A
Thesis
on

Replication Attack Detection using Independent Node Approach in WSN

Submitted in partial fulfillment of
the requirements for the award of the degree of

Master of Technology
in
Computer Science & Engineering

Submitted by

Akant Sharma
2015PCP5053

Under the Supervision of
Manoj Singh Gaur

Department of Computer Science & Engineering
Malaviya National Institute of Technology Jaipur

Declaration

I hereby declare that the work presented in this dissertation entitled, **"Replication Attack Detection using Independent Node Approach in WSN"** in partial fulfilment for the award of degree of "**Master of Technology**" in Computer Science and Engineering with specialization in Computer Science and Engineering and submitted to Department of Computer Science and Engineering, Malaviya National Institute of Technology is a record of my own investigation carried under the supervision of Dr. Manoj Singh Gaur, Professor, Department of Computer Science and Engineering, Malaviya National Institute of Technology.
I have not submitted the matter presented in this dissertation anywhere else for the award of any other degree.

Akant Sharma
2015PCP5053
MNIT Jaipur

Certificate

This is to certify that the project report entitled **"Replication Attack Detection using Independent Node Approach in WSN"** done by Akant Sharma, Enrollment No. 2015PCP5053 is an authentic work carried out by him at Malaviya National Institute of Technology,Jaipur under my guidance. The matter embodied in this project work has not been submitted earlier for the award of any degree to the best of my knowledge and belief.

Place: MNIT Jaipur
Date:

(Dr. MS Gaur)
Prof. MNIT Jaipur

Acknowledgement

This thesis I present here would not have been possible without the support of several person whom I would like to thanks. First of all, I would like to thank my thesis supervisor **Dr. Manoj Singh Gaur** for his patience and abundance of encouragements. I am grateful for his support and for the freedom he provides me to explore many opportunities in WSN and the guidance when I am overwhelmed.

My sincere thanks to Dr. Neeta Nain,D.P.C.G. Coordinator, (Professor, Department of Computer Science and Engineering, MNIT Jaipur), for her constant support to pursue this work. Furthermore, I owe thanks to my friends and staff of CSE Dept. for their time to listen, for their support, encouragement and valuable discussions during my thesis.

Finally, most importantly, I am indebted to my family for all the encouragement and moral support all this time. I am highly thankful to them due to whom I am here today. It has been a long journey up until finishing the thesis and this is just a beginning for more adventure ahead.

Akant Sharma
2015PCP5053
MNIT Jaipur

Abstract

Wireless Sensor Network(WSN) is a network of sensing devices/nodes whose purpose is to sense the data and report the result back to the network coordinator. These nodes are generally placed in hostile environment and thus left unattended. This give the possibilities that anyone can capture these nodes, extract the keys from them, make their replicas and put them back into the network. This will help the attacker to gain control over the whole network.
Since these devices are very small and cost-constraint, the hardware solution(Trusted Platform Manager - TPM) is not feasible on these devices. So, one has to totally depend on software solutions. Though there are several solutions proposed for detecting clones in the network, they assume very unrealistic assumptions or they diverge from the most important aspect of these network ie. low power consumption and low computation power. For example, Witness based approach are based on sending location claim but getting the location itself consumes too much power thus defeating the purpose of these network.
To deal with the above shortcomings, we propose a basic simple method of clone detection which is based on the assumption that two different nodes that are two different identities will be independent of each other. They don't know what other one is doing. And even if they communicate between themselves, they can be caught because verification is done each time they want to communicate with any node in the network including themseleves. Also even if they are successful in synchronizing, synchronization can break next time they want to communicate with any other node, thus they will again need to be synchronized which is not easy. The advantage of this algorithm are : very high rate of detection; low power consumption; works in extreme cases too like the one described just above in which clone nodes tries to communicate. Tests are done on real hardware and performance comparison is done with different types of algorithms.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Wireless Sensor Network

Wireless Sensor Network is basically a network of devices/nodes which are used for sensing data and return the result back to coordinator. A typical WSN network is shown in Fig: 1.1

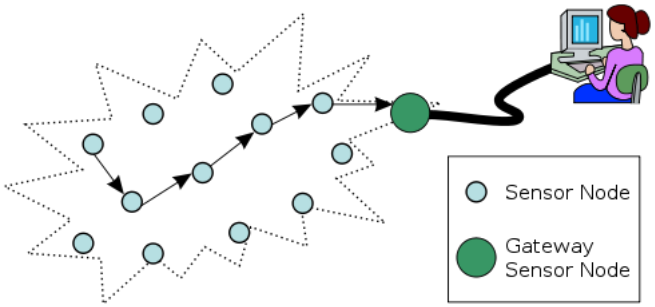

Figure 1.1: A typical Wireless Sensor Network[2]

In a typical Sensor Network, there are several nodes connected to each other and all are sending the data to a computer or collector via the gateway. Generally the node connected to the collector is referred as sink node.

Generally WSN is based on the standard of 802.15.4 Wireless PAN. This standard defines the physical layer and MAC layer. Rest of the above layers are left on the device vendors.

1.1.1 Application of WSN

- Sensing Devices can be installed in the forests to raise the alarm in case of fire.
- These devices are also placed at borders where any suspicious movements can be sensed and the border forces can be alarmed.

- These devices can also be placed in agriculture fields where the soil information can be used by scientists for research.
- These devices can also be used to determine pollution levels at various places.

1.1.2 Features of WSN

- WSN generally requires very few bytes to send. Thus it requires low bandwidth for communication.
- WSN devices are placed in hostile environment and are left unattended for several months. Thus these devices are battery-powered and the prime concern for them is that they should consume least amount of power.
- Since these devices just have to perform very simple task of sensing data they don't require any fancy hardware support or high storage power. Thus WSN devices have low-computational power and less storage.
- All these features are supported by IEEE 802.15.4 Wireless PAN standard[1]. The features of this standard are explained in next section.

1.1.3 IEEE 802.15.4 standard and Wireless Sensor Network

The Key features of WSN devices are:

- low-cost and ultra low-power consumption
- 16 channels, 250Kbps data rate in 2.4 GHz Band (Worldwide)
- low duty cycle ie. maximum time the devices are in sleep mode/doze mode to consume power
- Two types of devices are supported viz. FFDs(Full Functional Devices) and RFDs(Reduced Functional Devices). RFD's can only talk to FFDs and can't participate in Routing.
- Two types of topologies are used in this standard viz. point to point and peer to peer. RFDs are only involved in point to point while FFDs can be a part of any topology
- For Network setup a device/node first starts scanning the channels and selects the quietest channel on which network will be setup. This device is known as PAN coordinator as this setup the required network. Any other node that wants to join the network will ping all the channels and will associated to one of the channel to which PAN is connected.
- For synchronization/data transfer two types of mode of communication are defined in standard.

- **Beacon-based:** In this communication, PAN continuously transmits the beacon at regular interval and all the nodes in the network will align/synchronize to this beacon before begin the communication within this interval. All the data transfer will be done as per the beacon-based communication.
- **Non beacon-based:** In this communication, all the devices are continuously in RX mode. They don't require specific slots/time interval before they want to send/receive the data.

- This standard uses CSMA/CA mechanism as a communication medium

1.2 Attacks on WSN

Since the WSN devices are placed in hostile environment, they are can be captured and attacker can make copies of it with the help of reverse-engineering and placed the nodes in the network back. There are several types of attacks that are possible in WSN[7]. These are defined as follows:

1. **Active Attacks:** These are the attacks whose consequences/ill-effects starts showing as soon as they are launched. These are enumerated as follows:

   (a) **Routing Attacks:** These are the attacks in which the attack is being done on the data being routed. These are defined as:

      i. **Sinkhole Attack:** In this attack, all the data is being re-directed to a particular node with the help of a malicious node. Thus that particular node becomes a center of attraction for all the data in the network and behaves like a sink node.

      ii. **Selective Forwarding:** In this attack, the malicious node only forwards the data for a particular destination and drops all the other data. If there is no end-to-end communication between the sender and receiver or the data is of UDP packets then this attack may go unnoticed and can be very harmful. Like in a hot chamber a machine automatically shuts down if the temperature inside the room exceeds a threshold. The sensing node is constantly sending data to machine as non-acknowledge service. In this case selective forwarding can be very harmful.

      iii. **Wormhole Attack:** In this attack, attacker stores all the data packets and use them at later place/time in order to replay the same situation.

      iv. **Hello Flood Attack:** In this attack, false hello packets are generated into the network to make congestion or keeps the node busy to drain their power in processing.

   (b) **Sybil Attack:** In this attack, a single node possesses multiple identities to fool the other node in the network that there are multiple nodes/devices in the network. The consequences of this attack is very dire as it can affect

multi-path routing, voting, distributed storage etc. Thus a single node will gain majority of share of network. Example is fire alarm system in which if more than half of nodes report that the temperature exceeds the threshold then it will be treated as fire. If a node possesses multiple identities then it can affect the voting.

(c) **Node Replication:** In this attack, an attacker can get the secret credentials of the node, make their copies and place it back in the network. This attack can be very harmful as it allows the attacker to generate ambiguous information like two different clone nodes place at two different places, one is returning temperature value as $40°C$ while other is returning $23°C$.

(d) **Message Corruption:** In this attack, the attacker modifies the message and maligns the integrity of the network.

(e) **Denial of Service(DOS):** In this attack, the attacker tries to flood the network with unnecessary requests/ fake messages thus creating a load on the network.

(f) **Physical Attack:** In this type of attack, the attacker physically tampers the device in order to malfunction/destroy the device.

(g) **Node Subversion:** If the node is captured by an attacker, attacker can get secret credentials from the node which can then be responsible for compromising the whole network.

(h) **False Node:** If an attacker adds an extra node in the network which is used to inject false data, then this will be termed as false node.

2. **Passive Attacks:** These are the attacks which goes unnoticed until something bigger has happened or network administrator examines the network. These are the attacks which are hard to find and generally goes unnoticed. These are enumerated as follows:

   (a) **Traffic Analysis:** One is continuously studying the traffic pattern, this can give attacker sufficient information to attack that section/nodes that are under heavy traffic.

   (b) **Camouflages Advisories:** In this attack, Attacker can divert the whole traffic to a malicious node which then with some modification sends it further thus this node hides its identity.

   (c) **Monitoring and Eaves Dropping:** One can listen the communication between the nodes and try to find out the important information(like secret credentials) which can be use for subsequent attacks.

1.3 Motivation

Generally devices used in WSNs are placed in hostile environment like forests, borders, mountains, etc. These can be captured and one with the help of reverse engineering can

take the keys from them and make the copies of these nodes and place them back in the network. this will help the attacker to monitor or control the whole network.
Since WSN devices are constraint with low-battery-power, low computation power, cost-constraint, so these devices don't include some features like TPM(Trusted Hardware Module) that can help them in resisting these types of attacks. So, Due to constraints on these network or devices, hardware solutions are not effective.
There are several software solutions but those assume several unrealistic assumptions or they don't consider the special fetaures of these cost-effective network like low computation power and low-battery powered.

1.4 Objectives

The main objective behind working on this work is:-

- To provide an algorithm that is cost-effective and in-line with the features of these sensor devices like it uses very low computation power and low battery power.
- To provide high detection rate.

1.5 Thesis Organisation

The content of the thesis proceeds as follows:
Chapter 1: This section includes the introduction to wireless sensor network, features of wireless sensor network, motivation, Objective and thesis organization enclosed at the end.
Chapter 2: This section contains exhaustive literature survey about the previous algorithms for clone detection.
Chapter 3: This section explains the proposed work which includes attack detection approach, security control approach, network model, assumption and proposed algorithm.
Chapter 4: This section present the devices used, experiment parameters, results and comparisons with the other approaches.
Chapter 5: This section consists of conclusion and future work.
Appendix A/B: This consists of the actual data taken from the experiment.

Chapter 2

Replication Attack and Related Work

2.1 Definition

Replication attack is basically an attack in which an attacker is successful in getting the secret credentials from a captured node and with the help of these credentials, he makes copies of the node and place them back in the network. In simple words, he has taken the identity of a node and by using this identity he is able to successfully barged in the secure network. Once he is successful in gaining access to the network, he can launch variety of attacks discussed in next section.

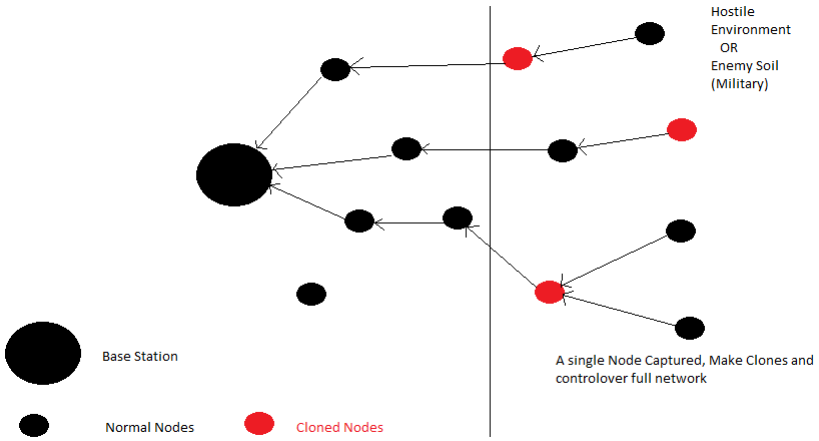

Figure 2.1: Replication Attack in WSN

2.2 Consequences of Replication Attack

Since Attacker was able to enter in the network using the identity of a legitimate node, he can successfully launch variety of attacks. These are listed as :

- **Ambiguous information:** Cloned nodes are placed at different locations and they are reporting the data to the central coordinator, it may be possible that the data will vary, so the coordinator will receive different data from same node ID and it will be impossible for him to differentiate which data is correct. Example is: Two cloned nodes with same node ID returning two different temperature values with large difference. So, it will be ambiguous information to coordinator whether to take a decision or not based on these values.
- **Sinkhole Attacks:** An attacker can placed cloned nodes at the places from where the large amount of traffic is forwarded. In such case all the traffic will be converged to one node ID and it can change the data before forwarding or forward it to some malicious destination.
- **Selective Forwarding:** There may be cases where the trust levels of some nodes may be higher ie. they are preferred over other nodes. In such cases, an attacker can capture such nodes make clones, place them in the network. Since these nodes will be preferred over other nodes, all the data may be routed through these nodes. In this case these nodes can do selective forwarding of data. Also it will help in masking other replicated nodes for which it will not send the verification claim.
- **False Data:** Since these cloned nodes have taken the identity of trusted nodes, these can be used to inject false data in the network. Since these have verified identity, no one will believe that they are cloned/malicious and are injecting false data.
- **Wormhole Attacks and Hello flood Attack:** Since these cloned nodes have verified identities, these nodes can store the data packets from the original nodes and can replay them later. Also they can continuously request Base Station or other nodes which will keep them busy and drain their already under constraint battery power.

2.3 Algorithm Design Challenges

The detection challenges are simply the special features of WSN network[8]. These are defined as:

- **Low Battery-Power Consumption:** Since the main goal of WSN is to use very little power as possible, the algorithm should consume as little power as possible.
- **Low computation-power:** The algorithm should be simple and not complex as these devices are not expensive and are equipped with very little processing power. Also, complex algorithms takes too much battery-power.

- **Low storage space:** These devices are also constraint in storage, so one must also takes into account the space complexity
- **Communication cost:** The message exchanges between the nodes during the execution of algorithm must not be very large in terms of both the number and sizes. As, during the communication time the node is in RX mode and it will consume power too if there are exchange of very large number of messages during the execution of algorithm.

2.4 Detection Approaches

In this section we will take a look at variety of detection algorithms. These detection algorithms are classified broadly in two classes viz. for Mobile network in which nodes are constantly moving and for Static network in which the position of nodes are fixed. These detection algorithm are further classified on the basis of centralized or distributed approach. There are variety of algorithms for detection like Neighbor list, Voting mechanism, Promiscuous Mode, SET, EDWA, Monitor based, Sequential Probability Ratio Test(SPRT), XED, EDD, RSSI, witness based(RM,DM,RED,LSM), challenge-response,etc. We will gonna look only at most important ones.

2.4.1 Mobile WSN

WSN network can be mobile in which nodes are continuously moving from one place to another. Detection in case of mobile network is hard as nodes are changing their position in the network very fast. There are several algorithms that are described as follows[10][9][5]

- **Centralized Approach:**

2.4.1.1 Speed based

In this algorithm, each node is loosely time-synchronized. each node sends it's location with time to Base Station, if there is considerable difference in between two successive location claim with the speed limit is set to v; then it is known to be cloned node and the revocation procedure starts.

2.4.1.2 Voltage level

In this algorithm each node broadcast its voltage level for claim. Since a malicious node will node to be doing some extra work like, so it will be using more power than the original one, so it can be used for detection of clones on the basis of behavioral difference between clone node and the original node.

- **Distributed Approach:**

2.4.1.3 eXtremely Efficient Detection - XED

In this approach, whenever the two nodes first met with each other, they exchange a random nounce value. when they encounter again nodes will ask for the re-confirm the nounce value. If it's different then it means the node is cloned and the new node is not the previous one which encounters. this approach depends on the probability of meeting of two nodes again. If two nodes don't encounter again then the clone will not be detected.

2.4.1.4 Efficient Distributed Detection - EDD

In this approach we put a fixed bound that how many maximum times a node i can encounter node j in its whole time. If it exceeds this value then there are high chances that the node is cloned

2.4.1.5 Neighbor Based Detection Scheme - NBDS

In the neighbor based detection, node keeps track of its neighbor. once this node is moved in the network. It will exchange its neighbor list with new neighbor. The new neighbors which are not in the previous list can send the claim to one of the previous neighbor of the claiming node. The previous node first checks whether the claiming node was its neighbor previously and right now it has moved from the network or not?
If the claiming node wasn't a neighbor then it means the claiming node is giving the false information thus it can be cloned. If the claiming node is still in the neighborhood means that there is single node at two different places which is replication attack.

2.4.2 Static WSN

Many a times, WSN devices are stationary, replication attack detection in case of static WSN is easy as compared to mobile WSN. So, there are several algorithms defined in the literature for static WSN that are very much different from those of mobile WSN algorithms[4][11][3].

- **Centralized Approach:**

2.4.2.1 SET based

In this approach we divide the network in several disjoint sets. The list of nodes in the particular set is sent to Base Station. Base Station then calculates the intersection of the sets. If it founds a node at two different location then it will be assume that node is cloned.

2.4.2.2 Location based

In this approach the location claim of each node is send to Base Station, if the Base Station founds a node at two different locations, then it will assume that the node is cloned.

- **Distributed Approach:**

2.4.2.3 Deterministic Multicast - DM

In this approach, witness for a given location claim are identified as a function of the node ID contained in the claim. Thus all claims with the same node ID are forwarded to same witnesses, therefore if two of the location claims are different for same node ID, then this algorithm will be able to detect. But in this approach the attacker can find the witness for a particular node ID thus he can further clone the witness to avoid the detection.

2.4.2.4 Randomized Multicast - RM

In the Randomized Multicast approach, the two claims with the same ID will have different set of witnesses. Thus the attack will be detected only if the two set of witnesses are not disjoint. If the cardinality is $\sqrt{n}$,then the birthday paradox guarantees a good rate of detection.

2.4.2.5 Line Selected Multicast - LSM

In the Line Selected Multicast approach, the neighbor node computes g random locations, and the location claim is forwarded to these random location with probability p. Each node in the path to the destination location will also act as witness. The clone will be detected only if the two different paths contains a common node. For this, claim is sent to $\sqrt{n}$ witnesses, so that clone can be detected with very high probability.
This leads to a high communication cost. Also, since the witnesses may be common for different IDs, the storage cost is also high as nodes have to store the location claim of many nodes.

2.4.2.6 Randomized Efficient and Distributed Protocol - RED

In this approach the location to which the claim is to be sent is chosen pseudo-randomly. The Base station will continuously transmit a random-number seed. Based on this random-seed and node ID, the location claim is forwarded to witness. Since the witnesses are chosen pseudo-randomly, the location claim of two nodes with same ID will go to same witness, thus the detection rate will be high.

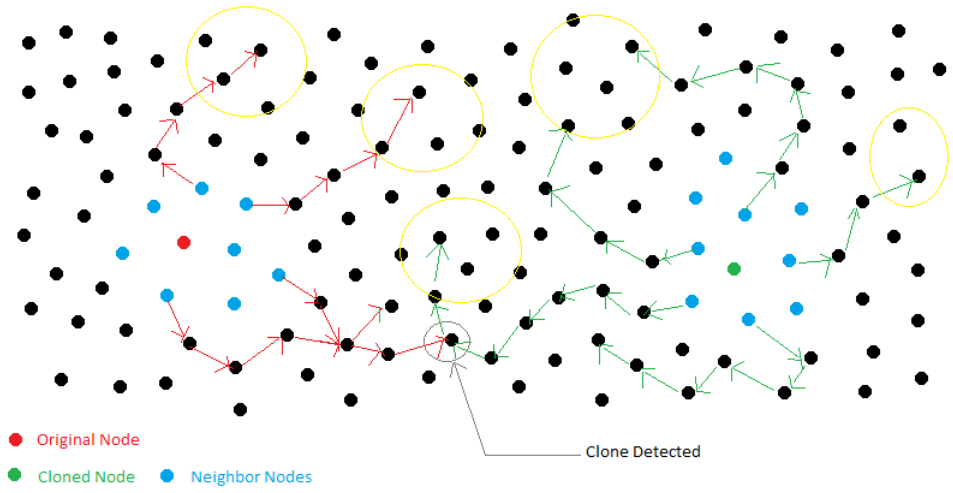

Figure 2.2: Witness Based Approach - LSM

Communication cost will still be high as claims are forwarded to many witnesses but the storage cost will be $O(1)$ as there is fixed number of nodes which will map to these witnesses.

2.4.2.7 Random Walk - RAWL

In this approach, each neighbor with probability p will send the claim to random locations, once the claim will reach the random location, a random walk of k hops will be done. In this walk, every node traversed will be a witness and store the claim. Clone Attack will be detected only if walk generated by two claims with same ID contains a common node.
It is different from LSM in that it enhances the detection rate because LSM takes the path only in the proximity of clone node while Random Walk takes the path far away from the clone node.

2.4.2.8 Localized Multicast

In this approach, we divide the regions into the cells and every location claim is then send to a particular cell. Once a claim reaches a cell, it is flooded in that particular cell. Here like RED approach cells are chosen pseudo-randomly.

2.5 Comparison of Algorithms

Wafa et al.[4] has shown that the above location based algorithms suffered with smart node positioning attack. If the attacker is smart he can place the nodes at the places where the detection rate will be low. Thus these algorithm suffers from node positioning and topology. They had experimented with several topologies with different number of nodes for the above algorithms LSM, RED, Localized Multicast and Random Walk(RAWL).

2.6 Performance Overheads of the algorithms

Wafa et al.[4] also shows the cost of the above distributed algorithms in terms of communication and storage used. Consider n is the number of nodes in the network.

Algorithm	communication	storage
LSM	$O(\sqrt{n})$	$O(\sqrt{n})$
RED	$O(\sqrt{n})$	$O(1)$
SDC	$O(\sqrt{n})$	$O(1)$
RAWL	$O(\sqrt{n}\log n)$	$O(\sqrt{n}\log n)$

Table 2.1: Performance overheads of witness based approach algorithms

Chapter 3

Proposed Work

3.1 Limitations of previous detection approaches

As can be seen from previous section, In case of static network clone detection(as we are going to deal with only static network) these are the following limitations:

- In case of centralized approach, the central server can become bottleneck, as all the verification data is send to central server.
- In case of SET based approach it is hard to divide the network into disjoint sets.
- Also performing operations on these sets to find the clones is power consuming which is not a desirable.
- In case of distributed approach, all are witness based approach in which location claims is sent to witnesses.
- The main drawback of these witness based approaches is that they require GPS or similar technology to tell the location. This operation itself consume power which is against the essential features of these network i.e. low power consumption.
- Also, in these algorithms the detection rate is dependent on the precision of location, if the location is very precise then less number of node belongs to a particular location and more chances that same node will be the witness for two different location claims with same ID.
- wafa et al.[4] also shows that communication cost and storage cost of these algorithms is very high if we want to detect the attack with very high probability.

3.2 Attack Detection Approaches

3.2.1 Need of another approach

From the previous section, we can say that the existing algorithm are not good when it comes to satisfying the desirable features of WSN i.e. one with low power consumption,

low storage, low processing power, etc. So, we need another algorithm that can atleast fulfill the features of WSN network and especially the low power consumption feature.
EvencCommunication cost and processing power also uses power. So, if there is high communication cost i.e. high numbers of packets are being sent and received then it will also consume power as the device in RX(Receive Mode) or TX(Transmit Mode) consumes power.
Also if algorithm requires more processing power i.e. algorithm computation complexity is high device will use more power for processing.
So, our goal is to devise an algorithm which is simple, don't require high computation or communication cost.

3.2.2 Idea!

The Idea for our new approach i.e. Independent Node Based Approach is based on the fact that two different devices or nodes are two different identities thus they are independent. Independent in terms of processing, battery-power, storage, and every thing else.
Since they are independent they don't know what is the status of the other nodes in the network i.e. their battery levels, what operation they are currently doing etc. until or unless they communicate with each other about their state. This is the basis of our algorithm, if we assign a sequence number for all the communications a node made with other nodes, then we can use it as a basis for clone detection, because the other nodes don't know how many communication a particular node has done till now.
Even if they synchronize or knows about the other sequence number, then we have the first hop node in the path of communication that is there to break the synchronization. Also, the claim will be made by the nodes involved in the communication. Any node in the communication path can send the verification of source to the witness. And witness have to check that the sequence number in the claim must not be less than the last seen sequence number for that source. If found less sequence number than the last seen, alert the network about the clone.

3.3 Network and Adversary Models

In next subsections we will present network and adversary models. Our algorithm is distributed algorithm for clone node detection, so we will consider the traditional models for such detection algorithms.

3.3.1 Network Model

Network is consists of many sensor nodes/devices that due to hardware-constraints are not tampered-resistant. These devices have a limited battery power and low computation power along with less storage and low transmission range. Each node is equipped with a 64-bit node ID but short addresses can also be used and a pair of ID based public and private keys. Like the traditional algorithms which uses location based detection

methods and thus equipped with a GPS system, our devices needn't to be equipped with a GPS sensor/system. One can also assume that the devices are loosely synchronized. All the communication by nodes will follow a unique structure as described in next few sections. Each node will have an sequence number associated with itself. It will send this sequence number along with any communication data, he wants to make with another node. It will also attach a hash of all the data including its sequence number and ID. When a first hop node receives the packet, it will first check whether the packet was sent from the same node that is claiming it has send the packet or not by checking its hash. If it founds that everything is correct, it will forward the data with probability p. If it forward the data it will send the ACK back to the source which will increase its sequence number for next communication otherwise not. The way in which the witness is chosen and how the verification claim is sent to witness will be described in algorithm. The verification claim for every communication made is to make sure that the attack won't go undetected. Also we are assuming that there is not very large exchange of communication messages apart from sending the sense data to the central server. In the algorithm we will explain, n is the number of nodes in the network, p is the probability of forwarding the data, q is the probability of forwarding the witness claim. Rest of the notations will be explained when necessary.

3.3.2 Adversary Model

Since sensor nodes are placed in hostile unattended environment, they are subject to captured and reverse-engineered. The attacker can be successful in extracting the keys from them, read data and based on this information he can make copies of the nodes and placed them in the network. We will treat the original node also the clone once it is compromised/cloned. Also, the attacker will be unable to produce the identity based key-pairs that can deceive honest nodes. Also, total number of nodes cloned is much smaller than the total number of nodes in the network.

3.4 Assumptions

Following are the assumptions that we are assuming:

- There will be no difference between the original node and its clone, once the clone is also placed in the network.
- The original node whose clone is placed in the network will now be treated as malicious and will be cut-off from the network once the attack will be detected along with its clones even if it is functioning correctly.
- The Node ID Pair of public-key and private-key will not reveal any other node public-key and private-key information i.e. it is impossible for the attacker to tell the node ID and their public-key private-key pair without capturing it.

- Total number of clone nodes is much smaller than the total number of nodes in the network. Infact, there is no detection approach possible if a large part of the network is compromised.
- Attacker can placed the clone nodes anywhere he wants.
- The hashing algorithm along with the encryption algorithm used will depend on one's taste. Also these two things are computationally expensive as well as power consuming. We will ignore these two facts for now!
- We will assume that there is no drop of the packet in the network. i.e. no congestion. If a Packet is forwarded by the first hop it will reach its destination. Same thing for verification request packets.

3.5 Proposed Algorithm

Our Algorithm will be based on sequence number, In this algorithm, if any node wants to communicate with any other node, it will append the header described in **??** to the data it wants to communicate. This new packet will then be handed over to routing layer which will then select the next hop for the destination and will send the packet to it.
On receiving the packet, the node if it is the first hop i.e. neighbor of the source, it will decide with probability p whether to forward the packet to destination or not. If it decides to forward it to destination, it will send an ACK back to source node that its communication to destination has been successful and it can increase its sequence number for next communication. If it decide not to send the packet to destination, it will silently discard the packet and will not send an ACK back to source.
Source node after sending the packet will wait for sometimes for the ACK. If it gets the ACK in stipulated time, it will increase its sequence number otherwise not.
Once decided that the packet will be forwarded to the destination, any node from the first hop to destination can send the verification for the source with probability q that generated the packet only if it is not already sent for verification by some other node. Here probability q is taken so that verification is not sent for every communication to avoid congestion in the network.
In the mean time, Base Station will generate a random-seed at regular interval which will be used for selecting the witness for a particular ID.
Here witness selection is based on pseudo-random function based on random-seed along with node ID.

3.5.1 Pseudo code

Each node plays 2 roles: i) Sender's Role ii) Receiver's Role. The role of the sender's is to forward the data along with sending the verification claim to witness. The role of the receiver is to receive the data for forwarding or receiving the verification claim. Also Base station will have work of broadcasting the random-seed value which will be used to update the witness to which verifcation claim has to be sent.
We assume that node are objects of same class that has fields ID, SEQ_NUMBER, K, $neighbors$, and $STORE$ and methods $Broadcast\_Random\_Number$, $Receive$, $Send\_Data$, $Send\_ACK$, $Send\_Claim$, $detect\_clone$, $put\_in\_store(C)$. Thus when executing these methods, the invocation object *this* denotes the node that is executing and $this.ID, this.K$, $this.SEQ\_NUMBER$, *etc.* The method *Receive* will have a role of receiver, all the other methods will have a role of sender's. The method $detect\_clone(C)$ checks C against the local store and will return true iff the store contains another claim with larger sequence number.
this $\rightarrow$ n:M means current node sends a message M to another node n. Also $=$ means assisgnment and $\leftarrow$ for pattern matching.
Also we will use some auxilary functions like $random(g), pseudo\_random(rand, ID, g)$, $is\_random\_seed\_msg$, $is\_claim(M), bad\_signature(C, S)$,$is\_data(M), is\_ack(M)$

```
Procedure Broadcast_Random_Number();

rnd = random()
C = <this.ID, is_random_seed_msg, rnd>
SC = <C,this.K^private(C)>
this→0xFFFF:<this.ID,0xFFFF,SC>
end procedure

Procedure Send_Data();

C = <this.ID,is_data,this.Destination,this.seq_number,this.verification,this.data>
SC = <C,this.K^private(C)>
L = nexthop(this.destination)
this→L:<this.ID,L,SC>
end procedure

Procedure Send_ACK(Source);

this.destination = Source
C = <this.ID,is_ack,this.Destination>
SC = <C,this.K^private(C)>
L = nexthop(this.destination)
this→L:<this.ID,L,SC>
```

```
end procedure

Procedure Send_Claim(Witness,verify_source,verify_source_seq_number);

  this.destination = Witness
  this.verify source = verify_source
  this.verify_source_seq_number = verify_source_seq_number
C = <this.ID,is_claim,this.destination,this.verify_source,this.verify_source_seq_number>
SC = <C,this.K^private(C)>
L = nexthop(this.destination)
this→L:<this.ID,L,SC>
end procedure


Procedure Receive(M);

<ID,-,SC>←M
<C,S>←SC
if bad_signature(C,S) then
 discard M
 return


if is_random_seed_msg(M)
    <ID,-,rnd>←C
this.random_number = rnd
this→0xFFFF:<this.ID,0xFFFF,SC>

if is_ack
this.seq_number++

if is_data
<ID,-,destination,source_seq_number,verification,verification_ID,data>←C

if ID == prev_Addr
do with probability $p$
   send_ACK(ID);
   do with probability $q$
   verification = true
   Send_Claim(witness,ID,source_seq_number);
   L = nexthop(destination)
   this→L:<this.ID,L,SC>
else
```

```
  //forward the packet and send to verification if not already sent
  L = nexthop(destination)
  this→L:<this.ID,L,SC>

  if verification = false
  do with probability $q$
   verification = true
   Send_Claim(witness,ID,source_seq_number);

if is_claim
<ID,-,destination,,verification_ID,source_seq_number>←C
  if this.nodeID == destination
  if verfication_ID.last_seen_sequence_number > source_seq_number
    clone detected
    trigger revocation procedure for ID

  else
     L = nexthop(this.destination)
       this→L:<this.ID,L,SC>

end procedure
```

3.5.2 Flowchart

The Flowchart3.1 here shows the source node has send the packet. After sending the packet it will wait for ACK packet. Once the first hop receive the packet, with probability p it will forward the packet. If it decide to forward the packet, it will also send an ACK back to source node.
After getting ACK back, the source node will increase its sequence number, otherwise it will use the same sequence number.

Once the first hop decides to forward the packet, from there on till the destination, any node can send the source for verification with probability q3.2, provided that some other node hasn't send the source for verification already i.e. the source which generated the packet can only be send for the verification at-most once while the packet is routed to the destination!

3.6 Security Control Approach

Each unicast communication is believed to be a part of verification process, be it data communication or control communication. But it should be unicast because this communication can be dropped by the neighbor nodes to break the sync between the original

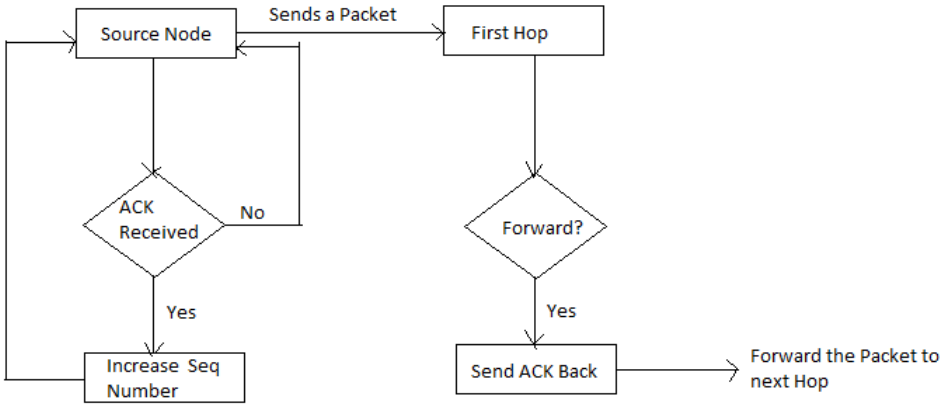

Figure 3.1: Algorithm Flow Chart - I (at Source Node)

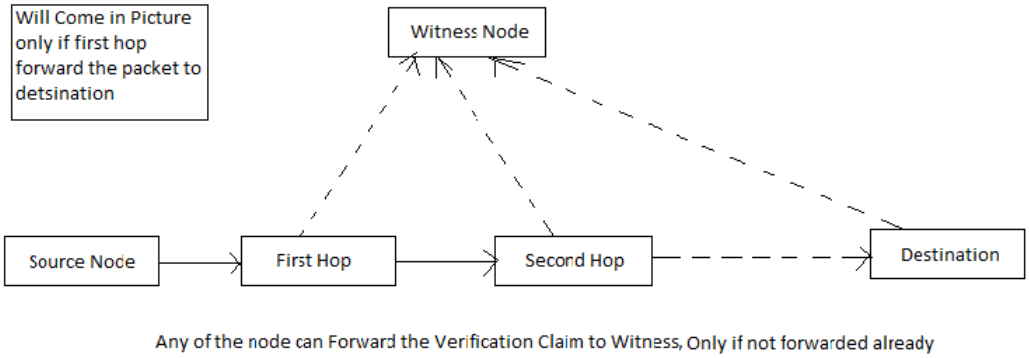

Figure 3.2: Algorithm Flow Chart - II (Verification)

nnode and clone node. In case of broadcast communication, many neighbors will receive the packets, we can't apply the same approach in that case.
Every communication will be prepend by a header consisting of node ID, destination ID and sequence number along with verification sent byte.
Since there are malicious nodes in the network, they can generate fake packets of any nodes. To deal with this situation, we will append some digital signatures which will allows a node to verify that the packet was actually generated by the node which is claiming and similarly packet was sent for verification which is claiming it has sent the verification. Packet Structure is shown in 3.3

3.7 Algorithmic Issues

There are various issues in the algorithm. These are described in detail along with their possible solutions:

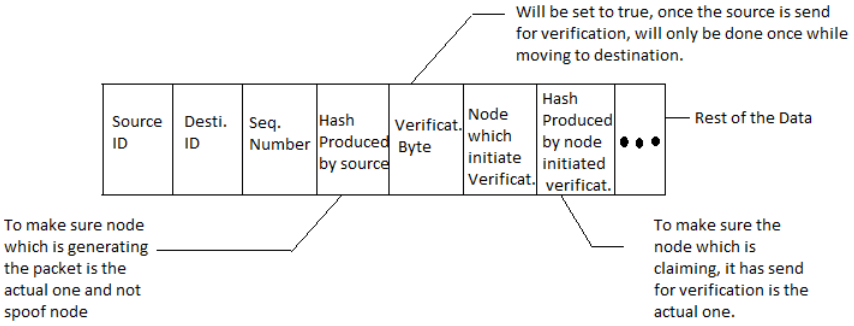

Figure 3.3: Communication Structure

- It may be possible that clone node may behave maliciously i.e. they increase the sequence number even without getting ACK. This case is considered as Case 2. The detection for this at the witness will be change only in the way that now the witness is expecting the verification claim for every sequence number. If it don't receives verification for any sequence number, this will be treated as clone detection. Theoretical as well as practical results are also shown for this approach.

```
Procedure Receive(M);
...
...
...

if is_claim
<ID,-,destination,,verification_ID,source_seq_number>←C
  if this.nodeID == destination
  if verfication_ID.last_seen_sequence_number + 1 != source_seq_number
    clone detected
    trigger revocation procedure for ID

  else
     L = nexthop(this.destination)
       this→L:<this.ID,L,SC>
...
...
...
```

- Cloned Node may be shielded by malicious neighbors[6]. They don't drop the packets. In this case, packet can be dropped by any node in the route to destination and not only the first hop node to break synchronization. The only difference will be that Probability of successful transmission will change $If, P(fwd) = 0.3$ and

destination is 10 hops away, then the probability of successful transmission will be given by $P(S) = 0.3^{10} = 0.000005905$. Now all the calculation shown in results section will be done on the basis of this probability.

- Malicious Neighbors always change the verification send byte to TRUE and don't even send the verification. It can be detected with the help of statistical analysis that verification for a particular source is always done by same node. In this case, neighbor can be put in suspicion.
- Malicious node falsely reports that the legitimate node is cloned. This problem can be alleviated as the node will not be decided clone until or unless it is verified by k different witnesses.

Chapter 4

Performance Evaluation Criteria

4.1 Performance Metrics

When one gives a new algorithm or approach, it is generally ask that how much better it is than the existing approaches. There are several parameters that are considered in case of distributed algorithms[4]. Some of them are given below on the basis of which we will compare our algorithm with the existing ones:

- Total number of messages exchanged.
- Power Consumption and Computation Cost.
- Storage Cost

4.2 Theoretical Estimations

Since our algorithm is based on the number of communication messages send by a particular node to other nodes. So, we have tried to find number of message to be send to guarantee that attack will be detected with probability p. We have consider both the cases i.e. Malicious node increase sequence number only if get ACK and the other in which Malicious node increases sequence number without getting ACK. The initial value of $P(fwd) = 0.3$ taken from the value which we set in practical attack detection.

4.2.1 Case 1: Malicious Node behaving benign

Let probability of forwarding data P(fwd) = p

Probability that both successfully transmitted P(S1)
= $p * p = p^2$

Similarly, Probability that both fails in transmission

P(S2) = $(1-p)^2$

Probability that one success and other one fails P(S3)
= $p * (1-p)$

Assume both the nodes are synchronized initially,
attack will be detected only if one is successful
and other one fails.

So,
Probability of Attack detection
$P(AD) = P(S3) + P(S3)$
Probability that Attack goes undetected
$P(AUD) = P(S1) + P(S2)$

Consider a snapshot:
S1: SSFSFS....S
S2: SSFSFS....F

So, probability of detection in n attempts
will be given as P(Detect)
= $P(AD) + P(AUD) * P(AD) + P(AUD)^2 * P(AD)$
$... + P(AUD)^{(n-1)} * P(AD)$;

P(Detect) = $P(AD) * \frac{(1-P(AUD)^n)}{(1-P(AUD))}$;
but, P(AUD) = 1-P(AD);
so, P(Detect) = $1 - P(AUD)^n$;
or, n = $\log_{P(AUD)}(1 - P(Detect))$;

So, for P(fwd) = 0.3; P(S1)=0.09;
P(S2) = 0.7*0.7 = 0.49; P(S3) = 0.3*0.7 = 0.21;

P(AD) = 0.21 + 0.21 = 0.42;
P(AUD) = 0.09 + 0.49 = 0.58;

If we want P(Detect) = 99% then,
n = 8.4540 $\approx$ 9 attempts to detect Clone

This value will be minimum when P(fwd) = 0.5
giving n = 6.6438 $\approx$ 7

4.2.2 Case 2: Malicious Node behaving malicious

Attack will be detected only if both fails as it will lead to skip in the sequence number at the witness node

So,
Probability of Attack detection
$P(AD) = P(S2)$
Probability that Attack goes undetected
$P(AUD) = P(S1) + P(S3) + P(S3)$

Consider a snapshot:
S1: SSFSSS....F
S2: SFSSFS....F

Rest of the formula will remain same,

or, n = $\log_{P(AUD)}(1 - P(Detect))$;

So, for P(fwd) = 0.3; P(S1)=0.09;
P(S2) = 0.7*0.7 = 0.49; P(S3) = 0.3*0.7 = 0.21;

P(AD) = 0.49;
P(AUD) = 0.09 + 0.21 + 0.21 = 0.51;

If we want P(Detect) = 99% then,
n = 6.8392 $\approx$ 7 attempts to detect Clone

This value will decrease if we lower the forward rate and will increase if we increase the forward rate

This can be set according to the chances of clone in the network, if there are high chances of clone in the network forwarding probability can be kept low.

4.3 Practical Results

4.3.1 Parameters

Following are the parameters of the actual implementation of the attack:

- Sensenuts Devices from Eigen Technologies, New Delhi has been taken for attack implementaion and detection.
- Total 10 nodes are taken from which 1 node is PAN and the rest are coordinators.
- Out of 9 nodes, 1 node(ID BD54) was clone of node ID BFD0.
- Our algorithm is built on top of DYMO(Dynamic Manet On Demand) Routing which we build from scratch on sensenuts devices.
- The transmission power was set to -20dBm so that multi-hopping can work.

4.3.2 Results

Detection was done for both the cases. Results are calculated for only case 2

- Only data from 8 nodes was received at PAN instead of 9 as one node was clone in the network.
- The expected value for (case: 2) $\approx$ 2 packets which is very less than $\approx$ 7 packets calculated using theoretical approach. This is because of the low transmit power = -20 dBm to implement multi-hopping which causes a high packet drop ratio.

In our algorithm, the communication cost is $O(1)$ as the verification claim is sent to only one witness not a bunch of witness as in other algorithm. Also, each node is uniquely mapped to a witness for a fixed interval, so the witness node is just to store the claim of only particular witness to which it is mapped for a fixed amount of interval.

Algorithm	communication	storage
LSM	$O(\sqrt{n})$	$O(\sqrt{n})$
RED	$O(\sqrt{n})$	$O(1)$
SDC	$O(\sqrt{n})$	$O(1)$
RAWL	$O(\sqrt{n}\log n)$	$O(\sqrt{n}\log n)$
Our Algo.	$O(1)$	$O(1)$

Table 4.1: Comparisions of our own algorithm with location based algorithms

Chapter 5

Conclusions

WSN is now-a-days becoming a very popular network due to its very special features. The devices of these networks are cheap, small and very easy to maintain. Thus these devices are widely used in various applications randing from military to agriculture, forests to streets, homes, etc. But as every coin has two-sides, these network devices are very easy to capture, and once captured they can be reverse-engineered, secret credentials can be taken out and malicious nodes can be made using those secret credentials. These can be placed into the network which will help the attacker in controlling the whole of the network. One of the most dangerous attack on these networks are node replication attack. Majority of the algorithm are based on the fact that no device can be at two places at a particular time, So in case of static WSN, they form this as a basis of attack detection approach. There are several algorithms but the core of all of them is same. Each node will broadcast its location claim, neighbors with probability p will forward this claim to witnesses. If a witness founds two different location for same node ID, it will treat it as replication attack and thus will raise the alarm.
The above approach is good but it is against the very special feature of these network i.e. low power consumption. These Location based algorithms uses GPS to determine their location which consumes battery-power. Our algorithm Independent Node based approach is an attempt to alleviate this problem. The idea is simple that two nodes are independent and don't know what the other one is doing until or unless they communicate with each other. In our algorithm, every communication between two nodes is timestamped with a sequence number which increase automatically if the communication is successful otherwise not. The neighbor of the source will try to break the synchronization even if the two cloned nodes are synchronized and any node in the route to destination can initiate a verification claim for that source only if the verification claim is not already sent. We have consider detection for both the cases, in which malicious node is behaving correctly and where the malicious node is behaving maliciously i.e. increasing its sequence number even if the ack is not received.
Our algorithm is better than the previous approaches in terms of battery-power consumption as our devices won't need to use GPS, number of messages exchanged and number of claims needs to be stored.

References

[1] IEEE Standard for Information technology Telecommunications and information exchange between systems Local and metropolitan area networks Specific requirementspart 15.4: Wireless medium access control (mac) and physical layer (phy) specifications for low-rate wireless personal area networks (wpans). *IEEE Std. 802.15.4-2006*.

[2] Wireless sensor network. https://en.wikipedia.org/wiki/Wireless_sensor_network. Accessed: 2017-06-15.

[3] P Abinaya and C Geetha. Dynamic detection of node replication attacks using x-red in wireless sensor networks. In *Information Communication and Embedded Systems (ICICES), 2014 International Conference on*, pages 1–4. IEEE, 2014.

[4] Wafa Ben, Mauro Jaballah, Gilberto Conti, Mohamed File, Akka Mosbah, and Zemmari. Smart node positioning in clone attack in wireless sensor networks. University of Padua Italy and University of Bordeaux France.

[5] Xiaoming Deng, Yan Xiong, and Depin Chen. Mobility-assisted detection of the replication attacks in mobile wireless sensor networks. In *Wireless and Mobile Computing, Networking and Communications (WiMob), 2010 IEEE 6th International Conference on*, pages 225–232. IEEE, 2010.

[6] Wazir Zada Khan, Mohammed Y Aalsalem, and Mohamad Naufal Mohamad Saad. Detection of masked replication attack in wireless sensor networks. In *Information Systems and Technologies (CISTI), 2013 8th Iberian Conference on*, pages 1–6. IEEE, 2013.

[7] Sachin Umrao, Arun Kumar, and Praveet Umrao. Security attacks and their countermeasures along with node replication attack for time synchronization in wireless sensor network. In *Advanced Nanomaterials and Emerging Engineering Technologies (ICANMEET), 2013 International Conference on*, pages 576–581. IEEE, 2013.

[8] Sachin Umrao, Deeksha Verma, and Arun Kumar Tripathi. Detection and mitigation of node replication with pulse delay attacks in wireless sensor network. In *Innovation and Technology in Education (MITE), 2013 IEEE International Conference in MOOC*, pages 390–392. IEEE, 2013.

[9] Chia-Mu Yu, Chun-Shien Lu, and Sy-Yen Kuo. Mobile sensor network resilient against node replication attacks. In *Sensor, Mesh and Ad Hoc Communications and Networks, 2008. SECON'08. 5th Annual IEEE Communications Society Conference on*, pages 597–599. IEEE, 2008.

[10] Chia-Mu Yu, Yao-Tung Tsou, Chun-Shien Lu, and Sy-Yen Kuo. Localized algorithms for detection of node replication attacks in mobile sensor networks. *IEEE Transactions on Information Forensics and Security*, 8(5):754–768, 2013.

[11] Bo Zhu, Venkata Gopala Krishna Addada, Sanjeev Setia, Sushil Jajodia, and Sankardas Roy. Efficient distributed detection of node replication attacks in sensor networks. In *Computer Security Applications Conference, 2007. ACSAC 2007. Twenty-Third Annual*, pages 257–267. IEEE, 2007.

Appendices

Appendix I - Malicious node only increases sequences number after receiving ACK

This is the data taken from experiment done on SenseNuts devices. This data is for case 1 in which malicious node is not increasing its sequence number without getting ACK, thus the attack is detected whenever a less sequence number is received than the last one.[1]

A Sensor Data - I

```
Node Light Seq. Number

BFD0 33.0 1.0
BFD0 33.0 2.0
BFD0 16.0 1.0
BFD0 33.0 4.0
BFD0 16.0 2.0
BFD0 33.0 5.0
BFD0 33.0 6.0
BFD0 33.0 7.0
BFD0 33.0 8.0
BFD0 33.0 11.0
BFD0 33.0 12.0
BFD0 33.0 13.0
BFD0 33.0 14.0
BFD0 33.0 15.0
BFD0 33.0 17.0
BFD0 33.0 18.0
BFD0 33.0 20.0
BFD0 33.0 22.0
BFD0 33.0 26.0
BFD0 33.0 28.0
BFD0 33.0 30.0
BFD0 33.0 31.0
BFD0 33.0 33.0
BFD0 33.0 34.0
BFD0 33.0 37.0
BFD0 33.0 39.0
BFD0 33.0 40.0
BFD0 33.0 42.0
BFD0 33.0 43.0
```

[1]Note: It is not a complete data, it is just a snapshot.Complete Data can't be included as it is very large. Also Don't combine the debug data with sensor data, they can be snapshots of two different time.

B Debugging Data - I

```
07/06/2017, 03:31:15. : 30080021BFD001010000BD36C0E2
07/06/2017, 03:31:12. : BD36
07/06/2017, 03:31:12. :  Send to
07/06/2017, 03:31:12. : 0000
07/06/2017, 03:31:12. :  Requested
07/06/2017, 03:31:12. :  Verified
07/06/2017, 03:31:12. :  Data Packet Received
07/06/2017, 03:31:12. : 30070021BFD001010000BD36C0E2
07/06/2017, 03:31:11. : BD36
07/06/2017, 03:31:11. :  Send to
07/06/2017, 03:31:11. : C0E2
07/06/2017, 03:31:11. :  Requested
07/06/2017, 03:31:11. :  Verified
07/06/2017, 03:31:11. :  Data Packet Received
07/06/2017, 03:31:11. : 30060021BFD00101C0E2BD36C0E2
07/06/2017, 03:31:11. : BD36
07/06/2017, 03:31:11. : BFE9
07/06/2017, 03:31:11. : BFD0
07/06/2017, 03:31:11. :  Replication Attack Detected! 1. Verified Node;
  2. Verification Made by Node; 3. Verified By
07/06/2017, 03:31:11. :  Replication Attack Detected
07/06/2017, 03:31:11. :  Verify Packet PAN Received
07/06/2017, 03:31:11. : 34BFD004BFE9BD36
07/06/2017, 03:31:09. : BD36
07/06/2017, 03:31:09. :  Send to
07/06/2017, 03:31:09. : 0000
07/06/2017, 03:31:09. :  Requested
07/06/2017, 03:31:09. :  Verified
07/06/2017, 03:31:09. :  Data Packet Received
07/06/2017, 03:31:09. : 30050021BFD001010000BD36C0E2
07/06/2017, 03:31:09. :  Not Verified
07/06/2017, 03:31:09. :  Data Packet Received
```

Appendix II - Malicious node increases sequences number without receiving ACK

This is the data taken from experiment done on SenseNuts devices. This data is for case 2 in which malicious node is increasing its sequence number even without getting ACK, thus the attack is detected only when a witness node receives a skip in the sequence number. In this case, we are assuming that source is send for verification for every packet transmitted.[1]

C Sensor Data - II

```
Node  Light Seq. Num
C0CC 9.0 1.0
BFD0 17.0 2.0
C0E2 13.0 2.0
BFF4 9.0 2.0
C0CC 9.0 2.0
C012 26.0 2.0
BFD0 9.0 2.0
BFD0 17.0 3.0
BFD0 9.0 3.0
C0E2 13.0 4.0
BFF4 10.0 4.0
C0CC 9.0 4.0
BFF4 9.0 5.0
BD36 14.0 5.0
C012 26.0 5.0
BFE9 5.0 6.0
C0E2 13.0 6.0
BFF4 10.0 6.0
C012 26.0 6.0
C081 15.0 7.0
BD36 14.0 7.0
C012 26.0 7.0
BFD0 17.0 8.0
C081 15.0 8.0
BFF4 10.0 8.0
C0CC 9.0 8.0
C0CC 9.0 9.0
BFF4 10.0 10.0
```

[1]Note: It is not a complete data, it is just a snapshot.Complete Data can't be included as it is very large. Also Don't combine the debug data with sensor data, they can be snapshots of two different time.

```
C0CC 9.0 10.0
C012 26.0 10.0
BFD0 9.0 10.0
C081 15.0 11.0
C0E2 14.0 11.0
C0CC 9.0 11.0
BFD0 17.0 11.0
C012 26.0 11.0
BFE9 5.0 12.0
BFF4 10.0 12.0
C0CC 9.0 12.0
BD36 14.0 12.0
C012 26.0 12.0
BFD0 9.0 12.0
C081 16.0 13.0
BFF4 10.0 13.0
BD36 14.0 13.0
BFD0 9.0 13.0
C081 16.0 14.0
BFF4 10.0 14.0
BD36 14.0 14.0
BFD0 17.0 14.0
```

D Debugging Data - II

```
08/06/2017, 02:34:48. : 300A0009BFD002BFF4C081
08/06/2017, 02:34:44. :  Data Packet Received
08/06/2017, 02:34:44. : 300A001AC01201BD36
08/06/2017, 02:34:41. :  Data Packet Received
08/06/2017, 02:34:41. : 300A0009C0CC01BD36
08/06/2017, 02:34:41. :  Data Packet Received
08/06/2017, 02:34:41. : 300A000ABFF401C081
08/06/2017, 02:34:31. : 07
08/06/2017, 02:34:31. :  Last Sequence Number Packet received is
08/06/2017, 02:34:31. :  Replication Attack Detected! because of missing
  sequence number
08/06/2017, 02:34:31. :  Data Packet Received
08/06/2017, 02:34:31. : 30090009C0CC01BD36
08/06/2017, 02:34:21. :  Data Packet Received
08/06/2017, 02:34:21. : 30080009C0CC01BD36
08/06/2017, 02:34:21. :  Data Packet Received
08/06/2017, 02:34:21. : 3008000ABFF401C081
08/06/2017, 02:34:19. : 3008000FC08100
08/06/2017, 02:34:17. :  Data Packet Received
08/06/2017, 02:34:14. : 3007001AC01201BD36
08/06/2017, 02:34:11. :  Data Packet Received
08/06/2017, 02:34:11. : 3007000EBD3600
08/06/2017, 02:34:09. : 05
08/06/2017, 02:34:09. :  Last Sequence Number Packet received is
08/06/2017, 02:34:09. :  Replication Attack Detected! because of missing
     sequence number
08/06/2017, 02:34:09. :  Data Packet Received
08/06/2017, 02:34:09. : 3007000FC08100
08/06/2017, 02:34:04. :  Data Packet Received
08/06/2017, 02:34:04. : 3006001AC01201BD36
08/06/2017, 02:34:01. :  Data Packet Received
08/06/2017, 02:34:01. : 30060009C0CC01BD36
08/06/2017, 02:34:01. : 3006000ABFF401C081
08/06/2017, 02:34:00. : 3006000DC0E202BFF4C081
08/06/2017, 02:34:00. :  Data Packet Received
08/06/2017, 02:34:00. : 30060005BFE902BFF4C081
08/06/2017, 02:33:57. :  Data Packet Received
08/06/2017, 02:33:57. : 30060011BFD003BFE9BFF4C081
08/06/2017, 02:33:54. :  Data Packet Received
08/06/2017, 02:33:54. : 3005001AC01201BD36
08/06/2017, 02:33:51. : 3005000EBD3600
08/06/2017, 02:33:51. : 03
```

```
08/06/2017, 02:33:51. :  Last Sequence Number Packet received is
08/06/2017, 02:33:51. :  Replication Attack Detected! because of missing
 sequence number
08/06/2017, 02:33:51. :  Data Packet Received
08/06/2017, 02:33:41. :  Data Packet Received
08/06/2017, 02:33:41. : 30040009C0CC01BD36
08/06/2017, 02:33:41. : 3004000ABFF401C081
08/06/2017, 02:33:40. :  Data Packet Received
08/06/2017, 02:33:40. : 3004000DC0E202BFF4C081
08/06/2017, 02:33:35. :  Data Packet Received
08/06/2017, 02:33:35. : 30030009BFD000
```