Chapter 1: Introduction

1.1 Wireless Sensor Network

Various networks consisting of the wireless sensor devices are being used to collect data and distribute information about a variety of phenomena of interest .A wireless sensor device is a battery operated small device which can perform activities like sensing the physical quantities, storing the data and limited computation and signal processing. Advances in the technology had made it possible to reduce the cost, size, weight of sensors and had considerably increased their efficiency and accuracy. So making use of sensors with the advance networking technology gave rise to a new term called Wireless Sensor Network (WSN) which can be described as the network consisting of large numbers of wireless capable sensor devices that are capable of sensing, processing and transmitting information by working collaboratively to achieve the common objective.

Since each node in the WSN is a battery operated device thus have a limited capabilities in order to achieve the maximum through the nodes the nodes must send their data to a common location. A WSN has one or more sinks or base stations which collect the data from all the sensor devices These sinks are the interface through which the WSN interacts with the outside world. All the sensor nodes in the WSN can be arranged in two architecture namely flat and hierarchical WSN. In the flat WSN all the sensor nodes sends the sensed information to the sink directly by multi hop communication. In the hierarchical WSN the network is divided into number of clusters and a node in every cluster is selected for the communication with the base station or sink, this node in every cluster is called cluster head which sends all the sensed information received from the

nodes to the base station or sink. The difference between the two architecture is that in hierarchical sensor network the nodes can unicast, multicast, and broadcast the information to the sink but in flat sensor network all the nodes need to broadcast their information to the sink,. The two architectures are shown in the figure 1

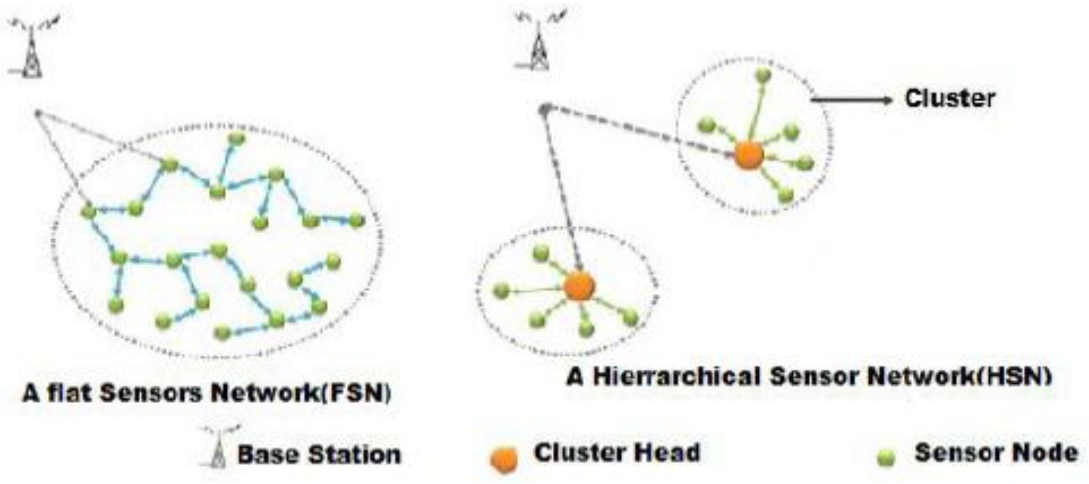

Figure1: Wireless Sensor Network Architecture

The prerequisite of WSN is to perform networked sensing using a large number of simple unsophisticated sensors this task can be performed by using the specialized and costly sensors but the WSN is preferred as it provides greater coverage, accuracy and reliability at a lower cost.

1.2 Salient features of WSN

The WSN have the following salient features which makes it stand in a different class of networks.

1. Large Density of nodes: In WSN sensor nodes can range from small numbers (10-20) to large numbers (100-1000) depending upon the application and area to be covered. Node density refers to the number of nodes in the communication range of a single node.

2. Tight limitations in energy, processing power and memory: Nodes in WSN are battery operated so there is limited power due to which there are tight limitations in energy, processing power and storage.
3. Collaborative objective: Each node in WSN senses the area of interest and informs the base station or sink about the activities in the region. To achieve this all the nodes work in a collaborative way rather than competing with each other as they use a multi hop communication to inform the base station about the activities in the region
4. Many to one communication paradigm: The main objective of the WSN is to monitor the signal of interest. These observations are then used by base station to decide the future course of action. Thus data flows in two direction UPSTREAM (Many –to- one) in this the sensor nodes sends the information to the base station or sink and other is DOWNSTREAM (One- to -many) in this the sink sends the queries or updates to the sensors nodes in the network. There is no any to any communication in the WSN (except during the packet forwarding).
5. Node mobility and dynamic topology: Mobility in WSN depends upon the application. There are applications like wildlife monitoring, military monitoring etc which requires high mobility due to which the topology of the network changes rapidly. The topology of network also changes due to the reasons like node failures, radio duty cycling (nodes entering into power saving mode) and environmental reasons like shadowing of surroundings, time fading etc.

1.3 Applications of WSN

WSN finds its application in many fields which can be grouped into the following categories

1. Event detection and reporting: It includes applications like intrusion detection as part of military surveillance, detecting anomalous behavior or failures in the manufacturing process and forest fire detection. All these applications deal with the infrequency of occurrence of the event of interest.
2. Data gathering and periodic reporting: It includes applications like monitoring the environmental conditions affecting the crops or livestock, monitoring temperature, humidity, lighting in the office building etc. All these applications require the constant monitoring of the area of interest which produces some amount of data which is sent to sink.
3. Sink initiated querying: Rather than each sensor node constantly reporting its measurements the sink could query the set of nodes for their measurement .This helps the sink to extract the information at a different resolution from different regions. All the monitoring applications can be included into the sink initiated querying.
4. Tracking based applications: Applications like border surveillance, movements of suspicious objects, tracking movements and patterns of insects, birds or small animals are included in this class of applications.

Overview of all the WSN applications can be shown by the following figure 2

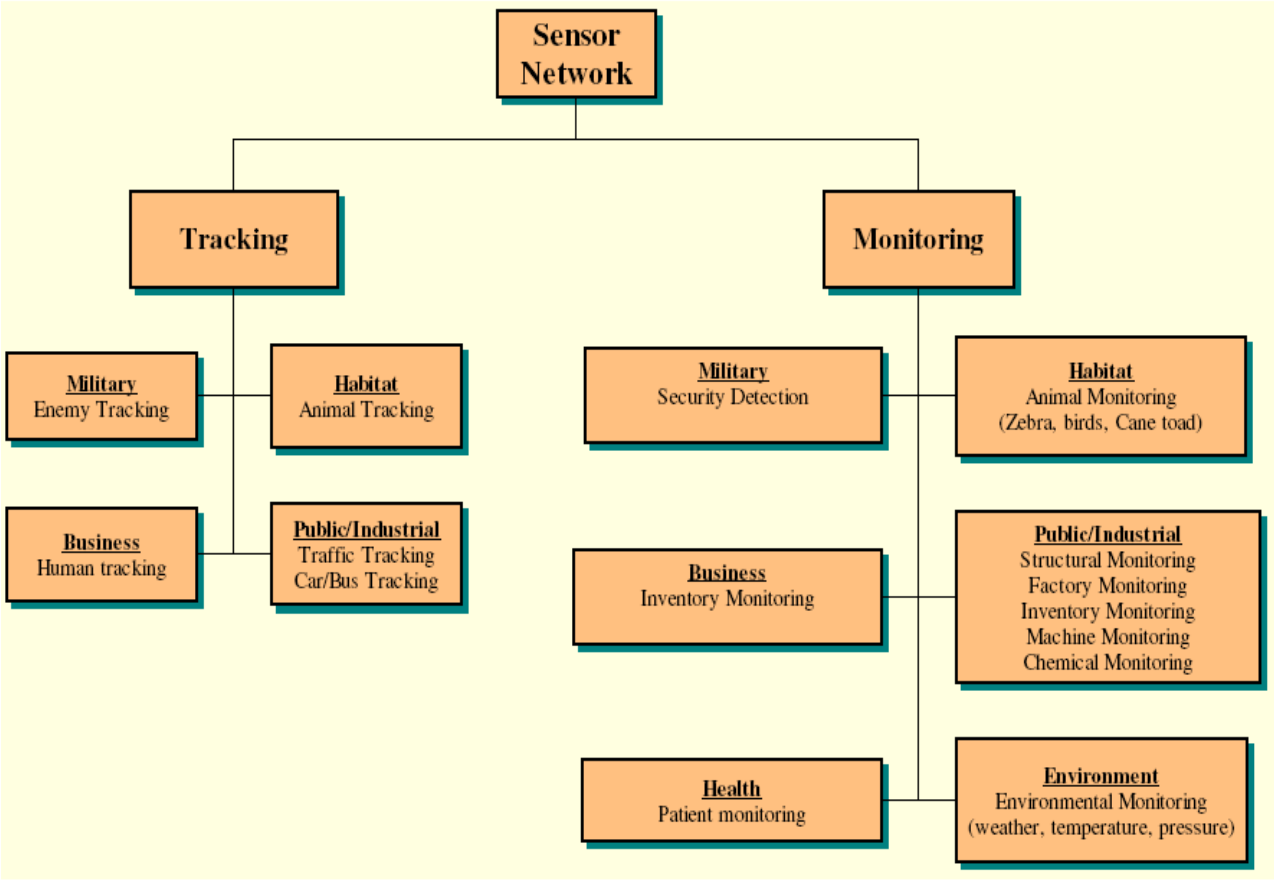

Fig. 2. Overview of sensor applications.

1.4 Security in WSN

Cryptography is used to provide the security for the two or more communicating parties. With the growth of the communication the risk of data theft and security has immensely increased. Cryptography uses the mathematical operations and algorithms for securing and preserving the secrecy of the data between the two or more parties communicating over the insecure or open channel. The four goals of cryptography are

1. Confidentiality: It is used to provide the security to the information from the unauthorized users i.e. hiding information from the unauthorized users.

2. Data Integrity: It is used to make sure the information sent to the authorized user is not being tampered or altered during the communication process ie protection of data from the alteration.
3. Authentication: It is used for providing the proof of the identity of a user or system to the other communicating party which doesn't have any knowledge about the user or system i.e. verification of the sender.
4. Non repudiation: It is used to prevent from the malicious users from hiding their inactivity.

Keys in cryptography play a very important role in securing the communication between the parties. On the basis of the keys used for providing the security the cryptography is of two types

1. Symmetric key cryptography: The both parties in communication uses the same keys for encryption and decryption.
2. Public key cryptography: The both parties use the different keys for encryption and decryption.

The WSN consists of small inexpensive battery operated wireless sensors which uses the wireless communication to interact with the sink. As the WSN is deployed where human observation or wired system can be inefficient, expensive, and dangerous or otherwise untenable [28] this application of WSN exposes them to various security threats which need to address for making WSN sustainable. The WSN constraints on the sensor nodes expose them to many security threats. The security in WSN is unique due the tight restriction on the nodes in the network so using the single key for whole of the network is not a good idea as the adversary can easily extract the key and can then attack the network. The public key cryptography is not used

in the WSN security context as it increases the storage, consumes more power [29] and increases the cost of the network due to these reasons it is not used in the WSN for providing the security. The constraint on the WSN exposes them to various security threats. The WSN constraints and their implications on the security of the network can be viewed from the table 1

No	Constraints	Implications
1	A node has severe hardware and resource constraints.	A node can't use the traditional cryptographic algorithms as they consume more energy and requires large storage.
2	A node has to operate in open environment	An adversary can compromise any node.
3	Nodes are not tamper resistant	An adversary can extract all the key material from the node once captured.
4	Lack of fixed infrastructure	A node can't assume a special purpose node in its vicinity.
5	No predefined topology	A node doesn't know its neighbors in advance.
6	Wireless communication	The communication channel is open to all and can be accessed by any one.

Table1: Constraints of WSN and their security implications.

1.5 Overview of the thesis

In this chapter we have given the brief introduction about the wireless sensor network and various security implications which arises due to the inherent features of the sensor network.

In the Chapter 2 we have described the key management concepts in the sensor network security and introduced various types of attacks which are vulnerable to the sensor network performance. The security mechanism is also discussed in details

In Chapter 3 we have given the brief introduction to the key management schemes and their problems in hierarchical wireless sensor network.

Chapter 4 is the literature review of the different work that has been done so far in the field to wireless sensor network security.

Chapter 5 describes our approach to enhance the security in the wireless sensor network. We described the weakness of the traditional approaches and further described our approach to overcome these weaknesses.

Chapter 6 discusses about our implementation, experiment, results and analysis. We have discussed the experimental setup and the various parameters for the network creation and functioning. Chapter 7 discusses about the conclusion and future scope of work to improve the accuracy of the proposed system. The next section shows the references of our work.

Chapter 2: Key Management and Security in WSN

2.1 Security in WSN

In WSN the security requirements like data confidentiality, integrity and authentication can be resolved by making a solid framework for key management. So in order to provide the solution for the basic security requirement the cryptographic keys are to be exchanged and that's where the efficient key management comes into play. Key management is one of the most important and basic aspect of WSN which provides the base for the various other secure mechanism like secure routing , secure localization etc.

2.2 Security vulnerability and requirements in WSN

The basic characteristics of WSN like wireless communication , open environment deployment etc posed them to various security vulnerabilities which can be capitalized by the adversary to compromise the node and affect the network security. So to provide a robust security to the network the following issues are needs to be handled carefully. There are six challenges in providing the security for the communication between the nodes in the network in WSN. The challenges are as follows:

1. No physical security: WSN is deployed in an open environment and consists of non tamper resistant material thus attacker can capture sensor node to extract the information or even reprogram them to perform the malicious activities. Due to the collaborative self configuring nature of the nodes other nodes in the network can also be significantly affected if the node is being compromised. In the worst case whole of the network can fall into the adversary's hand.

2. Wireless radio communication: Nodes in the WSN uses wireless communication to communicate with each other and since wireless communication is open to all and can be accessed by adversary and a false data can be injected into the system.
3. Unique traffic pattern: The flow of traffic in WSN is different from the wired network. In WSN communication is many-to-one due to which adversary can easily analysis the data flow which would be heavier near the base station can direct his attacks towards base station which would be devastating.
4. Limitation of conventional cryptography: The conventional cryptographic algorithms can't be used in case of WSN as they require more energy, storage and are expensive to implement in the scenarios of WSN application.

5. No fixed infrastructure: The random infrastructure of the WSN makes it difficult to devise a fixed and robust strategy for providing the security to the network. There is no prior knowledge of the topology before the node is being deployed. So to develop a scheme that addresses the lack of infrastructure while providing security to the network is a daunting task.
6. High node density: Node density in WSN means the number off nodes in the communication range of the other nodes. Since the node density is high in WSN it arises the issue of authenticating the nodes and restricting the entry of malicious node in the network.

2.3 Security Requirements in WSN

WSN are more vulnerable to attacks as compared to the wired networks due to the inherit characteristics of the sensor network which are discussed above. As the characteristics of WSN are unique to them similarly there security requirements are also unique to them. The WSN has close resemblance to the ad-hoc networks thus the general security requirements of WSN are as follows:

1. Availability: It ensures that service offered by the complete WSN or by any part of it, or by a single sensor node must be available whenever required.
2. Authentication: It ensures that all the nodes in the network and base stations are authenticated before granting a limited resource, or revealing any information.
3. Integrity: It ensures that message or the entity under consideration is not altered in any way.
4. Confidentiality: It provides the privacy of the wireless channel to prevent eavesdropping.
5. Non-reputation: It ensures that no malicious node is able to hide their activities in the network.

Apart from the general requirements the WSN has some specific requirements which are limited to them only. They are as follows:

1. Survivability: It is the ability of the network to provide the minimum level of service even in the presence of the faults like power loss, failures or attacks etc.
2. Degradation of security services: It is the ability of the network to change security level with the change in the resource availability.

These security requirements can be provided by a solid key distribution mechanism and are also used as the metric for evaluating the key distribution scheme in the sensor network. They are as follows:

1. Scalability: It is the ability to support larger networks. Key distribution mechanism must support large networks, and should be flexible enough against the substantial increase in the size of the network even after deployment.
2. Efficiency: It should consider the limitations of the sensor nodes like storage, processing and communication limitations.
3. Storage complexity: It is the amount of memory required to store security material in the nodes.
4. Processing complexity: It is amount of processor cycles required to establish a key among the nodes in the network.
5. Communication complexity: It represents the number of messages exchanged during a key generation process
6. Key connectivity (probability of key-share): It shows the probability that two (or more) sensor nodes store the same key or keying material. Enough key connectivity must be provided for a WSN to perform its intended functionality.
7. Resilience: It is resistance against node capture. Compromise of security credentials, which are stored on a sensor node or exchanged over radio links, should not reveal information about security of any other links in the WSN. Usually higher resilience means lower number of compromised links.

In general, resource usage, scalability, key connectivity and resilience are conflicting requirements; therefore, trade-offs among these requirements is done depending upon the goal of the application.

2.4 Security Attacks on WSN

A lot of attacks have been devised for WSN[25] which harasses the security vulnerabilities of the sensor network. Some attacks are similar to the other wireless and ad-hoc networks and some are specific to the sensor network. The attacks on sensor network can be active or passive. The active attacks are those that cause harm to the data in the network i.e. they modify or delete some data. Passive attacks are those that don't cause the any harm to the data but affects the availability of the data to the network and hinders the normal functioning of the network. The various attacks on the WSN[research paper on security survey] are as follows:

1. Eavesdropping: This is the passive attack on the wireless sensor network in which the attacker tries to capture some data by listening to the network. If the data is sent without any encryption the attacker can easily read it. Since this attack doesn't modify the data in the network they are hard to detect.
2. Radio jamming: This is an active attack which falls under the category of Denial Of Service (DOS) attacks. In this type of attack the attacker broadcasts the high energy radio signals at the same frequency as that of the network and prevents the nodes to communicate to each other over the wireless channel.
3. Message injection: This is an active on the network under which the attacker sends the false information messages in the network to corrupt the information or to saturate the network performance.

4. Message replication: It falls under the active attack category. In this type of attack the attacker captures the already sent message and resent those message back in the network for example an attacker catches a packet containing the fire detection information and some days later he resends the same packet back in the network to create a false alarm for the fire detection in the sensing area.
5. Node compromise (destruction or theft): Since the sensor nodes are deployed in an open environment, there is chance for the node to be physically attacked and destroyed or may be even theft. An attacker can extract the cryptographic data from the nodes and can reprogram them to work according to him. The reprogrammed nodes that the attacker puts in the network are named as malicious node.
6. Denial of service (DOS) attack: This is an active attack under which the attacker uses one or more malicious nodes or devices with high signal strength to flood the network with some message. It is similar to the denial of service attack in the traditional networks. The attack drains the energy of the nodes in the network and makes the network go out of order.
7. Hello flooding attack: Many routes discover protocols in the ad hoc network uses the Hello message to discover the neighbors and automatically creates the network. Every node who receives this hello message marks its sender as their parent this process goes on recursively. In this attack an attacker spoof or replay one of the hello message thus making itself the root of the tree and potentially excluding the base station from the data flow. This attack creates the illusion among the nodes that they are in the normal transmission range of the sink and sends their data to the malicious node for example in the figure the malicious node with the powerful transmission device sends the hello

message packets to the neighbors nodes V. The node believes that they are in the normal transmission range of the sink and passes the information to the malicious node.

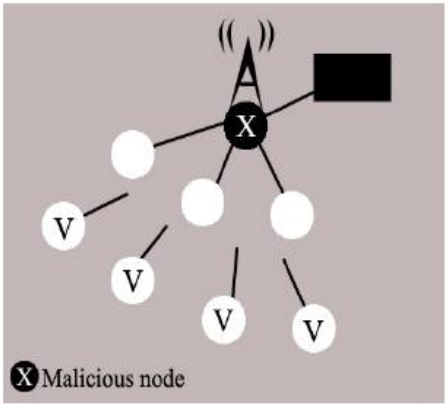

Figure 3: Hello flooding attack

8. Black Hole Attack: This is the devastating attack on the sensor network in which an attacker first insert a malicious node in the network and then by some means updates the routing table of the node by becoming the cluster head of the cluster. The main motive of the attacker is to attract maximum number of nodes to send the data to the malicious node planted in the network. Once the malicious node received the data from the nodes it doesn't forwards it to the sink. For example see figure 4 in which the malicious node x which created the black hole attack by updating the routing table of the cluster 1, 2, 3 and 4.

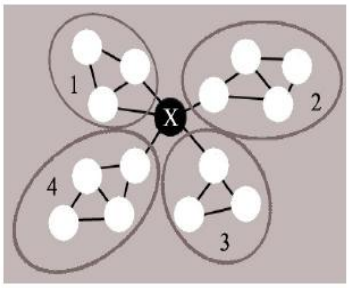

Figure 4: Black hole attack on cluster network

9. Selective Forwarding Attack: This is also known as grey- hole effect and it's a variant of the black hole attack. In this type of attack the malicious node selectively forwards some packets while dropping others. An adversary interested in suppressing or modifying packets originated from selected nodes can reliably forward the remaining traffic and limit suspicion of its misbehavior. These types of attacks are most effective when the attacker is explicitly included on t he path of a data flow.
10. Wormhole attack: This type of attack requires the adversary to insert two connected malicious node in the network. These two nodes create an illusion among the other nodes regarding the distance between the sink. The malicious node projects that they have the shortest distance to route the information and fools the neighbors about the actual distance. Usually the routing protocols look out for the route which has the small number of hops, in the wormhole attack the two malicious nodes successfully projects that they have the shortest distance route and thus collect the data sent from the neighbors. The wormhole attack is shown in figure 5 in which two malicious nodes X1 and X2 are connected by a powerful connection, making a wormhole. The nodes A and B choose the

shortest path provided by the wormhole for sending their data. Data will be captured by the malicious nodes and then by the attacker.

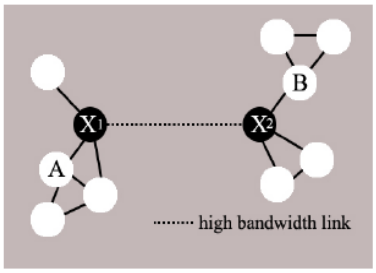

Figure 5: Wormhole attack

11. Sinkhole attack: In this attack the adversary attack the data near the sink directly by projecting the shortest path to the sink, since the communication is heavier near the sink so by offering a smaller route to the sink the malicious node gets hold of the data. The sinkhole attack can be seen from the following figure 6

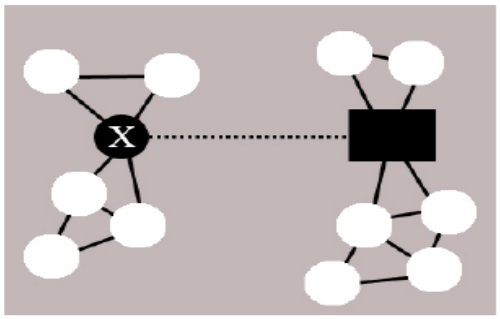

Figure 6: Sinkhole attack

The attacker can use the sinkhole attack with the wormhole attacks. The wormholes can be used to cover all the nodes in the network and a malicious node near the sink can be planted to execute the sinkhole attack. In figure 7 the malicious nodes X1, X2, and X3 are connected with powerful connections and make wormholes. X3 is connected to the sink with a powerful

connection to make a sinkhole attack. This is known as a sphere of influence exerted by the attacker on the network, because it is then able to recover all the information circulating in the wireless sensor network.

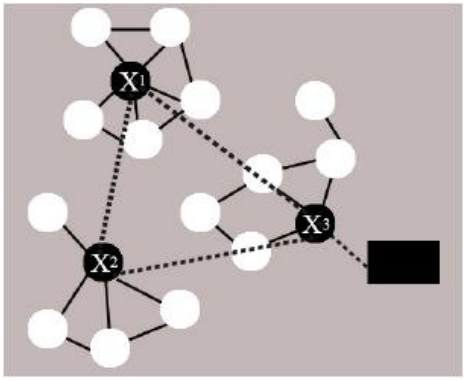

Figure7: Sinkhole attack with wormhole attack

12. Sybil attacks: In this attack the adversary pretends to be several different nodes. These attacks are easy to implement in the sensor network as there is no trusted identifier to authenticate the node in the network.
13. Sleep deprivation torture: This is an active attack on the senor network in which the attacker makes sure that the nodes doesn't go into the sleep mode by sending unnecessary data or by making them do calculations in this way the sensor node goes out of battery soon.

2.5 Security Mechanism

To counter these attacks various measures have been proposed by the researchers that takes into account the various constraints of the sensor network. The list of measure is exhaustive specific for the application for which the sensor network is deployed. Some of the state of the art measures that are simple and consume less energy are as follows:

1. Data partioning: It is a simple scheme under which the information of the network is secured by dividing it into several segments and routing these segments through the different routes. By dividing and sending the packets through the different routes makes it difficult for the adversary to capture whole of the information and increases the complexity for the attacker as to catch all the information the attacker has to listen to complete network. The only drawback for this solution is that it requires more computation which consumes the energy.
2. Cryptography: The use of cryptography is computationally too expensive for the sensor network so the measure that includes the asymmetric key cryptography is usually not applied to the sensor network. Though the symmetric key cryptography scales up with the constraints and thus is used to provide the counter measures.
3. Key management: The security measures using the key management can be achieved by applying the following four techniques [26].
    1. Global key: It is the single shared key for the entire network. Nodes use this key to the send information. All the information is encrypted using the global key. It is an energy efficient solution but provides the limited security; if the attacker gets the key by compromising a node then the security of whole network is at risk.
    2. Pair-wise key: It provides the unique key for communicating with each of the neighbors of the node i.e. if the node has n neighbors then it will require n unique keys to establish communication path between each of the node in the vicinity. It provides the increased security in the network but the storage overhead is more and also it is not scalable for the large network which includes thousands of nodes in the network.

3. Group keys: In this each group or cluster is given a key which is used to provide the security within that group. All the nodes belonging to a same group uses the same group key to communicate with each other. Cluster-heads use a single key for all cluster-heads to communicate or use a pair wise key to communicate between two cluster-heads. This solution is a hybrid solution to the first two techniques of encryption and offers a compromise between security and energy efficiency.

4. Individual key: In this solution each node as key which is shared only with the sink. The node can encrypt the information with their unique key and send the data to the sink which can later be verified by the sink as a result the message from the node goes hidden on the network until it reaches the sink. This solution provides security only between the sink and the node.

4. Stenography: The stenography is the technique for hiding the data from the attacker. This technique is not suitable to be implemented in the sensor network as the data in the network is not big enough.
5. Key generation: In this measure the sink generates a key for the entire network for a limited period or generation. Every node in the network that has this key certifies that this node belongs to this network. If any new node joining the network does not have the key it will not be allowed to join the network.
6. Localization: In this measure a node equipped with the GPS device called beacon node is used. The beacon node receives the request of the node who wants to join the network and it evaluates its location with their hearing area this will restrict attacks like

wormhole. Beacons will make a grid of their respective hearing area, and each beacon node, which received the request for entry in the network, will vote for an area of the grid that is able to hear. The area which receives the greatest number of votes will be supposed to be the area where is the new sensor.

2.6 Key management in WSN

Cryptography is the first line of defense to provide the message confidentiality, integrity and authentication since the public key cryptography is computationally too expensive to be used in the WSN .So most of the system uses the symmetric key cryptography to provide the security to network. In symmetric key cryptography key distribution and management is of prime importance. Key management is used to settle various kinds of keys in the network like individual keys, pair wise keys, group keys etc. It is the most important part of the network security upon which other security primitives are made. Most of the security requirements such as privacy, authenticity, and integrity can be addressed by making a robust key management framework. Moreover, in some deployment scenarios sensor nodes need to operate under adversarial condition. Security solutions for such applications depend on existence of strong and efficient key distribution mechanisms [25]. It is infeasible, or even impossible in uncontrolled environments, to visit large number of sensor nodes, and change their configuration. Moreover, use of a single shared key in whole WSN is not a good idea because an adversary can easily obtain the key. Thus, sensor nodes have to adapt their environments, and establish a secure network by:

1. Using pre-distributed keys or keying materials.
2. Exchanging information with their immediate neighbors.

3. Exchanging in-formation with computationally robust nodes.

Key distribution and management problem in WSN is difficult one, and requires new approaches.

2.7 Key management phases

Key management architecture in WSN consists of the four phases [28] which are described as follows:

1. Key Generation: It is the process in which a key pool is generated. It is usually done by a trusted authority in a offline mode.
2. Key Establishment: It is an important phase of key management process. It is the process by which shared keys are made available to the two or more legitimate users for secure communication. Basically it deals with creating the session keys between the two communicating parties. Key establishment can be done in many ways. Keys can be sent to the each user via a secure channel by the Trusted Authority. But such mechanism is usually not used in WSN as it is a costly .So, in WSN the Key Pre-distribution is used wherein the key rings are installed in the nodes before deployment of network in offline mode. It reduces the communication and doesn't require any centre for key distribution.
3. Key Updation: It is the process by which the keys used for the communication between the two or more parties are updated periodically so that the security of the communication is maintained.
4. Key Revocation: It is the process wherein the deletion of compromised keys is done.

2.8 Key management schemes for WSN

There are basically three general types of key management schemes [29] that are used in the sensor networks. They are as follows

1. Trusted server scheme: This scheme is implemented taking into an account a secure and trusted server such as base station for the key distribution among the nodes in the network. The server can assumed to be key distribution centre(KDC) .In this scheme the base station is most obvious choice for the trusted server and all the sensor nodes stores a implanted key so that if they get compromised not much information about the security can be reveled from the nodes. The drawback of this scheme is that if server is compromised the network will be insecure.
2. Self-enforcing scheme: These schemes make use of the asymmetric key cryptography approaches for key management since they are computationally too expensive to be incorporated into the WSN these schemes are rarely used. Algorithms likes RSA, Deffi-Hellman etc are used for exchanging the key in the network. The only advantage of these schemes is that if the nodes get compromised no information of the keys is revealed other then the ongoing present keys.
3. Key pre-distribution scheme: [26] In these schemes the key information is distributed to all the sensor nodes before the deployment. The key material is stored in their memory prior to the deployment. These schemes are used extensively in WSN as they can be used when the network topology is not known before the deployment. It is of three types

    1. Probabilistic key pre-distribution: In probabilistic solutions key material is randomly selected from a key-pool and distributed to sensor nodes.

2. Deterministic key pre-distribution: In deterministic schemes, deterministic processes are used to design the key-pool and the key-chains to provide better key connectivity.
3. Hybrid key pre-distribution: The hybrid solutions use probabilistic approaches on deterministic solutions to improve scalability and resilience.

Chapter 3: Key Distribution in Hierarchical WSN

A hierarchical wireless sensor network consists of one or more base station and the sensing area is divided into smaller regions called clusters. A node from every cluster is selected as the cluster head which is responsible for communication between the base station and the cluster. All the nodes in the cluster send their data to the cluster head which is further sent to the base station. Since the HWSN supports unicast, multicast and broadcast form of communication, the distribution of keys in such a communication scenario is difficult task.

So to provide a efficient solution for key distribution in HWSN three approaches are used to distribute keys in the network. Every scheme has its own advantages and disadvantages. The schemes are namely pair-wise key distribution, group-wise key distribution and network-wise key distribution.

3.1 Pair-wise key distribution

In hierarchical WSN, base station to sensor or sensor node to base station unicast communication requires pair-wise key. Distributing keys in such a scenario requires base station to establish a distinct pair-wise key with each of the sensor node in the network[22]. In this type of keying model N-1 keys are required for each node where N is the size of the network. Pair-wise key distribution provides ultimate robustness against the node capture attack because compromise of the single node does not affect any other node in the network as every node has distinct key. This key distribution scheme is not scalable for the large network, considering that each node stores N-1 keys so the total number of unique keys in the network will be N(N-1)/2 which grows at the rate of N^2 ;this is not manageable when N becomes considerably large. Another issue with pair-wise key distribution is that it lacks the flexibility when a new node

enters the network every node requires a new key to communicate with it. The pair-wise key distribution is a resource intensive process that requires more energy of the sensor node for the distribution of keys. Similarly the key revocation and key updation suffers from the same problem. Additionally the accessibility requirement is in jeopardy as nodes cannot passively monitor event signals. Lastly, in the case of some pair-wise key distribution schemes, self-organization comes into question, because they tackle the scalability problem by reducing the number of shared keys, resulting in some nodes being unable to communicate with others and compromising the self-healing and self-organizing abilities of the network. There are many schemes which have been proposed as solution to the problems faced by the pair-wise key distribution. Some of the state of art schemes proposes the similar solution like in Perimeter protection scenario[22], Base station authentication protocols [15], and Localized encryption and authentication protocol(LEAP) [19]. Since the base station shares pair-wise keys with sensor nodes, it can intermediate establishment of a pair-wise key between any pair of sensor nodes. Similar approach is used in ESA [29] where sensor nodes are separated into domains which are supervised by base stations. SNEP [28] proposes each pair of communicating party S_A and S_B to share a master secret key $X_{A,B}$ and a PRF. S_A and S_B can then generate encryption keys $K_{A,B}$ = PRF($X_{A,B}$,1) and $K_{B,A}$ = PRF($X_{A,B}$, 3), and MAC keys $K'_{A,B}$ = PRF($X_{A,B}$, 2) and $K'_{B,A}$ = PRF($X_{A,B}$, 4).

Localized encryption and authentication protocol (LEAP) [19] proposes that each sensor node can establish pair-wise keys with its immediate neighbor. In the key setup phase, nodes receive a general key K_I . A node Su can use K_I and one-way hash function H to generate its master key Ku = HK_I (IDu). In shared key discovery phase, node Su broadcasts (IDu,RN_u) and a neighbor Sv responds with (IDv,MAC Kv (RN_u) |IDv)). Node Su can then generate the key by

Kv = HK_I(IDv), and both nodes Su and Sv can generate the session key by using

Ku,v = HKv (IDu). Multi-hop pair-wise keys may be required to reach cluster heads. In that case, node Su generates secret Ku,c, and finds m intermediate nodes. It divides the secret into shares Ku,c = sK_1 sK_2 ….sK_m, and sends each share through a separate intermediate node Svi (1 ≤ i ≤ m). Basically, node Su sends ENC$K_{ui,v}$(ski), H_{ski}(0) to node Svi , and Svi sends ENC$K_{vi,c}$(ski), H_{ski}(0)to cluster head Sc. Solution has high communication cost because Su sends m messages through m intermediate nodes to increase resilience. However, security of the system depends on the general key K_I which can be compromised by capture of a sensor node. It is possible to compromise all the session keys generated by LEAP once K_I is compromised.

3.2 Group-wise key distribution

In hierarchical wireless sensor network the group-wise keys are required by the nodes for secure multicast communication. The communication between the different groups requires different pair-wise keys which is similar to the pair-wise key distribution scheme. The asymmetric key cryptography can also be used in the group-wise key distribution but it is usually avoided due to its extensive computation which consumes the energy of the nodes. The pair-wise key structure can be used to generate the group keys in this manner the group of nodes have their accessibility requirement satisfied because the data aggregation can occur with no additional cost while maintaining some degree of robustness[27]. In this scheme if the node is compromised than in the worst scenario the entire cluster to which the node belongs will be compromised which is considerably more isolated than the entire network. The scheme is scalable because the number of keys increases with the number of groups, not with the size of the network. The main problem with this scheme is that it is difficult to set as the formation of the groups depends upon the

application and to efficiently distribute the keys the scheme would require the information regarding the group which is not an easy task.

It is also possible to use existing pair-wise key structure to establish groups-wise keys. In a hierarchical network, where a base station shares pair-wise keys with all the sensor nodes, base station can intermediate establishment of group-wise keys .Localized encryption and authentication protocol(LEAP) [19] provides a mechanism to generate group-wise keys which follows LEAP pair-wise key establishment phase. Node Su, who wants to establish a group key with all its neighbors S_{v1} , S_{v2} , . . . , S_{vm}, first generates a unique group key K_{gu}. It then sends K_{gu} to its neighbors S_{vi} as ENC$K_{u,vi}$ (K_{gu}). Security of the scheme depends on security of the pair-wise keys which in turn has very low resilience.

3.3 Network-wise key distribution

The network –wise key distribution scheme has advantage over the other two schemes. It is simple and easy to implement and uses less resources as compared to the other schemes. It is usually used to secure the broadcast traffic from the base station to sensor node. The use of single key for the whole network is an insecure approach and can jeopardize whole network if key is leaked. The best part of this scheme is that it makes it easy for the nodes in the network to collaborate as neighboring nodes can read and interpret each other's data easily and satisfying the self organization and accessibility requirements. It provides the excellent scalability and flexibility due to the use of the single key for the entire network which doesn't changes with the addition of nodes.

However the only unacceptable problem with this scheme is that it lacks robustness. Suppose one node is compromised by the adversary and the network wise key is exposed then adversary

using this key can eavesdrop on all messages in the network and can also inject fake messages to degrade the performance of the network or halting the proper functioning of the network.

The problems for the network-wise keys can be addressed by using the Timed Efficient Stream Loss-tolerant Authentication(TESLA)[28] which is multicast stream authentication protocol which uses delayed key disclosure mechanism where the key that is used to authenticate the ith message is disclosed along with the (i+1)th message. A version of TESLA which is used in hierarchical WSN is µ-TESLA which provides the authentication for data broadcasts and requires that the base station and the sensor nodes be loosely time synchronized. It introduces the needed asymmetry through the delayed disclosure of the symmetric keys. To send an authenticated packet the base station computes the MAC on the secure checksum by using the secret key and sends it to all the nodes in the network. Nodes stores this packet in there buffer and after some time which is estimated on the basis of the communication to all nodes the base station sends broadcasts the key to all the receivers .When the node receives the disclosed key it can verify using the previous key as each key is the part of the key chain that was generated by a public one-way function. SPINS[16] uses µ-TESLA[29] and employs the base station as the key distribution centre. The base station randomly selects the last key K_n of chain and applies the one-way public function H to generate the rest of the chain $K_0, K_1, K_2 \ldots K_{n-1}$ as K_i = H(K_{i+1}).Given K_i,every sensor node can generate the sequence of $K_0, K_1, K_2 \ldots\ldots K_{i-1}$. However given K_i no one can generate K_{i+1}. At ith time slot the base station sends the authentication message MAC K_i(Message).Sensor nodes stores the message in their buffer until the base station sends them a verification key in (i+i)th time slot. The sensor nodes can verify the disclosed verification key K_{i+1} by using the previous key K_i as K_i = H(K_{i+1}). In µ-TESLA the nodes are required to store the message until the authentication key is not disclosed by the base

station this creates the problem for storage in the sensor nodes which exposes them to attacks like DoS. An adversary can jam the key disclosure message to saturate storages of sensor nodes. The µ-TESLA [29] requires to bootstrap the sensor nodes from the base station i.e. they receive the first key of the chain which is called the key chain commitment. Bootstrapping procedure requires unicast communication and can be secured with pair-wise keys. µ-TESLA is used in LEAP [19]to update the pre deployed network-wise key in case of the node compromise. Another variant of TESLA is TESLA Certificate[28] in which the base station is used as the certification authority(CA) which generates the certificate for the senor node at any given time which consists it id and the MAC and the authentication key is disclosed after the time (i+d) when the certificates expires.

All these key distribution schemes have their advantages and disadvantages which can be trade off as per the application for which the wireless sensor network is being used. The overview of all these schemes can be viewed from the table 2

Model	Description	Benefits	Problems
Network	The entire network uses one shared secret key.	1. Simple 2. Allows data aggregation and fusion 3. Scalable 4. Able to self-organize 5. Flexible/accessible	Compromise of one node compromises the entire network (lacks robustness)
Pairwise	Each specific pair of nodes shares a different key.	1. Best robustness 2. Authenticates each node	1. No scalable – storage, energy, computation 2. Unable to self-organize 3. Not flexible for addition/removal of nodes
Group	Each group uses a different shared key.	1. Allows multicast 2. Allows group collaboration 3. Better robustness than network-wide keying 4. Adjustable scalability 5. Addition/removal of nodes possible 6. Able to self-organize within the cluster	1. Lacks efficient storage method for group keying in IEEE 802.15.4 2. Difficult to set up securely 3. Cluster formation information is application-dependent

Table 2: Overview of the Key distribution schemes

Chapter 4: Literature Review

Before the WSN can exchange the data securely the cryptographic leys must be established among the sensor nodes in the network. Key distribution refers to the distribution of multiple keys among the sensor nodes in the network which is typical in a non-trivial scheme .Key management is the broader term for the key distribution which consists of key setup , the initial key distribution and the key revocation. In this chapter we provide a brief overview of the literature on the key distribution.

The Eschenauer and Gligor[1] were the first one to propose a scheme for key management in the WSN. Their scheme was simple, elegant and provides an effective trade-off between the robustness and scalability. In their scheme they generated a key pool with total of n keys and every node prior to its deployment is randomly chosen with m keys from the key pool. The number of keys in the key pool are large enough for establishing the communicating path between the nodes i.e. (n>>m).After the deployment the nodes communicate with its neighbors to discover the common keys by exchanging the key ID's without actually giving up the cryptographic secrets. The variable m is tunable parameter which can be set as per the desired probability of two neighbors sharing a common key. Nodes that discover that they share a key can verify whether or not their neighbor actually holds the key through challenge /response. The shared key then becomes the key for that link. If two nodes don't share a common key then can establish a path key through the neighbors they share keys with. Also, neighbors without shared key can generate one and pass them to each other via an already secured path.

Chan et al [2] proposed a scheme on the key distribution which was the extension of the Eschenauer and Gligor scheme [1] but provided the three potential improvements to RKP.

The first is the q-composite random key pre-distribution scheme in which q common keys are hashed together to compute the shared key between the two nodes. The advantage of this approach is that since there are more possible variations of shared keys an adversary needs more compromised nodes as compared to the RKP[1]. However to maintain the same probability of connectivity requires large value for m. As a result single compromised nodes expose more keys of the set to the adversary.

Second they propose a multipath key reinforcement scheme in which message the divided into several fragments and each fragment is routed through a different path. Thus adversary needs at least one node in every path to collect the all the data. This scheme provides the security but on the other hand increases the overhead as compared to the base scheme [1].

Thirdly they proposed a random pair-wise key distribution scheme which provides the node- to-node authentication and also offer resilience against the node capture attack. In this scheme each sensor node stores a random set of N_p pair-wise keys to achieve probability p that two nodes are connected. At key setup phase, each node ID is matched with N_p other randomly selected node IDs with probability p. A pair-wise key is generated for each ID-pair, and is stored in both nodes' key-chain along with the ID of other party. Each sensor uses $2\,N_p$ units of memory to store its key-chain. At shared-key discovery phase, each node broadcasts its ID; therefore, each node ends one message, and receives one message from each node within its radio range. Neighboring nodes can tell if they share a common pair-wise key. This solution has very good

key resilience. It is more scalable in the sense that efficient use of memory spaces helps support larger WSNs. However, it sacrifices key connectivity to decrease the storage usage.

Another scheme which modifies the above scheme was proposed by **Du, Han and Varshney**[15]. There scheme was based on the pair-wise keying model which extends the Esch et al and Bloom's work by using the same paradigm as Esch et al[1] but instead of individual keys it uses the concept of Bloom's key matrix which is array of keys. In Du's scheme there are k key matrices in each node which are distributed randomly. Bloom's model is based on the idea of symmetric matrix multiplication where row*i* column *j* is equivalent to row *j* and column *i*. Thus when node calculates key ji the keys are identical leading to a commonly shared key. Bloom's scheme distributes the information for this calculation by the means of two matrices public matrix and private matrix. The Du's scheme instead of using only one private matrix the sink generates *i* private matrices and each node stores the subset of this matrix in a similar fashion as proposed by Esch[1]. In order to make two nodes communicate with each other they broadcast there ID's, indices of key matrices they have and the seed column of the public matrix. If they share a common key matrix then they can compute the pair-wise key using the Bloom's scheme. If they don't share common key matrix they will go in the path-key discovery phase to find a common third party to route the data. The advantage of the Du's scheme is that it provides stronger robustness against the node capture at a reasonable scalability cost. Their scalability analysis shows that the energy cost remains reasonable and on par with the energy cost of using advanced encryption standard for a WSN consisting of 264 nodes is 48 times higher than the maximum number of nodes defined in IEEE 802.15.4. The only problem with their scheme is the complexity which makes it hard to implement and increases the overhead cost.

Another scheme for the key distribution was proposed by **Liu and Ning**[20] which combines the polynomial based key pre distribution with the key pool idea of the RKP[1] and q-composite RKP[2] to improve the resilience and scalability of the sensor network. In their scheme they generated a set *F* of bivarte polynomial of degree *t.* In the key setup phase each node in the network receives the subset of *F* There are several ways to store the subset of the set in the sensor node the Liu's scheme gave two such ways .One approach is store the polynomial subset along with the list of sensor ID's with which it shares the polynomial. In another approach, a grid-based key pre-distribution scheme is employed. They also propose the location based pair-wise key distribution scheme [21] which was the alternative to the Chan's scheme [2]. The scheme takes the advantage of the location information to improve the key connectivity. The sensor nodes are deployed in two dimensional area where each node has an expected location that can be predicted. The motive behind was to have shared a pair wise keys among the c closet neighbors of the every node in the network. In the key set up phase for the each sensor node Sa a unique key Ka and c closet neighbors for the nodes are selected and for each pair a pair-wise key is established by using a PRF(Kb|Ida). Node a stores all the pair wise keys while the neighbor node stores only the key Kb and PRF. Thus each sensor node in the network uses 2c+1 memory units to store its key chain. This makes deployment of the new nodes in the network easy as compared to previous work. This scheme provides the better resilience than the Chan's q composite RKP and it is also scalable.

SPINS[16] proposed a scheme for the security for the wireless sensor network. In their scheme each shares a secret key with the base station. To establish a new key two communicating nodes uses the base station as the trusted third party to set up the new key.

In their scheme they proposed SNEP which provides Confidentiality, authentication, integrity and freshness to the data in the network. In SNEP the two communicating nodes share a master key. The two communicating parties S_A and S_B to share a master secret key $X_{A,B}$ and a PRF generate encryption keys $K_{A,B}$ = PRF($X_{A,B}$,1) and $K_{B,A}$= PRF($X_{A,B}$, 3), and MAC keys $K'_{A,B}$ = PRF($X_{A,B}$, 2) and $K'_{B,A}$= PRF($X_{A,B}$, 4).

They also proposed a µ-TESLA[29] which is a modified version of the TESLA and is used to provide the authenticated broadcast in the wireless sensor network. It employs the base station as the key distribution centre. The base station randomly selects the last key K_n of chain and applies the one-way public function H to generate the rest of the chain $K_0, K_1, K_2 \ldots\ldots K_{n-1}$ as K_i = H(K_{i+1}).Given K_i,every sensor node can generate the sequence of $K_0, K_1, K_2 \ldots\ldots K_{i-1}$. However given K_i no one can generate K_{i+1}. At ith time slot the base station sends the authentication message MAC K_i(Message).Sensor nodes stores the message in their buffer until the base station sends them a verification key in (i+i)th time slot. The sensor nodes can verify the disclosed verification key K_{i+1} by using the previous key K_i as K_i = H(K_{i+1}). In µ-TESLA the nodes are required to store the message until the authentication key is not disclosed by the base station this creates the problem for storage in the sensor nodes which exposes them to attacks like DoS. An adversary can jam the key disclosure message to saturate storages of sensor nodes. The µ-TESLA requires to bootstrap the sensor nodes from the base station i.e. they receive the first key of the chain which is called the key chain commitment [29]. Bootstrapping procedure requires unicast communication and can be secured with pair-wise keys

Basagani[23] proposed a scheme for securing the wireless sensor network by updating the keys periodically. In his approach he used a traffic encryption key(TEK) which is periodically changed depending upon the level of security needed to secure the network from the

cryptanalysis attack. The scheme divides the network into cluster and the cluster head is elected in every cluster. The network of cluster head's is created which is termed as the backbone network which remains active even in the case of node failures and high node mobility. The cluster head from the backbone is selected to generate the keys for the entire network. The proposed scheme has several disadvantages like the scheme assumes the tamper resistant sensor nodes which are not the usual case and also the key generation scheme have collision problems.

Zhu et al[19] proposed a key management protocol for the sensor network which uses the hybrid approach. It supports the in-network processing and at the same time restricting the security impact of a node compromise to the immediate neighborhood of the compromised node. It offers network-wide, cluster/group and pair wise keying capabilities. It uses the four types of keys: individual, group, cluster and pair wise shared keys. The individual keys are unique for each senor node in the network and are used to communicate with sink. The group key is network-wide key which is used for communicating between the sink to all the sensor nodes in network. It uses µ-TESLA [29] for broadcast authentication of the sink nodes which ensure that packets are sent with the group keys are from the sink node only. The cluster key is used within cluster for cooperation. One-way hash key chain is used as authentication mechanism for authenticating the source packet. The pair-wise shared key is used for secure communication between the neighbor sensor nodes. All the keys established in the network by using the key pre-distribution mechanism. The individual keys in the LEAP [19] are established using the seed function and the ID's of the node in the network. In the pair wise shared key phase the neighbor discovery takes place and all the nodes broadcasts their ID's. In order to calculate the shared key between the node and its neighbor it uses a function which is seeded with the initial key. After this the initial and the intermediate keys are deleted. The cluster key is distributed to the entire sensor network

by cluster head using the pair wise communication with each node. The group key is distributed in the whole network by using the multi hop broadcast using cluster-by-cluster manner starting with closet cluster.

SHELL [18] The scalable, hierarchical ,efficient, location aware and light weighted(SHELL) protocol is a complicated key management scheme proposed by Younis et al in 2006. It's a key management scheme for large scale clustered sensor network. It is influenced by LEAP [19] and uses multiple keys and a proposes a new key distributed key management entity in the network. The distributed key management is handled by the non cluster node which separates the operational responsibility from the key management responsibility. This increases the resilience in the network. The scheme uses multiple entities and over seven keys for communication in the sensor network due to his reason it requires multiple cluster heads from the vicinity. It uses the EBS matrix for maintain the global information about the keys stored in every sensor node. Every node in the network is made aware of only k keys out of the total k+m keys. In case of the node compromise only k keys are reveled and m keys which the node doesn't know are used to refresh its compromised keys to evict the node from the adversary.

In SHELL, cluster head node of a cluster generates the EBS matrix, breaks it up into different parts and sends those parts to its neighboring cluster head nodes. Neighboring cluster head nodes manage keys for the cluster. The EBS matrix is divided in such a way that the compromise of a neighboring cluster head node does not compromise too many keys. On a cluster head's request, neighboring cluster heads generate keys and refresh them. However, the cluster head node does not get to know the actual key values.

Panja *et al.* [17] introduced a hierarchical group keying scheme which uses the Tree- based Group Diffie-Hellman(TGDH) protocol. The main feature of this scheme is that each key in the network is made from the many partial keys. The scheme makes the rekeying process simple by breaking the keys into smaller components. The sensor nodes in a group don't use pre-deployed keys but dynamically generate partial keys using a function that takes partial keys of its children as input. The partial keys in a group are used for computing the group key in a bottom up fashion. Groups of sensors at different levels are secured by using multiple level securities. The group key management protocol supports the establishment of two types of group keys: intra-cluster and inter-cluster. Intra-cluster group keys are used for encryption/decryption of messages for the sensor nodes within a group while Inter-cluster group keys are used within groups of cluster heads. The protocol handles freshness of the group key dynamically, and eliminates the involvement of a trusted third party (TTP). The hierarchical sensor node architecture consists of multiple levels consisting of sensor nodes, cluster heads and relay nodes. The data collection starts from a sensor node within a particular geographical area which then sends it to the nearest sensor node. If the receiving nodes are relay nodes, they further forward the data using appropriate routing path. The cluster head aggregates the data coming from different sensor nodes within its group and forwards it to the next higher level of cluster heads. This process is repeated until the data reaches the sink node.

Various scheme for secure key management has been proposed which uses high energy sensors along with the ordinary sensors for key management. These schemes uses the high end sensors for storing the large pre distributed keys and thus uses their high storage and energy capacity to reduce the storage overheads in sensor network. Some of the scheme proposed

RKPH [4] It is based on the random key distribution for the heterogeneous sensor network. It uses the separate keys for different clusters while taking the distance of the sensor node from its respective cluster head into consideration. It uses the cluster information and the distance of the node from the cluster head for key management. The cluster is divided into several levels and different seeds are used for generating keys in each of the level. The base keys are preloaded in the sensor node and after deployment the new keys are generated. The scheme takes into consideration the high storage and computing capacity of the high end sensor(H-sensors) which are limited in number and are made the cluster head for reducing the storage and computation over head of the network. The low end sensor nodes or the normal sensor nodes (L sensors) have the usual storage and computation power. The H sensors are assumed to be of tamper resistant material which provides security against the node capture attacks. After each H-sensor becoming the cluster head they obtain their location by the GPS and sends it to the base station .Base station after receiving this location information for each cluster head estimates the maximum distance of a point in cluster can have by making the grid of the deployment area.

The cluster head initially sends the seeds to the sensor node in the cluster depending on the distance from the cluster head. Different seeds are sent for the different distance. After each cluster head takes the seed for the base station considering the maximum distance of a point to that cluster head and the distance of seed utilization. After the deployment cluster heads sends the HELLO message to the all the nodes in the cluster and computes there distance from the cluster head the node can join the cluster which has the minimum distance to its cluster head. After this the nodes can discover the common keys between them.

The scheme uses the H-sensors with tamper resistant material equipped with the GPS increases the cost of the network and also drives the scheme away from the practicality.

Bafagi et al proposed [24] a scheme which was similar to the RKPH[4] but with some improvements. Its design was based on the random key distribution in the heterogeneous sensor network and uses the separate keys in different clusters while taking the distance of the sensor nodes from their cluster head into consideration. As compared to RKPH, the ARKPH considers the multiple shared keys between pair wise nodes. When a key used for establishing the secure path between the two nodes gets revealed and the link gets expired and the communication of the network is hindered, it changes the alternative shared key to replace the revealed key and establish a new secure path between the nodes in the network.

The ARPKH[24] is an improved version of RPKH[4] which considers multiple shared keys between pair-wise nodes on connectivity. However, ARPKH needs alternative shared key replacement, which makes sensors pre-distribute more keys and occupy larger storage. Moreover, ARPKH also needs the anchor nodes as the cluster head, which makes it impracticable.

Zang et al proposed [5] Distance based key management for hierarchical wireless sensor network in which they considered the normal nodes for the network as compared to the previous schemes. In their scheme they distribute the keys on the basis of the distance of the node from their respective cluster head. The cluster is further divided into belts based on the TTL .they uses the random numbers for generating the path keys between the nodes in the cluster. The problem with this scheme is that it fails to provide the inter cluster security and exposes the cluster head to many security threats.

Chapter 5: Proposed Work

Key management is one of the most important and basic aspect of WSN which provides the base for the various other secure mechanism like secure routing , secure localization etc

Security in WSN has six challenges

1. Wireless nature of communication,
2. Resource limitation on sensor nodes,
3. Very large and dense WSN,
4. Lack of fixed infrastructure,
5. Unknown network topology prior to deployment,
6. High risk of physical attacks to unattended sensors.

Moreover, in some deployment scenarios sensor nodes need to operate under adversarial condition. Security solutions for such applications depend on existence of strong and efficient key distribution mechanisms. It is infeasible, or even impossible in uncontrolled environments, to visit large number of sensor nodes, and change their configuration. Moreover, use of a single shared key in whole WSN is not a good idea because an adversary can easily obtain the key. Thus, sensor nodes have to adapt their environments, and establish a secure network by:

1. Using pre-distributed keys or keying materials.
2. Exchanging information with their immediate neighbors.
3. Exchanging in-formation with computationally robust nodes.

Key distribution and management problem in WSN is difficult one, and requires new approaches.

5.1 Framework for enhanced secure key management for hierarchical wireless sensor network

The proposed framework makes use of keys generated by the base station and the sensor nodes in the cluster to provide the security for both inter and intra cluster communication. This section describes the work in detail.

The framework uses three keys for providing the security in the network. Out of these three keys two are computed by the base station and the third key is computed by the senor nodes in the network by using the localized key material provided to them by the their respective cluster head. One key is used to provide the security between the cluster head and the base station another key is used to provide the inter cluster security i.e. between cluster head and sensor nodes and the third key is used for the securing path between the two neighbor nodes. The path key is generated by the nodes using the nounces supplied to them by the cluster head. Nounce are send on the basis of the distance of the sensor node from their respective cluster head which keeps on changing with the increase in the distance. So the nodes at the same distance will have same nounces and the one which is far off will have lesser nounces.

Our framework consists of four phases which are

1. Pre Key distribution
2. Pair wise key establishment
3. Computing Path key
4. Re- keying.

All these phases are described in details. The assumptions which made for the network are as following.

- The BS is a control center and connects the WSN with external network for processing of the sensed data. Further, it is assumed that the base station has unlimited computational, communication, and memory resources and it is considered trustworthy and it can also transmit directly to every sensor node.
- Sensors nodes collect information of surrounding environment and transmit them to their respective cluster head.
- Cluster heads are responsible for the coordination, the data retransfer and the management of all the nodes in the cluster.
- We assume that WSNs are homogeneous and symmetric. Nodes are deployed randomly in the network. Sensor nodes keep stationary after deployment during the network operation.
- All the sensor nodes have a unique ID.
- Use of non-tamper resistant hardware in sensor node.
- If a node is compromised all the key material can be taken out by the adversary.
- The sensor node should be a part of at least one cluster.

5.1.1 Pre key distribution

The WSN is a resource constrained network so to provide a efficient key management the keys should be pre-loaded in the nodes before they are being deployed[11].In our proposed work to authenticate each sensor node the BS computes a unique key K_{net} and pre loads this key into every sensor node before deployment. This key is deleted after the first round and is used in the cluster formation phase.

5.1.2 Pair wise key establishment

5.1.2.1 Pair wise key establishment between the cluster head and base Station

After the nodes are being deployed and pre distributed with K_{net} the BS needs to establish pair wise keys with every cluster head to secure the communication between them so to achieve this BS makes an array V consisting of the id's of all the sensor nodes. After this a cluster head is elected. A node can volunteer himself for being the cluster head else randomly any node is selected as cluster head. After the first round the cluster head can be elected using various scheme like[9,10]After becoming the cluster head for the first time the node sends the authentication message encrypted with K_{net} to base station which includes its id.The contents of message includes the following

idCH,idBS	E K_{net} (M|N)	mac K_{net} (M|N)

Figure 8: Message from CH to BS

M= |idCH,idBS| K_{net} |

Where M is the message from the CH and N is the timestamp and Mac is generated using the key K_{net} . After obtaining the message from the CH, BS computes the new key which would be used to for communication between the CH and BS.The new key is generated by applying the one way hash function on the id's of BS and CH ie

K_{BS-CH} = H K_{net} (V[idCH]+V[idBS])

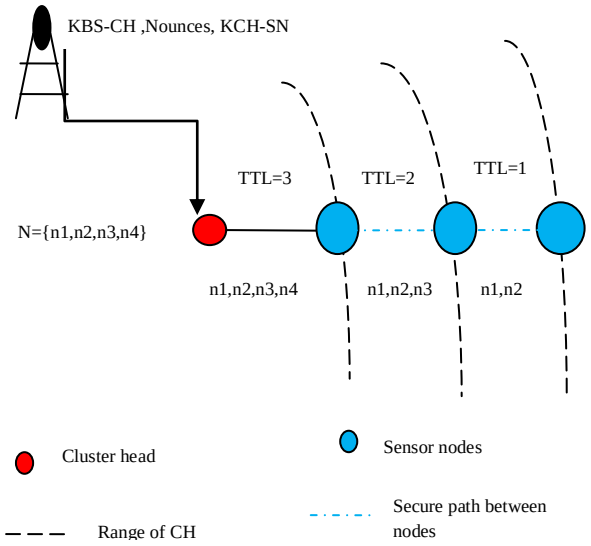

Figure 9: The network setup for TTL=3

This new key is sent to CH by encrypting the message with the K_{net}.The message from the BS to CH will have

idCH	idBS	E K_{net} (M|N| K_{BS-CH})	Mac K_{BS-CH} (M|N)

Figure 10: Message from BS to CH

After receiving the message the cluster head can decrypt he message and obtains the key K_{BS-CH}

5.1.2.2 Pair wise key establishment between cluster head and sensor nodes

After becoming the cluster head for the first time the node broadcasts a beacon message to all the nodes in the cluster. The range of beacon message is restricted by using a variable TTL (Time to live) which is initially set to a predefined value and then it gradually decreased with the every forwarding and stops when it becomes zero. The TTL helps in segregating the cluster into different belts based on the distance of sensor nodes from their respective cluster head. Initially the TTL is set to small number like say TTL=3.The beacon message consists of CH id, TTL and Time stamp (to avoid replay attack) all encrypted by the key K_{net} .

idCH	TTL	Timestamp

Figure 11: Beacon Message from CH to SN

The nodes in the network can receive several beacon messages from the different cluster heads but can join only one cluster. After receiving the beacon message the node sends a ACK message to cluster head which includes its ID, CH ID, value of TTL when it is received by the sensor node and MAC of the message from the cluster head is computed using the key K_{net} .

idCH,idSN	TTL(Value of TTL When SN received it)	MAC of message from CH

Figure 12: ACK Message from SN to CH

After receiving the ACK message from all the member nodes in the cluster, the cluster head calculates the hop count

Hop count = TTL_{CH} - TTL_{SN}

to every member node in the cluster. At this point the cluster head has the id's of the every senor node in the cluster. The cluster head sends these id's to BS encrypted by the encrypting it with the key K_{BS-CH} to obtain the cluster key. After receiving the id's of SN's the BS computes the cluster key K_{CH-SN} by using the one way hash function and the pair wise key K_{BS-CH} and then sends the key along with the nounces to the cluster head by encrypting it with the key K_{BS-CH}

idCH,idBS	E K_{BS-CH} (M|N $\|K_1(CH-SN)$ |)	mac K_{BS-CH} (M|N)	Nounces

Figure 13: Message from BS to CH for group key

After receiving the message from the BS, CH forwards the cluster key K_{CH-SN} and nounces as per the hop count to every sensor nodes in the cluster by encrypting it with the K_{net}. All the nodes decrypts the message to obtain the cluster key and the deletes the key K_{net}.At this point all the nodes in the cluster have the cluster key and the nounces according to their distance from the cluster head i.e. node have TTL as 3 will have 3 nounces(n1,n2,n3) and node having TTL as 2 will have 2 nounces(n1,n2).The algorithm for generating the nounces is given below.

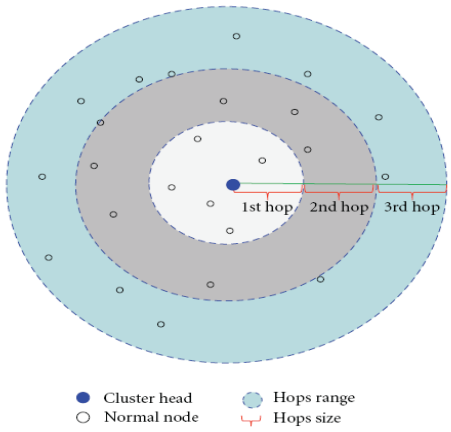

Figure 14: The different hope ranges of nodes from the CH

5.1.3 Computing the path key

The sensor nodes that are in the same distance range from cluster head receive the same beacon message, so there are matching keys with the nodes. The structure can be viewed from the figure3.These keys can be used to generate the path keys among the sensor nodes in cluster to provide a secure communication between the nodes within the cluster. As per the principal of key generation the two nodes MNi and MNj the set of common keys can be obtained as

S= LMNj $\cap$ LMNi= LMNj as (i<j)

Each MN generates a list which stores the keys as follows

LMNj= { $K_{MNj}^{i} | K_{MNj}^{hops}$ ……., $K_{MNj}^{TTL\ mnj}$ }

Since all the messages are sent to cluster head and thereafter to the base station so both the cluster head and base station should be able to decrypt the message and get assured for the message authenticity so CH knows the function to calculate the path key which can be shown by the following

KMNi j = $f^{abs(i-j)}(K_{CH-SN}, Ni)$

Algorithm for key generation among the sensor nodes to establish a path key

- CH broadcasts the beacon message with different nounces received from the BS CH->{idCH,N,TTL, K_{CH-SN} } K_{net}
- SN decrypts the {idCH,N,TTL, K_{CH-SN} } K_{net}
- SN deletes the K_{net}
- Key pool for the SN=Null
- K_length= TTL SN
- For i=1 to TTL SN

  {

  $k_j^i = f(K_{CH-SN}, ni)$ //generation of key pool using one way hash function on the nounce with the key K_{CH-SN}

$K^i = K^i \cup \{k_j^i\}$

}

- End

Sensor nodes use the key K_{CH-SN} to communicate with the CH thus provide enhanced security as communication between the sensor nodes and communication between the CH and SN are both secure. The path key provides the security for communication between the two nodes in the network and the cluster key provide the security for the entire cluster.

Since to provide the authenticity to the nodes in communication CH can verify the keys of the sensor nodes by applying the one way function and get the id's of the nodes. The CH can decrypt the message received from the sensor node by using the following procedure

- Sensor node sends the information to CH encrypted by the key K_{CH-SN}

  SN->{idSN,idCH,M} K_{CH-SN}

- CH can obtain the hop count as Nhops=$TTL_{CH} - TTL_{SN}$

- According to the Nhops, nounce and the one way function the CH can get the idSN and key information

- End

5.1.4 Re-key process

As WSN has limited battery life so to enhance the life of the cluster the cluster head must be changed and to maintain the security all the keys must be re-keyed as it is known that after

receiving the certain amount of encrypted messages the use of same key is no longer safe ie after receiving more than $2^{\frac{2k}{3}}$ where k is the length of the key[13].So the re-key process starts from the re election of the cluster head. In this way the process of re key starts from the scratch as the new cluster head will require new keys to communicate with the base station and the sensor nodes and more over the distance of all the nodes from the new cluster head will be different from the previous cluster head. So with the new CH the hop count from the cluster head to sensor nodes, nounces and the key material will change.fig4

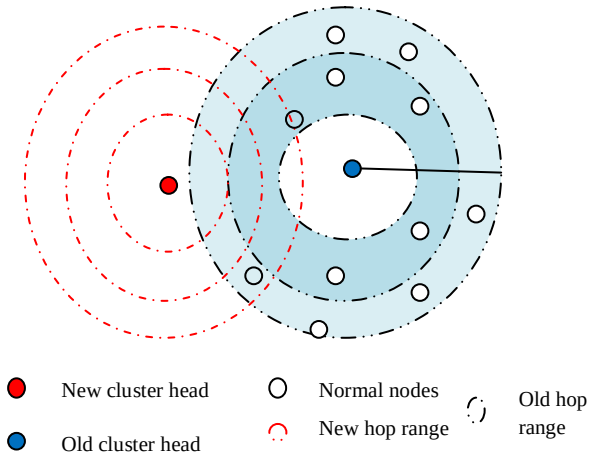

Figure 15:Re-election of the cluster head

Chapter 6: Security Analysis and Experimental Results

In this section we describe the experimental setup and the results in support for the proposed work.

6.1 Environmental setup

We have used the following configuration while finding the experimental results

6.1.1 Hardware Configuration

Processor: Intel Core 2Duo-T5870

Processor Speed: 2.0 Hz

Main Storage: 2GB RAM

Hard Disk Capacity: 360GB

Monitor: Compaq 14”

6.1.2 Software Configuration

Operating System: Windows 7 Enterprise Edition

Programming Language: MATLAB

Software: MATLAB R2010a

6.2 Security Analysis

As compared to the all the previous work [4,5,6,8] done in enhancing the secure key management the advantage of our work is that it provides the enhanced security services for both the inter and intra cluster communication in a hierarchical wireless sensor network and also address the challenging runtime security by using the localization of key material and design of robust key generation mechanism. All the previous work done has failed to address the intra cluster communication at the runtime due to this various loop holes in security can be utilized by the adversary to attack the network. Moreover the proposed work doesn't requires any special kind of nodes like RKPH[4],LDK[14] which makes it easy to deploy in practical scenario and also doesn't requires GPS devices. The use of hierarchical network for localizing the key material to nodes and easy hop count mechanism adds advantage to our scheme as compared from other proposed work.

The inter cluster communication is secured by using the path keys which are generated by the nounces and distance. According to the different distance, cluster is further divided into several small security belts and nodes in different security belts have different set of keys to communicate. Since the keys are computed by the set of nounces supplied by the base station the neighboring nodes have some common keys which make them to communicate with other by discovering common keys. As the keys are generated on the basis of hop count the node near to cluster head will have more keys then the other nodes that are far away from the cluster head which makes the far nodes only to submit the message. Moreover all the communication within the cluster is secured by the cluster key which is computed by the base station which makes only legitimate nodes to be a part of communication. This type of security model for inter cluster

communication prevents various attacks like eavesdrop, selective-forwarding and hello flood attacks.

The communication is secured from the initial phase i.e. before sending any message it is encrypted using the hash function. Our work also provides the freshness to key management by using the timestamp and nounces. It is always a worrisome to get the key material reveled to adversary after the node has been compromised but our scheme provides a dynamic solution to it as we except that attacker will take some fixed amount of time to compromise the node since the keys are re keyed after some time so by the time attacker will compromise the node the keys would have been changed. The intra cluster communication is secured by using the shared key between the cluster head and the base station which makes the base station to verify the authenticity of the cluster head and this shared key is re keyed every time when a new cluster head is elected for the cluster. By using this scheme for the intra cluster communication the attacks like Sybil attack, acknowledgement spoofing can be prevented. It also prevents the black hole attack to a large extent and makes sure that the entire network doesn't falls into the hands of attacker.

6.3 Experimental Setup

We have used the MATLAB R2010a version for evaluating results. MATLAB is very widely used numerical computing environment and the fourth generation programming language. It is developed by MathWork industry. It allows various functions like matrix multiplication, creation of user interface, implementation of algorithms and many more. The additional packages like Simulink adds graphical multi-domain simulation and model based design for dynamic and embedded system.

The network is created consists of 100 nodes which are randomly distributed in the area of 100x100 in simulation environment.

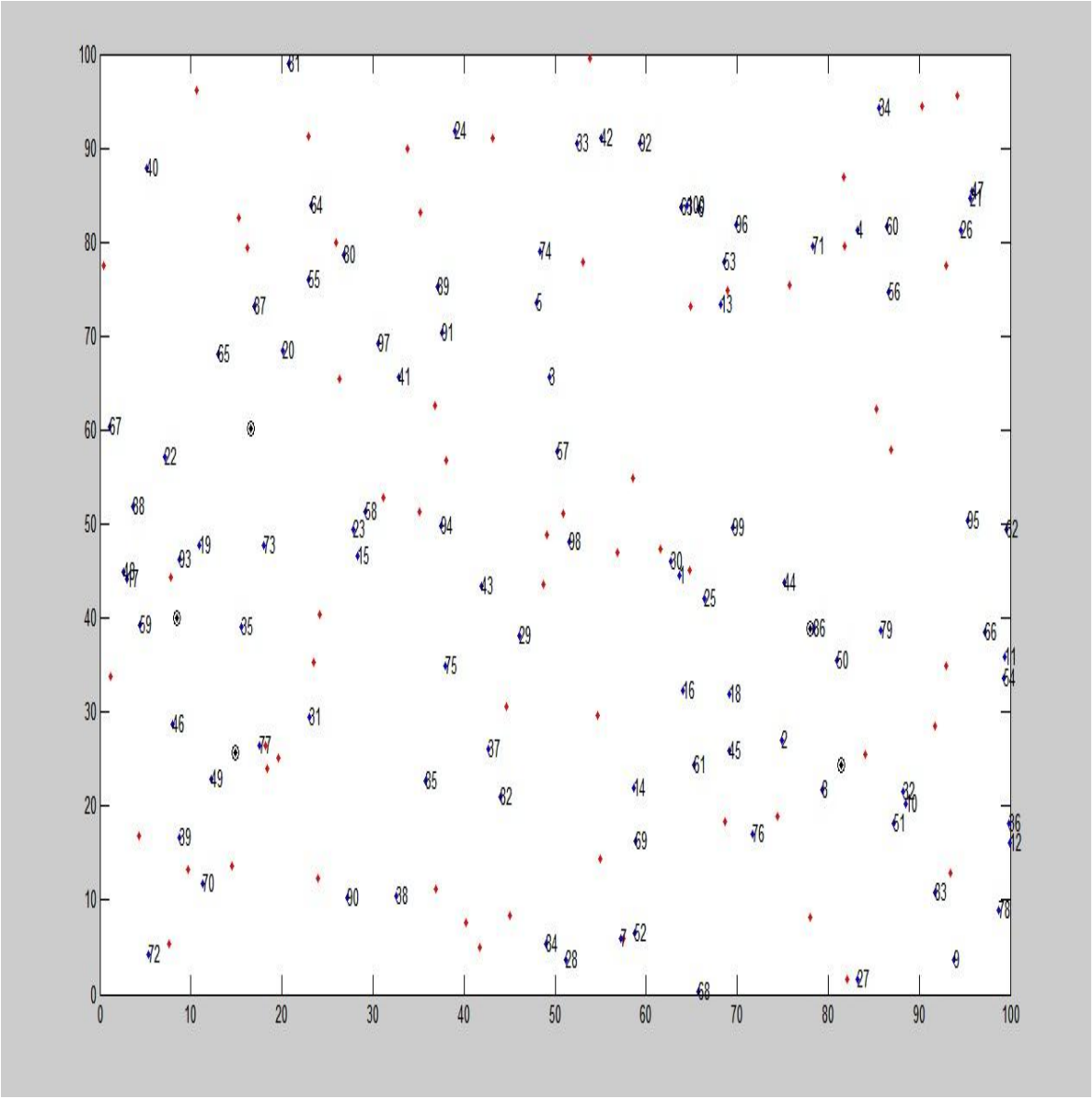

Figure 16: MATLAB Network Creation

The performance of the proposed work is evaluated on the capability of the network to securely send the message in the network during its entire lifetime. The sensor nodes sends their information in packets to sink using a multi-hop transmission, thus for sustaining the life of the

network the distance between the sink and the network should be optimal but this is not the usual case in practical. The sink is the located at the dead end of the simulation area environment. The five malicious nodes are introduced in the network whose main aim is to divert the traffic of the network to false destination. These intrusion nodes have the high energy capacity then the usual nodes in the network due to this they succeed in diverting the traffic to false destinations.

The cluster head in the network is elected using the Low-Energy Adaptive Clustering Hierarchy protocol (LEACH).The LEACH takes into account the energy of the nodes in the network for choosing the cluster head for a cluster. The various parameters used for the simulation can be viewed from the following figure.

Name	Value	Min	Max
Distance	<100x100 double>	0.6861	Inf
EDA	5.0000e-09	5.0000e-09	5.0000e-09
ERX	5.0000e-08	5.0000e-08	5.0000e-08
ETX	5.0000e-08	5.0000e-08	5.0000e-08
Efs	1.0000e-11	1.0000e-11	1.0000e-11
Emp	1.3000e-15	1.3000e-15	1.3000e-15
Eo	0.1000	0.1000	0.1000
LowRange	25	25	25
NodePara	<1x101 struct>		
PacketACT	<1x309 double>	100	17688
PacketSEC	<1x501 double>	95	16566
PacketW_SEC	<1x499 double>	95	17341
change	40	40	40
ctrPacketLength	100	100	100
do	87.7058	87.7058	87.7058
inf	1.0000e+15	1.0000e+15	1.0000e+15
n	100	100	100
p	0.0500	0.0500	0.0500
packetLength	4000	4000	4000
packetLimit	10	10	10
rmax	500	500	500
sink	<1x1 struct>		
switchC	3	3	3

Figure 17: MATLAB Parameters for Network

The GUI of the setup can be viewed from the following figure

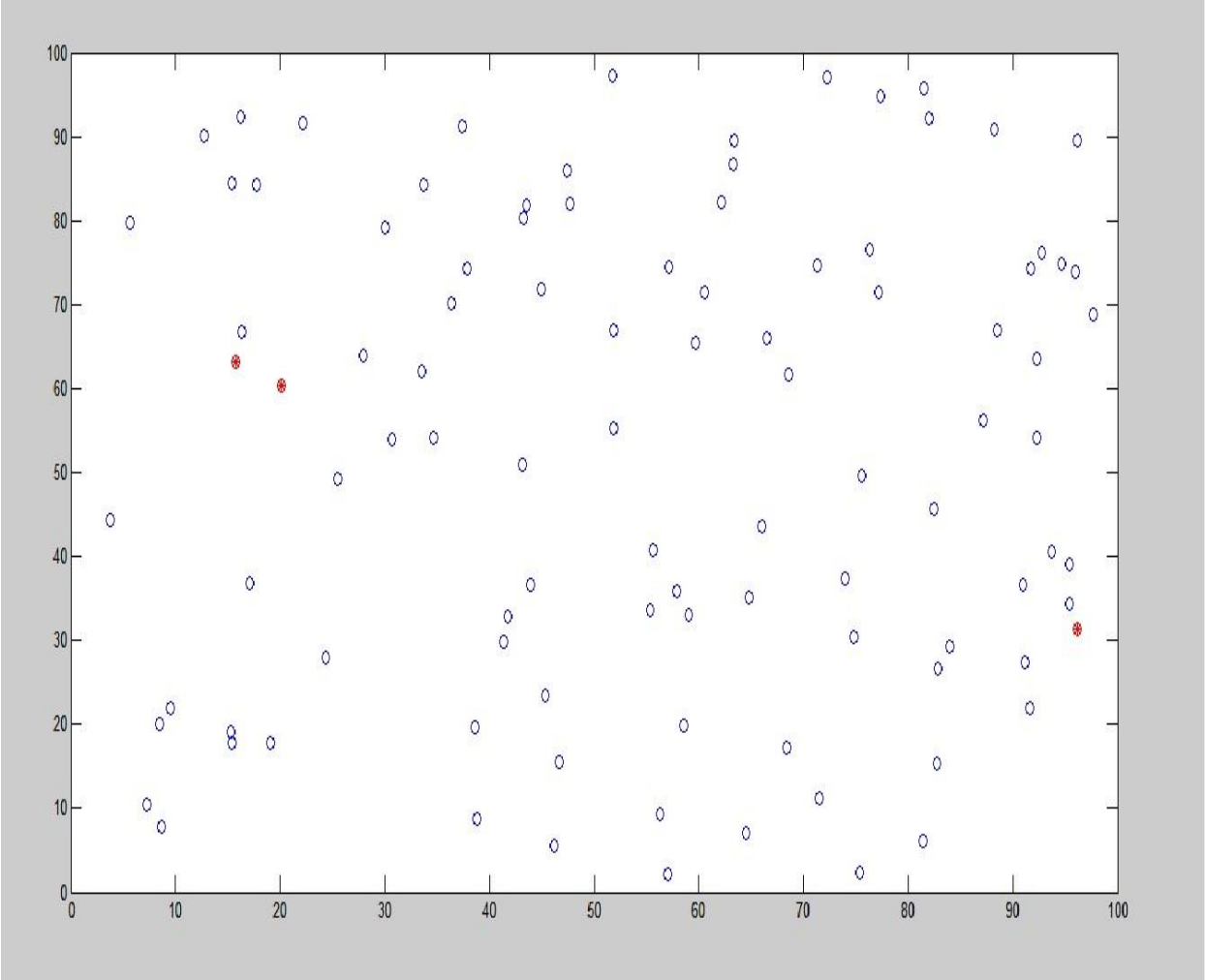

Figure 18: MATLAB Network GUI

The red dots in the figure shows the cluster heads in the network and blue dots shows the normal sensor nodes. The network also has the intrusion nodes which are shown by the black dot inside a circle

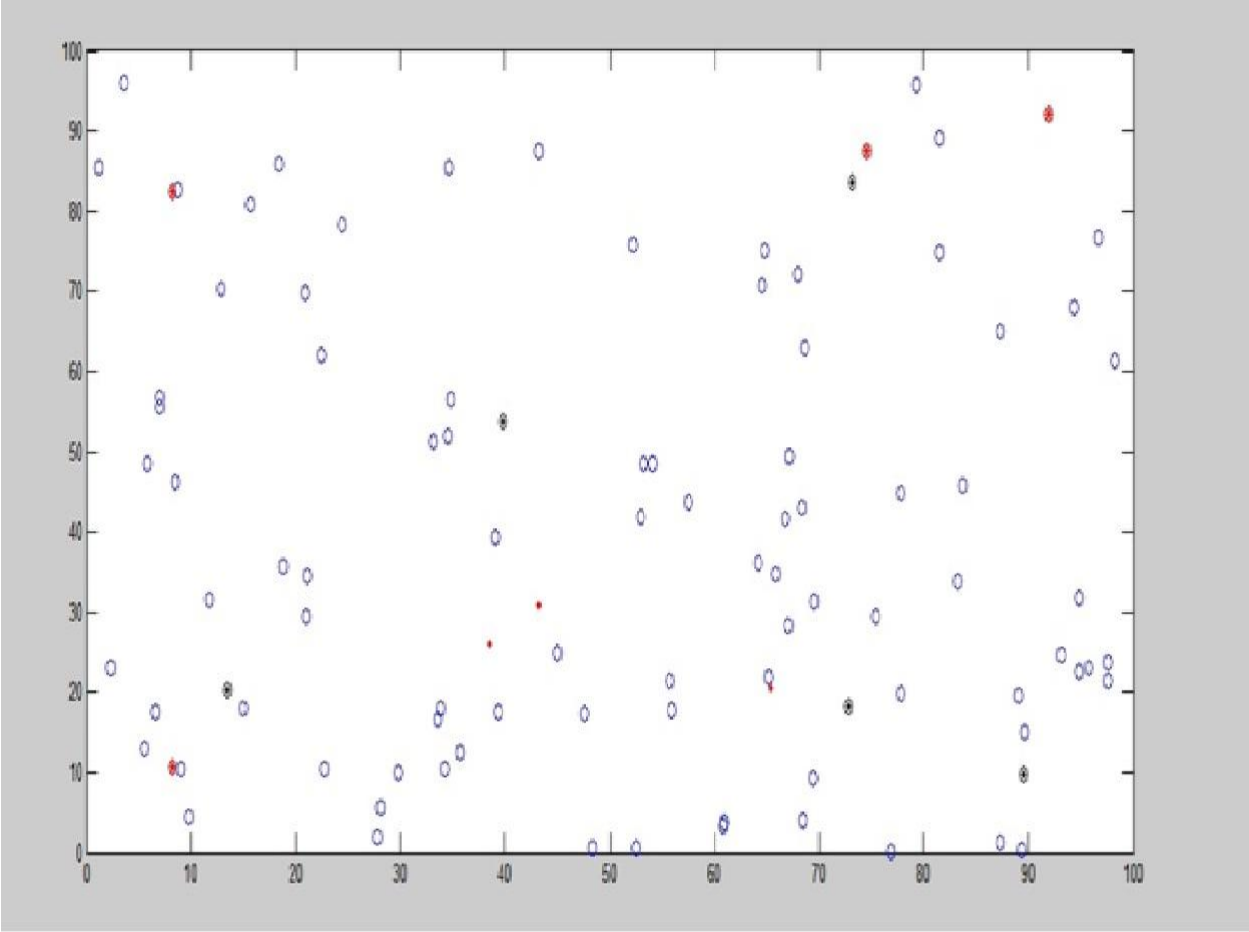

Figure 19: MATLAB Sensor Network with Intrusion Nodes

The simulation runs for the 10 minutes depending upon the parameters chosen. The simulation is triggered by using the main script which calls all the functions like GloVare, CreateNetwork etc and sequentially performs all the functionality. The final result after running the simulation can be viewed from the figure 20.

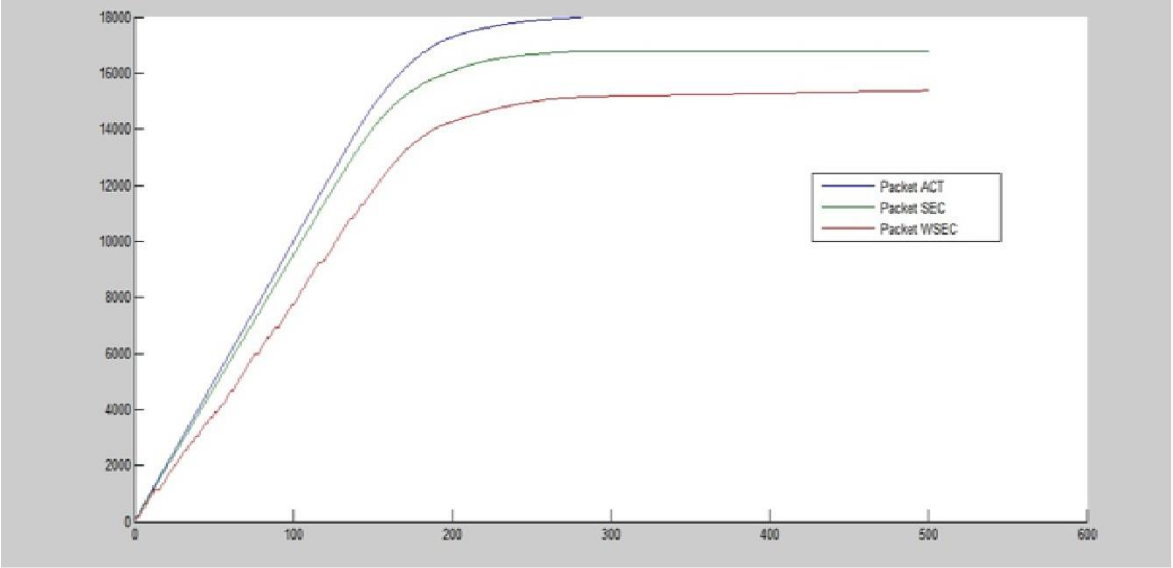

Figure 20: MATLAB Graph for Experiment

The graph shows the number of messages sent to the sink in the three different environment .One in blue line shows the actual messages that can be sent to sink. It represents the ideal situation where in we don't consider any intrusion or anomalies in the network operations. The red line shows the messages sent to the sink in the actual scenario i.e. when there are intrusion nodes in the network which try to divert the traffic to different destination. The green line shows the messages sent to the sink using the security options in the proposed work. The graph shows that the number of the packets successfully received by the sink increases.

Chapter 7: Conclusion and Future Scope

7.1 Conclusion

In this thesis we have discussed about the security problem in the wireless sensor network and have tried to find a solution by using the key management and distribution in chapter 5.We have proposed a framework for the enhance secure key management for a hierarchical wireless sensor network which computes the key based on the distance of the sensor nodes from their respective cluster heads. We used the symmetric keys for providing the security to the information that is to be sent to the sink by the sensor nodes in the network. We have got better resilience of the network by our experiment. In our work we tried to address the challenging run time security issues and a robust mechanism for continuous authentication of the sensor nodes in the network. Our work divides the cluster into smaller isolated geographical belts on the basis of the distance from the cluster head which provides the tight security inside the cluster. The uses of separate unique keys for both inter and intra cluster communication makes our work more resilient against the various attacks that can be carried out on the network. All the keys used for communication are rekeyed after the fixed time which makes it difficult for the adversary to know the keys from the network.

7.2 Future Scope

In our work we have considered less number of nodes in the network thus future research can be done on mobile and scalable wireless sensor network. The future research can be done on making the security mechanism in the wireless sensor network energy efficient.

References

[1] L. Eschenauer, V_D. Gligor, A key management scheme for distributed sensor networks. In: Proceedings of the Ninth ACM conference on Computer and communication security(CCS '02.), pp.41-47,2002.

[2] H. Chan, A. Perrig, and D. Song, "Random key predistribution schemes for sensor networks," in Proceedings of the IEEE Symposium on Security And Privacy, pp. 197–213, 2003

[3] S Zhu, S Setiaand ,S Jajodia "LEAP: efficient security mechanisms for large-scale distributed sensor networks" in the proceeding of 10th ACM conference on Computer and communication security(CCS'03),pp.62-72,2003.

[4] S. Banihashemian and A. G. Bafghi, "A new key management scheme in heterogeneous wireless sensor networks," in Proceedings of the 12th International Conference on Advanced Communication Technology (ICACT '10), pp. 141–146, February 2010.

[5] Y.Y. Zhang, X.Z. Li, J.P. Cao, L.K. Zeng, Y. Zhen and D.Q Gao "Distance-Based Key Management in Hierarchical Wireless Sensor Network" in the proceeding of Automatic Control and Artificial Intelligence (ACAI 2012), International Conference on Digital Object Identifier, pp. 915 – 918,2012

[6] Yiying Zhang,,Xiangzhen Li, Jianming Liu, Jucheng Yang, and Baojiang Cui "A Secure Hierarchical Key Management Scheme in Wireless Sensor Network" International Journal of Distributed Sensor Networks, Volume 2012 ,pp 1-8, 2012.

[7] X. Du, Y. Xiao, M. Guizani, and H. H. Chen, "An effective key management scheme for heterogeneous sensor networks," Ad Hoc Networks, vol. 5, no. 1, pp. 24–34, 2007.

[8] AbdoulayeDiop, Yue Qi, Qin Wang, and ShariqHussain "An Efficient and Secure Key Management Scheme for Hierarchical Wireless Sensor Networks" International Journal of Computer and Communication Engineering, Vol. 1, No. 4, November 2012,pp 365-370,2012.

[9] Heinzelman, WendiRabiner, Anantha Chandrakasan and Hari Balakrishnan" Energy-efficient communication protocol for wireless sensor networks" In System Sciences,2000.Proceedings of the 33rd Annual Hawaii International Conference on IEEE, pp 10-pp,2000.

[10] Manjeshwar and D. P. Agrawal, "TEEN: a protocol for enhanced efficiency in wireless sensor networks," in Proceedings of the 15th international workshop on parallel and distributed computing issues in wireless networks and mobile computing, pp. 2009–2015, 2001.

[11] S. Zhu, S. Setia, and S. Jajodia, "LEAP+: efficient security mechanisms for large-scale distributed sensor networks," ACM Transactions on Sensor Networks, vol. 2, no. 4, pp. 500–528, 2006.

[12] Martins,David and HerveGuyennet "Wireless sensor network attacks and security mechanism:a short survey". In the Proceedings of 13th International Conference on Network –Based Information System(NBiS),pp.313-320.IEEEE,2010 .

[13] H. N. Seyed, H. J. Amir, and D. Vanesa, "A distributed group rekeying scheme for wireless sensor networks," in Proceedings of The 6th International Conference on Systems and Networks Communications (ICSNC '11), pp. 127–135, 2011.

[14] Anjum and Farooq" Location dependent key management in sensor network without using deployment knowledge" in Wireless Networks 16,no.6(2010),pp.1587-1600,2010.

[15] W. Du, J.Deng, YS Han, S. Chen "A Pairwise Key Predistribution Scheme for Wireless Sensor Network," In the Proceeding of the 10th ACM Conference on Computer Communication Security, 2003, pp. 42–51.

[16] A. Perrig *et al.*, "SPINS: Security Protocols for Sensor Networks," *Wireless Network*, vol. 8, 2002, pp. 521–34.

[17] B. Panja, S. K. Madria, and B. Bhargava, "Energy and Communication Efficient Group Key Management Protocol for Hierarchical Sensor Networks," SUTC'06:In Proceeding of International Conference on Sensor Networks ,Ubiquitous, and Trustworthy Comp., 2006, pp. 384–93

[18] M. F. Younis, K. Ghumman, and M. Eltoweissy, "Location Aware Combinatorial Key Management Scheme for Clustered Sensor Networks," IEEE Transactions on. Parallel and Distributed System, vol. 17, 2006, pp. 865–82.

[19] Zhu, Sencun, Sanjeev Setia, and Sushil Jajodia. "LEAP: efficient security mechanisms for large-scale distributed sensor networks-10th ACM Conference on Computer and Communications Security (CCS'03)." Washington DC, October (2003).

[20] Liu, D. and Ning, P. 2003b "Establishing pairwise keys in distributed sensor networks". In 10th ACM conference on Computer and communications security CCS'03.

[21] Liu, D. and Ning, P. 2003c. "Location-based pairwise key establishment for static sensor networks". In 1st ACM Workshop on Security of Ad Hoc and Sensor Networks.

[22] Camtepe, Seyit A., and Bülent Yener. "Key distribution mechanisms for wireless sensor networks: a survey." Rensselaer Polytechnic Institute, Troy, New York, Technical Report (2005): 05-07.

[23] Basagni, Stefano, et al. "Secure pebblenets." In Proceedings of the 2nd ACM international symposium on Mobile ad hoc networking & computing. ACM, 2001

[24] S. Banihashemian and A. G. Bafghi, "Alternative shared key replacement in heterogeneous wireless sensor networks," in Proceedings of the 8th Annual Conference on Communication Networks and Services Research (CNSR '10), pp. 174–178, May2010.

[25] Martins,David and HerveGuyennet "Wireless sensor network attacks and security mechanism: a short survey". In the Proceedings of 13th International Conference on Network –Based Information System(NBiS),pp.313-320.IEEEE,2010

[26] Lee and C.Johnson C "Key management issues in wireless sensor networks: current proposals and future developments." Wireless Communications, IEEE 14.5 (2007): 76-84.

[27] You, Xuemei, and Fanchang Hao. "A key management method of wireless sensor network." In Computer Engineering and Technology (ICCET), 2010 2nd International Conference on. Vol. 2. IEEE, 2010.

[28] Chapter 19 and Chapter 20 from "Guide to Wireless Sensor Networks" by Sudip Misra,Isaac Woungang and Subhas Chandra Misra, Springer Publication 2009 edition.

[29] Section IV from "Wireless Sensor Networks and Applications" by Yingshu Li,My T Thai and Weili Wu, Springer Publication 2008 edition.

Appendix Notations

Table.1

Notations	Description
idSN	Identification number of sensor node in network.
idCH	Identification number of cluster head.
idBS	Identification number of base station.
K_{net}	Network key stored in each sensor node before deployment.
K_{BS-CH}	Pair-wise key shared between the base station and cluster heads.
N_{hops}	Number of hops
EK(M)	Encryption of the message M with key K.
V	Array of the node id's.
Mac M(K)	Message authentication code for message M using key K.
MNi	Member node i in the cluster.
Ni	ith nounce in the set of nounce N.
$f()$	One –way function
k_j^i	ith key for the member node MNj.
K_{CH-SN}	Pair-wise key shared between the cluster heads and the sensor nodes