Secure Clustering and Data Sharing in WSN Network

Sonal Jain, Pawan Malviya, Santosh Kushwaha

***Abstract*— The Wireless Sensor Network is a type of ad-hoc network consisting of hundred or thousand of resource constant tiny sensor nodes. These low cost sensors are deployed either randomly or manually to sense and gather data from the application environment. WSN is now a center of attraction for research due to smaller node size and the limited resources like fixed power battery, computation power and memory associated with it. It is used to monitor environment, gather information from disastrous places. Most of the WSN applications require high-level security. Because of the wireless mode of communication in WSN, an adversary node within the radio range of a particular node can easily intercept the message sent by that node. WSN is usually deployed in unattended sensitive area and therefore the sensor nodes can easily be captured physically be the attackers. Attackers can also change the network topology. Providing security solutions to WSN with limited resources associated with the nodes is a challenging issue for WSN designers. The objective of this dissertation is to identify and isolate adversary to join and alter the fabricated data in WSN. We have proposed a symmetric key based security technique using Hash function for cluster based low energy multi hop data gathering in WSN. In this technique every node uses three types of keys: Individual key (preloaded), Pair wise key (computed at cluster set up phase), Hash function (preloaded) and Group key (preloaded). The technique used for establishing and updating these keys is communication. It is also an efficient technique for inter-node traffic authentication based on the use of hash function. The performance and the security of the proposed work are analyzed and the technique is proven to be very efficient in defending against many attacks. In this work we have tried to minimize and localize possible attacks.**

***Index Terms*—About four key words or phrases in alphabetical order, separated by commas.**

I. Introduction

WSN is composed of a large set (hundreds to a few thousand) of homogeneous nodes with extreme resource constraints. Each sensor node has wireless communication capability plus some level of intelligence for signal processing and data networking. These nodes are usually scattered over the area to be monitored to collect data, process it, and forward it to a central node for further processing. Military sensor networks might detect and gather information about enemy movements of people and equipment, or other phenomena of interest such as the presence of chemical, biological, nuclear, radiological, explosive materials [26]. WSNs can support a myriad of uses including military, commercial, environmental, and medical applications. Natural environments such as remote ecosystems, disaster sites, endangered species, agriculture conditions, and forest fires can also be monitored with sensor networks.

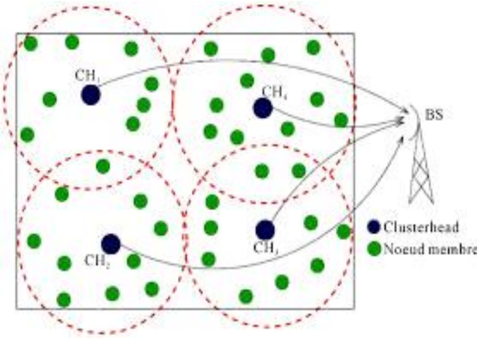

Figure 1.1 Architecture of a WSN

Short distances, sense environmental data, and perform limited data processing. A typical node might have only 4MHz of processing power, 4KB of RAM, and a short transmission distance of less than 100 feet. Tiny OS is a small, open source operating system developed to support most WSN applications. Wireless sensor networks often contain one or more base station that provides centralized control. A sink typically serves as the access point for the user or as a gateway to another network [35]. The sensor nodes communicate using RF, so broadcast is the fundamental communication primitive.

Security is one of the most difficult problem faced by these networks. For certain applications of sensor networks, like military applications, security becomes very important due to three major reasons. First, wireless communication is difficult to protect since it is realized over a broadcast medium. In a broadcast medium, adversaries can easily eavesdrop on, intercept, inject, and alter transmitted data. Second, since sensor networks may be deployed in a variety of physically insecure environments, adversaries can steal nodes, recover their cryptographic material, and pose as authorized nodes in the network. Third, Sensor networks are vulnerable to resource consumption attacks. Adversaries can repeatedly send packets to drain a node battery and waste network bandwidth. In these and other vital or security-sensitive deployments, secure transmission of sensitive digital information over the sensor network is essential [26]. The use of encryption or authentication primitives between two sensor devices requires an initial link key establishment process, which must satisfy the low power and low complexity requirements.

1.2 Security Requirements in WSNs

The most important security requirements in WSN are listed below:

1.2.1 Data confidentiality

The security mechanism should ensure that no message in the network is understood by anyone except intended

First Author name, CSE, SIST, Bhopal, India, 9691104389.

Second Author name, , CSE, SIST, Bhopal, India, 9752202008,

recipient.

1.2.2 Data Integrity

With the implementation of confidentiality, an adversary may be unable to steal information. However, this doesn't mean the data is safe. The adversary can change the data, so as to send the sensor network into disorder.

1.2.3 Availability

This requirements ensures that the services of a WSN should be available always even in presence of an internal or external attacks such as a data integrity attack.

1.3.4 Self-organization

Each node in a WSN should be self-organizing and self-healing. This feature of a WSN also poses a great challenge to security. The dynamic nature of a WSN makes it sometimes possible to deploy any preinstalled shared key mechanism [29] among the nodes and the base station.

1.3.6 Secure localization

In many situations, it becomes necessary to accurately and automatically locate each sensor node in a WSN. For example, a WSN designed to locate faults would require accurate locations of sensor nodes identifying the faults. A potential adversary can easily manipulate and provide false location information by reporting false signal strength, replaying messages etc. if the location information is not secured properly. In multilateration, the position of a device is accurately computed from a series of known reference points. In [14] have described a technique called verifiable multilateration .

II. Literature Review

A wireless sensor network consists of a large number of sensor nodes which are inherently resource-constrained. These nodes have limited processing capability, very low storage capacity, and constrained communication bandwidth. These limitations are due to limited energy and physical size of the sensor nodes. Due to these constraints, it is difficult to directly employ the conventional security mechanisms in WSNs Security is one of the most difficult problem faced by these networks.

In" Secure and Efficient Data Transmission for Cluster-Based Wireless Sensor Networks" [1], Author Proposed a scheme, they studied a secure data transmission for cluster-based WSNs (CWSNs), where the clusters are formed dynamically and periodically. They proposed two secure and efficient data transmission (SET) protocols for CWSNs, called SET-IBS and SET-IBOOS, by using the identity-based digital signature (IBS) scheme and the identity-based online/ offline digital signature (IBOOS) scheme, respectively. In SET-IBS, security relies on the hardness of the Diffie-Hellman problem in the pairing domain. SET-IBOOS further reduces the computational overhead for protocol security, which is crucial for WSNs, while its security relies on the hardness of the discrete logarithm problem. They show the feasibility of the SET-IBS and SET-IBOOS protocols with respect to the security requirements and security analysis against various attacks. The calculations and simulations are provided to illustrate the efficiency of the proposed protocols. The results show that the proposed protocols have better performance than the existing secure protocols for CWSNs, in terms of security overhead and energy consumption.

In "Constrained Function-Based Message Authentication (CFA) for Sensor Networks" [2], Author Proposed a scheme, this can be thought of as a hash function directly supporting the en-route filtering functionality. Obviously, the crux of the scheme lies on the design of each sensor to have en-route filtering capability guaranteed. Together with the redundancy property of sensor networks, Which it means that an event can be simultaneously observed by multiple sensor nodes, the devised CFA scheme is used to construct a CFA-based en-route filtering (CFAEF) scheme. In addition to the resilience against false data injection and PDoS attacks, CFAEF is inherently resilient against false endorsement-based DoS attack. In contrast to most of the existing methods, which rely on complicated security associations among sensor nodes, this design, which directly exploits an en-route filtering hash function, appears to be novel. Those examine the CFA and CFAEF schemes from both the theoretical and numerical aspects to demonstrate their efficiency and effectiveness. Moreover, prototype implementation on TelosB mote demonstrates the practicality of this proposed method.

In "Providing End-to-End Secure Communications in Wireless Sensor Networks" [3], Author Proposed an end to end secure communication protocol in randomly deployed WSNs. Specifically, this protocol is based on a methodology called differentiated key pre-distribution. The core idea is to distribute different number of keys to different sensors to enhance the resilience of certain links. This feature is leveraged during routing, where nodes route through those links with higher resilience. Using rigorous theoretical analysis, these works derive an expression for the quality of end to end secure communications, and use it to determine optimum protocol parameters. Extensive performance evaluation illustrates that the proposed solutions can provide highly secure communications between sensor nodes and the sink in randomly deployed WSNs. We also provide detailed discussion on a potential attack (i.e. biased node capturing attack) to our solutions, and propose several countermeasures to this attack. The work present an end to end secure communication protocol based on the above methodology by extending well known location centric (GPRS) and data centric (minimum hop) routing protocols. Detailed theoretical analysis and performance evaluations demonstrate the strengths of these techniques.

In "Noninteractive Pairwise Key Establishment for Sensor Networks" [4], Author Proposed a Constrained Random Perturbation-based pairwise key establishment (CARPY) scheme and its variant, a CARPY+ scheme, for WSNs. Compared to all existing schemes which satisfy only some requirements in so-called sensor-key criteria, including 1) resilience to the adversary's intervention, 2) directed and guaranteed key establishment, 3) resilience to network configurations, 4) efficiency, and 5) resilience to dynamic node deployment, the proposed CARPY+ scheme meets all requirements. In particular, CARPY+ is the first noninteractive key establishment scheme with great resilience to a large number of node compromises designed for WSNs.

Together with a comprehensive comparison, theoretical and experimental results are provided to validate the performance of the CARPY and CARPY+ schemes. These schemes have also been practically implemented on the TelosB compatible mote to evaluate the corresponding performance and overhead.

III. Possible Attacks against the WSN

Most of the routing protocols proposed for ad hoc and sensor network are not designed to handle security related issues. Therefore there is a lot of scope for attacks on them. Different possible attacks [15][16][17][21][31] on the flow of data and control information can be Categorized as follows:

3.1 Routing Attacks in Sensor Networks

In fig. 3.2 this section we have tried to explore different types of attack that can affect sensor network protocols. The attacks against the sensor network can be broadly categorized into following groups:

1) Spoofed, altered, or replayed routing information
2) Selective forwarding
3) Sinkhole attacks
4) Sybil attacks
5) Wormholes
6) HELLO flood attacks
7) Acknowledgement spoofing
8) Sniffing
9) Data integrity
10) Energy drain
11) Black hole

3.4.1.1 Spoofed, altered, or replayed routing information

This is the most common attack against a routing protocol. This attack targets the routing information exchanged between the nodes [12]. Adversaries may be able to create routing loops, attract or repel network traffic, extend or shorten source routes, generate false error messages, partition the network, and increase end-to-end latency.

3.4.1.2 Selective forwarding attack

Multi-hop mode of communication is commonly preferred in wireless sensor network data gathering protocols. Multi-hop networks assume that participating nodes will faithfully forward and receive messages [13]. However a malicious node may refuse to forward certain messages and simply drop them, ensuring that they are not propagated any further.

3.4.1.3 Sybil Attack

Most protocols assume that nodes have a single unique identity in the network. In a Sybil attack, an attacker can appear to be in multiple places at the same time. This can be convincing by creating fake identities of nodes located at the edge of communication range. Multiple identities can be occupied within the sensor network either by fabricating or stealing the identities of legitimate nodes. Sybil attacks can pose a significant threat to geographic routing protocols [13].

3.4.1.4 Black hole/Sinkhole Attack

By sinkhole attack, the adversary tries to attract nearly all the traffic from a particular area through a compromised node. A compromised node which is placed at the centre of some area creates a large “sphere of influence”, attracting all traffic destined for a base station from the sensor nodes. The attacker targets a place to create sinkhole closer to the base station so that the malicious node could be perceived as a base station.

3.4.1.5 Hello Flood Attack

Many protocols require nodes to broadcast HELLO packets for neighbor discovery, and a node receiving such a packet may assume that it is within (normal) radio range of the sender. A laptop-class attacker with large transmission power could convince every node in the network that the adversary is its neighbor, so that all the nodes will respond to the HELLO message and waste their energy.

3.4.1.6 Wormhole Attack

In this attack an adversary could convince nodes who would normally be multiple hops from a base station that they are only one or two hops away via the wormhole. The simplest case of this attack is to have a malicious node forwarding data between two legitimate nodes. Wormholes often convince distant nodes that they are neighbours, leading to quick exhaustion of their energy resources.

3.4.1.7 Acknowledgement spoofing

Several sensor network routing algorithms rely on implicit or explicit link layer acknowledgements. Due to the inherent broadcast medium, an adversary can spoof link layer acknowledgments for “overheard” packets addressed to neighboring nodes. Protocols that choose the next hop based on reliability issues are susceptible to acknowledgments spoofing.

3.4.1.8 Sniffing attack

Sniffing attack is a good example of interception or listen-in channel attack. In this attack an adversary node is placed in the proximity of the sensor grid to capture data. The collected data is transferred to the intruder by some means for further processing.

3.4.1.9 Data integrity attacks

Data integrity attacks compromise the data travelling among the nodes in WSN by changing the data contained within the packets or injecting false data. The attacker node must have more processing, memory and energy than the sensor nodes.

3.4.1.10 Energy drain attacks

WSN is battery powered and dynamically organized. It is difficult or impossible to replace/recharge sensor node batteries. Because there is a limited amount of energy available, attackers may use compromised nodes to inject fabricated reports into the network or generate large amount of traffic in the network.

3.4.2 Denial of Service

The simplest DoS attack tries to exhaust the resources available to the victim node, by sending extra unnecessary packets and thus prevents legitimate network users from accessing services or resources to which they are entitled. DoS attack is meant not only for the adversary’s attempt to subvert, disrupt, or destroy a network, but also for any event that diminishes a network’s capability to provide a service

IV. PROPOSED METHOD

The objective of this dissertation is to propose a symmetric key based security technique using Hash function for cluster based low energy multi hop data gathering in WSN. In this technique every node uses three types of keys: Individual key (preloaded), Pair wise key (computed at cluster set up phase), Group key (preloaded) and Hash function (preloaded). In this technique we have tried to minimize and localize possible attacks.

4.1 Our Assumptions

This is a cluster based multi-hop data gathering technique for WSN. Sensor nodes form clusters of different levels with the support of Base Station. Leaf nodes in the cluster hierarchy send sensed information to their cluster head and finally root level cluster head sends information to Base Station. The base station (which acts as a key server), is assumed to be a mobile device and is considered to be trust worthy. Every node has memory for storing up to hundreds of bytes of keying materials. If a node is compromised, the adversary can tamper the memory content of the node. Before deployment of the nodes, base station shares three common values (group key (GK), Seed value (SBS) and Hash function) with each node.

4.2 Security Technique

This technique is based on symmetric key cryptography. We are using Hash function along the keys to securing network. The keys and hash function are:

i) Individual key: Used for communication between BS and a node.

ii) Pair wise key: Used for communication between cluster head and cluster member.

ii) Group key: Used for broadcast in a group.

iv) Hash function: Used for message and node authenticity.

4.2.1 Individual key

Individual key (UKi) is used for communication between BS (base station) and a particular node with id i. Initially before deployment of nodes, BS loads a seed value (SBS). Using its own node-id and SBS each node generates the Individual key and deletes SBS. This computation is done prior to deployment so it is difficult for an adversary to generate the Individual key. The nodes interested for first order cluster head sends a request message (RM) to BS. A node calculate Hash code to request message(RM) and encrypt RM using IKi and group key(GK) as EGK{(i, UKi,((RM) +HRM))} & send it to Base Station. Request message contains Source-id, Energy level, Destination-id. BS decrypts this message using group key (GK). From source-id, BS calculates the Individual key using seed SBS for communicate with the node. Base station stores SBS for future Computation. Instead of storing Individual key of all the nodes, BS may use above approach to calculate it so as to reduce space complexity. The Individual key UKi for a node i is calculated as f (SBS, i). Here f is a pseudo-random function and SBS is a random value known only to the BS.

4.2.2 Pairwise key

Pair wise key (PK CH,i) is used for secure communication between cluster head (CH) and a member node i of that cluster. The node decided to be a CH generates a seed value and broadcast it in a advertise message to form the cluster. This message is encrypted using group key (GK). After receiving and decrypting this message a node calculates a pair wise key (PK CH,i) by using cluster head id (CH-id), node id and the received seed value. Once the pair wise key is calculated the node i sends joining message(JM) after encrypting it using PKCH,i and GK as EGK(i,PKCH,i (JM)+HJM) to CH. CH decrypts this message using GK. From CH-id, node id and seed value pair wise key is calculated. After that rest part of the joining message is decrypted for authentication. When battery power of first order cluster head goes below the threshold value, a backup node takes the responsibility of first order cluster head. So the cluster head transmits all the stored pair wise keys to the backup node by encrypting with the pair wise key of backup node and cluster head, so that the backup node will act as the first order cluster head. Pair wise key between cluster head and node i PKCH,i is calculated as f(CH-id, SCH,i). Once pair wise key is calculated, key SCH is erased.

4.2.3 Group key

Group key (GK) is shared by all the nodes in the network. This key is used for encryption and decryption of queries and control messages. A group key for a sensor network is pre-load on every node. Group key needs frequent updation for security point of view. BS periodically broadcast a group key updating instruction. Every node generates a new group key based on old GK. $GK^1 = f\ (GK^0)$. The old key GK is erased.

4.2.4 Hash Function

Hash function is shared by all the nodes in the network. The purpose of a MAC (hash code) is to authenticate both the source of a message and its integrity. The keyed-hash message authentication code (HMAC) or hash function have two functionally distinct parameters, a message input and a secret key(ik, pk, gk) known to the message originator node and intended receiver node only. An HMAC function is used by the sender node to produce a value (the MAC) that is formed by condensing the secret key and the message input. The MAC is typically sent to the receiver node along with the message. The receiver node computes the MAC on the received message using the same key and HMAC function as was used by the sender, and compares the result computed with the received MAC. If the two values match, the message has been correctly received and the receiver is assured that the sender is a member of the cluster of nodes that share the key.

4.3 Secured Cluster Formation

The main idea of level wise cluster formation is as follows. Sensor nodes are deployed randomly in the sensor field. In figure 4.1 the nodes which are interested to be the root level cluster head (RCH) calculate hash code to request message (RM) and encrypt (RM) using UKi and GK as EGK{(i, UKi,((RM) +HRM))} & send a request packet to base station node.

The request packet contains sensor node_ID, residual battery power, coordinate position and BS_ID. The base station selects one RCH from among those nodes based on the remaining battery power. It also selects a node as a backup RCH node within the neighboring area of RCH based on the nodes co-ordinate information and remaining battery power. The backup RCH will take the responsibility of RCH once its battery power goes down. This will reduce cluster formation overhead. Then the base station replies back to the selected cluster head and backup head node with an acknowledgement. Base station BS encrypt Acknowledgement (ACK) packet using UKi and GK and calculate hash code to Ack as GK{(Ch_id, UK_{CH},((Ack)+hash code))} and send it to the cluster head node and backup node.

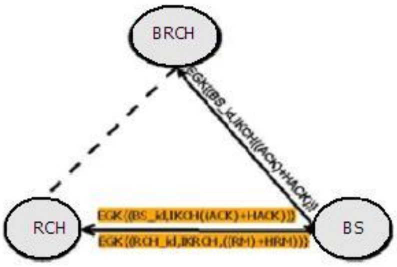
BRCH
RCH
BS
EGK((BS_id,IKCH((ACK)+HACK))
EGK((BS_id,IKCH((ACK)+HACK))
EGK((RCH_id,IKRCH,((RM)+HRM)))

Figure 4.1 Root level cluster head selection

The Acknowledgement (ACK) packet contains BS_ID, Sensor node_ID, Backup node_ID and a hop-count value. The hop count value gives the idea about maximum level of cluster that can be formed. When a CH receives the ACK packet, it stores the hop-count value reducing it by one. Then the cluster head nodes broadcast the advertisement message to the neighbor's nodes (which are within the radio-range of RCH) to form a cluster. The advertisement message contains CH_ID, Sensor node_ID, Backup node_ID and hop-count. In the advertisement message CH sends the hop-count decreasing by one. Nodes within the neighboring area of CH who decide to join the cluster acknowledge this advertisement to send joining message to CH. The cluster formed with this RCH is known as root level cluster, first order cluster or one hop clusters. Once this cluster formation is over, non cluster head nodes broadcast CH advertise message again to form next level clusters. These advertise message contains same fields as described earlier.

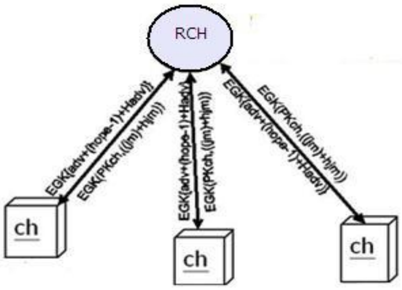

Figure 4.2 Cluster formation

Table 4.1 Key terms

Notation	meaning
BS	Base station
GK, PK, UK	Group key ,Pair wise key , Individual key
f	Pseudorandom function
S_{CH},S_{BS}	Seed value of cluster and base station node
Rm,jm	Request message, joining message
Ack	acknowledgement
Pk_{ch}	Pair wise key of cluster head node
RCH	Root level cluster head
BRCH	Backup RCH
H	Hash code

It may happen that a node will receive more than one cluster advertisement message. In figure 4.2 nodes decide to join that cluster whose hop-count value is highest among received advertisement message as higher the counter value, smaller is the distance from the BS. The new sub clusters formed are known as second order cluster or two hop cluster. This process continues till all the nodes in the network are grouped in to the clusters. In case, after a predefined time period a node could not receive any cluster advertisement message, it tries to find the nearest cluster and associates with it.

In figure 4.3 once the cluster setup phase is over, non cluster-head nodes start transmitting data to the head node of a cluster. The head node receives data from all the members within the cluster and combines them with its own data. Finally it sends the combined data to its upper level node. This process continues till the data reaches to BS. The Data transmission phase continues till the root level cluster head has sufficient battery power to transmit.

Every cluster head checks its energy level periodically and informs to the cluster members while acknowledging the received data. If the remaining battery power of the root level cluster head goes down a predefined threshold value then it informs to the backup node to carry out further communication. Member nodes of the first order cluster check, if the cluster heads energy level goes below a predefined threshold value. In that case, they start transmitting to backup node. When the residual battery power of the backup node goes below the threshold value then it informs the base station to initiate re-clustering. The backup node helps reducing clustering overhead to some extent.

4.4 Proposed Algorithm for Secured Clustering

The proposed security technique is a symmetric key based secured data gathering, we are using three types of keys and hash function to provide security in the network.

Step 1 Create base station (BS) and sensor nodes (i), BS preload GK and Hash function to each sensor node.

Step 2 Group key (GK) generation

(a) BS create a GK
(b) BS broadcast GK in the network.
(c) BS periodically update GK instruction

New GK^1 = f(GK^o)old. : Where f = Pseudorandom function

(d) Erase old $GK^{o.}$

Step 3 Individual key generation (UK)

(a) Initially BS loads seed (S_{BS}) to all nodes.

(b)Individual key (UK $_{i}$) = node id$_{i}$ + seed$_{BS.}$

UK =f ($S_{BS, i}$)

(c) Node deletes $S_{BS.}$

Step 4 Root level cluster head (RCH) selection.

(a) A node calculate hash code to request message(RM) and encrypt (RM) using IKi and GK as EGK{(i, UKi,((RM) +HRM))} & send to BS.

(b) RM = Source id$_{(i)}$+ Energy level$_{(i)}$+ destination id$_{(BS)}$ + Co-ordinate position.

(c) BS decrypt RM using GK & Individual key as (UK$_{(i)}$) = Sourse_id$_{(i)}$+ seed(S_{BS}) and calculate hash code to RM and compare it to received hash code.

(d) BS encrypt Ack using UKi and GK and calculate hash code to Ack as EGK{(Ch_id, UK$_{CH}$,((Ack)+HACK))} and send it to the cluster head node.

(e) Ack = Bs_id + node_id + RCH_id +hop_count.

Step 5 Pair wise key generation (PK)

(a) CH generate a seed, calculate hash code to seed (S_{CH}) & encrypt it using GK as EGK{(Ch_id+((S_{CH})+HSch)) & broadcast it in the cluster.

(b) A node calculate pair wise key(PK) First a node decrypt broadcast message using GK and calculate hash code to seed and compare it to received hash code.

(c) (PK$_{CH, i}$) = CH_id + node_id +S_{CH} : f(Ch_id,S_{CH} ,i)

(d) Cluster member node deletes $S_{CH.}$

Step 6 Cluster formation

(a) A node encrypt joining message (jm) using (PK$_{CH,}$ i) and GK as EGK(i, PK$_{CH, I,}$((jm)+HJM)) and calculate hash code to JM and send it to the CH.

(b) JM = Source_id$_{(i)}$ + energy level$_{(i)}$ + CH_id.

(c) Cluster head (CH) decrypt JM using GK and pair wise key as (PK$_{CH, i}$) = S_{Ch}+ Source_id + CH_id and calculate hash code to JM and compare it to the hash code of jm.

Step 7 Cluster reformation

(a) Erase all pair wise key.

(b) Each node store Individual key and pair wise key.

(c) Repeat step 4 and step 6.

V. Result Analysis

Survivability of our work

When a sensor node u is compromised, the adversary can launch attacks by utilizing node u's keying materials. If the compromise event is detected somehow, our scheme can reform node u from the group efficiently. Basically, every neighbor of node u deletes its pairwise key shared with u and updates its group key. The group key is also updated efficiently. After the reclustering, the adversary cannot launch further attacks. However, compromise detection in sensor systems is more difficult than in other systems because sensor systems are often deployed in unattended environments. Thus, we believe survivability under undetected node any sensor networks. Below we first consider in general what the adversary can accomplish after it compromises a sensor node. First, the unique key of a node is only used for communication between first order cluster head node and base station. A node calculates unique key from the seed value given by BS and then erases the seed. Therefore it is almost impossible to generate the key by an attacker. The attacker may also try to extract the key from a cluster head failed node. In our technique the first order cluster head nodes fail earlier than other nodes. By keeping track of the failed nodes, BS can find out if an attacker is trying to use key of a failed node.

Second, we use pair wise key for communication between member nodes and its CH .Possessing the pairwise shared keys and group keys of a compromised node allow the adversary to establish trust with all the neighboring nodes. Thus the adversary can inject some malicious routing control information or erroneous sensor readings into the network. However, in our thesis the adversary usually has to launch such attacks by using the identity of the compromised node due to the use of our inter-node authentication, Therefore even if a non head node of a cluster is captured, other nodes will not be affected as pair wise key is unique for each node and it is difficult to detect a CH. We note a salient feature of our technique is its ability in localizing the possible damage, because after the network deployment, every node keeps a copy of secret key of trusted neighboring nodes. Thus, the compromised node cannot establish trust relationship with any nodes except its neighbors, which means the adversary, cannot risk the secure links among any other nodes.

Third, the group key is used for broadcast message from base station and for encryption/decryption of control packets during cluster setup phase. Possessing the group key allows the adversary to decrypt the messages broadcast by the base station. Since a broadcast message, by its nature, is intended to be known by every node, compromising one single node is enough to reveal the message. Moreover, possessing the group key does not enable the adversary to flood the entire network with malicious packets impersonating the base station, because any messages sent by the base station are authenticated using HMAC. Finally, because we deploy a periodic group rekeying scheme, the adversary can decrypt only the messages being encrypted using the current group key.

Finally we analyzed security mechanism in various attacks described below. We found most of the attacks can be prevented using good encryption technique(Like Spoofed, altered, or replayed routing information, Acknowledgement spoofing, Sniffing, Data integrity) and some of the attacks can be prevented or minimized by protecting adversary to join the network.(like Selective forwarding, Sinkhole attacks, Sybil attacks, HELLO flood attacks, Energy drain, Black hole).

VI. Conclusion

Although research efforts have been made on cryptography, key management, secure routing, secure data aggregation, and instruction detection in WSNs, there are still some challenges to be addressed. From an authentication perspective, the HMAC function is the most popular method of providing authentication for symmetric key based algorithms. All of the previously mentioned security threats, the Hello flood attack , wormhole attack, Sybil attack , sinkhole attack, serve one common purpose

that is to compromise the integrity of the network they attack. Although some solutions have already been proposed, there is no single solution to protect against every threat. In this technique, we mainly focus on the security threats in WSN.

We've presented the summary of the WSNs threats affecting different layers along with their defense mechanism. We have proposed a symmetric key and hash function based security technique and the technique used for establishing and updating these keys. It is also efficient for inter-node traffic authentication based on the use of hash function. We conclude that the defense mechanism presented just gives guidelines about the WSN security threats. The exact solution depends on the type of application the WSN is deployed for. We found most of the attacks can be prevented using good encryption technique (Like spoofed, altered, or replayed routing information, acknowledgement spoofing , sniffing, data integrity) and some of the attacks can be prevented or minimized by protecting adversary to join the network.(like selective forwarding, sinkhole attacks, Sybil attacks, hello flood attacks, energy drain , black hole) we analyzed the performance and the security of our technique and found it to be very efficient in defending against many attacks.

APPENDIX

Appendixes, if needed, appear before the acknowledgment.

ACKNOWLEDGMENT

The preferred spelling of the word “acknowledgment” in American English is without an “e” after the “g.” Use the singular heading even if you have many acknowledgments. Avoid expressions such as “One of us (S.B.A.) would like to thank ... .” Instead, write “F. A. Author thanks ... .” **Sponsor and financial support acknowledgments are placed in the unnumbered footnote on the first page**.

REFERENCES

[1] Huang Lu, Student Member, IEEE, Jie Li, Senior Member, IEEE, and Mohsen Guizani, Fellow, IEEE, “Secure and Efficient Data Transmission for Cluster-Based Wireless Sensor Networks” IEEE Transactions On Parallel And Distributed Systems, Vol. 25, No. 3, March 2014.

[2] Chia-Mu Yu, Student Member, IEEE, Yao-Tung Tsou, Chun-Shien Lu, Member, IEEE, And Sy-Yen Kuo, Fellow, IEEE, ”Constrained Function-Based Message Authentication For Sensor Networks”, IEEE Transactions On Information Forensics And Security, Vol. 6, No. 2, Pp. 407- 425, June 2011.

[3] Wenjun Gu, Neelanjana Dutta, Sriram Chellappan, And Xiaole Bai, “Providing End-To-End Secure Communications In Wireless Sensor Networks”, IEEE Transactions On Network And Service Management, Vol. 8, No. 3, Pp. 205- 218, September 2011.

[4] Chia-Mu Yu, Student Member, IEEE, Chun-Shien Lu, Member, IEEE, And Sy-Yen Kuo, Fellow, IEEE, ”Noninteractive Pairwise Key Establishment For Sensor Networks”, IEEE Transactions On Information Forensics And Security, Vol. 5, No. 3, Pp. 556- 569, September 2010.

[5] Khadija Rasul1, Nujhat Nuerie1, And Al-Sakib Khan Pathan2, “Securing Wireless Sensor Networks With An Efficient B+ Tree-Based Key Management Scheme”, International Journal Of Communication Networks And Information Security (IJCNIS), Vol. 2, No. 3, Pp. 162-168, December 2010.

[6] Xu Jianbo, GUO Jian, Long Jing, Zhou Xinlian, ”Mobile Sink-Based Data Gathering Protocol”, 2010 International Forum On Information Technology And Applications, Vol.2, Pp. 427-430 .

[7] Saeed Rasouli Heikalabad, Nasrin Firouz, Ahmad Habibizad Navin, Mir Kamal Mirnia, ”HEECH: Hybrid Energy Effective Clustering Hierarchical Protocol For Lifetime Prolonging In Wireless Sensor Networks”, 2010 International Conference On Computational Intelligence And Communication Networks, Pp.325-328.

[8] Rabindra Bista, Hye-Kyeom Yoo, And Jae-Woo Chang, ”A New Sensitive Data Aggregation Scheme For Protecting Integrity In Wireless Sensor Networks”, 2010 10th IEEE International Conference On Computer And Information Technology (CIT 2010), Pp.2463-2470.

[9] Shu Qin Ren, Khin Mi Mi Aung, Jong Sou Park, ”A Privacy Enhanced Data Aggregation Model ”, 2010 10th IEEE International Conference On Computer And Information Technology (CIT 2010), Pp.985-990.

[10] Suat Ozdemir, Member, IEEE, And Hasan Çam, Senior Member, IEEE 2010, ”Integration of False Data Detection With Data Aggregation And Confidential Transmission In Wireless Sensor Networks”, IEEE/ACM Transactions On Networking, Vol. 18, No. 3, Pp. 736-749, June 2010.

[11] P. Mohanty, S. Panigrahi, N. Sarma And S. Satapathy, “HCEPSN: A Hierarchical Cluster Based Energy Efficient Data Gathering Protocol For Sensor network”, Anti-counterfeiting, Security, and Identification in Communication, 2009. ASID 2009. 3rd International Conference on, pp.138-143.

[12] Siahaan, I. And Fernandes, L.(2008), “Secure routing In Wireless SensorNetworks”,UniversityofTrento.Http://Dit.Unitn.It/Fernand/Downloads/IWSNS Lides. Pdf

[13] Fernandes, L. L., (2007) “Introduction To Wireless Sensor Networks Report”,UniversityOfTrento. ttp://Dit.Unitn.It/~Fernand/Downloads/Iwsn.Pdf

[14] S. Capkun And J.-P. Hubaux, “Secure Positioning In Wireless Networks”, IEEE Journal On Selected Areas In Communications, Vol. 24, No. 2, Pp. 221-232, 2006.

First Author name, CSE, SIST, Bhopal, India, 9691104389.
Second Author name, , CSE, SIST, Bhopal, India, 9752202008,