www.ijecs.in
*International Journal Of Engineering And Computer Science ISSN:2319-7242*
*Volume 4 Issue 7 July 2015, Page No. 13452-13456*

# Cost Effective Method for Detecting Clone Nodes in Wireless Sensor Network

*[1]Dilna V., [2]Sajitha M.*

[1] M-Tech Student, Computer Science and Engineering, MES college of Engineering
Malappuram, Kerala, India
dilnavijay0@gmail.com

[2] Assistant Professor, Computer Science and Engineering, MES college of Engineering
Malappuram, Kerala, India

**Abstract:** ***In most of wireless sensor application security is one of the prime concern. Generally sensor nodes are not equipped with any tamper resistant hardware and they are deployed in a hostile environment, so the chance of occurring attacks should be greater. In node clone attack adversary will capture few nodes from the network, retrieving its credentials and creating large number of clones by reprogramming the nodes. And these clones may have the ability to subvert the whole network. Thus the detection of node clone attacks in a wireless sensor network is therefore a fundamental problem. In distributed environment many protocols are available to detect the clone attack. Thus far, various schemes have been proposed to detect replicas; however, most of them require expensive hardware like global positioning system (GPS) to obtain the location of a sensor node. In general, sensor nodes are equipped with limited set of resources, to suit for resource constraint sensor application; hence it is not practical to employ additional devices like GPS in them for the detection process. In Cost Effective Method for Detecting Clone Nodes in WSN (CEMDCN) protocol introducing a low priced and energy efficient solution for detecting clone nodes in wireless sensor network without using GPS in them. Extensive simulation shows that proposed method is efficient in terms of detection probability, memory and communication overhead. Also this is a better clone detection scheme in resource constraint sensor application.***

**Keywords: Clone node, Distributed detection, Node replication, Wireless Sensor Network, Witness nodes.**

## 1. Introduction

Nowadays WSN are invaded in most of the areas of our daily life. Typically a WSN consist of large number of spatially distributed autonomous sensor node, with ability to sense environment, doing computation of sensed data and providing wireless communication. All the nodes in WSN collaborate to accomplish a common task, for example, earth sensing, military surveillance, health care monitoring . Here in this network sensor nodes collect data within their sensing environment and send this data to the sink node. These types of networks are generally heedless because sensor nodes are unattended and deployed in a hostile environment; hence there is a high chance of various attacks on sensor nodes. Normally WSNs are employed for some critical application, so one of the primary concerns of this type of system should be considered as its security. Generally sensor nodes are not equipped with any tamper resistant hardware. So it is easy for an attacker to capture and compromise a sensor node. In node clone attack an attacker captures a sensor node; retrieve the information about the node and produces copies of the captured node. And also all the cloned node will be having the same ID of the captured node. Clone nodes are treated as statutory nodes and hence it will be difficult to detect them. Once the clone nodes acquire the trust of other sensor nodes, they can perform various attacks on these sensor networks. For example they may provide false sensor reading, drop packet while communication, spy for confidential information and leak it to an adversary. In order to overcome these difficulties it could be efficient to identify the replicas in a static WSN.

### 1.1 Node Clone Attack

In node clone attack also called as node replication attack, an attacker will physically capture a node from its deployed location. Then the attacker will access the it’s memory, communication and processing unit of the captured node, and they also steals the relevant information including its secret key, identity and intrusion detection characteristics. After that by using the stolen information attacker will generate a number of clones having the same ID of the captured nodes, and deploy them back into the network. These clones operate under the control of the attacker. Also clones will then try to behave like a legitimate node, and participate in the process of communication using the stolen keying materials. The aim of an attacker in node clone attack is to control the network activities by using clones. With the help of clones, an attacker can launch a variety of insider attacks likes selective forwarding, wormhole, hello flooding and false data injection. An attacker can perform all of the above mentioned attacks only by compromising a single node from the network. Therefore, node clone attack is considered as one of the most serious threats in WSN. After creating replicas it is a great challenge to differentiate between the statutory node and its clones. Since, clones execute the same network protocols and they use the same keying materials as that of a original node, they pass in all authentication and verification process during transmission [8]. Most of the schemes discussing in the literature recommends for the identification of existence of clones in the network. These schemes mostly use the parameters such as unique set of neighboring nodes, position etc., to differentiate a clone from its original node.

## 2. Related works

Approaches for detecting clone node in static WSNs are broadly categorized into centralized and distributed techniques [7]. In Centralized scheme [1] each node sends a list of its neighbors and their location claim to the base station, and the base station checks whether there exist same node ID with different location information. If such nodes exist, it could be revoked from the network by flooding an authenticated revocation message.

In distributed method [1] one or more nodes are responsible for to identify the replica. These nodes are called witness node. When a new node joins in the network its ID and location information is send to witness node, and witness node check for clones. Preliminary approaches to detect clone node in distributed environment are, Node to Network broadcasting (N2NB) [1] and Deterministic multicast (DM) [1]. In Node To-Network Broadcasting every node in the network uses an authenticated broadcast message to flood the network with its location information. Each node stores the location information for its neighbors. If it receives a conflicting claim, it revokes the offending node. In this method the total communication cost for each node should be very high. In Deterministic multicast a nodes location claim is shared with a limited subset of deterministically [1] chosen witness nodes. Since deterministic, the attacker can also determine the witness nodes. Also it cannot afford a large number of witness nodes. Other distributed detection techniques are,

### 2.1 Randomized multicast (RM)

In Randomized multicast [1] each of the node's neighbors probabilistically forwards the location claim to a randomly selected set of witness nodes. If any witness node receives two different location claims for the same node ID, it can revoke the replicated node. The birthday paradox[9] ensures that two conflicting claims have a high probability of sharing a common witness node. Its drawbacks are higher communication cost and lower detection probability. Randomized multicast improves the resiliency of the deterministic multicast by randomizing the witnesses for a given node, so that the adversary cannot anticipate their identities.

### 2.2 Line Selected Multicast (LSM)

Line Selected Multicast uses the routing topologies to detect and to identify the clones in sensor network. It is an improved version of RM. In addition to the witness nodes of RM, LSM checks all the intermediate node within the path for clone nodes. Here all intermediate nodes from a node to a destination node will also store location claims as a line. When location claim is transferred, any node on the path verifies the signature of the claim and checks for the conflict, by using the location information stored in its buffer. If there is a conflict, it revokes offending node from the network. Otherwise store the claim and forwards to next node. Here a node on the line-crossing point will detect a conflict, if conflicting location claim line crosses the node. So LSM has lower communication cost and better detection level as compared to Randomized Multicast. But it suffers from higher memory overhead, cross over problem [1] and crowded center problem [1].

### 2.3 Memory Efficient multicast using Bloom filters and cell forwarding (BC-MEM)

Memory Efficient Multicast using Bloom filters and cell forwarding (BC-MEM) [2] is introduced to overcome the memory overhead problem occurred in LSM. In this protocol, the deployment area is divided into virtual cells. In each cell, an anchor point is assigned for every node in the network. The node close to the anchor point is called anchor node. The location claim is forwarded to the anchor point of the next cell where the line segment interacts. The claim is then forwarded from one anchor node to another until it reaches at the last cell. The anchor nodes in the intermediate cells are watchers and the anchor nodes in the first and last cells are witnesses. Here the location claim is only transmitted through the watcher nodes, and the witness nodes store the claim message. Watcher node uses bloom filter [2] for storing claim message in memory, so it takes lesser memory than LSM. This protocol also avoids the cross over problem [1] in LSM.

### 2.4 Localized Multicast

There are two variants of localized multicast [3] are introduced: Single Deterministic Cell (SDC) [3] and Parallel Multiple Probabilistic Cells (P-MPC) [3]. In these two protocols witness nodes are selected from a geographically limited region of node, called cell. By using a deterministic function each node ID should be mapped to one or more cells. To increase the resilience and security of the scheme randomization is using within the cells. In SDC, each cell is mapped into a single destination cell by using a geographical hash function. Each node in the destination cell independently decides whether to store the claim. On reception of different location claims with the same ID, destination cell can detect the presence of clones. In the P-MPC scheme, the location claim is mapped and forwarded to multiple deterministic Cells with various probabilities [3].

### 2.5 Random Walk based Approach

Random walk based approach is a modified version of RM. Here, after reaching the random destination in the RM, this starts a random walk to obtain the witness node. Thereby the adversary cannot easily find out the critical witness nodes. Two approaches: Random Walk (RAWL) [4] and Table Assisted Random Walk (TRAWL) [4]. In Random Walk approach neighbors of a node forward the location claim of the node to random destination with some probability. And these random nodes send a message containing the claim to start a 't' step random walk in the network, where 't' is a system defined parameter, and these passed nodes are acts as witness node for to identify the clones. Table Assisted Random walk is employed to minimize the memory costs of RAWL protocol. Traces of random walk are recorded at each node using a trace table. Each passed node does not want to store the location claims. Random walk based approaches increase the replica detection level of RM.

### 2.6 Randomized Efficient and Distributed (RED) protocol

RED [5] combines both the merits of RM and DM. RED execute in a fixed interval of time. Every execution of the protocol consists of two steps. In the first step of the protocol a random value called ‘rand’ is shared among all the nodes in the network by a trusted third party or by the base station. In second step which is also called detection phase, each node broadcast its ID and location claim to its neighboring node. Each neighbor node probabilistically forwards the location claim to a pseudo-randomly selected ‘g’ number of network location. Input parameters of the pseudo-random function are node ID, ‘rand’ and ‘g’. Every node in the path (from claiming node to the witness destination) forwards the message to its neighbor nearest to the destination. Hence, the replicated nodes will be detected in each detection phase. Here witness node should be different in the next interval since the random value which is broadcasted by the trusted entity is changed.

### 2.7 RDB-R detection scheme

In [6] introduced a new detection scheme for to find out the replicas in wireless sensor network called RDB-R detection scheme. RDB-R provides low-priced replica detection solution for static WSNs by using Bloom filter and sequential delivery approaches. RDB-R detection scheme avoid the use of GPS in the detection process because here, neighboring node IDs of a node is used as the proof for identifying clones, instead of location information like the existing schemes. Neighboring node IDs are presented with a constant size using a Bloom filter. The bloom filter output (BFO) [6] is used as a proof of identification. A newly deployed node generates different proofs according to the collected neighboring node IDs, until collecting the entire neighboring node IDs. The proofs are delivered to a randomly selected node called witness node to check the witness conflict.

RDB-R detection scheme consist of 3 steps

- Proof generation
- Proof delivery
- Proof validation

In proof generation node 'α' creates its bloom filter output by compressing its neighboring node information and store it into the bloom filter. In proof delivery which checks the neighboring nodes are registered to proof and they are the two hop neighbors of node 'α', if both the conditions are satisfied the proofs are delivered to a randomly selected node in the network called witness node. In proof validation step it check whether a conflicting bloom filter output are received for any witness node, if yes which identify the replica node in the network.

## 3. Proposed work

In this section introducing a cost effective method for detecting clone nodes in wireless sensor network (CEMDCN) by combining merits of RDB-R [6] and the RED [5] protocol. So in the proposed method which uses neighboring information of node to find out the replica, i.e. when a node is replicated, the original node and replicated node has different set of neighbors. Neighboring node IDs are presented with a constant size using a Bloom filter. The Bloom filter output (BFO) is used as a proof of identification. Here the witness nodes are selected pseudo-randomly by using a pseudo-rand() function. Pseudo-rand function select same set of witness node in one run of the protocol, and it takes different set of witness in different run of the protocol. So it highly improves the detection level of this protocol.

### 3.1 Bloom filter [6]

In general, a Bloom filter is used for member checking. A Bloom filter for representing a group G= { $x_1$, $x_2$,....$x_n$ } of ‘n’ members is described by an array of ‘m’ bits, initially all the bits are set to 0. A bloom filter uses k independent one way hash functions $h_1$, $h_2$,..$h_k$ with range 1,....,m. For mathematical convenience, it makes the natural assumption that these hash functions map each item in the universe to a random number uniform over the range 1,..., m. For each member x Є G, the bits $h_i(x)$ between 1 and m are set to 1 for $1\leq i \leq k$. A location can be set to 1 multiple times, but only the first change has an effect. To check if an item y is in G, we check whether all $h_i(y)$ are set to 1. If not, then undoubtedly y is not a member of G. If all $h_i(y)$ are set to 1, we infer that y is in G, although we are wrong with some probability. Hence, a Bloom filter may yield a false-positive error, whereby it suggests that a member x is in G even though it is not.

### 3.2 CEMDCN Protocol

The CEMDCN protocol is described here,

Here every run of the protocol consist of four steps.

**Step 1:** Random value distribution

**Step 2:** Proof generation

**Step 3:** Proof delivery

**Step 4:** Proof validation

**Step 1: Random value distribution**

In this step, a random value 'rand' is shared among all the nodes in the network. This can be done either by using a centralized broadcasting or distributed mechanism. In next run of the protocol ‘rand’ value broadcast by the base station could be changed [5].

**Step 2: Proof generation**

In proof generation step, each node generates its bloom filter output (BFO) by adding neighboring node ID into its bloom filter. After generating BFO, it is sending to its neighboring nodes.

**Step 3: Proof delivery**

In this step neighboring nodes calculate witness node by using pseudo-rand function. Parameters of pseudo-rand functions are ID of the node, current ‘rand’ value and number of witness nodes. So pseudo-rand map a node into same set of witness node in one run of the protocol, but in next run ‘rand’ value broadcast by the base station is changed and also witness nodes also changed. After calculating witness node each neighbor node sends ID and BFO of the node to witness location.

**Step 4: Proof validation**

In fourth step, when a witness node receives two different proof information for the same ID it do a subset checking process, to identify whether it is a replica or not. If the subset checking result is false it indicates that the received nodes are replicas, if it is true this is not a replica.

**Subset checking [6]:** Let U is a witness node selected by node C. We assume that node A’ and A” are replicas of A. If U collects two different proofs $BFO_{A'}$ and $BFO_{A''}$ on the same ID (here, $ID_A$), it checks whether either proof is a subset of the other (i.e., $BFO_{A'}\subseteq BFO_{A''}$ or $BFO_{A'} \supseteq BFO_{A''}$ ). Here, a BFO collision occurs when a node receives two different BFOs on the same ID. If the check is failed, the newly inserted node is regarded as a replica.

Subset checking is expressed as

$$(BFO_{A'}\subseteq BFO_{A''}) \vee (BFO_{A'} \supseteq BFO_{A''});$$

$$(BFO_{A'} \subseteq BFO_{A''}) \Rightarrow \{(b_{BFOA',\, i=}\, b_{BFOA'',\, i}) \vee (b_{BFOA',i}=0 \wedge b_{BFOA'',i}=1)\},$$

$$(BFO_{A'} \supseteq BFO_{A''}) \Rightarrow \{(b_{BFOA',i=}\, b_{BFOA'',i}) \vee (b_{BFOA',i}=1 \wedge b_{BFOA'',i}=0)\}$$

If the subset checking result is true, U decides that A is not a replica. Otherwise, U reports revocation of A to the base station. Finally, the base station broadcasts the revocation message of A to the entire network, and then each node ignores all messages from nodes having $ID_A$. Accordingly, the replica attack is nullified. Frame work of the proposed system is shown in fig 1.

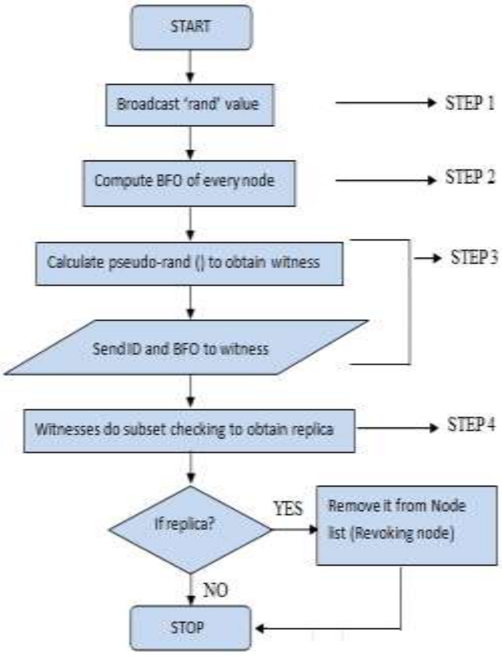
START
Broadcast 'rand' value
STEP 1
Compute BFO of every node
STEP 2
Calculate pseudo-rand () to obtain witness
STEP 3
Send ID and BFO to witness
Witnesses do subset checking to obtain replica
STEP 4
If replica?
YES
Remove it from Node list (Revoking node)
NO
STOP

Fig.1 Framework of the proposed system

## 4. Results and Analysis

The project is implemented using NS2. Here nodes are static. In the simulation result number of legitimate node taken as 30 and number of replicated node is taken as 1, 2, 3, 4 and 5. Initial energy of a node is taken as 250000 mJ. Witness nodes are calculated by using pseudo-rand function. Doing subset checking witness nodes identify the replica present in the network. After identifying replica, these nodes are revoked from the network. The proposed CEMDCN in WSN are compared with the existing RDB-R detection scheme. Here replica detection ratio, average remaining energy and memory overhead are used as parameters.

Once the modification is done it is found that the energy overhead in the proposed method is slightly reduced, because in proposed method intermediate nodes are not storing proof message. So avoid the use of extra energy for computation overhead for replica detection in intermediate node. This is shown in fig 2.

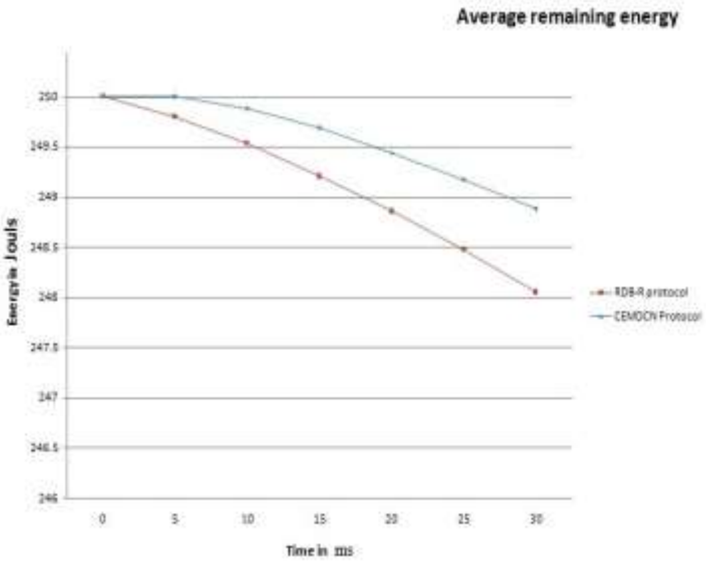

Fig. 2 Average remaining energy of the two protocols

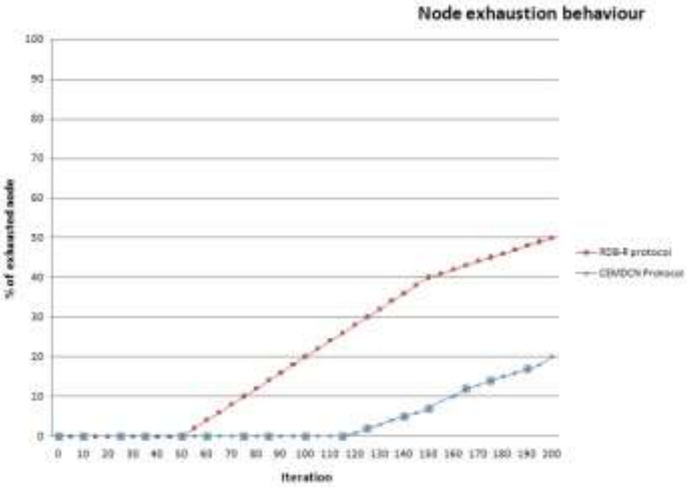

Fig. 3 Node exhaustion behavior

The operating life of a sensor node depends on its energy. Here in fig 3 show the node exhaustion behaviour. After completing 50 iteration of RDB-R detection scheme nodes are started to exhaust, but in proposed method nodes are started to exhaust after 110 iteration of the protocol. This is because that in existing method nodes located at the central region drains their energy very easily. So nodes at central region exhaust very easily. The graphical results fig 4 show that replica detection ratio in the proposed method is higher than the existing method. In existing method witness nodes are selecting randomly from the network. But in the proposed method one run of the protocol select same set of witness node for a node. In existing method when number of replicated node increases replica detection ratio also increase.

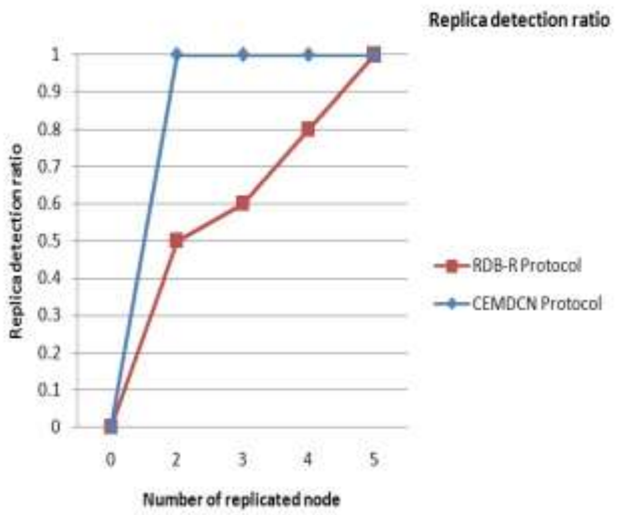

Fig. 4 Replica detection ratio, here legitimate node=30, witness location=1

Memory overhead in the existing method is higher than the proposed method, because in existing method, intermediate nodes are also storing proof information in its memory. Here number of witness node is selected as one and the total number of nodes in a network is taken as 30. And graph is plotted according to this. Graphical result in fig 5 shows that, in proposed method most of the nodes are storing smaller number of messages in its memory. But in existing method some nodes are storing large number of messages in memory. It may incur memory overhead problem in existing method.

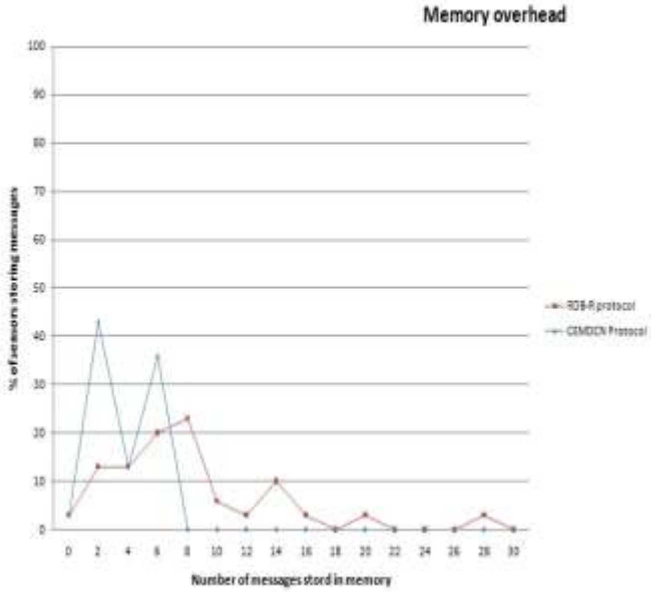

Fig. 5 Memory overhead of two protocol

## 5. Conclusion

Wireless sensor network is an emerging area which has wide applications. Hence the security in wireless sensor network is of great concern. Node replication attacks are an important attack against a wireless sensor network in which an adversary compromises a sensor node and creates copies of that node and deploying it in strategic areas. Various methods have been developed in order to detect the node replication attacks. Low priced and energy efficient detection of node replica in WSN introduced RDB-R detection scheme and it is a low cost and efficient solution of replica detection in wireless sensor network. But RDB-R protocol has problems in terms of its memory overhead and detection level. To overcome these difficulties a CEMDCN protocol is introduced. Implementation result shows that the proposed method reduces the memory and communication overhead and also improves the detection level of RDB-R detection scheme. And this protocol is best suited for resource constraint sensor application. Because it uses neighbouring information instead of location information for detecting replica, so it avoids use of GPS and reduce sensor node cost. So this scheme is a cost effective mechanism for detecting clone nodes in wireless sensor network.

## References

[1] Bryan Parno, Adrian Perrig, Virgil Gligor ,"Distributed detection of node replication attacks in sensor networks" in Proceeding of the IEEE Symposium on Security and Privacy,(IEEE S and P`05),pp49-63, May 2005.

[2] Ming Zhang, Vishal Khanapure, Shigang Chen, Xuelian Xiao, "Memory efficient protocols for detecting node replication attacks in wireless sensor networks", in Proceedings of the 17th IEEE International Conference on Network Protocols (ICNP'09), pp 284-293, Princeton, NJ,USA, October 2009.

[3] Bio Zhu, Sanjeev Setia, Sushil Jajodia, Sankar das Roy and Lingyu Wang, "Localized multicast: efficient and distributed replica detection in large scale sensor networks", IEEE Transactions on Mobile Computing, Vo1. 9, No. 7, pp 913-926, 2010.

[4] Yingpei Zeng, Jiannong Cao, Shigeng Zhang,Shanqing Gao and Li Xie ,Random walk based approach to detect clone attacks in wireless sensor networks, IEEE Journal on selected areas in Communications, Vol. 28, No.5, pp 677-691, 2010.

[5] M. Conti, R. Di Pietro, L. Mancini, and A. Mei, A Distributed detection of clone attacks in wireless sensor networks, IEEE Transactions on Dependable and Secure Computing, vol. 8, no. 5, pp. 685698, 2011.

[6] Kwantae Cho, Byung-Gil Lee, and Dong Hoon Lee, "Low Priced and Energy Efficient Detection of Replicas for Wireless Sensor Networks", IEEE Transactions on dependable and secure computing, Vol. 11, NO. 5, September/October 2014.

[7] S. S. Koshy and M. Sajitha, "Zone based node replica detection in wireless sensor network using trust", International Journal of Computer Trends and Technology, vol. 4, no. 7, pp. 2316-2320, 2013

[8] Wazir Zada Khan, Mohammed Y. Aalsalem, Mohammed Naufal Bin Mohammed Saad, and Yang Xiang, "Detection and Mitigation of Node Replication Attacks in Wireless Sensor Networks: A Survey", International Journal of Distributed Sensor Networks, March 2013.

[9] https://en.wikipedia.org/wiki/Birthday_problem