Defense Mechanisms against Hello Flood Attack in Wireless Sensor Network

Siddhartha Choubey[1], Abha Choubey[2] ,M.Abhilash[3], Kamal K Mehta[4]

siddhartha00@rediffmail.com , niceabha1@rediffmail.com , abhilash576@gmail.com , *kkmehta28@yahoo.com* ,

[1] *Reader, CSE Dept, SSCET, Bhilai*
[2] *Sr. Lecturer, CSE Dept, SSCET, Bhilai*
[3] *ISTE member , CSE, SSCET, Bhilai*
[4] *Asstt.Professor,CSE , SSCET , Bhilai*

Abstract : In a large-scale sensor network individual sensors are subject to security compromise. Where the nature of communication is broadcast and, hence, an attacker can overhear messages posted by any sensor node; security is an important issue here.

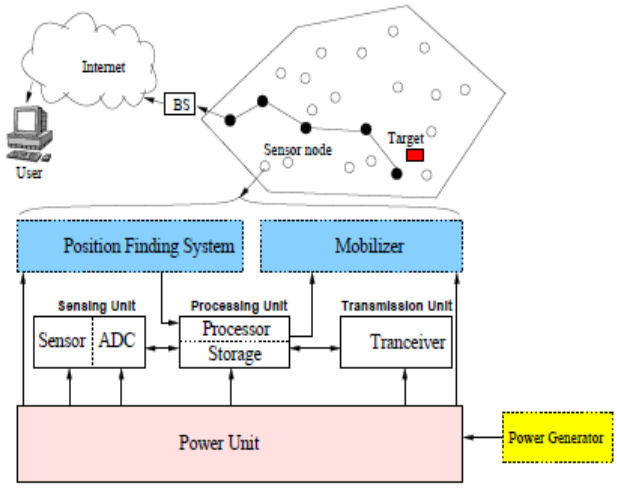
In this paper we consider Wireless Sensor Network (WSN) security and focus our attention to tolerate damage caused by an adversary who has compromised deployed sensor node to modify, block, or inject packets. We adopt a probabilistic secret sharing protocol using the concept of cryptography where secrets shared between two sensor nodes are not exposed to any other nodes. Adapting to WSN characteristics, we incorporate these secrets with bidirectional verification and multipath routing to multiple base stations to defend against HELLO flood attacks. We then analytically show that our defense mechanisms against HELLO flood attack can tolerate damage caused by an intruder.

Keywords: wireless sensor networks, cryptography, hello flood attack.

1. INTRODUCTION

1.1 What is a Wireless Sensor Network(WSN)

A wireless sensor network is a network of multiple sensing nodes that perform a certain task .The network can consist of any number of sensing nodes ,and each sensor node has the ability to store and send information across the network . Every sensing node has its own battery , memory , processor , transceiver and sensing device.

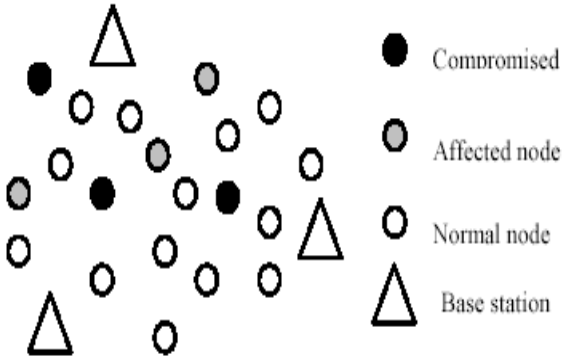


(Fig 1: Components of a sensor node)

1.2 How and why WSN are used

Wireless Sensor Networks (WSNs) are comprised of many small and resource constrained sensor nodes that are deployed in an environment to gather sensed data and forward that data to interested legal users.

Advances in micro-electro-mechanical systems (MEMS) technology allow sensors to be reprogrammable, selflocalizing, and to support low-energy, wireless, multi-hop networking, while requiring only minimal pre-configuration. To support the reliability of coordinated control, management, and reporting functions, the sensor networks are selforganizing with both decentralized control and autonomous sensor behavior, resulting in a sophisticated processing capability.



(Fig 2. In a sensor network, compromised nodes spoof, inject, modify, or represents false identity to affect normal sensor node to collect sensed data.)

- The sensing nodes have the ability to communicate with each other and collect information about the area of interest.
- The information can be stored in a special node called the sink node, or it can be sent to a neighbor node (a node with short distance).

1.3 Usefullness of WSN

- Gather information of the area in which it is located , usually environmental data.
- Also used because of their amal size and ability to exist in discrete areas.
- Currently they have provided usefulness to several important fields such as:

1. Environmental
2. Industrial
3. Medical
4. Military

2. CONTRIBUTION OF THE PAPER

The main contributions of the paper are as follows:

• We present probabilistic secret sharing protocol adopted from [1] where, a small increase in the number of secrets maintained by a user substantially reduces the probability of privacy compromise. And it is beneficial for the case where the sensor nodes do not have the capability to hold sufficient secret to ensure privacy. We show how these secrets can be used to route packets in a secured way.

• Then we propose defense mechanisms against HELLO flood attack using the secrets that nodes share among themselves.

3. ORGANIZATION OF THE PAPER

This paper is organized as follows. Section 4 discusses the network assumption and threat model and capabilities of sensor nodes. In section 6, the key sharing protocol is described in brief. Section 7 describes the defense against HELLO flood attack and addresses problem associated with this defense and section 8 addresses further defense to tolerate the damage. In section 9, we discuss about our counter measures against HELLO flood attacks and section 10 concludes the paper.

4. NETWORK ASSUMPTION AND THREAT MODEL

We consider a network composed of moderately large number of resource constrained sensor nodes[4]. We further assume that the sensor nodes are deployed in high density, e.g battlefield deployments. Each sensor node has a communication range such that if the distance between two sensors is more than this range, they can not communicate. We also assume that the communications channels are bidirectional, i.e. if a node *y* can receive a message from *z*, then it can also send a message to *z*. We assume that an adversary can pose the following threat:

• An attacker can cause a HELLO flood attack by advertising a very high quality route to the base station. So, every node in the network could cause a large number of nodes to attempt to use this route thereby sending the legitimate packets beyond the actual destination.

5. SUSPICIOUS NODE DETECTION AND SIGNAL STRENGTH

As mentioned in [3] ,assume that all the nodes are homogeneous (have same hardware and software) , symmetric (only communicate with a node that can communicate with them) and static. It concludes that a node is malicious if the signal strength is different from the signal strength agreed upon by all the nodes in the network.

• If a node Is thought to be suspicious, it Is classified as such.
• Every node contains a table that keeps track of the number of suspicious and unsuspicious votes of other nodes.
• The table in constantly updated when a new vote Is made.

This is a way to protect against a malicious node getting classified as unsuspicious or vise versa.

6. KEY SHARING PROTOCOL

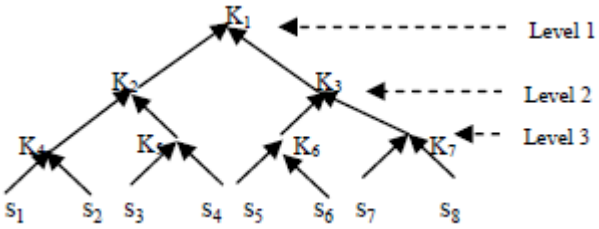
In this section, we present the probabilistic protocol, the tree protocol, for assigning the initial secrets. We will describe the single tree protocol and then compute the multiple trees based key assignment.

6.1 Secret instantiation by Tree Protocol

We present single tree and then multiple tree protocol as described in [2]. For each of these versions, we first identify the secret distribution protocol that determines the secrets that each user should get. Then, we present the secret selection protocol; when two users need to communicate, they use this protocol to determine a shared secret that they should use. Subsequently, we compute the probability of compromise. We organize the secrets in a tree . Each non-leaf node is associated with a secret and each leaf is associated with a sensor node. Each sensor node is assigned an ID that identifies its location in the tree. Finally each sensor node is provided the secrets along the path towards the root. Thus, node *s1* has the secrets, *k1*, *k2* and *k4*.

When two nodes, say, *s1* and *s2*, want to exchange messages during their effective communication, they first exchange their identities. Then, they identify their least common ancestor and based on the secret distribution mechanism, the common secret associated with this ancestor will be available to both *s1* and *s2*. So, the secret associated with the ancestor will be used for communication between *s1* and *s2*. For example, two

nodes *s1* and *s2* want to communicate then they will use secret key *k4* whereas if *s1* and *s5* want to communicate then they will use secret key *k1*.



(Fig 3: Single tree key assignment)

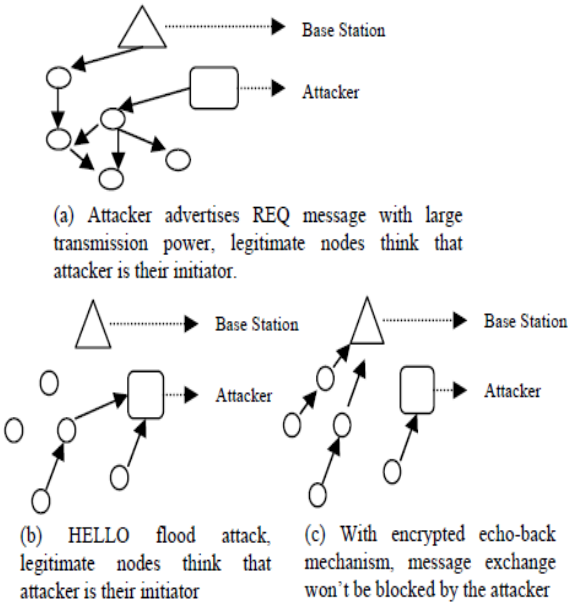
7.COUNTER MEASURE AGAINST HELLO FLOOD ATTACKS (BIDIRECTIONAL VERIFICATION)

Many protocols require nodes to broadcast HELLO packets to announce themselves to their neighbors, and a node receiving such a packet may assume that it is within (normal) radio range of the sender. This assumption may be false: a laptop-class attacker broadcasting routing or other information with large enough transmission power could convince every node in the network that the adversary is its neighbor. To launch this kind of attack, an adversary's packet sending range must be bigger than a normal node's sending range. If each sensor node constructs a set of reachable neighbor nodes, and is only willing to receive REQ messages from this set of neighbor nodes, then REQ messages from an adversary transmitted with larger power will be ignored. Thus, the damage from a HELLO flood attack can be restricted within a small range. To defend against attack, each request (REQ) message forwarded by a node is encrypted with a key. As we have shown from the tree protocol that any two sensor nodes share some common secrets, the new encryption key is generated on-the-fly (i.e. during communication). In this way, any node's reachable neighbors can decrypt and verify the REQ message while the attacker will not know the key and will be prevented from launching the attack. We show that the new key combined with the echo-back mechanism can well protect this attack. Each node locally broadcasts an echo message to its neighbor with format:

***s1*□: ECHO||Enew-key (IDs1||nonce)**

Where, ECHO is the message type, ID is the ID of the sensor node *s1*, nonce is the random number. If a node, say, *s2* receives this message, it sends echo reply with format:

***S2*□*s1*: ECHOBACK||Enew-key (IDs2||nonce).**



(Fig 4 : gives a pictorial view of how HELLO flood attacks can be initiated and the defense against the attack. We see that the message exchange won't be blocked by an adversary when bidirectional verification is applied.)

When node *s1* receives this message, it records node *s2* as its verified neighbor. If an attacker obtains the shared secrets after a node has received its new

encrypted key, it can not know the new pairwise key. Computing the pairwise key is more robust and secure in multiple tree protocol as we have described earlier, where we have shown that the probability of compromise of a secret is very low. However, if an attacker obtains the new key, it can initiate echo back many times by sending several echo messages. The attacker can generate false identities and can initiate Sybil attack, adding new nodes with false identities. To prevent such attacks, node should destroy its new key from memory after a certain time that is long enough to set up pairwise keys with all its neighbors. Again, during communication, it can calculate new key from the secrets they share.

7.1 Problem of Bidirectional verification

As we have stated that this defense against "HELLO flood" attack is to verify the bidirectionality of a link before taking meaningful action based on a message received over that link. But, this defense gets less effectiveness when an attacker has a highly sensitive receiver as well as a powerful transmitter. If an attacker compromises a node before the feedback message, it can block all its downstream nodes by simple dropping feed back messages. And thus, such an attacker can easily create a wormhole to every node within range of its transmitter/receiver. Since the links between these nodes and attacker are bidirectional, the above approach will unlikely be able to locally detect or prevent a "HELLO flood". We propose a different way of reliable exchange of messages among nodes and base stations. We show that when any particular node has different route to send data, this problem is solved.

8. MULTI-PATH MULTI-BASE STATION DATA FORWARDING

We describe how a sensor node can forward its sensed data to multiple routes i.e. multiple base stations in case where an attacker manages to compromise a sensor node. We assume that, there are a number of base stations in the network who have control over specific number of nodes and also, there are common means of communications among base stations. Each base station has all the secrets those are shared by all the sensor nodes according to the key assignment protocol described earlier. Given the shared secrets and the generated new key between two sensor nodes, the operation of setting up different routing paths is as follows:

Step 1: As each sensor node shares some common keys according to the secret distribution protocol (i.e. Multiple Tree Protocol), every node uses the echo-back scheme to identify its neighbor nodes and sets up pairwise new key with its verified neighbor nodes. Then it uses its new key to exchange messages among them.

Step 2: Each base station broadcasts its request (REQ) message to its neighbor nodes with the following format:

REQ||IDs||Ekey(IDB||HCN)

Here, REQ is the message type, IDs is the ID of the sending node *s*, IDB is the base station ID who generated this request message, Ekey is the key that is common between any node to which base station floods the message and HCN is the base but does not forward message to it, rather it sends message to its station's one-way hash chain number. Receiving node verifies that the REQ comes from the base station, then it forwards the REQ to its neighbor node, say, *y*, with the format:

REQ||IDy||Enew-key(IDB||HCN)

Step 3: When any ordinary node say, *y*, receives this REQ message, it checks the sender ID. If *s* is *y*'s verified neighbor, *y* decrypts and authenticates the sender with computed new key Enew-key. If the message sender is valid, it replaces the HCN with the new value and encrypts the REQ message with its Enew-key and broadcasts the newly encrypted message.

As we know, where four base stations with their communication range and sensor nodes with their communication range, if any message comes from a malicious node, the message won't be forwarded to that node, instead, the sensing node will take a different route to send data. Any base station, when receives the sensed data, it can cooperate with other base stations to interpret the sensed data as base station is powerful enough to communicate among themselves.

9. DISCUSSION

In simple defense, we have shown every node to authenticate identity with shared secret by the means of bidirectional verification. We have shown that if the protocol sends the messages in both directions over the link between the nodes, HELLO floods are prevented. We have shown a different approach when bidirectional verification does not prevent a compromised node. We present multi-path multi-base station routing. The flooding of REQ messages can securely establish direction without feedback to each base station. By setting up a new pairwise key from secret shared by nodes, multi-path routing improves intrusion tolerance. Specific one-way hash chain number (HCN) is addressed to defend against replay attack.

10. CONCLUSION

Our work described the defense against HELLO flood attack by introducing bidirectional verification and multi path routing using shared secret between sensor nodes. We have adopted a probabilistic key assignment among sensor nodes and during communication, each node can calculate a pairwise key using these common secrets and hence improving the network resilience against security threats. The key objective of our approach is to tolerate damage caused by an adversary who has captured deployed sensor nodes and is intent on injecting, modifying or blocking packets.

11. REFERENCES

[1] Routing Security in Sensor Network: HELLO Flood Attack and Defense Md. Abdul Hamid, Md. Mamun-Or-Rashid and Choong Seon Hong*
Department of Computer Engineering, Kyung Hee University Seocheon, Giheung, Yongin, Gyeongi 449-701 KOREA hamid, mamun.

[2] S. S. Kulkarni, M. G. Gouda, and A. Arora, "Secret instantiation in adhoc networks," *Special Issue of Elsevier Journal of Computer Communications on Dependable Wireless Sensor Networks.*

[3] Survey of Wireless sensor network security, author:Matthew N.Vella,Texas A&M University-corpus Christi,Computer Science Program: Mentor Dr.Ahmed Mahdy,Texas A&M University-corpus Christi,computer science faculty