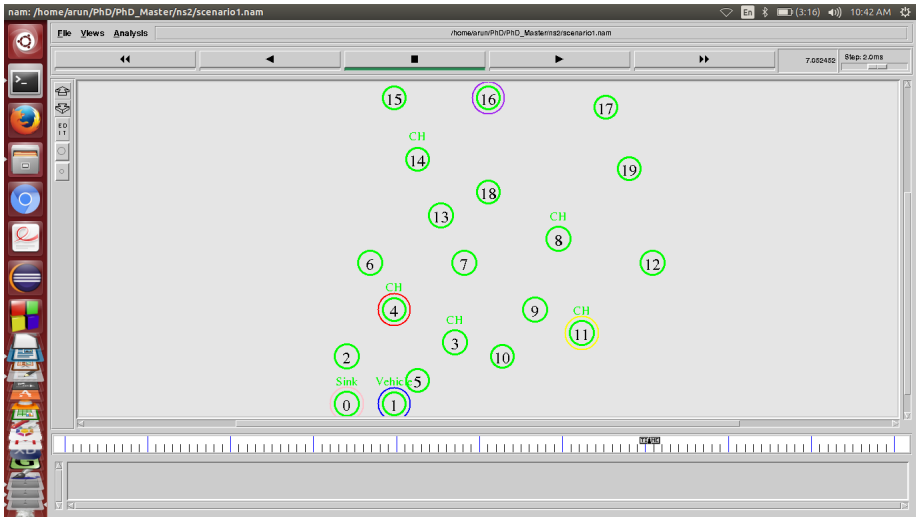

Figure 5.6: Cluster Head election round 2

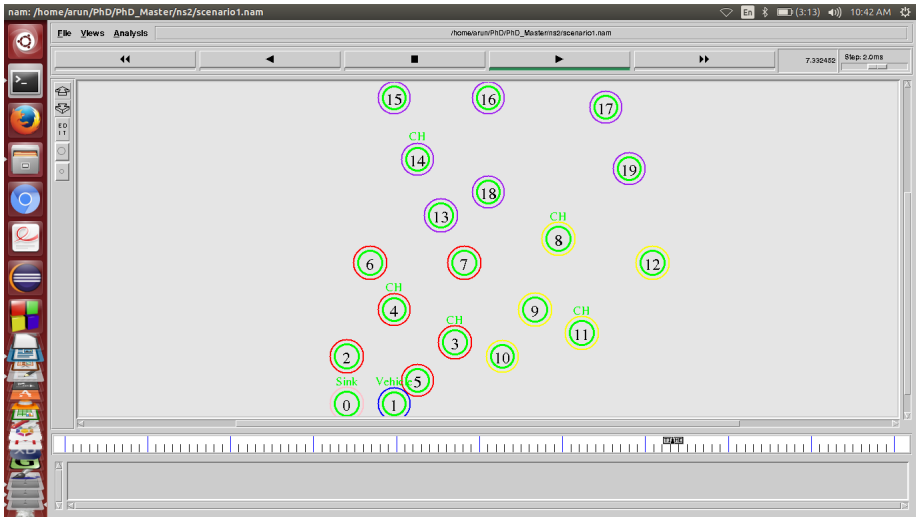

Figure 5.7: Cluster formed after Steady phase of round 2

5.4.2 NS2 Simulation of Vehicle Navigation in GARUDA

The hardware implementation of vehicle navigation system is already discussed in chapter 2. The implementation had been done on a toy car having four functions. By utilizing these functions, the vehicle can have forward motion, reverse motion, left turning and right turning. The WSN node placed inside the vehicle can control the movements of the vehicle by using the above functions. The navigation of the vehicle through the network is not easy and is restricted due to certain complexities. Firstly the driver mote which controls the vehicle has no idea of the network topology and this poses difficulty in vehicle navigation. Yet another difficulty is posed by the different modes of vehicle navigation. The three different modes of vehicle navigation are given below.

1. The vehicle needs to have a periodic routine travel to cover the entire network region.

2. The Base station may navigate the vehicle to a particular location in the network.

3. The vehicle mote can have independent navigation to cross check the anomalies.

5.4.2.1 Vehicle Navigation in the Entire Region

The initial vehicle navigation in GARUDA covers the entire region to check the network. This navigation depends upon the shape and area of the region that the vehicle is going to cover. One of the main limitations faced by the vehicle is that it needs to cover maximum distance in a straight line. The Vehicle which is implemented using a toy car has the property that with the given current position and destination, it has to navigate to the desired destination. An algorithm

which covers and monitors all the nodes in the region has to be devised. Consider a rectangle covering the entire set of nodes, that is the region the vehicle has to cover. The vehicle's initial location is near the base station and parallel to x axis. Three basic functions present in the vehicle are.

1. goto(x,y)- moves the vehicle from current location to x,y through straight line

2. turn left(90^0)- Turns the vehicle 90^0 to left

3. turn right(90^0)- Turns the vehicle 90^0 to right

hm

The variables used in this algorithm are

1. x_{llimit}= least x value in the rectangular area
2. x_{rlimit}= largest x value in the rectangular area
3. y_{llimit}= least y value in the rectangular area
4. y_{ulimit}= largest y value in the rectangular area
5. y_{incr}= This is the value incremented on each scan of the vehicle. This value depends on the density of nodes

Algorithm for vehicle navigation in the entire region is given in the table 5.1.

The Algorithm for Vehicle Navigation in the entire region is simulated using NS2. The vehicle covered the entire region and came back to the initial position.The figure 5.8 depicts the vehicle navigation parallel to x axis and reaches the right most border. Figure 5.9 shows the vehicle movement in which the vehicle takes a left turn from the previous position and performs a small translation in y direction. Figure 5.10 shows that the vehicle takes a left turn and reaches x left limit. Figure 5.11 illustrates the steps 12,13 and 14 mentioned in the algorithm. Figure 5.12 shows the completion of

Vehicle navigation algorithm
Step 1. Assume a rectangle to cover the entire region
Step 2. Assign values x_{llimit} and x_{rlimit}
Step 3. Assign values y_{llimit}, y_{ulimit} and y_{incr}
Step 4. x= x_{llimit} and y=y_{llimit}
Step 5. while(y<=y_{ulimit}) Repeat steps 6 to 15
Step 6. goto(x_{rlimit},y)
Step 7. turn left(90^0)
Step 8. y=y+y_{incr}
Step 9. goto(x_{rlimit},y)
Step 10. turn left(90^0)
Step 11. goto(x_{llimit},y)
Step 12. turn right(90^0)
Step 13. y=y+y_{incr}
Step 14. goto(x_{llimit},y)
Step 15. turn right(90^0)
Step 16. turn right(90^0)
Step 17. goto(x_{llimit},y_{llimit})

Table 5.1: Algorithm for Vehicle Navigation in the entire region

round 1 and starting of round 2. The Figure 5.13 shows the vehicle movement after exiting from the loop. The vehicle will execute the steps 16 and 17 to reach initial position.

5.4.2.2 Vehicle Navigation by Base Station

The vehicle navigation can be controlled by the base station by continuously sending messages to the vehicle mote. The vehicle mote controls the vehicle based on the instruction from the base station. The base station has a clear idea about the topology of the network and the next movement of the vehicle. The vehicle node updates the

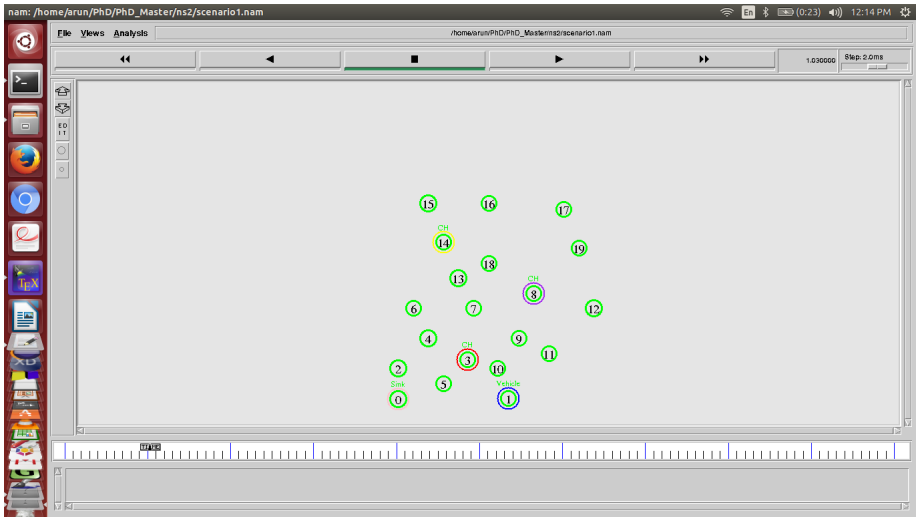

Figure 5.8: Vehicle navigation algorithm step 6

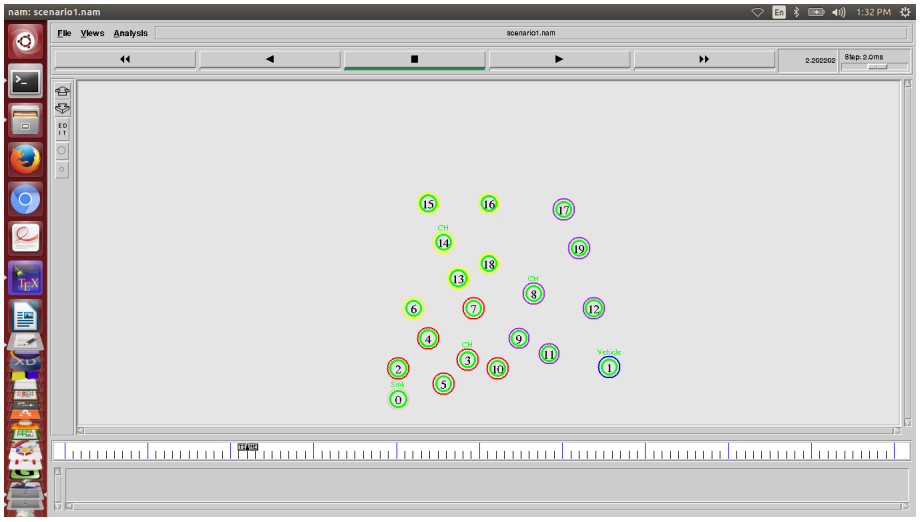

Figure 5.9: Vehicle navigation algorithm step 7,8 and 9

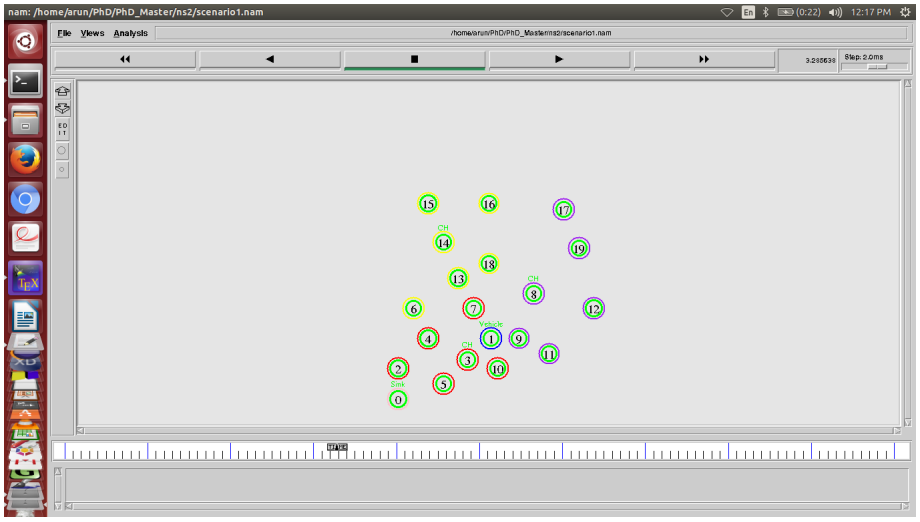

Figure 5.10: Vehicle navigation algorithm step 10 and 11

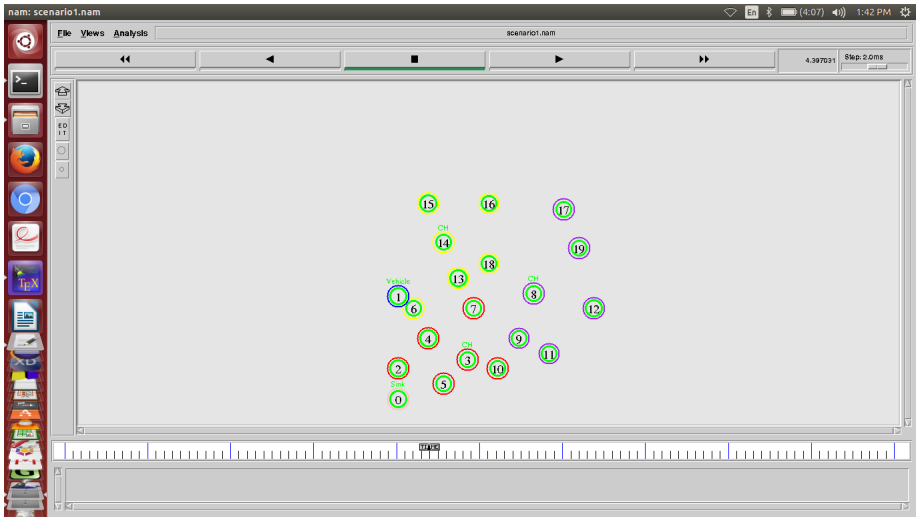

Figure 5.11: Vehicle navigation algorithm step 12, 13 and 14

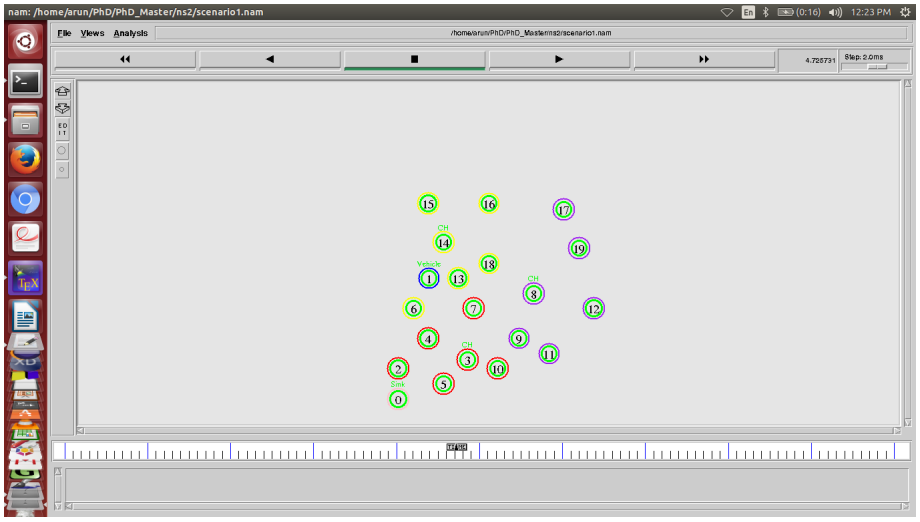

Figure 5.12: Vehicle navigation algorithm step 15 and 6

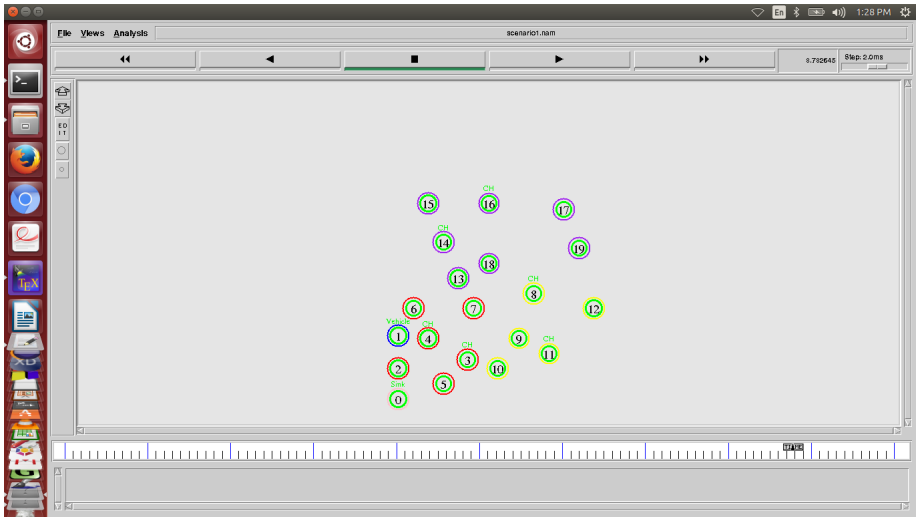

Figure 5.13: Vehicle navigation algorithm step 16 and 17

position to the base station and the base station decides the next movement based on the information.

5.4.2.3 Vehicle Navigation by Vehicle Mote

The vehicle navigation can be controlled by vehicle mote with the help of GPS reading and neighbour mote information. The GPS reading gives an overview about the global position. The surrounding mote position gives the exact location. These parameters aid the vehicle mote in navigating to the exact location.

5.5 Summary

This chapter gives an overview about NS2 simulation in networking. A detailed description of NS2 simulation for wireless sensor network has been supplied. The broader analysis of GARUDA that could not be physically implemented has been implemented through simulation. The GARUDA clustering is simulated by using NS2 simulation. The various vehicle navigation techniques were discussed and the vehicle navigation algorithm inside a rectangular region is simulated using NS2 simulation.

Chapter 6

Result Analysis and Discussion

6.1 Overview

This chapter delineates the results obtained during the research and discusses the correlation among them. Wireless Sensor Network security is the prime objective of this research. The conventional cryptographic approaches were not suitable since it required high computational cost and powerful processing unit for its implementation. The sensor nodes have limited power, processing capacity and memory which makes it difficult to implement the conventional cryptographic methodologies. For mitigating these issues, a potential security solution had to be implemented to secure the wireless sensor network. The proposed solution is feasible technically and financially. The suggested technique acts as the guard of defence against powerful intruder attacks. The prime motivation had been to build a secure fool proof architecture which

absolutely implements clustering, key-management, encryption and intrusion detection mechanisms to safeguard the wireless sensor network. Though implemented on wireless sensor networks, it can be applied to Internet of Things domain with limited processing power and memory. The unmanned vehicle navigation has been incorporated with security architecture to resolve the unattended and resource constraint nature of the network. The research work is divided into three sections.

- Implementing tiny Vehicle navigation controlled by WSN
- Implementing an Intrusion detection system using vehicle navigation in WSN
- Implementing a Guarding Architecture for WSN security

This chapter briefs on how the network security is achieved after implementing the IDS with the Tiny Vehicle Architecture. The second section of the chapter deals with major applications of vehicle navigation in wireless sensor network. The third section deals with an overview on navigation system for WSN security. The fourth section provides a comparison of Vehicle based IDS with the existing IDS systems. The fifth section gives an overview about GARUDA and the sixth section deals with GARUDA protection against attacks. Section seven analyses GARUDA with existing security architectures.

6.2 Vehicle Navigation System Applications in WSN

The vehicle controlled by wireless sensor network has been already implemented. The vehicle moves under the control of driver node attached with the vehicle. The driver node is controlled by the sensor readings from the sensors attached, by the control of Surrounding motes and by the control of Base station nodes. The vehicle navigation can be applied to solve various research problems of WSN. Some of them are mentioned below.

6.2.1 Unmanned Vehicle Based Routing

Multihop routing is a critical service required for WSN. Due to unreliable network and asymmetric links between nodes, WSN routing is complex from conventional network routing. There are various routing schemes implemented for WSN with different objectives. The aim is to incorporate unmanned vehicle for routing. The Vehicle can navigate through areas of the network where there is limited connectivity and can route the packets to the base station. The Vehicle can be used to broadcast messages from base station to the network nodes. Vehicle mote has the capability to store the network topology to optimize the neighbour discovery process.

6.2.2 Unmanned Vehicle Based Node Localization

Node localization is the problem of determining the geographical location of each node in the system. Localization is one of the most fundamental and difficult problems that must be solved for WSN.

Localization solutions should take into account the factors such as the cost of extra localization hardware, the degree of location accuracy required, whether the region is indoor or outdoor and whether there is a line of sight among the nodes. Unmanned vehicle can solve the localization without the burden of extra hardware on each node. When the unmanned vehicle is in close range, a node can take the location value from the vehicle. This is considered to be its initial location and later this value may be refined. If it is an indoor application,where GPS is not available, the vehicle can assign the relative position as the location of the network nodes. If range based methods are used to solve localization, the vehicle location at three different positions can be used to find the location of a node.

The vehicle navigation inside the wireless sensor network is based on the location information of the vehicle. The importance of localization is to relate the vehicle location with a local map or a global map. Various localization techniques can be used to determine the position of the deployed motes. If GPS is used, the global position can be obtained and with this position information the next movement of the vehicle can be determined. However, in the absence of GPS signal, relative position can be found out based on the location information of other motes. Initially, the WSN nodes self-configure to form a network and compute their relative position. The mote which is placed inside the vehicle utilizes the location information from the surrounding nodes and decides the next movement of the vehicle.

6.2.3 Unmanned Vehicle Based Time Synchronization

Time synchronization can be defined as the clocks of each node in a WSN should read the same time within epsilon and remain that way. Since clocks drift over time, they must be periodically re-synchronized and in some instances when very high accuracy is required, it is even important for nodes to account for clock drift between synchronization periods. The vehicle node can be used for time Synchronization task. The master time will be set on the clock travelling through the network. The network nodes will receive the time from vehicle navigating through the network. Based on received signal strength, the nodes will estimate the time delay and update the clock timing.

6.2.4 Unmanned Vehicle Based Power Management

Depending on the activity level of a node, its lifetime may only be a few days, if no power management schemes are used. Since most systems require much longer lifetime, significant research has been undertaken to increase the lifetime while still meeting functional requirements. At the software level, power management solutions are targeted at (i) minimizing communications since transmitting and listening for messages is energy expensive, and (ii) creating sleep/wake-up schedules for nodes or particular components of nodes. The vehicle can be periodically maintained to recharge or replace the batteries. So the life time of batteries is infinite for the vehicle. The major activities that can be done by the vehicle to save the power include

1. Assigning high power computations to vehicle mote.

2. Nodes can go to sleep mode in the presence of vehicle.

3. Broadcasting of messages can be handled by a vehicle navigating the entire region.

6.2.5 Unmanned Vehicle Based Data Aggregation

Data Aggregation is the process of aggregating messages from multiple nodes into a single message and reducing the number of packets which are sent to the base station. There will be some specific nodes to do the data aggregation and transmission to the base station. The vehicle navigating through a region can aggregate the data and send to the base station. The data aggregation requires processing and communication, since the vehicle mote has high computational power and battery life, it can be a perfect choice for data aggregation.

6.2.6 Unmanned Vehicle Based Network Maintenance

The Network maintenance include periodically monitoring the network for connectivity issues, faulty nodes, link failures and correcting those errors. Due to unattended nature of the network, it is very difficult to maintain the network. A specially designed vehicle can be used for network maintenance. While travelling through the network, vehicle may find absence of node or node failure. The vehicle can replace those nodes with new ones.

6.3 Unmanned Vehicle Based Security Solutions for WSN

The applications of the Unmanned vehicle navigation are given in the previous section. This section gives an overview about how the vehicle navigation have been applied to solve WSN security issues. The major security issues that can be solved by vehicle navigation are given in this section. The vehicle controlled by wireless sensor network was implemented. The vehicle moved under the control of driver node attached with the vehicle. The driver node was controlled by the sensor readings from the sensors attached, by readings from surrounding nodes and by the control of Base station nodes. The system implemented by us got 90 percentage accuracy on the location where the vehicle was planned to move. The wireless sensor network worked as a unit for the successful navigation of the vehicle. The unmanned vehicle navigation can be used to satisfy various WSN security requirements. Some of them are listed below.

6.3.1 Unmanned Vehicle for Cryptography

Selecting the most appropriate cryptographic method is vital in WSNs as all security services are ensured by cryptography. The WSN limitations such as code size, data size, processing time, and power consumption make it difficult to design Cryptographic algorithms. The vehicle can be involved in cryptographic solutions as a trusted third party. The complex cryptographic operations can be done by the vehicle and are sent to the base station. In the presence of vehicle a node will sent message to the vehicle and the vehicle will do the encryption and forward it to the base station. For certain public key methods, initial computation can be assigned to the

vehicle unit. The costly hardware required for Cryptography can be included inside the vehicle.

6.3.2 Unmanned Vehicle for Intrusion Detection

An intrusion detection system (IDS) monitors a host or network for suspicious activity patterns outside normal and expected behaviour . It is based on the assumption that there exists a noticeable difference in the behaviour of an intruder and legitimate user in the network such that an IDS can match those pre-programmed or possible learned rules. IDS has specific motes called watch dogs to monitor the network activities and find anomalies. The vehicle mote can monitor the network activities and report the anomalies to the base station. The vehicle mote is comparatively more secure than other motes. The vehicle mote can directly report the anomalies to the base station. The vehicle mote can be incorporated with existing IDS to make it more secure.

6.3.3 Unmanned Vehicle for Key Management

Key management is a core mechanism to ensure security in network services and applications in WSNs. The goal of key management is to establish the keys among the nodes in a secure and reliable manner. Most of the key management protocols for WSNs are based on symmetric key cryptography because public key cryptographic techniques are in general computationally intensive. In certain key management techniques, a trusted third party node is used to store and distribute the keys. The vehicle node can act as a trusted third party to distribute the keys. The nodes can establish the keys with the help of the vehicle mote.

6.3.4 Unmanned Vehicle for Protecting the Network

This vehicle can navigate through the network and monitor the network security. As the vehicle is tiny, it will not easily catch the the enemy's attention. The vehicle will be controlled by a mote inside it and if a particular event is reported from an area, the vehicle will move towards that particular area and cross check whether the event has occurred. The vehicle node is having unlimited power and maintenance of the vehicle can be easily done. Other facilities like camera, GPS and other costly sensors can be added in the vehicle. The importance of localization is to relate the vehicle location with local map or global map. If GPS is adopted, the global position may be obtained and with this position information and we can decide the next movement of the vehicle.

6.3.5 Unmanned Vehicle Defence Against Physical Attacks

The major physical attacks include jamming and tampering. Jamming the network traffic by continuously transmitting the radio frequency signal can be protected by vehicle mote using a different high frequency signal for transmission. Tampering is the process of directly taking the mote and destroy it or reprogramme it as an attacker. If the vehicle found such attacker it can destroy that attacker either physically or draining the battery by continuously sending messages. The vehicle used in military monitoring can be equipped with advanced attacking mechanisms such as guns or suicidal bombs.

6.4 Vehicle Navigation System Based IDS

The possible malignant behaviour and the lack of a trusted monitoring system in wireless sensor networks brained up the idea to propose a flawless effective system. The current approaches for intrusion detection make use of a dedicated set of nodes,termed the watch dogs which monitors the wireless environment. However, the watch dogs themselves can be a victim of a vicious attack. The focus is on building a much reliable and secure monitoring system without the burden of complex computational tasks. The proposed system contains an unmanned tiny vehicle which is controlled by a base station. Vehicle navigation eradicates the concept of unattended nature of the network. The vehicle can be maintained periodically to ensure the unlimited life time and power supply for the vehicle. The vehicle mote serves the purpose of watch dog to detect the malicious activities. The vehicle can independently or collaborate with the cluster head to detect the pernicious activities and report it to the base station. The vehicle mote enjoys the privilege of having a strong security mechanism and powerful communication system.The Vehicle can act as an additional security system along with the existing security solutions to safeguard the network.

The performance of an intrusion detection system depends on the intricacy of detection algorithm and the number of messages exchanged between the motes. The limited battery power of the motes may drain out the entire system. To tackle the problem, the approach is to use a cluster based hierarchical model for intrusion detection along with an unmanned tiny vehicle. The vehicle plays a pivotal role in detecting intrusion by analysing the network traffic and cross checking the data . The prime challenge is to detect the

intrusion by distributing duties to base station, vehicle mote, cluster head and cluster members. The basic version of the intrusion detection system works fine in the absence of a vehicle and the proposed system acts as a guarding shield to the system.

6.4.1 Intrusion Detection System Characteristics

The unattended and distributed operation has become slightly controlled and strictly monitored by the mobile driver node. It provides a secure environment for wireless network applications and the major results are listed below. The wireless network nodes were programmed to introduce various types of attacks. The system was able to detect the intruder motes by checking the unique Id and location information. The various types of attacks and how the system was able to prevent those attacks were listed in the Table 6.1. The major attacks were classified into Node replication, Compromised clusters, Packet Drop, False Messages, Jamming communication , Traffic analysis, Denial of service and Sink hole. An overview of defensive mechanism against each of the attack by using IDS is listed in Table 6.1.

The figure 6.1 shows the graphical analysis of Vehicle based Intrusion detection system. The figure 6.1 (a) shows the effect of average Intrusion detection time due to increasing the number of vehicles in the network. As the number of vehicles increase, the average intrusion detection time decreases and reaches a lower limit. By further increasing the number of vehicles there is no further decrease in the Average detection time. The figure 6.1 (b) shows the comparison between anomaly detection IDS algorithm and Proposed IDS. As the graph shows the proposed IDS outperforms anomaly based IDS.

Attack	Prevention Method
Node replication	Cross check unique Id, pairwise key and assigned location
Packet Drop	Cluster head and Driver mote monitoring
False Messages	Driver mote can move to that location and cross check the information
Compromised clusters	The vehicle can travel and driver mote can report the malicious information
Jamming communication	The driver mote can directly send information to the base station using wi-fi
Traffic analysis	The driver mote can inject packets to confuse the attacker
Denial of service	The driver mote can monitor and report the anomalies
Sink hole	Based on the location information provided by driver mote

Table 6.1: Intrusion Detection Defence against attacks

The Intrusion Detection System was able to provide a top-notch security mechanism without the burden of complex computational tasks. The movement of the vehicle through the system has vanquished the threats posed by the solitary nature of the network. The proposed system was able to perform splendidly in identifying and eliminating all possible kinds of threats which can be posed by an intruder. The vehicle mote and the driver mote certify the verity of the reported events. The Driver mote provides a classy solution to the localization and time synchronization of the entire network. The system can also be used as a trusted third party for key distribution. The attacks against the vehicle must be addressed in order to ensure the overall system security. The time required for Intrusion detection is substantially reduced. The system is a sure shot solution for the

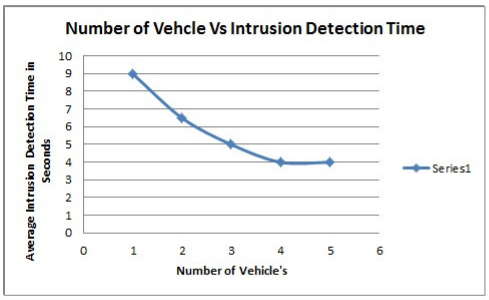

Fig 3.a

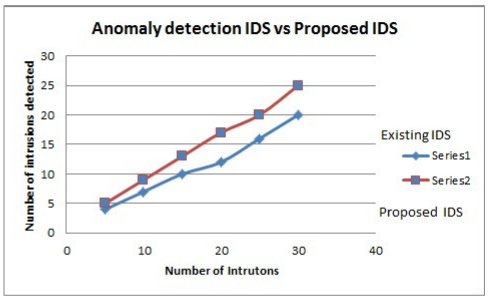

Fig 3.b

Figure 6.1: Graphical Result Analysis (a.)Number of Vehicle Vs Average Intrusion detection time (b.) Anomaly based IDS Vs Proposed Vehicle IDS

wireless sensor network threats, bettering the security ambiance of the system.

The increase in the number of wireless sensor nodes can improve the lifetime of the system. The increase in node density provides an option to increase the sleeping time of the node thus increasing the lifetime of the network. It can also prevent in the isolation of the network. The number of attacks to the WSN negatively affects the life time of the network. DOS like attacks can drain the battery and the node will eventually run out of power. However by proper monitoring, the lifetime of the network can be increased amidst attacks. The attack detection scheme continuously sends messages by encryption and decryption. This phenomenon leads to drain the battery which inversely affect the network life time.

6.5 GARUDA

The GARUDA brings together a general Guarding Architecture for Unattended Deployment Applications. A cluster-based approach is used to classify the network nodes based on functionality and priority. A key Pre-distribution technique is used to protect the key-management schema. The modified Localized Encryption and Authentication Protocol are used for hierarchical key management. The Rivest Cipher 5 (RC5) algorithm is used for encryption of sensitive data. The system has an unmanned vehicle with sensors to protect the network from attacks and report the malicious activities to the base station.

The architecture was successfully implemented in wireless sensor network, set up by micaz motes. The RC5 algorithm was programmed using nesC language in micaz mote for encrypting the sensitive data. The highly confidential data can be directly sent to the base station from the vehicle unit. The absence of key exchange and presence of individual key makes the system sheltered. The GARUDA architecture can be used in any resource constraint monitoring application of unattended nature to make it secure. The vehicle unit can resolve network problems like time synchronization and localization in Ad hoc Networks.

The GARUDA System has significant changes from the existing system due to an unmanned vehicle involved in security architecture. The vehicle driver mote can be used to cross check the validity of information and to detect the intruders. The intrusion detection system works along with security architecture so as to make it more secure. The table 6.2 shows a comparison of existing techniques with GARUDA. Due to the presence of the vehicle, GARUDA offers high security when compared to the existing techniques which are costlier.

Parameters	GARUDA	Existing System's
Security	Highly secure against all sort of attacks and security is scalable.	The single system which can prevent all sorts of attacks are not present.
Connectivity	The system can ensure connectivity to the entire region.	If the node battery life is over the connectivity will be lost.
Cost	Costlier	Less Costlier
Accuracy	More Accurate since it provides additional security	Less Accurate than GARUDA.
Costlier Sensors	Can be incorporated with GARUDA	Can't be incorporated with Existing system
Localization	Can be used to assign location information	Can't be used to assign location information
Time Synchronization.	Can be used to ensure Time Synchronization	Can't be used to solve Time Synchronization

Table 6.2: Advantages of GARUDA over Existing Systems

The accuracy of GARUDA is high when compared to the other systems. The GARUDA system can also solve issues like time synchronization and localization, while the other systems cannot ensure it.

The encryption and decryption is associated with the vital information which is to be share with the base station. After encryption, the cluster head aggregates the message and sends it to the base station. The number of encryption and decryption occurring in the network is minimal. The processor is barely affected by the imposition of additional security measures. The encryption and decryption operations offer an additional load over the processor as well as the node battery. However, as the light weighted algorithms used can reduce the impact on the battery life. In the long run, the

operations may have a negative effect on the nodes battery life.

6.6 GARUDA's Protection Against Attacks

The GARUDA has significantly improved the overall security of the network by improving the probability of detecting an attack. The unattended and distributed operation has become slightly controlled and strictly monitored by the mobile driver node. It provides a secure environment for wireless network applications and this section discusses how the new system can prevent the most common attacks in military wireless network.

6.6.1 Protection Against Physical Attacks

The major Physical attacks are tampering the nodes, reverse engineering to get the vital information, changing the legitimate node to attacker and introducing a new attacker node. The direct physical attacks on sensor nodes and vehicle can be prevented by providing a self-defence system for the vehicle as well as the nodes. In case of an attack, the vehicle can move away from the attacker using sensor information. The vehicle can even surprise the attacker by sound or an activity. If the attacker captures the vehicle, it can act as suicidal bomb to prevent it from losing the most valuable information. This will prevent the attacker from using the vehicle against the network. The encoding schema and keys shared among vehicle and normal nodes are different to make it more secure. The nodes can prevent the physical attacks by small earthing or vibration

in case of tampering attempt and diffuse the entire data in case of an attacker opening a sealed node.

If the vehicle is physically damaged by the attacker, or due to a collision, it will be removed from the network. The stored information and code in the driver mote will be automatically erased. In the absence of the vehicle, it works as a normal WSN without the vehicle component. In case of the multi-vehicle based system, the fellow motes takes up the role of the damaged node. Later, a new vehicle will be introduced for the smooth functioning of the system.

6.6.2 Protection Against Denial of Service Attacks

The major denial of service attacks is by physically jamming the network with same communication frequency wave or by introducing an attacker node in the network. The attacker node can introduce Hello flood or Replay attack and the attacker can drop the packets or modify the data contents by sending false information. The jamming attacks can be prevented by using the vehicle node to send the information directly to the base station using another radio frequency or Wi-Fi communication. The intruder sending false information can be cross checked by the driver node after reaching the region. By constantly monitoring the network traffic, the delayed replay and Hello flood attacks can be detected and prevented. The attacker dropping the packets can also be tracked and located by using the vehicle. Since the vehicle can travel to the interrupted region, the driver mote can directly send the information to the base station.

6.6.3 Protection Against Privacy Attacks

Monitoring and eavesdropping are the most obvious attacks against privacy of data. The RC5 encryption technique is used to make it difficult for the attacker. The Vehicle along with the driver node can act as a trusted third party to store and exchange the session key related information to the motes in the network. The asymmetric encryption can be used between driver mote and base station since both are having high computational power compared to the surrounding nodes. The surrounding nodes can send the encrypted information by using the secret key to the driver mote and driver mote can send it directly to the base station by using asymmetric encryption techniques. This makes the system less vulnerable for the attackers.

6.6.4 Protection Against Traffic Analysis Attacks

The main aim of these types of attacks is to find the location of the base station and behaviour of the network at times of a false attack. The common counter measures include random forwarding of packets and false forwarding of packets in order to confuse the attacker. The mobile driver mote can confuse the attacker by sending random messages.

6.6.5 Protection Against Node Replication Attacks

An attacker seeks to add a node to an existing sensor network by replicating the node ID of an existing sensor node. This type of attack can be prevented by adding location information to the

message. The vehicle with GPS can assign the location information to all legitimate nodes and the vehicle can verify whether the surrounding nodes are keeping the actual location information. The nodes can store the neighbour's Unique ID and Location information to cross check and find the attacker.

6.7 GARUDA Comparison with Existing Architectures

This section provides details of proposed and implemented security architectures optimal for use with wireless sensor networks. These architectures will be reviewed, contrasted and compared based on their individual characteristics. A comparative study of the existing techniques with GARUDA is shown here. We are going to consider SPINS, TINYSEC, LEAP+ and Security Manager(SM) against GARUDA Architecture.

6.7.1 SPINS

Perrig et al. (2002) proposed SPINS, a suite of security protocols optimized for sensor networks [46]. SPINS has two secure building blocks, namely Secure Network Encryption Protocol (SNEP) and TESLA, which run on top of the TinyOS operating system. SNEP is used to provide confidentiality through encryption and authentication; whilst also providing integrity and freshness. TESLA is used to provide authentication for broadcasted data.

SNEP has various unique properties. These include a low overhead (8 bytes per message), kept-state at each end point to remove the need for counter values to be transmitted, and semantic

security; a strong security property which prevents eavesdroppers from inferring any of the message content from the cipher text. Additionally, data authentication, replay protection and weak freshness are provided. Under SNEP, communicating nodes share a secret master key, from which they derive independent keys using a pseudo random function. Using the derived key data will be encrypted and along with that MAC of counter value encrypted with derived MAC key and encrypted data is sent to destination[24].

It is evident that data authentication is achieved through the use of a MAC. The use of the counter value implies that replay protection is invoked, in addition to weak freshness. Overhead is reduced by keeping counter state at each end; removing the need for it to be sent with each message. In order to achieve strong freshness, a nonce (random number long enough that an exhaustive search of all possible nonces is not feasible) is used by one of the communicating parties. Node A generates nonce N_A at random and sends it to B with a request message R_A . Strong freshness is achieved as B returns the nonce with the response message R_B in an authenticated protocol.

μTESLA is the micro version of TESLA (Timed Efficient Stream Loss-tolerant Authentication) proposed by Perrig et al. in 2002 [30]. It emulates asymmetry through the delayed disclosure of symmetric keys and serves as the broadcast authentication service of SNEP. μTESLA requires that the base station and the nodes be loosely time synchronized, and that each node knows an upper bound on the maximum error for synchronization.

For an authenticated packet to be sent, the base station computes a MAC on the packet with a key that is secret at that point in time. When a node gets a packet, it can confirm that the base station did not yet disclose the corresponding MAC key, using its loosely synchronized clock, maximum synchronization error and the time at which the keys

are to be disclosed. The node stores the packet in a buffer, aware that the MAC key is only known to the base station, and that no adversary could have altered the packet during transmission. When the keys are to be disclosed, the base station broadcasts the key to all receivers. The receiver can then verify the correctness of the key and use it to authenticate the packet in the buffer

6.7.2 TINYSEC

Karlof et al. (2004) designed the replacement for the unfinished SNEP, known as TinySec, a Link Layer Security Architecture for Wireless Sensor Networks[9]. Inherently, it provides similar services, including access control, message integrity and confidentiality. Access control and integrity are ensured through authentication, and confidentiality through encryption. Semantic security is achieved through the use of a unique initialization vector (IV) for each invocation of the encryption algorithm.

TinySec allows for two specific variants. The first of these, TinySec-Auth, provides for authentication only, and the second, TinySec-AE, provides both authentication and encryption. For TinySec-Auth, the entire packet is authenticated using a MAC, but the payload data is not encrypted; whilst using authenticated encryption, TinySec-AE encrypts the payload and then authenticates the packet with a MAC.

6.7.3 LEAP+

Localized Encryption and Authentication Protocol (LEAP) was proposed by Zhu et al. (2003) as a key management protocol for sensor networks, motivated by the observation that different types of messages propagated in wireless sensor networks have different

security requirements [67]. Lightweight, energy efficient operation and robustness and survivability in the face of node compromise, are the main design goals of this protocol.

There are four different keying mechanisms provided by LEAP, in keeping with the need for different security requirements for different types of messages. These include Individual Keys, Group Keys, Cluster Keys and Pairwise Shared Keys. The Individual Key is a unique key that every node shares with the base station. This allows for confidential communication between the base station and individual nodes, useful for special instructions or keying material etc., The Group Key is a globally shared key that is used by the base station for sending encrypted messages to the entire sensor network (or Group). A Cluster Key is similar but is shared between a node and its neighbours. A Pairwise Shared Key is a key which every node shares with each of its immediate neighbours. These keys are used under this scheme for secure communications that need privacy or source authentication.

An attractive property of this security architecture is the manner in which it handles inter-node authentication. μTESLA is used for broadcast authentication, but between nodes, one-way key chain based authentication is used. Every node creates a one-way key chain of a certain length, and transmits the first key of the chain to each neighbour (encrypted with the Pairwise Shared Key). A key from the one-way chain is known as the AUTH key, and whenever a node sends a message, it attaches the next AUTH key in the chain. The keys are disclosed in reverse order to their generation, and a receiver can verify the authenticity of the message based on the received initial key (commitment), or a recently disclosed AUTH key.

6.7.4 Security Manager

Heo and Hong (2006) proposed a new method of authenticated key agreement [24]. It is based on a Public Key Infrastructure (PKI) and Elliptic Curve Cryptography (ECC). The Security Manager (SM) gives static domain parameters such as the base point and elliptic curve coefficients to prospective network nodes. Devices use these initial parameters to establish permanent public keys and ephemeral public keys, which are in turn used for securing the network data. After calculating a public key, a node sends this to the SM, which could have a public key list for all nodes in the network.

ECC is an approach to public-key cryptography which is based on the algebraic structure of elliptic curves, over finite fields. Elliptic Curve algorithms provide reasonable computational loads and smaller key sizes for equivalent security to other techniques. Smaller key sizes reduce the size of message buffers and implementation cost of protocols. Authenticated key agreement is achieved via the SM, based on the EC-MQV algorithm. This algorithm is more advanced than Diffie-Hellman, eliminating the man-in-the-middle attack. Diffie-Hellman is included in the EC-MQV algorithm as a subset.

6.7.5 GARUDA

The proposed system uses a cluster based approach for data collection and management. An optimized version of Low Energy Adaptive Clustering Hierarchy (LEACH) is used for clustering. The cluster head aggregates the data among cluster nodes and manages the cluster. The cluster head is re-elected periodically [41, 54]. Localized Encryption and Authentication Protocol (LEAP) is modified and used for hierarchical key management [5, 67]. The need for key exchange is evaded by storing unique ID, Individual key and

Global Key which is stored in network nodes before their deployment. The Global key is used for pairwise key generation and cluster key formation. The Global key is obliterated after key formation.

The data is encrypted based on the previous agreement and nature of the information. The RC5 symmetric encryption technique used to encrypt the data [52, 53]. GARUDA consists of an additional unmanned vehicle unit along with the conventional system. The vehicle unit is controlled by a wireless sensor network mote known as the driver mote. The vehicle can navigate inside the network region without any human interaction. The vehicle can be equipped with sensors or other costly devices for monitoring the region. The vehicle can cross check the validity of information shared by network nodes and monitor the network traffic to detect the anomalies. The vehicle mote has direct high power communication with the base station to send the sensitive information without any hindrance. It can also interact with network motes to challenge the nodes to prove their identity without sharing the secret information. Table 6.3 shows the comparison of GARUDA with existing security systems.

6.8 Summary

The unmanned vehicle navigation was successfully implemented. The unmanned vehicle can solve various research problems in wireless sensor network. The unmanned vehicle solves the security issues in wireless sensor network. The Intrusion Detection system was successfully implemented in wireless sensor network. The GARUDA is successfully implemented in WSN and GARUDA architecture was compared with existing architectures.

Properties	SPINS [46]	TinySec [9]	LEAP+ [67]	SM [24]	GARUDA
Encryption	CTR	CBC	RC5	ECC	RC5
Freshness	Yes	No	No	No	No
Key Agreement	Delayed Disclosure	Any	Pre -deployed	EC-MQV	Pre -deployed
Clustering	No	No	No	No	Yes
Authentication	Yes	Yes	Yes	Yes	Yes
Localization	No	No	No	No	Yes
Time synchronization	No	No	No	No	Yes

Table 6.3: Comparison of Security Architectures

Chapter 7

Conclusion and Future Scope

This chapter summarizes the contribution towards this thesis and draws several useful inferences and suggests several problems for future investigation.

7.1 Conclusion

The awesome world of wireless sensor network with its security domain is a great field for novel research work. The nodes in WSNs suffer severe resource constraints due to their limited processing power, limited memory and energy. Since these networks are usually deployed in remote places and left unattended, they should be equipped with security mechanisms to guard themselves against attacks such as node capture, physical tampering, eavesdropping, denial of service, etc., Unfortunately, traditional security mechanisms with high overhead are not feasible for resource constrained sensor

nodes. The proposed approach is to bring a practical solution to the problem with out much overhead to the existing software and hardware.

A detailed review of the major constraints of WSNs had been performed. These constraints were the blocking factors for implementing a reliable security mechanism in a WSN. A detailed analysis of the security requirements in these networks were reviewed. Major threats and various possible attacks against wireless sensor network were described. The existing security mechanisms and the corresponding countermeasures were explained. A holistic view of the security issues were presented. These major research areas in WSN security can be classified into six categories: cryptography, key management, secure routing, secure data aggregation, intrusion detection and trust management. For each category there are various protocols to ensure the various objectives of wireless sensor network security.

The implementation of the Vehicle Navigation system controlled by wireless sensor network is a major achievement. The vehicle can travel around the network and monitor the network. This introduces a dynamic outlook to the otherwise static nature of a wireless sensor network. The wireless sensor network mote placed inside the vehicle is connected to the vehicle through an interface card. The mote is programmed to control all the vehicle movements using hardware interface unit. The crossbow mote acts like the driver of the vehicle controlled by base station commands and the neighbour node messages. The navigation of the vehicle is achieved by Location information and Neighbour mote sensor readings. The selection of Localization algorithm is based on the application usage. The Implementation is carried out by using a toy car with five controls.

The vehicle operations are controlled by a Micaz mote connected to it. The mote programming is done by using nesC language.

The premier concept of unmanned vehicle controlled by wireless sensor network in Military application had been introduced by this research work. This vehicle can navigate through the network and monitor the network security. As the vehicle is tiny, it does not catch the attention of the enemy. The vehicle controlled by a mote placed inside can monitor the truth of any event reported to have happened in an area. The vehicle node enjoys battery recharge and maintenance can be periodically performed. Other facilities like camera and costly sensors in the vehicle can be appended to the system. The importance of localization is to relate the vehicle location with local map or global map. If GPS is used, the global position is obtained and the next movement of the vehicle can be planned accordingly. The driver mote inside the vehicle send the location information to the other motes. The surrounding motes reply to the mote about the next movement.

The Intrusion Detection System designed is able to provide a top-notch security mechanism without the burden of complex computational tasks. The movement of the vehicle through the system has vanquished the threats posed by the solitary nature of the network. The proposed system is able to perform splendidly in identifying and eliminating all possible kinds of threats which can be posed by an intruder. The vehicle motes and the driver motes certify the verity of the reported events. The Driver mote provides a classy solution to the localization and time synchronization of the entire network. The system can also be used as a trusted third party for key distribution. The attacks against the vehicle must be addressed in order to ensure the overall system security. The time required for Intrusion detection is substantially reduced. The system is a sure

shot solution for the wireless sensor network threats, bettering the security ambiance of the system.

The proposed system brings together a general architecture called Guarding Architecture for Unattended Deployment Applications (GARUDA) for Ad Hoc network Security. A cluster-based approach is used to classify the network nodes based on functionality and priority. A key Pre-distribution technique is used to protect the key-management schema. The modified Localized Encryption and Authentication Protocol are used for hierarchical key management. The Rivest Cipher 5 (RC5) algorithm is used for encryption of sensitive data. The system has an unmanned vehicle with sensors to protect the network from attacks and report the malicious activities to the base station.

The GARUDA architecture has been successfully implemented in wireless sensor networks, set up by micaz motes. The RC5 algorithm was programmed using nesC language in micaz mote for encrypting the sensitive data. The highly confidential data can be directly sent to the base station from the vehicle unit. The absence of key exchange and presence of individual key makes the system sheltered. The GARUDA architecture can be used in any resource constraint monitoring application of unattended nature to make it secure. The vehicle unit can resolve network problems like time synchronization and localization in Ad hoc Networks. The following are the summary of the major contributions

- Development of tiny unmanned vehicle controlled by a wireless sensor network node.
- Unmanned tiny vehicle navigation inside the wireless sensor network region.

- Feasibility study on unmanned vehicle navigation for military application.
- Development of an Unmanned Tiny Vehicle Based Intrusion Detection System for Wireless Sensor Networks.
- Development of a Guarding Architecture for Unattended Deployment Applications of Ad Hoc Networks : GARUDA
- Implementation of RC5 algorithm in nesC.
- Development of Vehicle navigation algorithm inside a rectangular region.
- Simulation of LEACH clustering algorithm using ns2

7.2 Future Directions

A lot of research is still possible on the field of wireless sensor security. The field opens up a variety of unanswered questions to a challenging researcher. This section provides some future enhancement in purview of the thesis's future directions that are beyond the scope of this work.

- The GARUDA architecture implemented for wireless sensor network can be applied to other unattended deployment applications.
- The advanced unmanned vehicle proposed for military application can be built and used to secure the military troupes. The vehicle can be used as a watch dog in various monitoring applications.

- The unmanned vehicle can be used to solve other issues in wireless sensor network such as localization, Time synchronization and Data aggregation.
- Unmanned vehicle navigation based security in wireless sensor network is an un explored area. The vehicle can be used to store and distribute keys. The vehicle can act as a trusted third party to distribute the keys. The complex cryptographic functions can be computed by using the vehicle.
- Costly and unique devices or services needed throughout a WSN can be placed in the vehicle and the vehicle can navigate the entire network to provide the required service.

Appendix A

List of Notations

WSN	Wireless Sensor Network
GPS	Global Positioning System
MANET	Mobile Ad hoc Network
RF	Radio Frequency
GARUDA	Guarding Architecture for Unattended Deployment Applications
RBS	Reference Broadcast Synchronization
FTSP	Flooding Time Synchronization Protocol
TPSN	Time Synchronization Protocol for Sensor networks
RAM	Random Access Memory
ACK	Acknowledgement
GF	Geographic Forwarding
IDS	Intrusion Detection System
CH	Cluster Head
BS	Base Station
DM	Driver Mote
Mid	Mote Unique ID
Kg	Global Key

Ks	Secret Key
Kp	Pairwise Key
DMp	Driver mote position
Mp	Mote position
PKI	Public Key Infrastructure
ECC	Elliptic Curve Cryptography
SM	Security Manager

.

Appendix B

List of Publications Related to This Thesis

Part of the work presented in this thesis has been published in journals

1. Arun Madhu., and A. Sreekumar."Wireless Sensor Network Controlled Vehicle Navigation System and Its Applications", International Journal of Information Processing.Vol-7, No-2, P-32-40, 2013, ISSN: 0973-8215.

2. Arun Madhu., and A. Sreekumar."Guarding Architecture for Unattended Deployment Applications of Ad Hoc Networks: GARUDA", Indian Journal of Science and Technology.Vol-9, No-33, P-1-7, September, 2016, ISSN: 0974-6846.

3. Anu Kunjumon., Arun Madhu., and Shalin Elizabeth."Clone Detection in Wireless Sensor Networks using RSS and TOA.", International Journal for Scientific Research and Development.Vol-3, No-9, P-612-618, 2015, ISSN: 2321-0613.

Part of the work include in the thesis has been presented in various National/International conferences

1. Arun Madhu., and Sreekumar A. "Wireless Sensor Network Security in Military Application using Unmanned Vehicle". National conference in "Network Security "(NCNS), 2014, IHRD adoor, Kerala, India.

2. Anu Kunjumon., and Arun Madhu. "Survey of Spoofing Attack Detection in Wireless Networks". International conference in "Network Security"(ICCEECON-2k15), 2015.

3. Jini Raju., and Arun Madhu. "Secure PHR Sharing with Secure Data Transmission in Heart Monitoring Systems". International Conference, ACCIS-14, Published in ELSEVIER conference Proceedings. TKM, Kollam, Kerala, India.

4. Jerrin Sebastian., and Arun Madhu. "Secure PHR Sharing with Secure Data Transmission in Heart Monitoring Systems". International Conference, RAEREST-16, Published in ELSEVIER conference Proceedings. SJCET, Pala, Kerala, India.

Bibliography

[1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, *A survey on sensornetworks*, IEEECommunicationsMagazine **40** (2002), no. 8, 102–114.

[2] Javed Aslam, Zack Butler, Florin Constantin, Valentino Crespi, George Cybenko, and Daniela Rus, *Tracking a moving object with a binary sensor network*, ACM Conference on Embedded Networked Sensor Systems, ACM, November 2003.

[3] T. Aura, P. Nikander, and J. Leiwo, *Dos-resistant authentication with client puzzles*, 8th International Workshop on Security Protocols, Springer, 2001.

[4] U.C. Berkele, *Getting started with tinyos and nesc*, 2007.

[5] S. Blackshear and R. Verma, *R-leap+: Randomizing leap+ key distribution to resist replay and jamming attacks*, ACM Presss, 2010.

[6] R. Blom, *An optimal class of symmetric key generation systems*, Advances in Cryptology, Proceedings of EUROCRYPT'84, LNCS, 1995.

[7] P. Brutch and C. Ko, *Challenges in intrusion detection for wireless ad-hoc networks*, The Symposium on Applications and the Internet Workshops, SAINT'03 Workshops, 2003.

[8] Erdal C. and Rong C., *Security in wireless ad hoc and sensor networks*, John Wiley, 2009.

[9] Karlof C., Sastry N., and D Wagner, *Tinysec: A link layer security architecture for wireless sensor networks*, Proceedings of the 2nd International Conference on Embedded Networked Sensor Systems, ACM Press, November 2004, 162-175.

[10] D.W. Carman, P.S. Krus, and B.J. Matt, *Constraints and approaches for distributed sensor network security*, Technical Report NAI Lab, Glenwood, 2000.

[11] Xiangqian Chen, Kia Makki, Kang Yen, and Niki Pissinou, *Sensor network security: A survey*, IEEE Communications Surveys and Tutorials, IEEE, April 2009.

[12] J. Denga, R. Han, and S. Mishra, *Countermeasures against traffic analysis in wireless sensor networks*, Technical Report CU-CS, University of Colorado at Boulder, 2004.

[13] S. Duri, M. Gruteser, and X. Liu etc., *Framework for security and privacy in automotive telematics*, 2nd ACM International Workshop on Mobile Commerce, ACM, 2000.

[14] L. Eschenauer and V. Gligor, *A key-management scheme for distributed sensor networks*, Proc. of ACM, CCS, 2002.

[15] Yuguang Fang, Xiaoyan Zhu, and Yanchao Zhang., *Securing resource-constrained wireless ad hoc networks*, IEEE Wireless Communications, IEEE, April 2009.

[16] David Gay, Philip Levis, Robert von Behren, Matt Welsh, Eric Brewer, and David Culler, *The nesc language: A holistic approach to networked embedded systems*, ACM SIGPLAN, 2007.

[17] M. Gruteser and D. Grunwald, *Anonymous usage of location-based services through spatial and temporal cloaking*, 1st International Conference on Mobile Systems, Applications, and Services (MobSys), USENIX, 2003.

[18] M. Gruteser, G. Schelle, A. Jain, R. Han, and D. Grunwald, *Privacy-aware location sensor networks*, 9th USENIX Workshop on Hot Topics in Operating Systems, HotOS IX, 2003.

[19] N. Gura, A. Patel, A. Wander, H. Eberle, and S. Shantz, *Comparing elliptic curve cryptography and rsa on 8-bit cpus*, Workshop on Cryptographic Hardware and Embedded Systems (CHES '04), CHES, 2004.

[20] C. Hartung, Balasalle, and R. Han, *Node compromise in sensor networks: The need for secure systems*, Department of Computer Science, University of Colorado, 2004.

[21] M Healy, T Newe, and E Lewis, *Wireless sensor node hardware: A review*, 7th IEEE Conference on Sensors, 2008.

[22] Nayana Hegde, Dr.Sunilkumar, and S.Manvi, *Simulation of wireless sensor network security model using ns2*, International Journal of Latest Trends in Engineering and Technology **4** (2014), no. 1, 113–119.

[23] J. Hwang and Y. Kim, *Revisiting random key pre-distribution for sensor networks*, ACM Workshop on Security of Ad Hoc and Sensor Networks, SASN'04, December 2004.

[24] Heo J and Hong C.S, *Efficient and authenticated key agreement mechanism in low-rate wpan environment*, International Symposium on Wireless Pervasive Computing 2006, IEEE 2006, January 2006.

[25] D Jiagen, S Y Cheung, C W Tan, and V Pravin, *Signal processing of sensor node data for vehicle detection*, IEEE Intelligent Transportation Systems, 2004.

[26] Michael Johnson, Michael Healy, Pepijn van de Ven, Martin J Hayes, John Nelson, Thomas Newe, and Elfed Lewis, *A comparative review of wireless sensor network mote technologies*, IEEE SENSORS Conference, 2009.

[27] Jeonghoon Kang, Jongmin Hyun, Dongik Kim, Kooklae, Pil Mhan Jeong, Taejoon Choi, and Sukun Kim, *Tracking vehicles in a container terminal*, SenSys'11, November 2011.

[28] Tufan Coskun Karalar, *Implementation of a localization system for sensor networks*, University of California, Berkeley, 2002.

[29] C. Karlof and D. Wagner, *Secure routing in wireless sensor networks: Attacks and countermeasures*, 1st IEEE International Workshop on Sensor Network Protocols and Applications, IEEE, May 2003.

[30] L. Lazos and R. Poovendran, *Serloc: Robust localization for wireless sensor networks*, ACM Transactions on Sensor Networks **1** (2005), no. 1, 73–100.

[31] D Li, K D Wong, Y H Hu, and A M Sayeed, *Detection, classification, and tracking of targets*, IEEE Signal Processing Mag. **19** (2002), no. 1, 17–29.

[32] D. Liu and P. Ning, *Efficient distribution of key chain commitments for broadcast authentication in distributed sensor networks*, 10th Annual Network and Distributed System Security Symposium, San Diego, CA, 2003.

[33] D. Liu and P. Ning, *Multilevel tesla: Broadcast authentication for distributed sensor networks*, Transactions on Embedded Computing Systems **3** (2004), no. 4, 800–836.

[34] Arun Madhu and A. Sreekumar, *Wireless sensor network controlled vehicle navigation system and its applications*, International Journal of Information Processing **7** (2013), no. 2, 32–40.

[35] Arun Madhu and A. Sreekumar, *Wireless sensor network security in military application using unmanned vehicle*, IOSR Journal of Electronics and Communication Engineering, NCNS, December 2014.

[36] D.J. Malan, M. Welsh, and M.D. Smith, *A public-key infrastructure for key distribution in tinyos based on elliptic curve cryptography*, 1st IEEE International Conference on Sensor and Ad Hoc Communications and Networks, IEEE, 2004.

[37] MSI. Mamun and AFM. Kabir, *Hierarchical design based intrusion detection system for wireless ad hoc sensor network*, International Journal of Network Security Appl (IJNSA) **2** (2010), no. 3, 102–117.

[38] Guoqiang Mao, Barys Fidan, and Brian Anderson, *Wireless sensor network localization techniques*, University of California, Berkeley, 2003.

[39] Hangzhou Silan Microelectronics, *Tx2b/rx2b, toy car remote controller with five functions*, Hangzhou Silan Microelectronics, 2005.

[40] D. Molnar and D. Wagner, *Privacy and security in library rfid: Issues, practices, and architectures*, ACM CCS, ACM, 2004.

[41] Saeid Mottaghi and Mohammad Reza Zahabi, *Optimizing leach clustering algorithm with mobile sink and rendezvous nodes*, International journal of Elctronics and communication, October 2014.

[42] J. Newsome, E. Shi, D. Song, and A. Perrig, *The sybil attack in sensor networks: Analysis and defenses*, 3rd International Symposium on Information Processing in Sensor Networks, ACM, 2004.

[43] D Niculescu, *Communication paradigms for sensor networks*, IEEE Commun. Mag. **43** (2005), no. 3, 116–122.

[44] T. Onel, E. Onur, C. Ersoy, and H. Delic, *Wireless sensor networks for security: Issues and challengaes*, Springer, Springer, December 2006.

[45] Lynch J P, *Overview of wireless sensors for real-time health monitoring of civil structures*, Proceedings of the 4th International Workshop on Structural Control, 2004.

[46] A. Perrig, R. Szewczyk, V. Wen, D.E. Culler, and J.D. Tygar, *Spins: Security protocols for sensor networks*, Wireless Networks **8** (2002), no. 5, 521–534.

[47] Adrian Perrig, John Stankovic, and David Wagner, *Security in wireless sensor networks*, ACM Communication, 2004.

[48] R. Di Pietro, S Guarino, and J Domingo-Ferrer, *Security in wireless ad-hoc networks a survey*, Computer Communications, Elsevier, 2014.

[49] R/C Car Products, *Hpi racing high performance r/c car products*, 2007.

[50] Khanna R., Liu H., and H Chen, *Reduced complexity intrusion detection in sensor networks using genetic algorithm*, Proc. IEEE international conference on communications, IEEE, 2009, 598-602.

[51] C S Raghavendra, K M Sivalingam, and T Znati, *Wireless sensor networks*, Springer, 2006.

[52] RL Rivest, *The rc5 encryption algorithm old*, Workshop on Fast Software Encryption, 1995.

[53] RL Rivest, *The rc5 encryption algorithm*, MIT laboratory for Computer Science, 1997.

[54] Roberto, Blanca, and Laura, *Survey on clustering techniques for mobile ad hoc networks*, Revista Facultad de Ingenieria journal **2** (2007), no. 6, 145–161.

[55] M. Sa, MR. Nayak, and Rath AK., *A simple agent based model for detecting abnormal event patterns in a distributed wireless sensor networks*, International Journal Comput Sci Security (IJCSS) **4** (2011), no. 6, 580–588.

[56] P Santi, *Topology control in wireless ad hoc and sensor networks*, Wiley, 2005.

[57] H. Sedjelmaci and M. Feham, *Novel hybrid intrusion detection system for clustered wireless sensor network*, International Journal Network Security Appl (IJNSA) **3** (2011), no. 4, 17–29.

[58] Jaydip Sen, *A survey on wireless sensor network security*, International Journal of Communication Networks and Information Security **2** (2009), no. 1, 55–77.

[59] R. E. Shannon, *Introduction to the art and science of simulation*, 30th conference on winter simulation, WSC98, December 1989.

[60] B Sinopoli, C Sharp, L Schenato, S Shaffert, and Sh S Sastry, *Distributed control applications within sensor networks*, Proceedings of the IEEE, IEEE, August 2003.

[61] He T, Huang C, Blum B, Stankovic J, and Abdelzaher T, *Range-free localization schemes for large scale sensor networks*, MOBICOM, 2003.

[62] Joo Valente, David Sanz, Antonio Barrientos, Jaime del Cerro, ngela Ribeiro, and Claudio Rossi, *An air-ground wireless sensor network for crop monitoring*, Sensors, 2011.

[63] Jiafu Wana, B. Hui Suob, and Hehua Yanb, *A general test platform for cyber-physical systems: Unmanned vehicle with wireless sensor network navigation*, Int. Conf. Advances in Engineering, Elsevier Procedia Engineering, 2011.

[64] Michael Winkler, Klaus-Dieter Tuchs, Kester Hughes, and Graeme Barclay, *Theoretical and practical aspects of military wireless sensor networks*, The Journal of Telecommunications and Information Technology.

[65] J Yick, B Mukherjee, and D Ghosal, *Wireless sensor network survey*, The International Journal of Computer and Telecommunications Networking **52** (2008), no. 1.

[66] Y. Zhang, W. Lee, and Y. A. Huang, *Intrusion detection techniques for mobile wireless networks*, Mobile Networks and Applications **9** (2003), no. 5, 545– 556.

[67] S. Zhu and S. Jajodia, *Leap: efficient security mechanisms for large-scale distributed sensor networks*, 10th ACM conference on Computed Communications Security, ACM, December 2003, 500-528.