Asian Journal of Information Technology 15 (21): 4255-4268, 2016
ISSN: 1682-3915
© Medwell Journals, 2016

Prevention Protocol for Node Replication Attack

[1]V. Manjula,[2]C. Cheallappn and [3]R. Thalapathi Rajasekaran
[1]Department of Computer Science and Engineering, S.A. Engineering College, Anna University,
Chennai, Tamil Nadu, India
[2]Department of Computer Science and Engineering, GKM Engineering College, Anna University,
Chennai, Tamil nadu, India
[3]Department Computer Science and Engineering, SMK Fomra Institue of Technology,
Anna University, Chennai, Tamil nadu, India

Abstract: A naming or addressing scheme usually requires that each node be assigned a unique name or address called as an identity in order to avoid ambiguities. Notably, routing protocols need addresses to route packets. Services need names in order to be identifiable, discoverable and useable. Attacks against naming and addressing are address stealing, Sybil attack and node replication attack. The harmful attack against Wireless Sensor Networks (WSN) is node replication attack, where one or more node(s) illegitimately claims an identity, are also called clone attack due to identity theft. Naming and addressing are fundamental for networking. Indeed, it is not required to compromise a high number of nodes: the main cost for the adversary is to capture and to tamper just one sensor; making many clones out of the tampered sensor can be considered easy and cheap. Hence, we are concentrating on node replication attack where also basis for variety of attacks such as Sybil attack, routing attacks and link layer attacks etc. also called as denial of service attacks. As per the proverb, 'Prevention is better than Cure', This study proposes a prevention protocol to stop replicas before joining into the network, instead of after allowing the attack into the network and detecting them. This study discusses a novel approach for preventing the node replication attack by blocking the possible re-entry of captured node either cloned or compromised.

Key words: Security, clone, node replication attack, wireless sensor network

INTRODUCTION

A Wireless Sensor Network (WSN) is a collection of sensors with limited resources that collaborate in order to achieve a common goal. Sensor nodes operate in hostile environments such as battlefields and surveillance zones. Due to their operating nature, WSNs are often unattended, hence prone to several kinds of novel attacks.

The mission-critical nature of sensor network applications implies that any compromise or loss of sensory resource due to a malicious attack launched by the adversary-class can cause significant damage to the entire network. Sensor nodes deployed in a battlefield may have intelligent adversaries operating in their surroundings, intending to subvert damage or hijack messages exchanged in the network. The compromise of a sensor node can lead to greater damage to the network. The resource challenged nature of environments of operation of sensor nodes largely differentiates them from other networks. All security solutions proposed for sensor networks need to operate with minimal energy usage, whilst securing the network.

We classify sensor network attacks into three main categories (Baig, 2008). Identity attacks, routing attacks and network intrusion. Identity attacks intend to steal the identities of legitimate nodes operating in the sensor network. The identity attacks are Sybil attack and clone (Replication) attack. In a Sybil attack, the WSN is subverted by a malicious node which forges a large number of fake identities in order to disrupt the network's protocols. A node replication attack is an attempt by the adversary to add one or more nodes to the network that use the same ID as another node in the network.

Furthermore naming or addressing scheme requires that each node be assigned a unique name or address called as an identity in order to avoid ambiguities. Notably, routing protocols need addresses to route packets. Services need names in order to be identifiable, discoverable and useable. Attacks against naming and

Corresponding Author: V. Manjula, Department of Computer Science and Engineering, S.A. Engineering College, Anna University, Chennai, Tamil Nadu, India

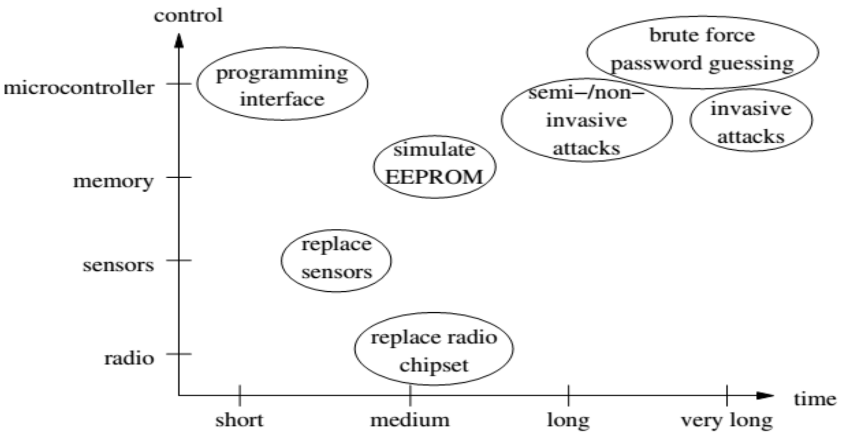

Fig. 1: Degree of control over the sensor node with time span

addressing are address stealing, Sybil attack and node replication attack. Adversary started using an address already assigned to and used by a legitimate node is called address stealing. Sybil attack is known as a single adversarial node uses several invented addresses to make legitimate nodes to believe that there are many other nodes around. The node replication attack is dual of the Sybil attack the adversary introduces replicas of a single compromised node using the same address at different locations of the network.The clone attack is very convenient (Conti *et al*., 2014) for an adversary. Indeed, it is not required to compromise a high number of nodes: the main cost for the adversary is to capture and to tamper just one sensor; making many clones out of the tampered sensor can be considered easy and cheap. Hence, we are concentrating on node replication attack where also basiss for variety of attacks such as Sybil attack, routing attacks and link layer attacks etc. also called as denial of service attacks. The detection of node replication attacks in a wireless sensor network is therefore a fundamental problem. A few centralized and distributed solutions have recently been proposed. However, these solutions are not satisfactory. First, they are energy and memory demanding. A serious drawback for any protocol that is to be used in resource constrained environments such as a sensor network. Further, they are vulnerable to specific adversary models introduced in this study.

MATERIALS AND METHODS

Stages of replication attack: An attacker captures a node, i.e., physically removes it from the network) eventually acquiring all the cryptographic material stored in it. Node capture represents the first step for further different types of attacks, node replication attack, thwart on confidentiality and Sybil attack. If we are able to detect the capture attack (Conti *et al* 2008), then able to prevent the node replication as well as other different types of attacks.

Becher *et al*. (2006) categorized based on the degree of control over the sensor node the attacker gains against time span as shown in Fig. 1 able to control the radio function of the node (including the ability to read, modify, delete and create radio messages) without having access to the program or the memory of the sensor node; allow to learn at least some of the contents of the memory of the node, either the RAM on the microcontroller, its internal flash memory, or the external flash memory. This may give the attacker, e.g., cryptographic keys of the node; adversary ability to have complete read/write access to the microcontroller. Becher *et al*. (2006) also categorized the time span as short, medium, long and very long during which regular operation of a node is interrupted to have control over radio, sensors, memory and microcontroller.

Short (less than five minutes): Attacks creating plug-in connections and making a few data transfers over programming interfaces - IEEE 1149.1 JTAG standard. Medium (less than thirty minutes): Most attacks which take this amount of time require some mechanical work, for instance (de-) soldering, eg., replace sensors. Replace radio chipset and simulate EEPROM. Long (less than a day). This might involve a non-invasive or semi-invasive attack on the microcontroller, e.g., a power glitch attack where the timing has to be exactly right to succeed, or erasing the security protection bits by UV light.

Very long (more than a day): usually invasive attacks on the electronic components with associated high equipment cost. Invasive attacks require sophisticated tools on or away from the site.

Khan *et al*. (2013) described the process or stages of node replication attack in the form of a flow chart as

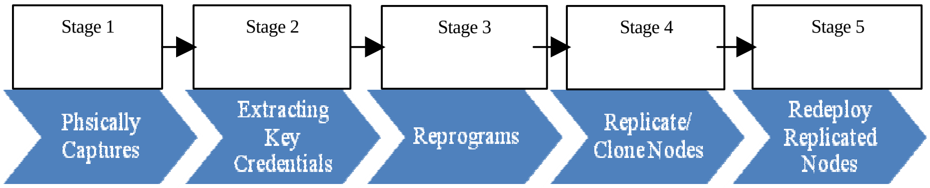

Fig. 2: Stages of replication attack

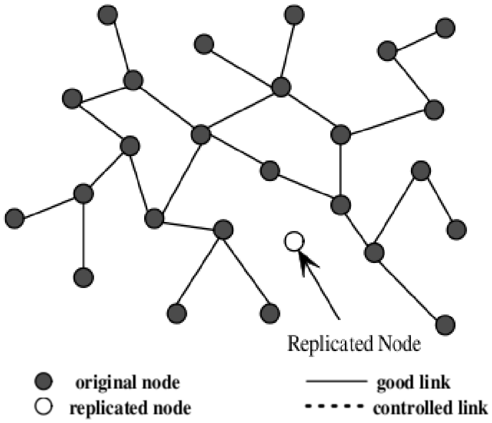

Fig. 3: Injecting replicated nodes

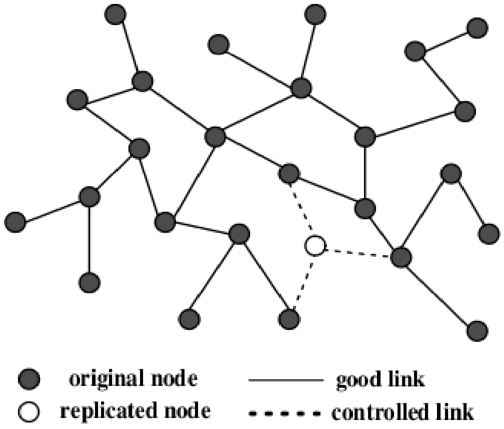

Fig. 4: Creation of controlled links

shown in Fig. 2 The flow chart concisely describes the initiation of a node replication attack from physical node capture, extraction of secret credentials, reprogram, replicating or cloning and redeployment.

At stage 1, an adversary physically captures a sensor node. After physical capture the sensor node remains absent from the network for a specific period of time. If this absence of a sensor node is detected or a tamper-proof hardware is used, the attack will be prevented. Otherwise, an attacker or an adversary starts extracting all the secret materials of the captured node at Stage 2. Hartung *et al*. (2004) demonstrate how to extract cryptographic keys from a sensor node using a JTAG programmer interface in a matter of seconds. At Stage 3, an adversary reprograms the captured node. If an adversary is unable to use a new hardware, it can compromise the node and then exploits the compromised node to disrupt the network operations by its misbehaving activities. On Stage 4, an adversary makes clones or replicas of the captured nodes by using new hardware and these replicas have the same ID and all other keying materials as that of the captured node. After making clones or replicas, an adversary redeploys or injects them at strategic positions of the network for further insider attacks at Stage 5 as shown in Fig. 3 and creates the controlled links as shown in the Fig 4. Once cloned nodes are deployed in the network, the adversary can use them in several malicious ways (Mohammadi and Jadidoleslamy, 2011). For instance, a clone could create a black hole, initiate a wormhole attack (Alarifi and Du, 2006) with a collaborating adversary, inject false data or aggregate data in such a way to bias the final result of a data aggregation procedure (Yang and Yacoub, 2006), collaborative false data injection (Wang *et al*., 2014) with replica cluster (Ho, 2012) or simply leak data.

The Resilience strategy is defined as a control loop, consisting of the processes is defend, detect, remediate and recover as short term (Jabbar, 2010). As per the proverb, ‘Prevention is better than Cure’, this study proposes a prevention protocol to stop replicas before joining into the network, instead of after allowing the attack into the network and detecting them. This study discusses a novel approach for preventing the node replication attack by blocking the possible re-entry of captured node either cloned or compromised

Literature review: To prevent physical node capture attack, the most effective technique is to shield each sensor node with tamper-resistant hardware (Anderson and Kuhn, 1996). However, this technique is generally considered as impracticable due to the high cost and large size of the tamper-resistance hardware. Alarifi and Du (2006) proposed to use the code diversifying technique. They proposed to use different memory locations to store sensitive information for different nodes. By doing so, even if an attacker cracks a sensor node by using the node capture attack suggested in (Hartung *et al*., 2004) and figures out the location of the sensitive information in the memory of that particular node, such memory location information cannot be used to obtain the sensitive information of other nodes in the network. This technique can greatly lengthen the time an attacker needed to compromise a node. Other Prevention Mechanism can be classified based on location based ID Key, Absence of node (might be captured - detection of node capture attack), node addition prevention, block the possible re-entry of captured node either cloned or compromised and proactive prevention. Existing works of the mechanisms have been discussed in the following subsections.

Liu *et al*. (2006) proposed a distributed location-aware key establishment protocol that achieves a high protection against nodes replication attacks under the assumptions made by the authors. The protocol assumes that the network is formed by simple nodes and some sufficiently dedicated nodes called the service nodes, which are elected amongst sensors after deployment. Services nodes play the role of trusted key servers in the network and are assumed to be non compromised. The protocol also assumes that once nodes are deployed, they know their exact location coordinates (x; y) and they remain static. After deployment, each service node creates a distinct t -degree secret bivariate polynomial and then securely initializes each neighbour node with its secret polynomial share, using the unique location coordinates of the node. Two nodes initialized by the same server and being in the transmission overhead of each other, can directly establish a pairwise key. As long as the service nodes are not compromised, the protocol guarantees that an attacker compromising an arbitrary number of sensors in the network, with at most t nodes initialized by the same server node, cannot launch nodes replication attacks. However, if a service node is compromised, which is a current threat because a service node is just a non tamper resistant sensor, an attacker can inject clones and new nodes with new positions and deploy them in the neighbourhood of the compromised service node, to establish pairwise keys with other nodes initialized by the same compromised server node..

Zhang *et al*. (2006) proposed the use of Location-Based Keys (LBK) to thwart and defend against several attacks, one of which is node clone attack. The identity-based cryptography is used in their protocol such that nodes private keys are bounded by both their identities and locations. Once nodes are deployed, some trusted mobile agents travel around the sensor network and issue the location-based keys to sensor nodes. Since these location-based keys cannot be used in nodes at other locations, the node clone attack is inherently frustrated.

Node capture detection: Yu *et al*. (2007) have used the concept of group deployment and proposed a scheme that is resistance to node capture attack. In their proposed scheme, they estimated the fraction of communication compromised by a node capture attack.

Tague and Poovendran (2008) have modeled a formal node capture attack characterization and it as integer-programming minimization problem. They also proposed strategies for node capture attack based on known heuristics. However, this work did not propose detection schemes against node capture attacks. Tague and Poovendran (2008) proposed a scheme for channel access that is robust against channel jamming by compromised node. They also proposed resilience metrics against node capture attack.

Conti *et al*. (2008), node capture attack detection scheme was proposed in mobile sensor networks. They leverage the intuition that a mobile node is regarded as being captured if it is not contacted by other mobile nodes during a certain period of time. However, this scheme will not work in static sensor networks where sensor nodes do not move after deployment. We use the fact that the physically captured nodes are not present in the network during the period from the captured time to redeployed time. Accordingly, captured nodes would not participate in any network operations during that period. By leveraging this intuition, we detect captured nodes by using the Sequential Probability Ratio Test (SPRT). The main advantage of our scheme is to quickly detect captured nodes with the aid of the SPRT.

Mobility based node capture detection, exploits emergent properties (Conti *et al*., 2008) of mobile sensor networks and given solutions based on the simple observation that if node a will not re-meet node b within a period t secs, than it is possible that node b has been captured. Hence, node a can autonomously know the probability that a “not yet re-met” node has been actually captured by the adversary. Hence, node a can locally decides that a “not yet re-met” node has been actually captured. This approach is called Simple Distributed Detection (SDD), since the attack is detected using only information local to the nodes. The second solution, the

Cooperative Distributed Detection CDD leverages network mobility, as SDD, but unlike SDD it also leverages node cooperation to improve node capture detection. In particular, two nodes a andb exchange information about the nodes (if any) that are tracked by both a and b, that is the nodes in TanTb; CDD considered node-exchange information only when cooperating nodes are in the same communication radius. We expect that node cooperation while requiring more energy consumption (due to the message exchange, absent in SSD), would allow to reduce the number of false positive alarms and consequently reduce the number of false positive revocations.

Kohno *et al*. (2009, 2011) proposed a resilient key distribution method against node capture attacks. In their work they have proved that their proposed method is more resilient against node capture attack in comparison to TinySec. When a WSN runs for a long time using a fixed key on a link, it enhances the probability for the adversaries to decrypt the key by analyzing adequate messages eavesdropped or by capturing some nodes. Under this circumstance, the entire network security might be threatened. Thus, it is necessary to update pair-wise keys periodically. In order to solve this problem, some rekeying protocols, Shen and Guo (2009) proposed a robust pair-wise re-keying protocol for hierarchical WSN to prevent node capture attack. Their proposed re-keying protocol exploits the characteristic of perturbation polynomial to achieve high robustness against node capture attack. Maala *et al*. (2009) have studied the impact of node capture attack on three different key management schemes: Asymmetric Pre-distribution key management (AP scheme), HERO and TLA, that are designed for heterogeneous wireless Sensor Networks (HWSN). They have also analyzed the resistance of these schemes against node capture attack.

Deng *et al*. (2010) proposed two implementations are Distributed FSD and SEFSD to detect node capture. First Stage Detection (FSD) scheme is decentralized approach which uses the successive hello messages to detect node capture which is faster, easier to execute. The detection is based upon the discovery of missing and malfunctions of nodes due to the physical capture. The monitor node the monitor floods a Captured message to the entire network to declare the node ID of the compromised node and deletes its key list in EEPROM. The approach is simple, reliable, energy-efficient, completely local and completely distributed. It detects node capture only, not possible to detect violent attacks like node compromise and node clone and it consumes more energy, causes more overhead. And the second is SEFSD (Sink Enhanced FSD) utilizes the base station to achieve energy efficiency and better security. In SEFSD, the monitor sends the Captured message to the base station only. SEFSD is having the problem of the single point of failure, the point is sink.

Jokhio *et al*. (2012) proposed Sensor Node Capture Attack Detection and Defence (SCADD) uses a simple and lightweight solution against node compromise and node capture attacks. SCADD have strategic-based attack detection to eliminate the possibility of misjudgement and the defence uses a self-destruction defence measure against node capture attack, without actually destroying the node's radio service, to avoid a major security breach. The possibility of misjudging the attack situation cannot be completely eliminated, the protocol performance was analysed with respect to possible misjudgment with respect to the type of node. It is equally important to identify the actual degree of threat to avoid unlawful node destruction.

Node addition prevention: In a node addition attack (Zhu *et al*., 2006) an adversary injects new nodes into a WSN. This differs from a node replication attack because the new nodes are not duplicates of an existing node, but rather are provisioned correctly as unique nodes in the network. To carry out this attack in a network with the pair wise key establishment, an adversary must have access to keying information such as a master key (m_A in our solution).While some solutions allow the master key to exist for a short period of time on individual nodes during the distribution of a new network (Zhang *et al*., 2006; Zhu *et al*., 2006), ours does not allow m_A to exist on any node or base station in the network. Without m_A, an attacker can only gain information such as IDx, Kx: A, IDA, etc. Even with the pairing components (IDx, Kx:A), m_A cannot be deduced, due to the properties of the pairing scheme.

Block the possible re-entry of captured nodes: Defending against nodes replication attacks by limiting the order of deployment and no knowledge deployment locations of nodes (Bekara and Laurent-Maknavicius, 2007) with symmetric polynomial for pairwise key establishment. In this scheme the sensors to be deployed progressively in successive generations and each node belongs to a unique generation. Only newly deployed nodes are able to establish pairwise keys with their neighbors and all nodes in the network know the number of highest deployed generations. Therefore, the clone nodes will fail to establish pairwise keys with their neighbours since the clone nodes were belong to an old deployed generation and block the re-entry of capture nodes.

Song *et al*. (2007) prevention of an attack at re-deployment stage have three phases are power-level tuple recording, guardian node marking and redeployment detection. During the power-level tuple recording, a sensor node sets up power-level tuple and it broadcasts to neighbouring nodes with special packet, called probes, using all possible power levels: from the minimum to the maximum (example: from 1-255 for Mica2 motes). The probe contains three fields: (the node id), p (the current power level that is used for transmission) and p_{data} (the power level that will be used for data communication after joining the network). The guardian nodes for a monitee are self-discovered and marked based on the recorded p_{min} tuples. For a monitee, its guardian nodes are those nodes whose p_{min} is g-level only the guardian nodes (for a monitee) need to monitor the monitee actively.

The reasons are twofold. First, since a monitee sends its regular packets using power level p_{data}, it is the guardian nodes that are capable of and more sensitive on detecting the redeployment than the other neighbors. Since only part of the neighbors are monitoring the monitee. Type I guardian nodes detect the redeployment of a monitee when they can no longer hear from the monitee, whereas Type II guardian nodes detect the redeployment when they receive a message from the monitee with a transmission power level smaller than the recorded p_{min} for it. Particularly, there is a problem with the approach of Song *et al* (2007). The detection procedure specified by Song *et al*. (2007) will not work if only data packets are monitored by guardian nodes, since data packets, are transmitted at fixed radio power level p_{data}. To make their approach work, the redeployed node has to repeat the booting-up process, that is, broadcasting probe messages at multiple power levels. The attacker certainly will not let the redeployed node do so. Doing so is equivalent to announcing "I am a compromised node". It is justifiable only when data packets are transferred at the different radio power level, which will certainly need much more overhead in message control. Most current sensor networks use fixed uniform radio power.

On node controlled link establishment prevention proposed by Tran (2010), the detection process is executed after each deployment of sensor nodes. All predeployed nodes in the neighborhood of a newly deployed node as well as the newly deployed node itself are under suspicion of being controlled nodes. This must be resolved. After having been deployed, a node makes a request for the link establishment by broadcasting its HELLO message locally. Upon receiving the request, each neighbour in the newly deployed node's neighbourhood first verifies the authenticity of the request and then checks the request freshness by comparing whether its counter value equals the one in the request. If the authenticity or freshness is unconfirmed the neighbour will then believe that the requester is a controlled node, otherwise it accepts the request by sending out an authenticated message (location claim) containing its ID and location (e.g., geographic coordinates) in response. Upon hearing this message, each neighbouring node of the neighbour follows the approach in LSM, for example, to detect the controlled nodes. Specifically, each location claim receiver probabilistically forwards the claim to a randomly chosen set of witness nodes. By requiring each node en route to store a copy of this claim, the scheme effectively draws r line segments per location claim across the network. If a conflicting location claim ever traverses the segments, the node at the intersection will detect the conflict and initiate an authenticated revocation broadcast.

Prevention of node replication attack: Khan *et al*. (2013) described the process or stages of node replication concisely describes the initiation of a node replication attack from physical node capture, extraction of secret credentials, reprogram, replicating or cloning and redeployment.A novel approach is proposed in this chapter, which aims to detect node replication attack to block the possible re-entry of captured node either cloned or compromised in the stage of redeployment. Prevention of node replication attack has been done in a distributed fashion, in which any node in the network can be a detector and block the replica.

Once cloned nodes are deployed in the network, the adversary can use them in several malicious ways and create a controlled links to its neighbours. The Main cost for the adversary is to capture and to tamper just one sensor to make many clones out of the tampered sensor which is easy and cheap. Detecting such attack is very challenging, since a clone cannot be easily detected with only local topology knowledge. The Fig. 1 depicts controlled link creation of the replica node with the benign neighbour nodes. Hence proposing of the prevention protocol is important and which will be discussed in detail, in the forthcoming sub sections.

Assumptions: The network may not have powerful base station. Each node should know their neighbours (routing table) and relatively stationary. Each node knows its own locations and all nodes using ID-based public key crypto system. At least one neighbour node should not be compromised. Choosing Witness location instead of ID, since if claim sent to witness ID no longer present in the network. So request sent to a network location to the node closest to this location.

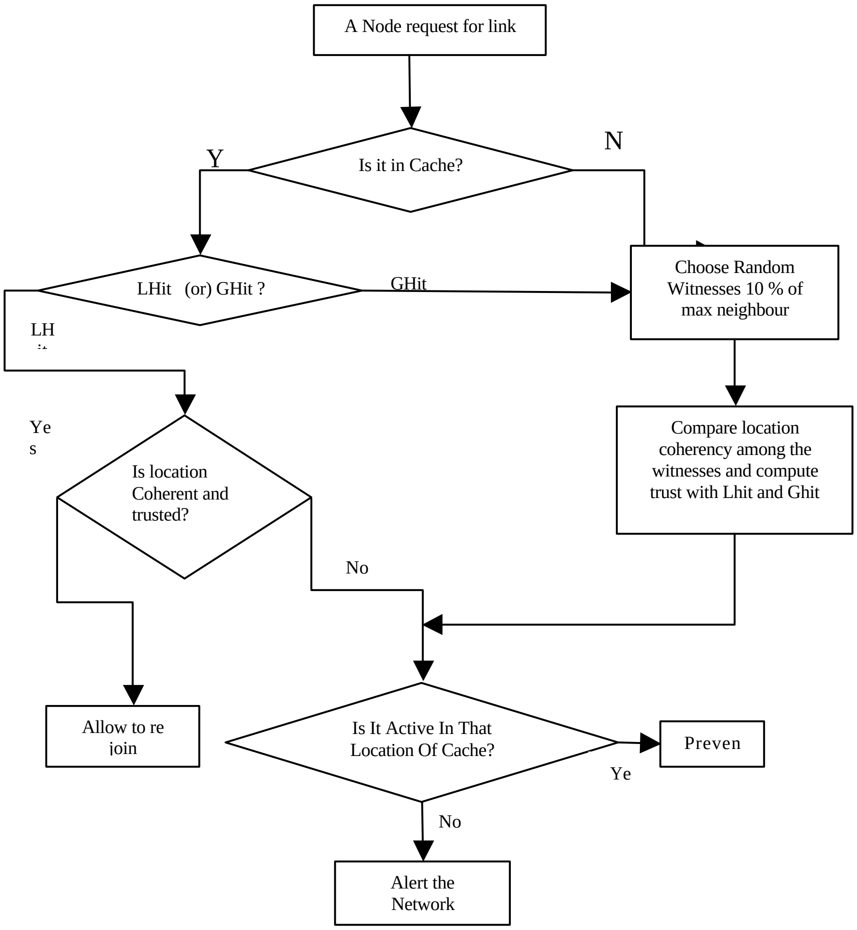
A Node request for link
Is it in Cache?
Y
N
LHit (or) GHit ?
GHit
Choose Random Witnesses 10 % of max neighbour
LH
Compare location coherency among the witnesses and compute trust with Lhit and Ghit
Ye s
Is location Coherent and trusted?
No
Allow to re join
Is It Active In That Location Of Cache?
Preven
Ye
No
Alert the Network

Fig 5: Flowchart for prevention protocol

Prevention Protocol description: WSNs are usually left to operate in an unattended manner for long periods of time, the attack should be detected and precluded at the first attempt, otherwise the attacker would be able to launch other malicious attacks using the compromised nodes and controlled links.

The proposed prevention protocol is a reactive type and which is invoked only when clone starts their communication between neighbours in the form of the link creation. Initially, each incoming node attempts to join the network by broadcasting a HELLO message. After injection of clone and before creating links between the neighbours, node replication attack has been detected during association or re-association of broadcasting a HELLO message. If it is genuine node which will be allowed to rejoin otherwise link is prevented.

Generally for genuineness of node is ensured in the following manner. Each neighbour checks whether any trace of ID about the requesting node is available or not in its own cache or routing table. If any trace of ID is found in the cache, the corresponding neighbour will check the trustworthiness of requesting node and it allows the requesting node to rejoin. Unless, the neighbour node chooses w witnesses and submits its request to w witnesses with a random number about A_1(clone of A) location and behaviour (local hit and global hit). After getting the info with the same random number from those witnesses, neighbour node will check the location coherency and activeness of the requesting node. If the requesting node is active in some other location the corresponding link is prevented . The flow chart for prevention protocol is shown Fig. 5.

Let p_c be the probability of compromising or creating a controlled link and Neighbour degree (d). Choosing w witnesses are doneby Witness selection Factor which is estimated as the product of probability of compromising (p_c) and d. To Check the activeness of a node, cumulative trust is evaluated. This cumulative trust calculated using

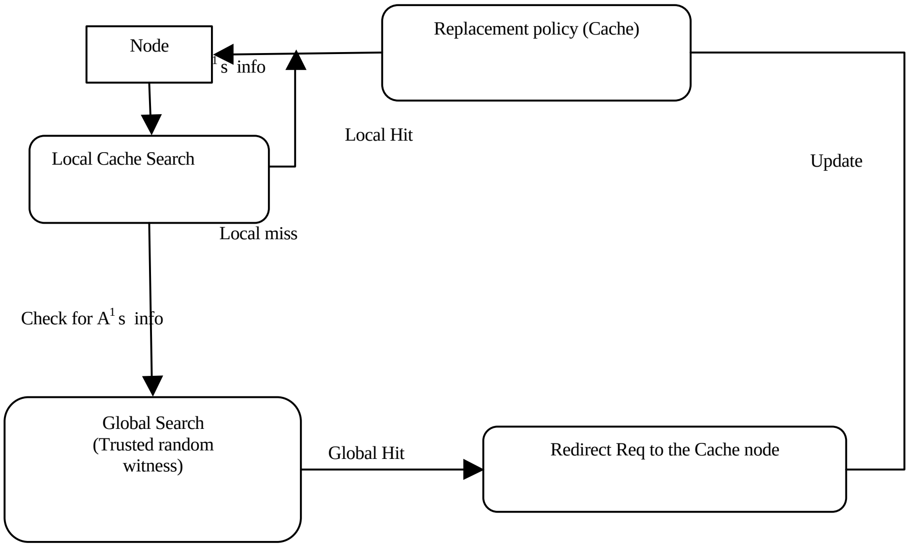
Node
s info
Replacement policy (Cache)
Local Hit
Local Cache Search
Update
Local miss
Check for A¹s info
Global Search (Trusted random witness)
Global Hit
Redirect Req to the Cache node

Fig. 6: Block diagram for cache verification and updating

Equation 1 and it depends on local hit and global hit. Local hit depends on community trust which will be collected from one hop neighbours and updated in the cache. W_1 and W_2 are the weightages given for local hit and global hit, respectively.

$$C_T^K(A)=\frac{1}{k}\sum_1^k\left(W_1 * LH_k * GH_k\right) \qquad (1)$$

Where:

$C_T^K(A)$	=	Cumulative trust of node A
k	=	No. of witnesses
W_1	=	Weightage of local hit
LH	=	Local hit
W_2	=	Weightage of global hit
GH	=	Global hit

Cache verification and updating: Cache verification and updating is part of the proposed prevention protocol. If any node receives a link request which initially searches its cache for information about requesting node whether there is trace as local hit or global hit. Local hit is a one hop neighbour transaction information and global hit is multihop remote node transaction information. This cache detail is used to decide whether the requesting node is replica or not. If the requesting node is replica then cache detail have conflict in location due to the global hit with requesting location. Also, the activeness of the multi-hop remote node is ensured and to confirm the requesting node is replica. In the case of local miss, the random w witnesses are chosen which initiates a global search. The cache will redirect requests if there is no required information is present in global search. Again the next set of random w witnesses is chosen to initiate the next global search. The cache replacement policy is initiated when the cache is filled. The status of the requesting node such as replica or benign and its activeness, witnesses activeness are updated in cache as when the state changes occurred. This is illustrated in the block diagram shown in Fig. 6.

Algorithm for prevention protocol: An algorithm for proposed prevention protocol for node replication attack is given in the following. This protocol is initiated with the input of hello messages from the incoming nodes containing their nodeID info. The output of this algorithm is either alerting the network to avoid replicas or granting permission to join in the network.

Algorithm 1:

```
Input: Hello Msg to create links
Output: Grant permission or not, if not, alert the network
Step 1: if (Link _Request ($A^1$)) then scan (Cache (Nei$_B$ ($A^1$)) //Local Hit
Step 2: if (is Cache (Nei$_B$ ($A^1$)) then
        if (is_ location _coherency (Nei$_B$ , $A^1$ )) then Link(Nei$_B$ , $A^1$) // it matches with current location - Rejoing of node($A^1$ )
        else Choose_ Rand_Trusted_ Witness(w)
        For all_Witness(w ) get_info($A^1$)
        Find_ location_coherency(among(w) and loc($A^1$)) if location_coherent(among) then
        Find Cumulative trust($A^1$)
        Compare with threshold //to decide on trust worthiness on witnesses
        If satisfies grant_link , update cache() else go to step 2- ii(to choose trust worthy witnesses)
        if location_incoherent(among(w) and loc($A^1$)) (alert the network(clone $A^1$ ) //malicious
Step 3:else if ( is ID found in Neighbour's Cache(Nei$_B$ ) ) // Community Hit
        if (is_ location _coherency (Nei$_B$ , $A^1$ )) then Link(Nei$_B$ , $A^1$) // it matches with current location-Rejoing of node($A^1$ )
        else ( id not found ) //mean is not a negibour (B)  (or) New node
        Choose_ Rand_Witness(w)
        For all_Witness(w ) get_info($A^1$)
        Find_location_coherency(among(w))
if  location_coherent(among)  then
        Find  Cumulative trust ($A^1$)
        Compare with threshold        // to decide on the trust worthiness
of witnesses
        If satisfies (alert the network (clone $A^1$), update the cache () else
        if  location_incoherent(among(w) ) go to step 3
```

RESULTS AND DISCUSSION

Simulation results and comparisons: The simulation setup for prevention protocols are: 4000 nodes are randomly deployed within a 1000×1000 m^2. The transmission range is set to 50 m. The trust threshold $Trust_{thres}$ = 75%. Assume, weightage of Consistency factor (C_i), Communication factor (Si) and Battery value are 0.35, 0.35 and 0.30, respectively. Average degree of neighbour 30. The Maximum neighbour degree is 53. Then the size of the cache is twice the maximum neighbour size.

Performance metrics: The various parameters that were estimated during the simulation are as follows:

- True Positive (TP): Equivalent with hit
- True Negative (TN): Equivalent with correct rejection
- False Positive (FP): Equivalent with false alarm, Type I error
- False Negative (FN): Equivalent with miss, Type II error

The Positive Predictive Values (PPV) are the proportions of positive results in statistics and diagnostic tests that are true positive results. The PPV statistic is often called the precisionwill be calculated Equation Precision is how close the measured values are to each other:

$$\text{Precision} = (\text{TP})/(\text{TP}+\text{FP}) \tag{2}$$

Accuracy is defined as how close a measured value is to the actual (true) value. It is calculated using Eq. 3:

$$\text{Accuracy} = (\text{TP}+\text{TN})/(\text{TP}+\text{FP}+\text{FN}+\text{TN}) \tag{3}$$

The false positive rate is the proportion of absent events that yield positive test outcomes, i.e., the conditional probability of a positive test result given an absent event. The false positive rate is equal to thesignificance level. Thespecificityof the test is equal to1minus the false positive rate. Fall-outor False Positive Rate (FPR) also called false alarm rate.

$$\text{Fall}-\text{out} = (\text{FP})/(\text{FP}+\text{TN}) \tag{4}$$

The False Discovery Rate (FDR) is the expected proportion of false positives among all discoveries and given Eq. 5:

$$\text{FDR} = (\text{FP})/(\text{TP}+\text{FP}) = 1-\text{PPV} \tag{5}$$

The prevention rate is the ratio between number of successful detection of replication attack with number of controlled link request and estimated by the Eq. 6:.

$$\text{Prevention rate} = \frac{\text{No.of successfulcontrolledlinkprevention}}{\text{No.of controlledlinkrequest}} \tag{6}$$

Case 1: Normally attacker would not deploy a replica in these areas as a neighbour of clone (or) one hop neighbour of clones. Since, it is easily detected with the aid of local search in this area.

Case 2: If the attacker deployed the clone area other than as a neighbour or one hop neighbour, this will be detected through random g witness selection using global search.

Let p_c be the probability of compromising or creating a controlled link and d is the neighbour degree , then the number of controlled links to be created by the entered cloned in the network is calculated using p_c*d. In general, this research work assumed that any cloned nodes entered into the network will try to compromise at least 10% of its neighbours. Similarly, each node in the deployed network contains average of minimum 30 neighbours (29.95). Hence, entered cloned node will try to compromise and create controlled links with atleast 3 of its neighbours among its total neighbours.

Performance comparison with respect to cloned node: Figure 7 illustrates Precision and False Discovery Rate (FDR) with respect to number of cloned Nodes. When varying the number of cloned nodes from 5-15, precision is increasing and false discovery rate is decreasing, since precision is inversely proportional to FDR.

In this scenario of the proposed prevention protocol, both precision and FDR values are estimated for the various genuine node requests for link creation such as 5 and 10. When the number of cloned nodes is increasing, the information about the respective cloned node requests are proportionately increasing. Hence, the proposed prevention protocol easily detects and prevents the controlled link creation using its algorithm.The simulated results shown in Fig. 7 proves that initially,

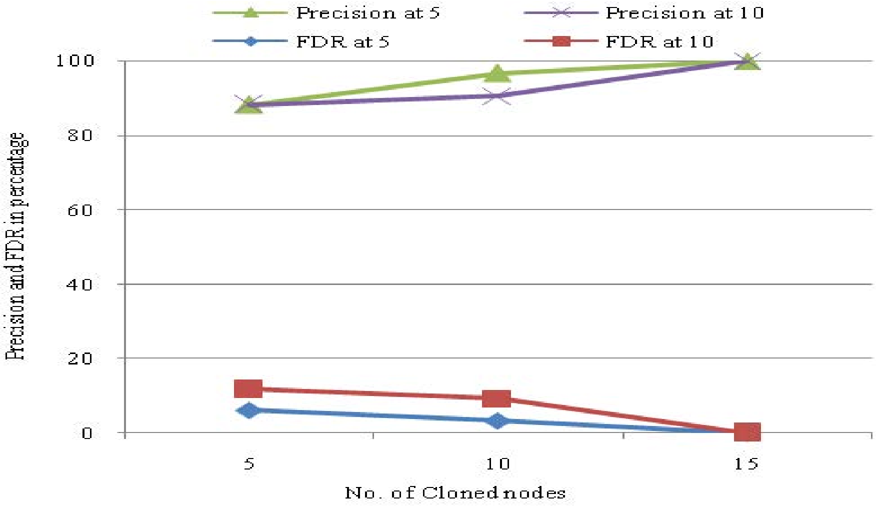

Fig. 7: No. of cloned nodes vs precision and FDR

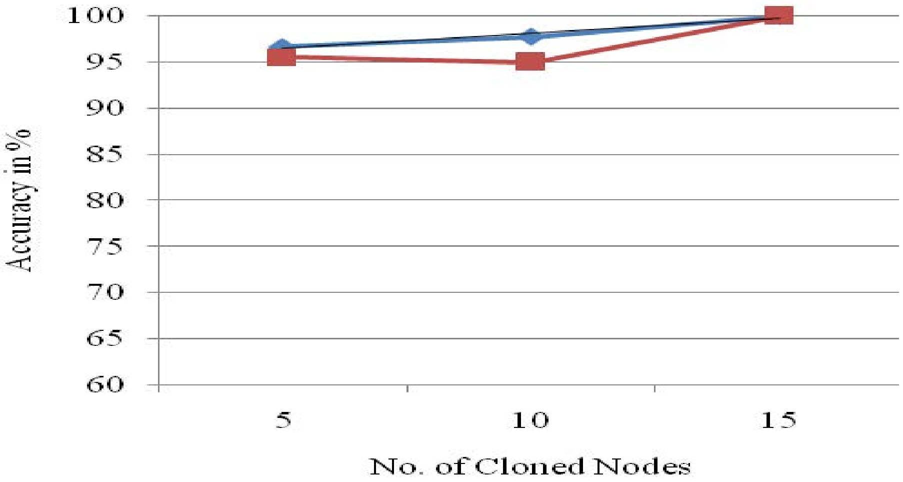

Fig 8: No. of cloned nodes vs accuracy

when the replicas are 5, the FDR is 6.25% and when the replicas are increased to 15, FDR reaches to 0% for the constant genuine nodes of 5.

This illustrates that the proposed prevention protocol, prevents the controlled link creation, though the cloned node increases for the various cases such as keeping constant genuine nodes 5 and 10. Similarly precisions estimated by the proposed prevention protocol are also better in both the cases of keeping constant genuine nodes 5 and 10.

Figure 8 illustrates Accuracy with respect to number of cloned Nodes varying from 5-15. When varying the cloned nodes, Accuracy is increasing, due to the availability of increased information about the respective cloned nodes in the network. The simulation results plotted in the graph ensures that the increased cloned nodes increases the Accuracy of the proposed prevention protocol by preventing the controlled links created by them. In this scenario of the proposed prevention protocol, accuracy values related to controlled link creation is estimated for the various constant genuine node requests for link creation such as 5 and 10. At 5 number of genuine and cloned node scenario, Accuracy of the proposed prevention protocol is 96.67%. When increasing cloned nodes from 5-15 numbers with 5 numbers of genuine nodes, the proposed protocol shows accuracy of 100% due to the availability of more information about the replicas. The similar performance is achieved for 10 numbers of genuine nodes along with varying cloned nodes from 5-15 and it is illustrated in Fig. 8.

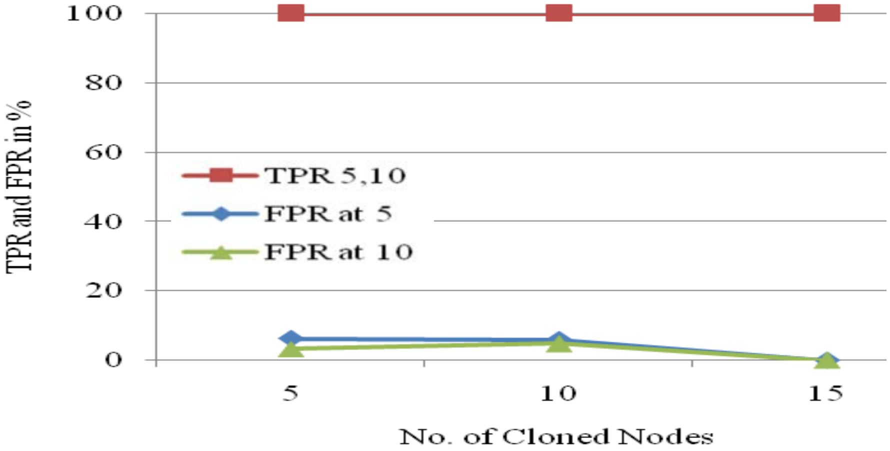

Fig. 9: No. of cloned nodes vs TPR and FPR

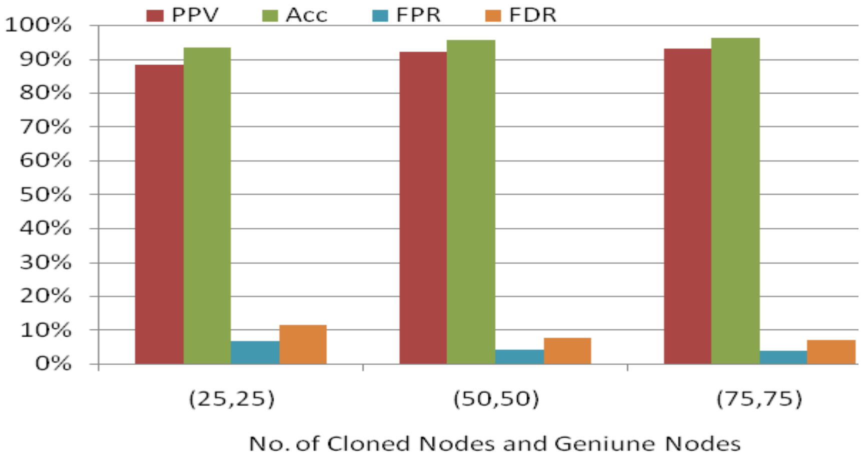

Fig. 10: No. of cloned and genuine nodes vs precision, accuracy, FPR and FDR

True positive rates and false positive rates against a varying number of cloned nodes are shown in Fig. 9. When cloned nodes are increasing from 5-15 numbers, FPR is decreased from 6.25-0% in the case of 5 numbers of genuine nodes and 3.3-0% case of 10 numbers of genuine nodes. As proposed prevention protocol strictly prevents the controlled links created by the cloned nodes, false negatives are always zero. Hence the TPR is 100%, though varying both the cloned nodes and genuine nodes from 5-15.

Performance metrics such as Precision, Accuracy, FPR and FDR with respect to varying cloned and genuine nodes for proposed prevention protocol is shown in Fig. 10. In this scenario instead considering of lower cloned and genuine nodes, which was already discussed, higher number cloned and genuine nodes are considered for performance evaluation of the proposed protocol. In this scenario, it is considered that 25-75 numbers of cloned nodes are trying to create the controlled links in the network. At the same time, same number of genuine nodes (25-75) are also interested to join in the network. From the past experience, though the increased number of cloned nodes increased the amount of information about them in the network, which enabled proposed prevention protocol to prevent the replicas. In the cases of smaller number genuine of nodes, the probability of getting false positives is less or negligible, which cause higher accuracy up to 100%. But this scenario considered, higher number of genuine node are interested to join in the network, which increases false positives by in turn slightly decreases the Accuracy of the proposed protocol for prevention of replicas.

From the simulated results it is inferred that the proposed protocol initially shows 88 and 93% precision and accuracy respectively for 25 numbers of cloned and genuine nodes. When increasing cloned and

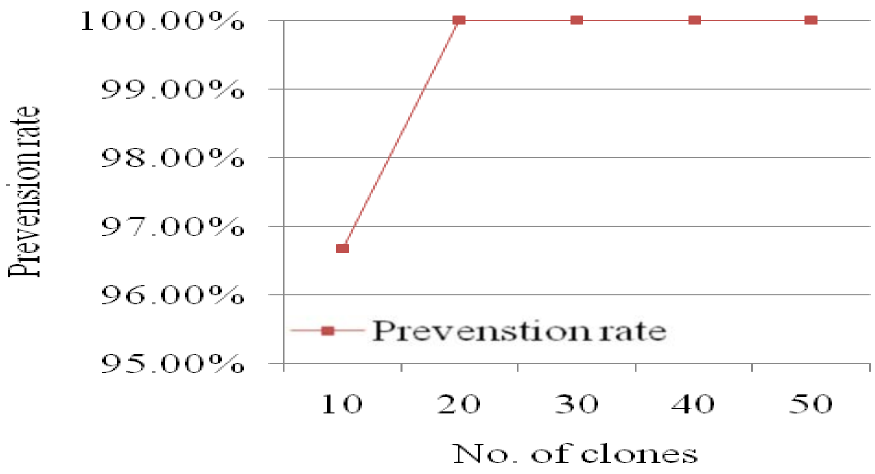

Fig. 11: Prevention rate with respect to number of cloned nodes

Table 1: Estimation prevention rate with varying cloned nodes

No. clone addition	Avg. no. of links to compromise	Avg. no. of witness	Witness chosen	Prevention success	Prevention rate (%)
10	30	90	89	29	96.67
20	60	180	188	60	100
30	90	270	286	91	100
40	120	360	404	125	100
50	150	450	154	154	100

genuine nodes from 25-75 numbers, the precision and accuracy are also increasing to 93% and 96, respectively for the proposed protocol. Though more information about replicas are present in the network, when compared to less number of genuine and cloned node scenario which was already discussed, as false positive increases, the precision and accuracy of the proposed protocol slightly decreases and not reaches to 100%. Similarly, when increasing cloned and genuine nodes from 25-75 numbers FPR and FDR of the proposed protocol are decreasing from 7, 12-4 and 7%, respectively.

Figure 11 shows prevention rate for the proposed prevention protocol when varying the number of cloned nodes from 10-50 irrespective genuine nodes. The information about the cloned node is initially less due to the less number of cloned nodes interested to create controlled links with their neighbours. Hence the prevention rate is 96.67%. When the number of cloned node increases due to the availability about them in the network will increase prevention rate to 100%. Table 1 inferred that the prevention rate of the proposed protocol increases with increasing the number of cloned nodes. Also the Table shows the Average number of links to be compromised, average number of witnesses chosen and prevention success.

CONCLUSION

This study proposed a prevention protocol to prevent the node replication attack when controlled links creation initiated by the any one or many of the cloned nodes in the network. The proposed prevention protocol was a reactive type and it was invoked only when clone starts their communication between neighbours in the form of the link creation. The prevention of the cloned nodes is achieved by tracing their information using local and global searches from the caches of their neighbours in the network. From the simulation results is inferred that proposed protocol has shown improved performance in precision, accuracy and prevention rate when varying number of cloned and benign nodes.

REFERENCES

Alarifi, A. and W. Du, 2006. Diversify sensor nodes to improve resilience against node compromise. Proceedings of the 4th ACM Workshop on Security of Ad Hoc and Sensor Networks, October 30, 2006, Alexandria, USA., pp: 101-112.

Anderson, R. and M. Kuhn, 1996. Tamper resistance-a cautionary note. Proceedings of the 2nd Usenix Workshop on Electronic Commerce, November 18-20, 1996, USENIX, Oakland, California, pp: 1-11.

Baig, Z.A., 2008. Distributed denial of service attack detection in wireless sensor networks. Ph.D Thesis, Monash University, Melbourne, Victoria,

Becher, A., Z. Benenson and M. Dornseif, 2006. Tampering with Motes: Real-World Physical Attacks on Wireless Sensor Networks. In: Security in Pervasive Computing, John, A.C., R.F. Paige, F.A.C. Polack and P.J. Brooke (Eds.). Springer, Berlin, Germany, ISBN:978-3-540-33376-0, pp: 104-118.

Bekara, C. and M. Laurent-Maknavicius, 2007. A new protocol for securing wireless sensor networks against nodes replication attacks. Proceedings of the 3rd IEEE International Conference on Wireless and Mobile Computing, Networking and Communications, October 8-10, 2007, White Plains, NY., pp: 59-59.

Conti, M., D.R. Pietro and A. Spognardi, 2014. Clone wars: Distributed detection of clone attacks in mobile WSNs. J. Comput. Syst. Sci., 80: 654-669.

Conti, M., R.D. Pietro, L.V. Mancini and A. Mei, 2008. Emergent properties: Detection of the node-capture attack in mobile wireless sensor networks. Proceedings of the 1st ACM Conference on Wireless Network Security, March 31-April 2, 2008, Alexandria, VA., USA., pp: 214-219.

Deng, X.M., Y. Xiong and D.P. Chen, 2010. Mobility-assisted detection of the replication attacks in mobile wireless sensor networks. Proceedings of the IEEE 6th International Conference on Wireless and Mobile Computing, Networking and Communications, October 11-13, 2010, Niagara Falls, Canada, pp: 225-232.

Hartung, C., J. Balasalle and R. Han, 2004. Node compromise in sensor networks: The need for secure systems. Technical Report CU-CS-988-04, Department of Computer Science, University of Colorado at Boulder.

Ho, J.W., 2012. Sequential hypothesis testing based approach for replica cluster detection in wireless sensor networks. J. Sensor Actuator Networks, 1: 153-165.

Jabbar, A., 2010. A framework to quantify network resilience and survivability. Ph.D Thesis, University of Kansas, Lawrence, Kansas. https://kuscholarworks.ku.edu/handle/1808/6770

Jokhio, S.H., I.A. Jokhio and A.H. Kemp, 2012. Node capture attack detection and defence in wireless sensor networks. IET. Wirel. Sensor Syst., 2: 161-169.

Khan, W.Z., M.Y. Aalsalem, M.N.B.M. Saad and Y. Xiang, 2013. Detection and mitigation of node replication attacks in wireless sensor networks: A survey. Int. J. Distributed Sensor Networks, Vol. 2013. 10.1155/2013/149023

Kohno, E., T. Ohta and Y. Kakuda, 2009. Secure decentralized data transfer against node capture attacks for wireless sensor networks. Proceedings of the 2009 International Symposium on Autonomous Decentralized Systems, March 23-25, 2009, IEEE, New York, USA., ISBN:978-1-4244-4327-7, pp: 1-6.

Kohno, E., Y. Kakuda and A.I.D.A. Masaki, 2011. Improvement of dependability against node capture attacks for wireless sensor networks. IEICE. Trans. Inf. Syst., 94: 19-26.

Liu, F., J.M. Rivera and X. Cheng, 2006. Location-aware key establishment in wireless sensor networks. Proceedings of the 2006 International Conference on Wireless Communications and Mobile Computing, July 3-6, 2006, ACM, Vancouver, British, ISBN:1-59593-306-9, pp: 21-26.

Maala, B., Y. Challal, H. Bettahar and A. Bouabdallah, 2009. Node capture attack impact on key management schemes for heterogeneous wireless sensor networks. Proceedings of the 2009 Global Symposium on Information Infrastructure, June 23-26, 2009, IEEE, Compiegne, France, ISBN:978-1-4244-4623-0, pp: 1-7.

Mohammadi, S. and H. Jadidoleslamy, 2011. A comparison of link layer attacks on wireless sensor networks. Int. J. Appl. Graph Theory Wirel. Ad Hoc Networks Sensor Networks Graph. HOC., 3: 35-65.

Shen, A.N. and S. Guo, 2009. A robust pair-wise rekeying protocol in hierarchical wireless sensor networks. Proceedings of the 5th International Student Workshop on Emerging Networking Experiments and Technologies, December 1-1, 2009, ACM, Rome, Italy, ISBN:978-1-60558-751-6, pp: 25-26.

Song, H., L. Xie, S. Zhu and G. Cao, 2007. Sensor node compromise detection: The location perspective. Proceedings of the International Conference on Wireless Communications and Mobile Computing, August 12-16, 2007, Honolulu, HI., USA., pp: 242-247.

Tague, P. and R. Poovendran, 2008. Modeling node capture attacks in wireless sensor networks. Proceedings of the 2008 46th Annual Allerton Conference on Communication, Control and Computing, September 23-26, 2008, IEEE, Washington, USA., ISBN:978-1-4244-2925-7, pp: 1221-1224.

Tran, D.T., 2010. Controlled link establishment attacks on distributed sensor networks and countermeasures. Ph.D Thesis, Kyung Hee University, Seoul, South Korea.

Wang, J., Z. Liu, S. Zhang and X. Zhang, 2014. Defending collaborative false data injection attacks in wireless sensor networks. Inform. Sci. Int. J., 254: 39-53.

Yang, G.Z. and M. Yacoub, 2006. Body Sensor Networks. Springer, Berlin, Germany, ISBN:978-1-4471-6373-2,.

Yu, C.M., C.C. Li, C.S. Lu, D.T. Lee and S.Y. Kuo, 2007. Attack probability based deterministic key predistribution mechanism for non-uniform sensor deployment. Proceedings of the 27th International Conference on Distributed Computing Systems Workshops (ICDCSW'07), June 22-29, 2007, IEEE, Taipei, Taiwan, ISBN:0-7695-2838-4, pp: 18-18.

Zhang, Y., W. Liu, W. Lou and Y. Fang, 2006. Location-based compromise-tolerant security mechanisms for wireless sensor networks. IEEE. J. Selected Areas Commun., 24: 247-260.

Zhu, S., S. Setia and S. Jajodia, 2006. LEAP: Efficient security mechanisms for large-scale distributed sensor networks. ACM. Trans. Sensor Networks TOSN., 2: 500-528.