Protocols for detection of node replication attack on wireless sensor network

Mrs: Suvarna Game, Mr. Chandrashekhar Raut
Pursuing M.E. in computer science from Datta Meghe College of Engineering Airoli, Navi Mumbai
Prof. Datta Meghe College Of Engineering Airoli, Navi Mumbai

ABSTRACT: Wireless sensor network has many small sensor nodes that work in collaborative manner to achieve a specific task. But it is deployed in unattended environment and that is why it is prone to attacks. These attacks mainly fall into two categories that is application dependent and application independent. In this paper the focus is on the node replication attack which falls under application independent attacks. In this paper a survey has been done related to node replication attack and existing techniques for solving this issue has been studied. The paper mainly focuses on the types of attacks on wireless sensor network and the two techniques centralized and distributed detection for detection of the node replication attack. Defending against this node replication attack is recently become a research topic in the security of wireless sensor network. The applications and advantages of centralized detection and distributed detection and their respective limitations has been studied.

KEYWORDS: Wireless sensor network, node replication attack, centralized detection, distributed detection, adversary

I. INTRODUCTION

1.1 About Wireless Sensor Network

WSN has many small sensor nodes. These nodes vary from several hundreds to thousands. These sensor nodes work in a collaborative manner to achieve a common goal. Sensor network is used for interaction between computer system and there environment. These autonomous sensors are used to monitor physical or environmental conditions, such as temperature, pressure, sound etc and cooperatively pass their data through the network to a main location. The more modern networks are bi-directional, also enabling control of sensor activity. The development of wireless sensor networks was motivated by military applications such as battlefield surveillance; today such networks are used in many industrial and consumer applications, such as industrial process monitoring and control, machine health monitoring, and so on.

Basic components of WSN nodes are Sensors, memory, processor, GPS, Radio transceiver and power source and major components of WSN are sensor node and base station. Sensor nodes are known as sensing cells and base station as brain of WSN.WSN consists of spatially distributed autonomous sensors to monitor physical or environmental conditions. The sensor node vary in size .It may be as large as shoebox or size of grain of dust.

Fig: Representative Wireless Sensor Network

WSN is of two types that is Stationary and Mobile WSN. In stationary WSN nodes are stationary while in mobile WSN nodes can move and after deployment can interact with physical environment. Depending on the type of WSN attacks can vary.
1.2 Attacks on Wireless Sensor Network

Attacks on stationary WSN can be classified in to three categories Identity attacks, routing attacks and Network intrusion.

![Fig: Types of attacks in WSN](image)

1.2.1 Identity Attack: This is classified into two types as

1.2.1.1 Sybil attack: superficially similar to the node replication attack where one physical sensor node gains an unfair advantage by claiming multiple ids. The Sybil attack is also application-independent and enables one malicious node to multiply its inputs to subvert many protocols like distributed storage, routing, data aggregation, voting/agreement, resource allocation, and so on.

1.2.1.2 Replication attack: also called as node replication attack where one logical node id is reused by multiple physical sensor nodes. The Sybil attack is On a high level, the detection of sensor node replication attacks can be either network-based or not. A typical (but perhaps the only) instance in the latter case is found in Hussain and Rahman (2009), where radio signal strength is utilized at a receiver node to detect node replication we only consider network-based detections. A WSN can be either stationary (which is the prevalent case) or mobile, and replication detection scenarios in stationary and mobile WSNs can be substantially different The detection philosophy for stationary WSNs, on a coarse level, is based on the “exclusiveness” of node location (Boukerche et al., 2007). That is, a sensor node should be related to a unique deployment position; if one logical node id is found to be associated with two or more physical locations, node replication is detected. Clearly, this principle is inapplicable to the emerging mobile WSNs, where sensor nodes may roam in the deployment field all the time

1.2.2 Routing Attacks: In this type of attack a rouge node is placed on a routing path from source to base station. This may attempt to tamper legitimate data packets and this may discard the data packets.

1.2.2.1 Sinkhole attack: In this case adversary creates large sphere of influence because of which all traffic destined for the base station will get attracted from nodes which are several miles away from compromised node.

1.2.2.2 False Routing: This means injecting false routing control packets in to the network.

1.2.2.3 Selective Forwarding: In this case compromised node may refuse to forward the packets or forward only selective packets.

1.2.3 Network Intrusion: It is an unauthorized access to a system. This can be done either an external perpetrator or by an insider having lesser privileges. Among these various attacks in this paper we are focussing on Node replication attack which comes under Identity Attack. The attacks on WSN can be classified as application dependent and application independent attacks. The Node replication attack is application-independent Attack. In this attack an adversary prepares its own sensor node and induces that node in the network in such a way that a network will accept it as a legitimate node. An attacker is able to obtain compromised node’s memory and data within one minute after discovering it.
II. Centralized Detection Technique

1.1 Straightforward scheme: It requires each node to send a list of its neighbours (more specifically, a list of their ids) and the positions claimed by these neighbours (and signed by them, e.g., with an id-based signature scheme) to the base station, which then examines every neighbour list to look for replicated sensor nodes. In a stationary WSN, conflicting position claims for one node id indicates a replication. Once the base station spots one or more replicas, it can revoke the replicated nodes by flooding the network with an authenticated revocation message for broadcast authentication. While conceptually simple, this approach suffers from several drawbacks (Parno et al., 2005) inherent in a centralized system. First, the base station introduces a single point of failure and can become a significant bottleneck. Nodes close to the base station are known as hotspots will receive the huge routing load and thus there power supply get depleted. Thus network connectivity get depleted. Beside this base station hotspots are also target for attacks.

1.2 Set Operations: This is a centralized detection scheme known as SET. This technique attempts to reduce the overhead by computing set operations such as Union, Intersection of exclusive subsets in the network. SET partitions the network logically in to non-overlapping regions called clusters. These clusters are managed by leaders known as cluster heads. These leaders report all the ids of the nodes in the region in the form of subset to base station. The intersection of an two subset of reports is empty that means no replication has occurred otherwise replication is detected. The actual set protocol is complex because of its complicated components. Which contribute to overload. SET protocol has to perform multiple rounds for countering the colluding replicas. One drawback is there for revoking an honest node an adversary can use SET protocol.

1.3 Detecting cloned keys: It is used for addressing cloned cryptographic keys rather than cloned sensor nodes. There is a random key distribution and key of a genuine nod should follow certain pattern. This helps for monitoring the key usage. Key usage refers to number of times key is used for setting the secure connection between neighbouring nodes. In this technique cloned keys are detected by analyzing node authentication statistics those keys whose usage exceeds a certain threshold (determined by the false positive rate) are considered cloned and erased from the network. Now each node will report its preloaded keys o base station. Now it is the task of base station to discover cloned keys. For collecting the key usage data bloom filter is used. Now this detection is effective when i) size of keys pre-distributed to each node is small, (ii) more clones exist in (i.e., are inserted into) the network, and (iii) a high false positive rate is set. Practically any WSN sensor node can communicate with only limited number of its neighbours within finite radius. This number is d known as average node degree or node density. The value of d can be adjusted by selecting appropriate transmission
range. There is one drawback that e.g., how to ensure that the participating clones report their keys honestly (and exactly) to the base station. All the Centralized solution bear the deficiencies such as any compromise of base station will make the solution useless. For another example, even if there are no attacks, the nodes surrounding the base station (i.e., the hotspots) will suffer an undue communication burden that may shorten the life expectancy of the WSN. Because of this distributed solution become necessity.

III. Distributed Detection of Node Replication Attacks

In all four protocols, we assume that nodes know their own geographic positions. We also assume that the nodes in the network remain relatively stationary, at least for the time it takes to perform one round of replication detection. If the network designers anticipate occasional mobility, they can schedule regular detection rounds. As long as a node successfully participates in a round, it can continue to communicate until the next round, even if its position changes in the interim.

Distributed detection of node replication attack can be done using following three techniques. These are:

- Node to network broadcasting
- Witness based strategy
- With deployment knowledge.

2.1 Node-To-Network Broadcasting: N2NB where each node floods the entire network with authenticated broadcast to claim its own location (instead of its neighbours’). Each node stores the location information for its neighbours, incurring a storage cost of O(d). Each node upon receiving a conflicting claim invokes a revocation procedure against the offending nodes, and eventually any replica will be cut off by all its neighbours (thus isolated from the WSN). Each node stores the location information for its neighbours and if it receives a conflicting claim, revokes the offending node. This protocol achieves 100% detection of all duplicate location claims under the assumption that the broadcasts reach every node. This assumption may not hold if the adversary can jam key areas or otherwise interfere with communication paths through the network. The challenge for detecting replication attacks has roots in the resource scarcity of sensor nodes. For stationary WSNs, such detection essentially requires network-wide comparison of location-dependent authentication information, and the limited memory capacity and energy supply places ever constraints on how much authentication information can be stored per node and exchanged in the network (Zhang et al., 2009). Hence it is reasonable to trade the detection rate (e.g., 100% for N2NB) for other major performance criteria like energy efficiency and memory efficiency. Note that the wireless transceiver is the biggest energy consumer for sensor nodes, while the communication cost for N2NB is O(\(n^2\)).

2.2 Witness based strategy has following different schemes:
1. Deterministic multicast (DM)
2. Randomized multicast (RM)
3. Line select multicast (LSM)
4. Randomized efficient distributed detection (RED)
5. Localized multicast (LM)

2.2.1 Deterministic Multicast: DM is based on claimer–reporter–witness framework. Main goal behind the design of DM is reduced communication cost and the main idea is to only send a node’s location claim to a limited set of deterministically chosen nodes serving as witnesses. For improving the communication cost of the N2NB protocol, we describe a detection protocol that only shares a node’s location claim with a limited subset of deterministically chosen “witness” nodes. When a node, referred to as the claimer, locally broadcasts its location claim to its neighbours, each neighbour, serving as a reporter, employs a function to map the claimer id to a witness. Then the neighbour forwards the claim to the witness, which will receive two different location claims for the same node id if the adversary has replicated anode. Now there is a immediate issue the adversary can also employ the function to know about the witness for a given claimer id, and may locate and compromise the witness node before she inserts the replica into the WSN so as to evade the detection. To solve this problem DM employs n instances of a function so that one claimer id is mapped to n different witnesses. Nevertheless, each of the d neighbours does not necessarily need to forward the location claim to each of all g witnesses. If t is assumed that each reporter behaves independently. The well- studied coupon collector’s problem (Cormen et al., 2001) tells us that if the reporters randomly select in all g\(\sum\) xi \(1\ leq i \ leq g\) \(\leq 1\) \(\leq n\) \(g\|\|0:58\) repeatable destinations from all g witnesses for the claimer, then More formally, in this protocol, whenever node \(\gamma\) hears a location claim \(\lambda\) from node \(\alpha\), it computes F(\(\alpha\)) = \{\(\alpha_1, \alpha_2, \ldots, \alpha_g\)\}, where F maps each node ID in the set of possible node IDs, S, to a set of g node IDs: F : S \to \{\sigma : \sigma \in \Sigma, |\sigma| = g\} (1) The nodes with IDs in the set \{\(\alpha_1, \alpha_2, \ldots, \alpha_g\)\} constitute the witnesses for node \(\alpha\). Node \(\gamma\) forwards \(\lambda\) to each of these witnesses. If \(\alpha\)
claims to be at more than one location, the witnesses will receive conflicting location claims, which they can flood through the network, discrediting α. In this protocol, each node in the network stores g location claims on average. For communication, assuming α’s neighbors do not collaborate, we will need each of α’s neighbors to probabilistically decide which of the ois to inform. If each node selects (g ln g) /d random destinations from the set of possible ois, then the coupon collector’s problem [7] assures us that each of the ois’s will receive at least one of the location claims. Assuming an average network path length of O(√n) nodes, this results in O( (g ln g√n/d ) messages. Unfortunately, this cost does not provide much security. Since F is a deterministic function, an adversary can also determine the ois. Thus, they become targets for subversion. If the adversary can capture or jam all g of the messages destined to the ois, then she can create as many replicas of α as she desires (limited only by the requirement that no two replicas share a neighbor). Since the communication costs of this protocol grow as O(g ln g), we cannot afford a large value for g, and yet a small value for g allows the adversary almost unlimited replication abilities after compromising a fixed number of nodes; in other words, if the adversary controls the g witnesses for α, she can create unlimited replicas of α and suppress the conflicting reports arriving at the witness nodes. These disadvantages make this protocol unappealing.

2.2.2 Randomized multicast (RM): RM also follows claimer–reporter–witness framework. But the witnesses become unpredictable for the adversary Therefore, both schemes can be regarded as improvements of the above DM. One major difference between RM and LSM lies in that in the former protocol the reporters randomly select several witnesses, while in the latter protocol nodes forwarding a location claim (i.e., on the path from a reporter to the corresponding witness) also save the claim for inspection, serving as additional witnesses. To improve the resiliency of the deterministic multicast protocol discussed in Section 4.2, we propose a new protocol that randomizes the witnesses for a given node’s location claim, so that the adversary cannot anticipate their identities. When a node announces its location, each of its neighbors sends a copy of the location claim to a set of randomly selected witness nodes. If the adversary replicates a node, then two sets of witnesses will be selected. In a network of n nodes, if each location produces √n witnesses, then the birthday paradox predicts at least one collision with high probability, i.e., at least one witness will receive a pair of conflicting location claims. The two conflicting locations claims form sufficient evidence to revoke the node, so the witness can flood the pair of locations claims through the network, and each node can independently confirm the revocation decision. At a high level, the protocol has each node broadcast its location claim, along with a signature authenticating the claim. Each of the node’s neighbors probabilistically forwards the claim to a randomly selected set of witness nodes. If any witness receives two different location claims for the same node ID, it can revoke the replicated node. The birthday paradox ensures that we detect replication with high probability using a relatively limited number of witnesses. More formally, each node α broadcasts a location claim to its neighbors, β1, β2, . . . , βd. The location claim has the format _IDα, lα, {H(IDα, lα)} where lα represents α’s location (e.g., geographic coordinates(x, y)). Upon hearing this announcement, each neighbor, βi, verifies α’s signature and the plausibility of lα (for example, if each node knows its own position and has some knowledge of the maximum propagation radius of the communication layer, then it can loosely bound α’s set of potential locations). Then, with probability p, each neighbor selects g random locations within the network and uses geographic routing (e.g., GPSR [19]) to forward α’s claim to the nodes closest to the chosen locations (as in GHT [30]). Since we have assumed the nodes are distributed randomly, this should produce a random selection from the nodes in the network. In Section 5.3, we show that the probability of selecting the same node more than once is generally negligible. Collectively, the nodes chosen by the neighbors constitute the witnesses for α. Each witness that receives a location claim, first verifies the signature. Then, it checks the ID against all of the location claims it has received thus far. If it ever receives two different locations claims for the same node ID, then it has detected a node replication attack, and these two location claims serve as evidence to revoke the node. It blacklists α from further communication by immediately flooding the network with the pair of conflicting location claims, lα and lα. Each node receiving this pair can independently verify the signatures and agree with the revocation decision. Thus, the sensor network both detects and defeats the node replication attack in a fully distributed manner. Furthermore, the randomization prevents the adversary from detecting the replications.

Security Analysis Let malicious node α claim to be at L locations, l1,l2, . . ., lL. We would like to determine the probability of a collision using the randomized multicast protocol outlined above, since a collision at a witness corresponds to detection of α’s replication. At each location li, p · d · g nodes randomly select g witnesses. If the neighbors coordinated perfectly, this would store α’s location claim at exactly p · d · g locations. However, since we prefer to have each neighbor act independently, there may be some amount of overlap between the it nesses each neighbor selects. To determine the impact of this overlap, we receive that will receive the location claim assuming the neighbors choose witnesses independently. If Pclaimis the probability that a node hears at least one claim and Pnone is the probability that a node hears no location claims, then we have:
Protocols for detection of node replication attack on wireless sensor network

\[ E[N_{\text{receive}}] = n \cdot P_{\text{claim}} \quad (2) \]
\[ P_{\text{claim}} = 1 - P_{\text{none}} \quad (3) \]

Since each neighbor is assumed to select \( g \) random, unique witness locations, the probability (\( P_f \)) that a
\[ P_f = 1 - \left( 1 - \frac{g}{n} \right)^{p \cdot d} \quad (4) \]

Since each neighbor decides independently whether to send out location claims, the number of nodes that actually send out location claims is distributed binomially, with mean \( p \cdot d \) and variance \( d \cdot (1 - p) \). For a network with \( d = 20 \) and \( p = 0.1 \), the variance will be less than 0.005, so we will approximate the number of neighbors that send out locations claims as \( p \cdot d \). Since the neighbors choose their destinations independently, we have:
\[ P_{\text{none}} = \left( 1 - \frac{g}{n} \right)^{p \cdot d} \quad (5) \]

Combining equations 2, 3 and 5, the number of witness nodes that receive at least one location claim is:
\[ E[N_{\text{receive}}] = n \cdot (1 - \left( 1 - \frac{g}{n} \right)^{p \cdot d}) \quad (6) \]

The Binomial Theorem allows us to approximate \((1 - x)y\) as \((1 - xy)\) for small \( x \), so as long as \( g \leq n \), we have \( N_{\text{receive}} \approx p \cdot d \cdot g \), so overlapping witness locations should not impact the security of the protocol. As an example, in a network with \( n = 10,000 \), \( g = 100 \), \( d = 20 \), and \( p = 0.1 \), perfect coordination would tell 200 nodes, while independent selection would tell 199. Thus, for the remainder of the analysis, we will assume that \( p \cdot d \cdot g \) nodes receive each location claim. If the adversary inserts \( L \) replicas of \( \alpha \), we would like to determine the probability that two conflicting location reports collide at some witness node, since this corresponds to the probability that a witness detects the node replication. Note that even if there are more than two replicas of \( \alpha \), we still only need two location claims to collide in order to completely revoke all \( L \) of the replicas, since one collision will prompt a network-wide flood of the duplicate claims (\( l_i \) and \( l_j \)) and any other node that has heard location claim \( l_k \) for \( k = i, j \) will also revoke \( \alpha \). Following the standard derivation of the birthday paradox [7], the probability \( P_{\text{nc1}} \) that the \( p \cdot d \cdot g \) recipients of claim \( l_1 \) do not receive any of the \( p \cdot d \cdot g \) copies of claim \( l_2 \) are given by:
\[ P_{\text{nc1}} = \left( 1 - \frac{p \cdot d \cdot g}{n} \right)^{p \cdot d \cdot g} \quad (7) \]

Similarly, the probability \( P_{\text{nc2}} \) that the \( 2 \cdot p \cdot d \cdot g \) recipients of claims \( l_1 \) and \( l_2 \) do not receive any of the \( p \cdot d \cdot g \) copies of claim \( l_3 \) is given by:
\[ P_{\text{nc2}} = \left( 1 - \frac{2 \cdot p \cdot d \cdot g}{n} \right)^{p \cdot d \cdot g} \quad (8) \]
\[ P_{\text{nc}} = \prod_{i=1}^{L-1} \left( 1 - \frac{i \cdot p \cdot d \cdot g}{n} \right)^{p \cdot d \cdot g} \]

Thus, the probability \( P_{\text{nc}} \) of no collisions at all is given by:

Using the standard approximation that \((1 + x) \leq e^x\), we have:
\[ P_{\text{nc}} \leq \prod_{i=1}^{L-1} \frac{e^{-1 \cdot p \cdot d \cdot g}}{n} \quad (10) \]
\[ \leq e^{- \frac{p \cdot d \cdot g}{n} \sum_{i=1}^{L-1} i} \quad (11) \]
\[ \leq e^{- \frac{p \cdot d \cdot g}{n} \frac{L(L-1)}{2}} \quad (12) \]

Unlike the deterministic proposal (Section 4.2), we no longer need to worry about the adversary using a limited number of captured nodes to enable an unlimited amount of replication. If the adversary captures neighboring nodes \( \alpha \) and \( \beta \), then the total number of claims about either node is reduced by \( 1d \), essentially reducing the neighbor count of each node by one. Since all of the protocol decisions are made locally by individual nodes, the adversary has only two options remaining: it can disrupt the routing of packets from the remaining legitimate neighbors or it can capture all of the legitimate neighbors. Routing disruptions create tell-tale signs of the adversary’s presence in the network and will be avoided by a prudent adversary.

2.2.3 Line-Selected Multicast

If we interpret RM as a randomized version of DM, LSM can then be regarded as a less expensive version of RM to decrease the communication cost. On a high level, when node \( \alpha \)'s location claim propagates from a reporter to a witness, all the intermediate nodes on the forwarding path also learn about the information,
and can serve as additional witnesses, as shown in Fig. 2. Hence whenever a conflicting location claim by a replica $a_0$ crosses the forwarding path for $a$, the intermediate node at the intersection of the two path scan detect the conflict, i.e., an intersection corresponds to a detection of the replication attack. This idea can be compared to the following geometric theorem: for $x$ randomly drawn lines within a circle, the expected number of intersections is about $0.339x \times 1P$, and thus we only need a few such lines to insure an intersection (e.g., with only $x/43$ random lines we expect 2collisions). Each location claim from a node $a$ is forwarded to $r$ random nodes following the claimer-reporter-witness framework, but the intermediate nodes on the $r$ forwarding paths also save a copy of the claim. Once another location claim is received by a witness (either selected by a reporter, or more likely, an additional one), it floods the network with the unforgivable evidence to exclude both $a$ and $a_0$. One may notice that LSM actually “draws” $r$ line segments (paths) that originate from reporters around a central point (the claimer node) and radiate out in random directions (to $r$ random witnesses), instead of random lines (as in the aforementioned geometric theorem). However, even so, simulations indicate that even if there is only one replica (i.e., $r$ random paths radiate from $a$, another $r$ paths from $a_0$), setting $r \leq 2$ assures that the probability for generating at least one intersection (i.e., the detection rate $P_d$) is above 56%, and $r \geq 5$ leads to 95%. Similar reliability holds for realistic WSN deployment fields with irregular topologies far different from circular domain. Compared with RM, LSM has the network communication costs scaling as $O(\sqrt{n} - n \cdot r)$ and an average node storage cost of $(1/n) \cdot r \cdot \sqrt{n} \cdot n$ location claims scaling as $O(\sqrt{n})$.

2.2.4 Single deterministic cell (SDC) and parallel multiple probabilistic cells (P-MPC)

Zhu et al. (2007) proposed two schemes SDC and P-MPC under the brand “localized multicast”. Essentially both are variants of DM (Section 4.2), and can be parsed as network-wide deterministic multicast, followed by in-cell broadcast and probabilistic storage. In both schemes, the WSN deployment field is considered as a geographic grid of cells, and a location claim from node $a$ is sent by its report, to $g/41$ (SDC) or $g/41$ (P-MPC) cells for in-cell broadcast, the cell id(s)of which is/are deterministically mapped from ID $a$; each node in the destination cell(s) then probabilistically chooses to be a witness by saving the claim. If there is a replica $a_0$, its location claim is sent to the same cell(s) for in-cell broadcast, and thus the witnesses can spot the conflict. One may remark that the concept “localized multicast” advocated in Zhu et al. (2007) is not very exact. Both schemes also bear a similar dilemma with DM (Section 4.2) that if the cell size $s$ is too large, they incur excessive communication cost like $2NB$ (Section 4.1); if $s$ is too small, they degenerate back to DM, and an adversary can defeat both schemes by compromising all nodes in the g deterministic “tiny” cells. Note that in the latter case (a very small $s$), all prospective witnesses in one cell are deployed close to each other with in a geographically limited region instead of sparsely spreading throughout the deployment field, and thus it is easy for an adversary to physically approach and compromise them once for all. Therefore, the practicality of SDC and P-MPC relies on careful selection of $s$. Unfortunately, in Zhu et al. (2007) the critical issue of choosing an appropriate cell size $s$ is over-looked; for all provided examples, $s$ is setto100nodes without any explanation/discussion. In practice, one needs to choose $s$ carefully to find an appropriate trade off between efficiency and security. Another problem omitted in Zhu et al. (2007) is what we term the indistinguishable dilemma. Take SDC for example. Once a location claim by node $a$ arrives at the destination cell, it should be flooded within! the cell so that each node in the cell independently stores the claim (i.e., becomes a witness) with probability $p$. To reduce the in-cell broadcast to overhead, SDC requires that the

2.2.5 Randomized, efficient, and distributed (RED) detection

Conti et al. (2007) proposed a randomized, efficient, and distributed (RED) protocol, which combines both merits of DM (Section 4.2) and RM (Section 4.3.1). The major motivations from the fairness so called “quality” of the detection protocol (Conti et al., 2006): resilience to attack scan be improved by designs that associate individual sensor nodes with equal risk level. For example, a protocol where the likelihood for a genuine node to serve as a witness node (known as the “node appeal”) is independent of the node’s geographical position is more favourable, because such an “area-oblivious” protocol actually associates sensor nodes with almost even responsibility. In RED, each of the $d$ neighbours of a claimer $a$ becomes a reporter with probability $p$, and each reporter sends $a$’s location claim to a set of $g$ pseudo-randomly selected network locations (hence to $g$ witnesses, like RM). The point is that these pseudo-random locations are computed from ID $a$ with $a$-to-$g$ deterministic mapping (like DM), which is seeded with a non received from centralized broadcasting (e.g., from a satellite). Once the random seed is shared network-wide at the beginning of each protocol iteration, the $g$ witnesses are actually deterministic, and the witness set selected by any reporter for $a$ is actually the same. Compared with RM and LSM (Section 4.3), RED’s philosophy lies in “just enough witnesses”, which is inherited from DM. The product $dp$ can be merely very small constant (5 ffiffifi n p ); it is even enough to set $g/41$. Clearly, then ode storage is $dp$ location claims. Importantly, the incurred overhead almost evenly balanced among sensor nodes. The probability that a claimer has no reporter is $(1-p)$, and thus the detection rate is $((1-(1-p)^g))$ assuming there are only two nodes sharing the same id. The pseudo-random
choice of witnesses leads to a uniform witness distribution ("area-oblivious"). On the contrary, in LSM (Parno et al., 2005) a very small central area (for a convex deployment field like a square) may accommodate a large portion of all the witnesses that spot non-coherent location claims, because two forwarding paths are more likely to intersect in the central area; these nodes are just another type of hotspots (recall Section 3.1), and may become appealing target so f attack and/or exhausted quickly. This is termed the crowded center problem in Zhang et al. (2009) to be reviewed in the next subsection. RED (Conti et al., 2007) following Conti et al. (2006) solves this problem justifiably, and an updated version is in Conti et al. (2011), we consider RED as one of the most promising replication detections in the state of the art. Never the less, in Zhang et al. (2009) it is also noted that the infrastructure for distributing RED’s random seed may not always be available. Moreover, since for each protocol iteration the witnesses set for any node is deterministic, there might exist a dilemma in selecting an appropriate g so as to balance between efficiency against node compromise (Zhu et al., 2010).

2.2.6 Memory efficient multicast: B-MEM, BC-MEM, C-MEM, and CC-MEM

Zhang et al. (2009) proposed four replication detection protocols in the name of memory efficient multicast (MEM). The first, B-MEM, is an extension of LSM (Parno et al., 2005), and is the basis of all other three schemes. It reduces the number of stored location claims per node by factor ffiffififi n p through the use of two compact Bloom filters, which are maintained by semi witnesses (known as watchers) and are reset right before each detection round. However, additional memory consumption per node has to be incurred for storing the two filters (essentially “compressed” location claims), and the overall node storage still scales as O ffiffififi n p P (i.e., of the same level with LSM). Moreover, simulations show that B-MEM may lower the detection rate of LSM due to so-called false verifications (Zhang et al., 2009) (essentially the intrinsic false positives of Bloom filters). The second, BC-MEM, employs a technique called cell forwarding to solve the cross over problem that unlike geometric line segments intersecting at a common point, in LSM even when two forwarding paths cross they may not intersect at a common node. One can indeed verify the problem by reconsidering the geographic routing (Section 2.2). The third, C-MEM, employs a technique called cross forwarding to address the aforementioned crowded center problem (Section 4.5) that in LSM random forwarding paths end to pass the central area of the deployment field more frequently, where the nodes suffer far worse over heads. For each claimer, C-MEM first select a random point called the cross point in the network, and forwards the location claim to that point. From there, the claim is then forwarded in four directions, along the horizontal and vertical lines that pass the cross point. Last, CC-MEM integrates cell forwarding and cross forwarding, and thus is a combination of BC-MEM and C-MEM. Simulation results show that the performance of C-MEM is comparable to BC-MEM, because two set so crossing lines have a very high probability to intersect at one or two locations. That is, C-MEM can also mitigate the crossover problem. However, cross forwarding achieves a high probability for intersection only for a convex deployment field, particularly a rectangle (the simulations in Zhang et al., 2009 actually employed a square). For the various irregular topologies considered by LSM such as “thin cross”, “large H”, etc. (Parno et al., 2005), the cross forwarding technique employed by both C-MEM and CC-MEM may work far poorer than in a rectangle; the detection rate may drop drastically. As to cell forwarding in BC-MEM, the basic idea is to divide the deployment field into virtual cells (like SDC and P-MPC Zhu et al., 2007). By employing a pseudo-random mapping similar to RED (Conti et al., 2007) but seeded with the detection round number W.T. Zhu et al. / Journal of Network and Computer Applications 35 (2012) 1022–1034 1030 in each cell an anchor node is assigned for each claimer in the network; one anchor node as a representative of the cell may “serve” different network-wide nodes. To solve the cross over problem, BC-MEM only chooses witnesses from these anchor nodes, which serve as definite intersections for forwarding paths. The price is increased energy expenditure, as any location claim is no longer forwarded along an approximately straight path but a zigzag path. A major problem with BC-MEM is that similar to Ho et al. (2009b), the cell division and anchor node selection ask for highly accurate localization, which may not be affordable for the current generation of WSNs. A less serious problem is that an adversary may circumvent BC-MEM by compromising certain deterministic anchor nodes, assuming the detection only runs for a few rounds. An unaddressed problem is the policy for cell size selection (as also observed in SDC and P-MPC Zhu et al., 2007, recall Section 4.4), which makes fair comparison with other schemes difficult. In all simulations the deployment field is always divided into 100 cells without further explanation/discussion.

2.2.7 Randomly directed exploration (RDE)

In Li and Gong (2009a), a simplified version of N2NB (Section 4.1) known as randomly directed exploration (RDE) is proposed, where a location claim along with the claimer’s neighbour list is forwarded in such a manner that the each of the forwarding paths are approximately a straight line segment. We notice such a “directed” (i.e., oriented) forwarding approach is just a special (yet simple) implementation of geographic routing (Section 2.2) that only works for a convex deployment field (them or regular the better). There al interesting part lies in its motivation: RDE tries to mimic N2NB while suppressing broadcast flood. The

www.iosrjournals.org 8 | Page
underlying idea can be interpreted as follows: if the WSN is small-scale but very densely deployed, a “thin” forwarding path can become a “thick belt” to cover sufficient over hearing nodes. Hence it is plausible to substitute such an “any cast” (as suggested in Li and Gong, 2009a) for broadcast. RDE’s node storage cost remains the same with N2NB (i.e., O(d)), while the network communication overhead is reduced from O(n^2) to O(d· n / √ n ) at the price of decreased detection rate. Note that we add the coefficient d to count for the cost of additionally forwarding claimer’s neighbour list (which is not forwarded in N2NB), whereas this cost is overlooked in the evaluation in Li and Gong (2009a). Actually, the communication reduction from O(n^2) to O(d· n / √ n ) is not very beneficial. Moreover, RDE only seems feasible for an ideal network model, and the detection rate may not be very significant even for a convex deployment field.

2.3 With deployment knowledge scheme can be done using two techniques. These are as follows:
1. Fingerprint with neighborhood: the best of our knowledge, the only approach that achieves real-time detection of clone attacks in WSN was proposed by Xing et al. [7]. In their approach, each sensor computes a fingerprint by incorporating the neighborhood information through a superimposed s-disjunctive code [16]. Each node stores the fingerprint of all neighbors. Whenever a node sends a message, the fingerprint should be included in the message and thus neighbors can verify the fingerprint. The messages sent by clone nodes deployed in other locations will be detected and dropped since the fingerprint does not belong to the same “community”.
2. Group deployment
Group deployment technique is again divided into two strategies
1. Order of deployment without location knowledge.
2. Deployment with location knowledge.

REFERENCES

Protocols for detection of node replication attack on wireless sensor network


Procedures for detection of node replication attack on wireless sensor network
