Hindawi Publishing Corporation
International Journal of Distributed Sensor Networks
Volume 2013, Article ID 149023, 22 pages
http://dx.doi.org/10.1155/2013/149023

Review Article

Detection and Mitigation of Node Replication Attacks in Wireless Sensor Networks: A Survey

Wazir Zada Khan,[1] Mohammed Y. Aalsalem,[2] Mohammed Naufal Bin Mohammed Saad,[1] and Yang Xiang[3]

[1] *Electrical and Electronic Engineering Department, Universiti Teknologi PETRONAS, Bandar Seri Iskandar, 31750 Tronoh, Perak, Malaysia*
[2] *School of Computer Science & Information System, Jazan University, Jazan 45142, Saudi Arabia*
[3] *School of Information Technology, Deakin University, 221 Burwood Highway, Burwood, Melbourne, VIC 3125, Australia*

Correspondence should be addressed to Wazir Zada Khan; wazirzadakhan@yahoo.com

Received 25 January 2013; Accepted 22 March 2013

Academic Editor: Muhammad Khurram Khan

Wireless sensor networks are a collection of a number of tiny, low-cost, and resource-constrained sensor nodes which are commonly not tamper proof. As a result, wireless sensor networks (WSNs) are prone to a wide variety of physical attacks. In this paper, we deem a typical threat known as node replication attack or clone node attack, where an adversary creates its own low-cost sensor nodes called clone nodes and misinforms the network to acknowledge them as legitimate nodes. To instigate this attack, an adversary only needs to physically capture one node, and after collecting all secret credentials (ID, cryptographic keys, etc.), an adversary replicates the sensor node and deploys one or more clones of the compromised node into the network at strategic positions, damaging the whole network by carrying out many internal attacks. Detecting the node replication attack has become an imperative research topic in sensor network security, and designing detection schemes against node replication attack involves different threatening issues and challenges. In this survey, we have classified the existing detection schemes and comprehensively explore various proposals in each category. We will also take a glance at some technical details and comparisons so as to demonstrate limitations of the existent detections as well as effective contributions.

1. Introduction

Advancement in technology has made it possible to develop tiny low-cost sensor nodes with off-the-shelf hardware. A wireless sensor network (WSN), which is a distributed and self-organized network, is a collection of such sensor nodes with limited resources that collaborate in order to achieve a common goal. These sensor nodes are comprised of low-cost hardware components with constraints on battery life, memory size, and computation capabilities [1]. Wireless sensor networks are often deployed in harsh and hostile environments which are inaccessible and even hazardous areas to perform various monitoring tasks. For example, they can be used to monitor factory instrumentation, pollution levels, freeway traffic, and the structural integrity of buildings [2]. Some of the other applications of WSNs include patient monitoring, climate sensing, control in office buildings, and home environmental sensing systems for temperature light, moisture, and motion.

WSNs are viable solutions for a wide variety of real-world challenges; however, a set of new security challenges arise in sensor networks due to the fact that current sensor nodes lack hardware support for tamper-resistance (because it is uneconomical to enclose each node in a tamper resistant hardware) and are often deployed in unattended environments where they are vulnerable to capture and compromise by an adversary. Taking an example of a battlefield, WSNs must tackle the threats and attacks from attackers because these areas are sometimes physically accessible to camouflaged enemies [3] who would like to acquire the private locations of soldiers from or inject wrong commands into the sensor network [4]. Similarly, an unattended WSN can be deployed

in hostile environments which imply the existence of an adversary. For example, WSN can be used to monitor firearm discharge, illicit crop cultivation, drug/weapons smuggling, human trafficking, nuclear emissions in a rogue region and other illegal activities [5]. Thus, it is very important to ensure the security of sensor networks in such scenarios.

The unattended nature of wireless sensor networks can be exploited by adversaries which are able to launch an array of different physical attacks including node replication attack, signal or radio jamming, denial of service (DoS) attack, node outage, eavesdropping, and Sybil attack. and other attacks like sinkhole, wormhole, and selective forwarding attack. Threats to sensor networks can be either layer dependent or layer independent. Attacks in the former category can be application dependant and are specific to different OSI layers targeting specific network functionalities such as routing, node localization, time synchronization, and data aggregation, while the attacks in the latter category are application independent affecting a wide variety of applications from object tracking and fire alarming to battlefield surveillance, and these attacks are not launched on any OSI layer. The attacks of the latter category are also application independent [2]. This attack taxonomy is also shown in Figure 1. In order to protect wireless sensor networks from layer dependent attacks, many schemes have been proposed. To alleviate the effects of routing disruption attacks, secure routing schemes have been proposed [6, 7]. Authentication schemes [8–10] are used to mitigate false data injection attacks. Data aggregation can be secured by using secure data aggregation protocols proposed in [11–14]. To defend localization and time synchronization protocols from different attacks, and threats many protocols have been proposed in [15–21]. Nevertheless, most of these schemes are attack resilient, rather than they can detect and remove the source of attack. Thus, there is a need to detect and revoke the sources of attacks as soon as possible to substantially reduce the costs and damages incurred by employing attack resilient approaches.

In this comprehensive survey, we consider a very severe and important physical attack on WSN which is called node replication attack or clone attack. It is also known as identity attack. In this attack, an adversary first physically captures only one or few of legitimate nodes, then clones or replicates them fabricating those replicas having the same identity (ID) with the captured node, and finally deploys a capricious number of clones throughout the network. This whole process of node replication attack and the various stages are shown in Figure 2. This vexing problem arises from the actuality that sensor nodes are unshielded. It is stated in [22] that an experienced attacker can completely compromise a typical sensor node by using only a few readily available tools, and it can then obtain copies of that node memory and data within 1 min of discovering it. The clones or replicas may even be selectively reprogrammed to subvert the network by launching further insider attacks like falsifying sensor data or suppressing legitimate data, extracting data from the network and disconnect the network by triggering correct execution of node revocation protocols that rely on threshold voting schemes and staging denial of service (DoS) attacks. Clone nodes may create a black hole, initiate a wormhole attack

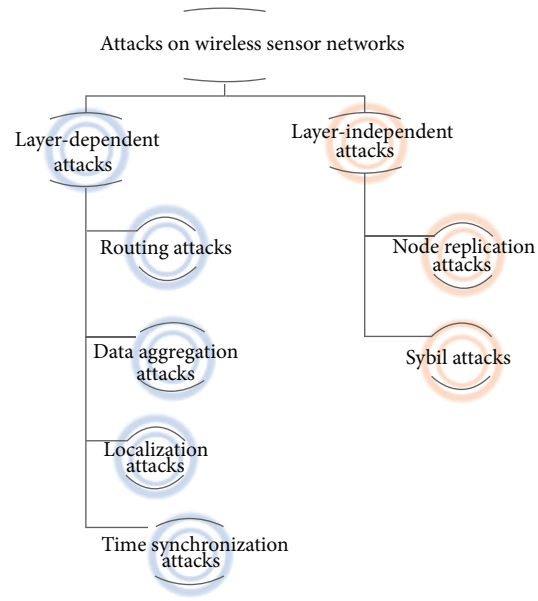

Figure 1: Classification of attacks on wireless sensor networks.

with a collaborating adversary, or may also leak data in an environment in which sensed data must be kept private [23]. If these replicated nodes or clones remain undetected or unattended for a long time, they can further commence the changes in protocol behavior and intrusion into the systems security [24]. It is easy for an adversary to launch such attacks due to the fact that the clones, created by an adversary, have legitimate information (codes, key materials, and credentials), and they may be considered as legitimate nodes and totally honest by its neighbors which are participating in the network operation in the same way as the noncompromised nodes.

The above mentioned traditional security schemes for WSNs are inept to detect and prevent node replication attack. Thus, in the last few years, a number of detection and prevention techniques/schemes have been proposed in the literature. According to [2], the detection schemes are classified on a high level as network-based or radio-based detection. Only one instance of radio-based detection is found in [25]. The former category is further categorized into two types as for mobile WSNs and for stationary WSNs. Both techniques for mobile and stationary WSNs are further divided into two broad categories, namely, centralized and distributed. This can be summarized with Figure 3 which shows a detailed classification of all replica detection schemes. This categorization provides a first step to better understand the node replication detection schemes.

A WSN can be either stationary or mobile. In static wireless sensor networks (SWSNs), the sensor nodes are stationary or static; that is, the sensor nodes are deployed randomly, and after deployment their positions do not change. On the other hand, in mobile wireless sensor networks (MWSNs), the sensor nodes can move on their own, and after deployment, they can interact with the physical environment by controlling their own movement. Advances in robotics have made it possible to develop such mobile sensors which

Figure 2: Steps of node replication attack.

are autonomous and have the ability to sense, compute, and communicate like static sensors. The prime difference between static and mobile WSNs is that mobile nodes are able to reposition and organize themselves in the network, and after initial deployment, the nodes spread out to gather information [26, 27]. Mobile nodes can communicate with one another when they are within the range of each other, and only then they can exchange their information gathered by them. Another important difference is that in static WSNs fixed routing or flooding is used for data distribution, while in mobile WSNs dynamic routing is used. As static and mobile WSNs differ in their characteristics hence replication detection schemes for stationary and mobile WSNs will be substantially different. In a static or stationary WSN, a sensor node has a unique deployment position, and thus if one logical node ID is found to be associated with two or more physical locations, node replication is detected. But this is inapplicable to mobile WSNs where sensor nodes keep roaming in the deployment field. So, replication detection in such mobile WSN involves different scenarios and techniques.

For mobile WSNs, both centralized and distributed techniques have been proposed in the literature. In the case of stationary WSNs, centralized techniques are further categorized into five types, namely, straightforward base station-based technique, key usage-based technique, SET operations techniques, cluster head-based techniques and neighborhood social signature-based techniques. The distributed techniques for stationary WSNs are further divided into four types naming Node to Network Broadcasting, claimer-reporter-witness-based techniques, neighbor-based and generation- or group-based techniques. On the other hand, mobile centralized detection techniques are further divided into two types including key usage-based and node speed-based techniques. The mobile distributed detection techniques are divided into three main types, namely, node meeting-based, mobility-assisted-based, and information-exchange-based techniques. This inclusive categorization can be summarized with Figure 3 which provides a first step in better understanding node replication detection schemes.

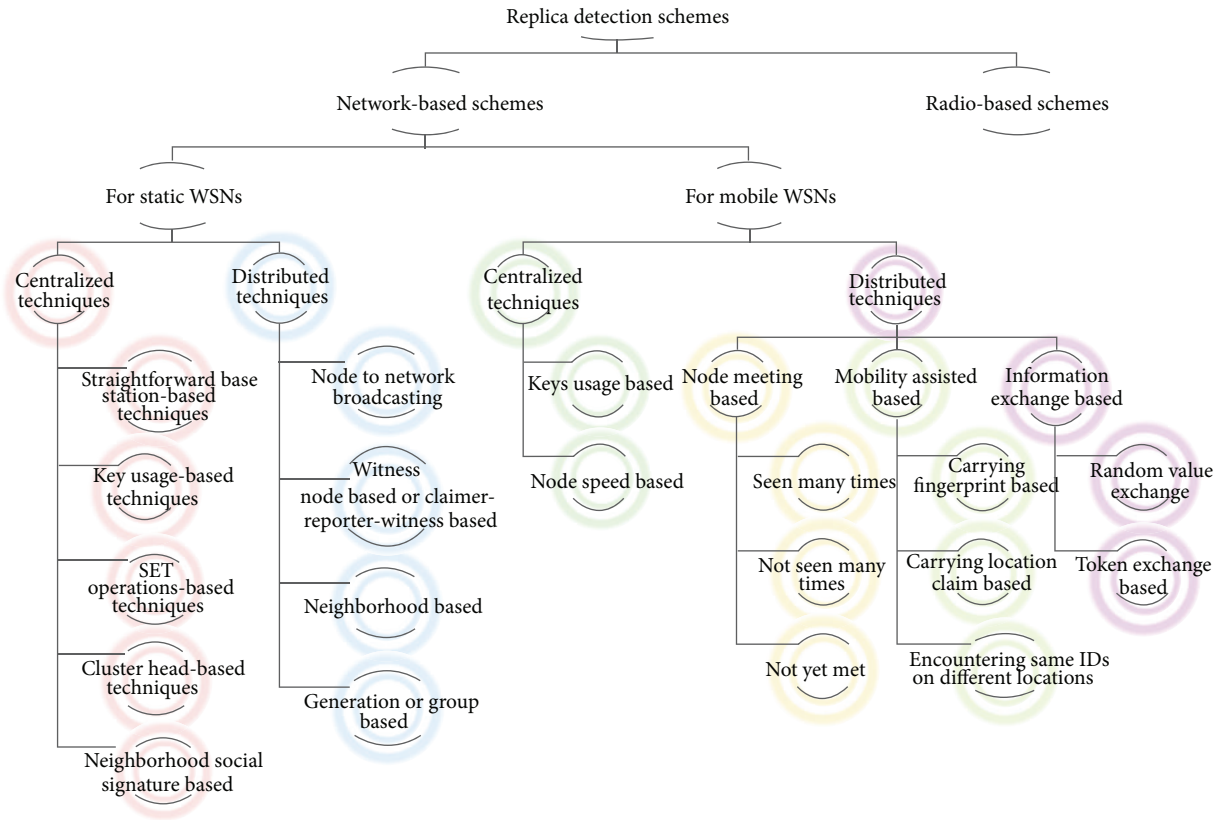

Figure 3: Taxonomy of replica detection schemes.

1.1. Motivation. With the rapid use of vast technologies in WSNs, the threats and attacks to WSN are escalating and are also being diversified and deliberate. A typical threat called node replication attack is a very severe and niggling problem in which an adversary replicates a sensor node after physically capturing it and then uses these replicas to disrupt the network operations by redeploying them at strategic positions of the network. Thus the research related to node replication attack in WSNs has been followed with much interest in recent years. The research of authentication and security techniques is already quite mature but such solutions fail to detect node replication attack and thus no longer provide WSN with adequate security from this attack. Furthermore, the detection of node replication attack in mobile WSN is far different and more challenging than in static WSNs.

The development of replica/clone detection techniques suitable for static WSNs and mobile WSNs is therefore regarded as an essential research area which will make WSN (either static or mobile) to be more secure and reliable. Most recently, Zhu et al. [2] did a survey on the countermeasures of node replication attack which has pointed out some valuable technical weaknesses and advantages of some of the techniques, but latest progress of replica detection schemes is absent, and it also lacks the detailed analysis of all existing techniques for mobile WSNs.

This motivates us to present our paper as a complete guideline of replica/clone detection schemes both for static and mobile WSNs. Moreover, in this paper we have identified the advantages and shortcomings of all the techniques/schemes. Finally, some variations of node replication attack are also identified and discussed. This paper is helpful in understanding all the replica detection schemes developed so far, and it can assist the researchers and developers in the development of new, robust, and effective detection schemes.

1.2. General Adversary Model. Conventionally, some assumptions are made about an adversary in order to scrutinize security of a sensor network. First of all, an adversary is a smart and powerful attacker who can launch a clone attack [4], and it has the ability to secretly capture a limited number of legitimate sensor nodes [28]. Secondly, an adversary can create replicas by using cryptographic information which is obtained from the compromised node. An adversary has also full control over the compromised and replicated nodes and can communicate with them at any time. Thirdly, the main goal of an adversary is to protect its replicas from being detected by the detection protocol used in the network because if any replicas are detected, besides starting a revoke process to revoke replicas, the network may start a sweeping process to sweep out [29] the compromised node and may also draw human intervention. Thus, it is mostly assumed that nodes controlled by an adversary still follow the replica detection protocol as an adversary always wants to be overlooked. Fourthly, an adversary is so powerful that it is able to subvert

the nodes that will possibly act as witnesses. To cope with such an adversary, it could be possible to assume that nodes are tamper-proof. But as tamper proof hardware is expensive and energy demanding, a large part of the literature has assumed that nodes in the network are not tamper resistant.

In case of mobile WSNs, the method of attack is the same but difference is that an adversary is mobile. The scenario of mobile WSN is that the sensors are unable to transmit sensed data at their will because the sink is not always present. Thus, the data accumulated in their memories become targets of many adversaries. In [30], a mobile adversary model is proposed in which mobile adversary visits and travels around the network trying to compromise a subset of sensors within the time interval when sinks are not present in the network. The time taken by a mobile adversary to compromise a set of sensors is much shorter than the time between two successive data collections of a sink.

1.3. Node Replication Attack and Its Effects on the Security Goals of WSNs. High level security issues are basically identical to the security requirements of both static and mobile WSNs. Thus, when dealing with security of WSNs, one is faced with achieving some of the following common security goals including availability, authenticity, confidentiality, and data integrity. When node replication attack is launched by an adversary, all of these security goals are affected severely because of two reasons. First, if any proper, specific, and efficient detection scheme is not used to identify and revoke these replicas because the existing general purpose security protocols would allow the replica nodes to encrypt, decrypt, and authenticate all of their communications as if they were original captured nodes. Second, when the detection probability of the detection technique used is very low to detect these clones or replicas. Node replication attack is significantly harmful to the networks because these replicas or clones have legitimate keys, and they are recognized as legitimate members of the network, since they carry all cryptographic materials extracted from the captured nodes so that an adversary can use them to mount a variety of insider attacks [2]; for example, it can monitor all the information passing through the nodes or monitor significant fraction of the network traffic that passes through the nodes, falsify sensor data, launch denial of service (DoS) attack, extract data from the network, inject false data to corrupt the sensor's monitoring operation, subvert data aggregation, and jam legitimate signals and can also cause continual disruption to network operations by undermining common network protocols.

Availability ensures the survivability of network services despite attacks [31]. In case of node replication attack, an adversary is able to compromise the availability of WSN by launching a denial of service (DoS) attack, which can severely hinder the network's ability to continue its processing. By jamming legitimate signals, the availability of the network assets to authorized parties is also affected.

Authenticity is a security goal that enables a node to ensure the identity of the sensor node it is communicating with. In case of node replication attack, an adversary creates clone nodes which are seemingly legitimate ones (identical to the original captured node) as they have all the secret credentials of the captured node; thus, it is difficult for any node to differentiate between a clone node and the original or legitimate node. Also the existing authentication techniques cannot detect clone nodes as they all hold legitimate keys. This is how the authenticity of the network is affected.

Confidentiality is the assurance that sensitive data is being accessed and viewed only by those who are authorized to see it. But when node replication attack is launched, confidentiality of data is not assured as clone nodes are the duplicated nodes of the compromised ones, and thus they behave like original compromised nodes. These clone nodes can have all the data that contains trade secrets for commercial business, secret classified government information, or private medical or financial records, and thus by misusing such sensitive data, it can damage the network or organization, person, and governmental body.

Data integrity ensures that the contents of data or correspondences are preserved and remain unharmed during the transmission from sender to receiver. Integrity represents that there is a guarantee that a message sent is the message received meaning that it was not altered either intentionally or unintentionally during transmission. But in case of node replication attack, an adversary can falsify sensor data or can inject false data to corrupt the sensitive data and thus subverting the data aggregation using the replicated or clone nodes.

1.4. Evaluation Metrics for Replication Detection Techniques. For the performance analysis and evaluation of replica detection protocols, four vital evaluation metrics are mostly used by all the detection schemes. These are communication overhead, storage or memory overhead, detection probability and detection time [26].

Communication overhead is defined as the average number of messages sent by a sensor node while propagating the location claims. *Storage overhead* defines the average number of the location claims stored in a sensor node. *Detection probability* is an important evaluation metric which shows how accurately a protocol can identify and detect the clones or replicas. The *detection time* is simply the delay between actual replica node deployment and detection.

To make the current survey more comprehensive and detailed, here in Section 2 we have discussed all the existing schemes for the replica detection in stationary WSNs which are accordingly compared in Section 3. Section 4 describes all the replication detection schemes in mobile WSNs proposed so far in the literature which are then compared in Section 5. In Section 6, we have highlighted some important issues and challenges associated with the node replication attack in both static and mobile WSNs. Finally, Section 7 concludes the paper.

2. Detection Techniques for Stationary WSNs

Many techniques have been proposed for the detection of node replication attack in static WSNs which are categorized mainly into two types as centralized and distributed techniques.

2.1. Centralized Techniques. In centralized techniques base station is considered to be a powerful central which is responsible for information convergence and decision making. During the detection process every node in the network sends its location claim (ID, Location Info) to base station (sink node) through its neighboring nodes. Upon receiving the entire location claims, the base station checks the node IDs along their location, and if it finds two different locations with the same ID, it raises a clone node alarm.

2.1.1. On the Detection of Clones in Sensor Networks Using Random Key Predistribution. This technique falls into the category of key usage based techniques. Brooks et al. [32] have proposed a cloned key detection protocol in the context of random key predistribution [33]. The basic idea is that the keys employed according to the random key predistribution scheme should follow a certain pattern, and those keys whose usage exceeds a threshold can be judged to be cloned. In the protocol, counting Bloom filters is used to collect key usage statistics. Each node makes a counting Bloom filter of the keys it uses to communicate with neighboring nodes. It appends a random number (nonce) to the Bloom filter and encrypts the result using base station public key; this encrypted data structure is forwarded to base station. Base station decrypts the Bloom filters it receives, discards duplicates, and counts the number of time each key used in the network. Keys used above a threshold value are considered cloned. Base station makes a bloom filter from the cloned keys, encrypts the list using its secret key and broadcasts this filter to the sensor network using a gossip protocol. Each node decrypts base stations bloom filter removes cloned keys from its keying, and terminates connections using cloned keys.

2.1.2. SET: Detecting Node Clones in Sensor Networks. This technique falls into the category of base station-based techniques. Choi et al. [23] have proposed a clone detection approach in sensor networks called SET. In SET, the network is randomly divided into exclusive subsets. Each of the subsets has a subset leader, and members are one hop away from their subset leader. Multiple roots are randomly decided to construct multiple subtrees, and each subset is a node of the subtree. Each subset leader collects member information and forwards it to the root of the subtree. The intersection operation is performed on each root of the subtree to detect replicated nodes. If the intersection of all subsets of a subtree is empty, there are no clone nodes in this subtree. In the final stage, each root forwards its report to the base station (BS). The BS detects the clone nodes by computing the intersection of any two received subtrees. SET detects clone nodes by sending node information to the BS from subset leader to the root node of a randomly constructed subtree and then to the BS.

2.1.3. Real-Time Detection of Clone Attacks in Wireless Sensor Networks. This technique falls into the category of neighborhood social signature-based techniques. Xing et al. [34] have proposed real-time detection of clone attacks in WSN. In their approach, each sensor computes a fingerprint by incorporating the neighborhood information through a superimposed s-disjunct code [35]. 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." The motivation behind their scheme for detection of clone attacks is exploring the social characteristics of each sensor. Once they are deployed, these sensors reside within a fixed neighborhood. The sensor and its neighborhood form a small "community," or a "social network." A cloned sensor can have the same legitimate credentials (ID, keys, etc.) as the original node, but cannot have the same community neighborhood. Thus, each sensor can be distinguishably characterized by its social community network. In a small community, a newcomer can be easily recognized if speaking with a different accent. Similarly, a clone node can be easily identified by its neighbors if carrying a "social signature" belonging to a different community.

2.1.4. Hierarchical Node Replication Attacks Detection in Wireless Sensor Networks. This technique falls into the category of cluster head-based techniques. Znaidi et al. [36] have proposed a cluster head selection-based hierarchical distributed algorithm for detecting node replication attacks using a Bloom filter mechanism including the network reactions. More precisely, the algorithm relies on a cluster head selection performed using the local negotiated clustering algorithm (LNCA) protocol [37]. Each cluster head exchanges the member node Ids through a Bloom filter with the other cluster heads to detect eventual node replications. The algorithm works in three steps. In the first step all the material required for Bloom filter computations and for cryptographic operations that will be performed in the network predistributed in each sensor node. The second step performs the cluster head election. In the third step, Bloom filter construction is performed by each cluster head, and the Bloom filter verification is performed by the other cluster heads.

2.1.5. CSI: Compressed Sensing-Based Clone Identification in Sensor Networks. This technique falls into the category of base station-based techniques. Yu et al. [38] have proposed a centralized technique called compressed sensing-based clone identification (CSI) for static wireless sensor networks. The basic idea behind CSI is that each node broadcasts a fixed sensed data (α) to its one hop neighbors. Sensor nodes forward and aggregate the received numbers from descendant nodes along the aggregation tree via compressed sensing-based data gathering techniques. Base station (BS), as the root of the aggregation tree, receives the aggregated result and recovers the sensed data of the network. According to the reconstructed result, the node with the sensory reading greater than α is the clone since a nonclone node can only report the number once.

2.2. Distributed Techniques. In distributed techniques, no central authority exists, and special detection mechanism called claimer-reporter-witness is provided in which the detection is performed by locally distributed node sending

the location claim not to the base station (sink) but to a randomly selected node called witness node. Distributed techniques are classified into four types and these are described below.

2.2.1. Node-to-Network Broadcasting (N2NB) and Deterministic Multicast (DM). This technique falls into the category of node-to-network broadcasting. The N2NB and DM protocols are two unappealing examples proposed by Parno et al. [28]. Both of protocols received relatively less attention. In N2NB, each node floods the entire network with authenticated broadcast to claim its own location (instead of its neighbors). Each node stores the location information for its neighbors, 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 neighbors (thus isolated from the WSN). The N2NB protocol achieves 100% detection rate as long as the broadcast reaches every node if the network size is assumed to be n and certain duplicate suppression algorithm is employed so that each node only broadcasts a given message once.

The DM protocol is a good example to illustrate the claimer-reporter-witness framework. The claimer is a node which locally broadcasts its location claim to its neighbors, each neighbor serving as a reporter, and employs a function to map the claimer ID to a witness. Then the neighbor forwards the claim to the witness, which will receive two different location claims for the same node ID if the adversary has replicated a node. One problem can occur that 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 the adversary inserts the replicas into the WSN so as to evade the detection.

2.2.2. Distributed Detection of Node Replication Attacks in Sensor Networks. Both RM and LSM fall into the category of witness node-based techniques. Parno et al. [28] have introduced two more distributed algorithms for the detection of clone nodes in wireless sensor networks which are quite mature schemes as compared to DM. The first protocol is called randomized multicast (RM) which distributes location claims to a randomly selected set of witness nodes. The birthday paradox [39] predicts that a collision will occur with high probability if the adversary attempts to replicate a node. Their second protocol, line-selected multicast (LSM), exploits the routing topology of the network to select witnesses for a node location and utilizes geometric probability to detect replicated nodes.

In RM, each node broadcasts a location claim to its one-hop neighbors. Then, each neighbor selects randomly witness nodes within its communication range and forwards the location claim with a probability to the nodes closest to chosen locations by using geographic routing. At least one witness node is likely to receive conflicting location claims according to birthday paradox when replicated nodes exist in the network. In LSM, the main objective is to reduce the communication costs and increase the probability of detection. Besides storing location claims in randomly selected witness nodes, the intermediate nodes for forwarding location claims can also be witness nodes. This seems like randomly drawing a line across the network, and the intersection of two lines becomes the evidence node of receiving conflicting location claims.

2.2.3. A New Protocol for Securing Wireless Sensor Networks against Node Replication Attacks. This technique falls into the category of generation- or group-based techniques. Bekara and Laurent-Maknavicius [40, 41] have proposed a new protocol for securing WSN against node replication attack by limiting the order of deployment using symmetric polynomial for pair-wise key establishment and defined group-based deployment model. Their scheme requires sensors to be deployed progressively in successive generations (or group). Each node belongs to a unique generation. In their scheme, only newly deployed nodes are able to establish pairwise keys with their neighbors, and all nodes in the network know the number of the highest deployed generation. Therefore, the clone nodes will fail to establish pair-wise keys with their neighbors since the clone nodes belong to an old deployed generation.

2.2.4. A Randomized, Efficient, and Distributed Protocol for the Detection of Node Replication Attacks in Wireless Sensor Networks. This technique falls into the category of witness node-based techniques. Conti et al. have proposed a randomized, efficient, and distributed protocol called RED [42, 43] for the detection of node replication attack. It is executed at fixed intervals of time and consists in two steps. In first step, a random value, $rand$, is shared between all the nodes through base station. The second step is called detection phase. In the detection phase, each node broadcasts its claim (ID and location) to its neighboring nodes. Each neighbor node that hears a claim sends (with probability p) this claim to a set of g pseudorandomly selected network locations. The pseudo random function takes as an input ID, random number, 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. When next time the RED executes, the witness nodes will be different since the random value which is broadcasted by the BS is changed.

2.2.5. Efficient Distributed Detection of Node Replication Attacks in Sensor Networks. These techniques falls into the category of witness node-based techniques. Zhu et al. [44, 45] have proposed two distributed protocols for detecting node replication attacks called single deterministic cell (SDC) and parallel multiple probabilistic cells (P-MPC). In both protocols, the whole sensor network is divided into cells to form a geographic grid. In SDC, each node ID is uniquely mapped to one of the cells in the grid. When executing detection procedure, each node broadcasts a location claim to its neighbors. Then, each neighbor forwards the location claim with a probability to a unique cell by executing a geographic hash function [46] with the input of node ID. Once any node in the destination cell receives the location claim, it floods the location claim to the entire cell. Each node in the

destination cell stores the location claim with a probability. Therefore, the clone nodes will be detected with a certain probability since the location claims of clone nodes will be forwarded to the same cell. Like SDC, in the P-MPC scheme, a geographic hash function [46] is employed to map node identity to the destination cells. However, instead of mapping to single deterministic cell, in P-MPC the location claim is mapped and forwarded to multiple deterministic cells with various probabilities. The rest of the procedure is similar to SDC.

2.2.6. (Space-Time)-Related Pairwise Key Predistribution Scheme for Wireless Sensor Networks. This technique falls into the category of base station-based techniques. Fei et al. [47] have proposed a polynomial based space-time-related pairwise key predistribution scheme (PSPP-PKPS, for short PSPP) for wireless sensor networks, which relates the keying material of a node with its deployment time and location. In PSPP, the keying material of a node can only work at its initial deployment location. If a node leaves its deployment location, its keying material will become invalid. By using this idea, their scheme provides resistance against the clone attack.

2.2.7. A Neighbor-Based Detection Scheme for Wireless Sensor Networks against Node Replication Attacks. This technique falls into the category of neighborhood-based techniques. Ko et al. [48] have proposed a real time neighbor-based detection scheme (NBDS) for node replication attack in wireless sensor networks. The main idea of their scheme is that when a person moves to another community, he will meet new neighbors and tell his new neighbors where he comes from through chatting. But new neighbors will not check if he lies or not. However, if some of his new neighbors ask his previous neighbors whether this newcomer really comes from the community that he claims, the identity of the newcomer can be implicitly verified. If previous neighbors say that this person still lives in the original neighborhood, the newcomer can be detected as a replica. This observation motivates their research on node replication attacks, and replicas are detected in the same way.

2.2.8. Distributed Detection of Node Capture Attacks in Wireless Sensor Networks. This technique falls into the category of base station-based techniques. Ho [49] has proposed a node capture detection scheme for wireless sensor networks. Their scheme detects the captured sensor nodes by using the sequential analysis. They use the fact that the physically captured nodes are not present in the network during the period from the captured time to the redeployment time. Accordingly, captured nodes would not participate in any network operations during that period. By leveraging this intuition, the captured nodes can be detected by using the sequential probability ratio test (SPRT) [50]. The protocol first measures the absence time period of a sensor node and then compares it to a predefined threshold. If it is more than threshold value, the sensor node is considered as a captured node. The efficient node capture detection capability depends on a properly configured threshold value.

2.2.9. Memory Efficient Protocols for Detecting Node Replication Attacks in Wireless Sensor Networks. These techniques fall into the category of witness node-based techniques. Zhang et al. [3] have proposed four memory efficient multicast protocols for replication detection, namely, memory efficient multicast with Bloom filters (B-MEM), memory efficient multicast with Bloom filters and Cell Forwarding (BC-MEM), memory efficient multicast with cross forwarding (C-MEM), and memory efficient multicast with cross and cell forwarding (CC-MEM). The first protocol B-MEM use Bloom filters to compress the information stored at the sensors and the location claim Cα of a node α is multicast via its neighbors to a number of randomly selected locations in the network. Each neighbor β has a probability p to participate in the multicast. If it does, it becomes a witness node and sends Cα to a random location in the network. The node closest to that location will be another witness node w to store Cα. The watcher nodes on the routing path P from β to w only store the membership of IDα and lα in the Bloom filters. Such membership information can help them detect any conflicting location claim C'α received later, and guide C'α along P to either β or w, which will then broadcast both Cα and C'α to the entire network in order to revoke node α and its replicas.

The second protocol BC-MEM is designed on top of B-MEM. It adopts a cell forwarding technique that not only solves the crossover problem but also reduces the memory overhead. The deployment area is divided into virtual cells. In each cell an anchor point is assigned for every node in the network. The anchor point for a node α is determined by α ID. The node closest to the anchor point is called the anchor node for α. In B-MEM, when a location claim is forwarded on a line segment, all intermediate nodes on the line serve as watchers, while the first node and the last node serve as witnesses. In contrast, in BC-MEM a claim is not forwarded on the line segment. It is forwarded to the anchor point in the next cell where the line segment intersects. The claim is forwarded from one anchor node to another until reaching the last cell. The anchor nodes in the intermediate cells are watchers, and the anchor nodes in the first and last cells are witnesses.

The third protocol C-MEM is designed on top of B-MEM. It incorporates a new cross forwarding technique to solve the crowded center problem. B-MEM stores the information about a location claim along randomly selected line segments, which are likely to pass the center area of the deployment. On the other hand, C-MEM first selects a random point (called the cross point) in the network and forwards the location claim to that point. From there, it forwards the claim along the horizontal and vertical lines that pass the cross point. While the node closest to the cross point is a witness node, the nodes along the horizontal and vertical lines are watchers. Since the cross points for all location claims are distributed uniformly at random in the network, it is no longer true that the lines pass the center area more frequently. C-MEM does not use cell forwarding.

The fourth protocol CC-MEM combines cross forwarding and cell forwarding to solve both the crowded center problem and the crossover problem, such that it can detect

node replication attack with high probability and low overhead.

2.2.10. Active Detection of Node Replication Attacks. This technique falls into the category of base station-based techniques. Melchor et al. [51] have proposed a distributed protocol for the detection of replication attack for wireless sensor networks, in which each node verifies at random a few other nodes in the network. The proposed protocol does not build a distributed database of location claims that will contain local conflicting claims when replicas exist. The idea is that each node will actively test if 1 k other random nodes are replicated or not; they call them the scrutinized nodes. In order to test whether a scrutinized node α is replicated or not, 2 k nodes are randomly chosen in the network and asked to forward to α a request for a signed location claim. If two replicas exist, each will probably receive a request, and if both answer, two conflicting claims will be obtained by the queerer.

2.2.11. Randomly Directed Exploration: An Efficient Node Clone Detection Protocol in Wireless Sensor Networks (RDE). This technique falls into the category of witness node-based techniques. Zhijun et al. [52] have presented a novel clone node detection protocol called randomly directed exploration. This protocol does not call for any unrealistic assumptions. Each node only needs to know its neighbor nodes. During the detection procedure, nodes issue claiming messages containing neighbor list with a maximum hop limit to randomly selected neighbors. The previous transmission of a claiming message forms a direction, and then the intermediate node tries to follow the direction to forward the message. During forwarding messages, the intermediate nodes explore the claiming messages for node clone detection. In such a simple way, the proposed protocol can efficiently detect clone nodes in the dense sensor networks. In addition, the protocol consumes almost minimum memory during detection, and communication payload is satisfactory. It can scale to large configurations. They have implemented the protocol in the OMNet++ simulation framework.

2.2.12. Random-Walk-Based Approach to Detect Clone Attacks in Wireless Sensor Networks. These two techniques fall into the category of witness node-based techniques. Zeng et al. [4] have proposed two protocols RAndom WaLk (RAWL) and Table-assisted RAndom WaLk (TRAWL) for the detection of clone attack in wireless sensor networks. The RAndom WaLk (RAWL) starts several random walks randomly in the network for each node a, and then selects the passed nodes as the witness nodes of node a. RAWL works in four steps in each execution. In the first step, each node broadcasts a signed location claim. In the second step, each of the node neighbors probabilistically forwards the claim to some randomly selected nodes. In the thirds step, each randomly selected node sends a message containing the claim to start a random walk in the network, and the passed nodes are selected as witness nodes and will store the claim. In the fourth step, if any witness receives different location claims for same node ID, it can use these claims to revoke the replicated node.

The second protocol, Table-assisted RAndom WaLk (TRAWL), is based on RAWL and adds a trace table at each node to reduce memory cost. Usually, the memory cost is due to the storage of location claims, but in TRAWL each node only stores $O(1)$ location claims (although the size of the trace table is still $O(\sqrt{n}\log n)$, the size of a table entry is much smaller than the size of a location claim). When a randomly chosen node starts a random walk, all the passed nodes will still become witness nodes. However, now they do not definitely store the location claim, instead, they store the location claim independently with probability $c_2\sqrt{n}\log n$, where c_2 is a constant. Also, each witness node will create a new entry in its trace table for recording the pass of a location claim.

2.2.13. CINORA: Cell-Based Identification of Node Replication Attack in Wireless Sensor Networks. Gautam Thakur [24] has proposed two distributed methods for detecting node replication attack based on intersecting sets called CINORA-Inset and restricted cell two-phase authentication model called CINORA-Hybrid. Initially, the sensor network is divided into geographical cells similar to the existing cellular network. However, their approach does not deterministically map a nodes identity to a cell. In CINORA-Inset, location claims from the nodes are distributed among a subset of cells to detect any replication. These cells are generated from a nonnull intersecting subset algorithm. The inherent property of this algorithm is for any two subsets C_i and C_j of total $1 \le i, j \le N$ cells, and $C_i \cap C_j \neq \emptyset$. Thus, during the authentication phase at least one cell receives conflicting location claims if adversary has ever attempted to replicate a legitimate node. In CINORA-Hybrid a base station-based two-phase authentication scheme is used in which a sensor node has a valid residence entry permit for a cell. If permitted and nodes current residing cell is different or two (or more) similar permits are detected with different location claims, then that identity node is removed from the network.

2.2.14. A Note-Based Randomized and Distributed Protocol for Detecting Node Replication Attacks in Wireless Sensor Networks. This technique falls into the category of witness node-based techniques. Meng et al. [53] have proposed a note-based randomized and distributed protocol called NRDP, for detecting node replication attacks, which introduces no significant overhead on the resource-constrained sensors. This protocol does not need the geographic locations of nodes as well. Three types of nodes are assumed in the network, namely, a *claimer node, a reporter node*, and *a witness node*. A node which broadcasts a claim message is a *claimer node*. Neighbor node which forwards a claim message is a *reporter node*. And the destination node of a claim message is a *witness node*. This protocol works in two phases: neighbor discovery period and replication detection period. In the beginning of NRDP, it is a neighbor discovery period in which each node in the network broadcasts a message within its one-hop neighbors. After neighbor discovery period, each node in the network gets a neighbor list. The replication detection period starts when the neighbor discovery period ends. Replication

detect period consist of two steps. The first step is called request-note step and the second step is called send claim step. In *request-note step*, node α randomly chooses a node γ from its neighbor list as its reporter node, and then sends a request-note message to the reporter node. Upon receiving α request-note message, node γ replies with a signature note message which contains a note. The parameter time is fresh time of the note. Nodes in the network use it to identify the validity of a note received in different iterations. Note is an evidence to prove that the reporter node of a claimer node is existing and valid. In the *send-claim step*, every node generates a claim message, which includes a signed subneighbor list and a note got from the corresponding reporter node. The parameter list in the claim message is an ID list, which consists of q α's neighbor node IDs. And the reporter node γ must be in the list. Each node α then broadcasts the claim message in one-hop neighbors. When the reporter node receives corresponding claim message, it first verifies the signature and the time fresh of the note contained in the claim message. Further, the reporter node verifies that the list in the claim message contains its ID. If all the verifications succeed, using a pseudorandom function, the reporter node calculates g witness nodes for the claimer node. This function takes in input, the ID of the claimer node, which is the first argument of the claim message, the current rand value, and the number g of witness nodes that has to be generated. The witness nodes of a certain node change in different iterations. A trusted entity broadcasts a seed *rand* to the network before each detection iteration starts. This prevents the adversary from anticipating the witness nodes in a given protocol iteration. The reporter node analyzes the claim message, then generates a forwarded claim message, and forwards the forwarded claim message to all the g witness nodes. The forwarded claim message just contains the subneighbor list signed by claimer node, without note. When a node receives a claim message, it first checks whether it is the corresponding reporter node. If it is the reporter node of the claimer node, it checks the signature, the fresh of the note, and the list in the claim message. If it is not the reporter node, with probability pc it does the checking jobs as the reporter node does. It is necessary for nonreporter node neighbors to do the checking jobs with probability pc. This can prevent a claimer node from specifying a nonexisting neighbor node as its reporter node. Each node in the network has to specify an actual neighbor node as its reporter node, or it will be detected as a replicated node by its neighbor nodes.

Each witness node that receives a forwarded claim message verifies the signature and time fresh firstly. Then, it compares the claim to each previously stored claim. If it is the first time received claim contains IDα, then it simply stores the claim. If a claim from IDα has been received, the witness checks whether the claimed neighbor list is the same as the stored claim. If a conflict is found, the witness detects a node replication attack. Then, the witness triggers a revocation procedure for IDα. Actually, because there is always only one reporter node for a claimer node, if the claimer node is a valid node, its corresponding witness nodes would never receive more than one forwarded claim message from the claimer node. Therefore, once a witness node receives two claims containing the same ID in one detection iteration, it detects a replication attack. The two signature claims become evidence to trigger the revocation of the replicated node. The witness node forwards both claims to the base station. The base station will broadcast a signature message within the network to revoke the replicated node.

2.2.15. Distributed Detection of Replication with Deployment Knowledge in Wireless Sensor Networks. This technique falls into the category of group-based techniques. Ho et al. [54] have proposed three group deployment knowledge-based schemes for the detection of node replication attack in wireless sensor networks. Their schemes are based on the assumption that nodes are deployed in groups. By taking advantage of group deployment knowledge, the proposed schemes perform replica detection in a distributed, efficient, and secure manner. The sensors can be preloaded with relevant knowledge about their own group's membership and all group locations. Then, the sensors in the same group should be deployed at the same time in the location given to that group. The three proposed schemes are basic, location claim, and multigroup approaches. The first scheme is the basic scheme in which each node only accepts the messages from the member's of their own group (trusted nodes) not from other groups (untrusted nodes). It stops intercommunication between groups. An advantage of this basic scheme is low communication and computational or memory overhead. But the problem that is even honest nodes suffer from communication due to the fact that the deployment points are far away from their group. The network becomes poorly connected and not suitable for high resilient applications. To solve this problem, second scheme is proposed which also forwards messages from untrusted nodes as long as they provide provable evidence that they are not replicas but based on only predetermined locations for replica detection. The second scheme achieves high replication detection capability with less communication, computational, and storage overheads as compared to the first scheme, but there is a risk of DoS by flooding fake claims.

The third scheme protects against this kind of aggressive adversary. Every sensor node sends its neighbor's location claims to multiple groups rather than a single group. This scheme has higher communication overhead. It can provide a trade-off between the overhead and resilience to attack. This scheme provides very strong resilience to node compromise, since attacker needs to compromise multiple groups of nodes to prevent replicas being undetected.

2.2.16. Distributed Detection of Node Replication Attack Resilient to Many Compromised Nodes in Wireless Sensor Networks. This technique falls into the category of group-based techniques. Sei and Honiden [55] have proposed a distributed protocol for the detection of node replication attack that is resilient to many compromised nodes. Their method does not need any reliable/trusted entities. To prevent an attacker from learning the location of a witness node of a compromised node, the protocol uses a one-time seed for each replicated node detection process; that is, each node has the role of

starting a detection process, and it is preloaded with the assigned turn number and seed for the turn.

When node has a turn starting detection process, it sends the seed and its ID with a signature. Other nodes verify the signature and execute the detection process if the verification succeeds. They divide nodes into groups to increase resiliency to fault nodes and compromised nodes. The role of the starting detection process is not assigned to each node but to each group. If at least one node of a group survives, the group can start the detection process during its turn. An attacker must compromise the first node of a group which has the next turn starting detection process if he wants to learn the location of the witness node in the next detection process.

2.2.17. A Resilient and Efficient Replication Attack Detection Scheme for Wireless Sensor Networks. This technique falls into the category witness node-based techniques. Kim et al. [56] have presented a distributed, deterministic approach to detect node replication attack. Their scheme works in three steps: initialization, witness node discovery phase, and node revocation phase. In initialization phase, before deployment, a base station (BS) associates a particular location coordinate (hereafter referred to as the verification point, vp) with each node id using geographic hash function F. A vp is the target location coordinate in the network where each sensor node will be verified, and it can be predetermined by a network operator to a certain extent with experience. In witness node discovery phase, the replicas with the same id but different deployment locations are detected through location claim message. In the last phase of node revocation base station BS floods the revocation node lists after checking out the revocation request message received from the witness nodes. Once a BS receives this revocation request message, it checks whether the revocation request message is correctly encrypted by witness node using a pair-wise key shared with witness node. If the key is correct, a BS floods a list of replica nodes including reporter node through the network. If the key fails, which means that an attacker sent the forged replica revocation message, the BS regards that reporter node has been compromised.

3. Comparison of Node Replica Detection Schemes for Static WSNs

In this paper, we have addressed an important attack on WSN referred to as node replication attack or clone node attack. So far, many techniques have been proposed to detect node replication attack in static WSNs which are broadly categorized into centralized and distributed techniques. We have compared all the techniques according to their year of publication, identifying their shortcomings.

3.1. Centralized Techniques. Centralized techniques are considered to be the first solutions for detecting replicated nodes which are simple but suffer from several common drawbacks. Some of the limitations of centralized techniques are found to be fairly serious like the base station which introduces a single point of failure, and any compromise of the base station will render the solution useless; also, even if there are no attacks the nodes surrounding the base station will suffer an undue communication burden which may shorten the lifetime of a network, and this approach also incurs an observable processing delay. Consequently, centralized detections have barely an advantage over distributed detections making a distributed solution a necessity. The asymptotic performance of centralized techniques (including their memory and communication cost) is shown in Table 1. Localized voting protocols are also considered as the first naïve solutions for the detection of clone nodes which are unable to deal with distributed node replication attacks, in which replicas are placed at least two hops away from each other. In order to detect replicas which are spreading anywhere in the network a fully distributed solution is needed that also incurs small memory and energy overhead.

In 2004, one of the first solutions for detecting replicated nodes was proposed by Dutertre et al., outlined in [57] which was based on a centralized base station for node replica detection. This scheme was the most straightforward one and a naive solution that provided a low defense against node replication attacks, suffering from several drawbacks as mentioned before.

In 2007, Brooks et al. [32] proposed a clone detection protocol which was based on random pairwise key predistribution schemes and used to tackle with detection of cloned cryptographic keys rather than clones sensor nodes. This solution seemed effective but only when the size of the keys predistributed to each node is small and more clones exist in the network, thus implying poor detection accuracy. Moreover, it is assumed in the protocol that the connections between all nodes are possibly equal, while practically in WSNs, any sensor node can only communicate with a limited number of neighbors within a finite wireless communication radius. Another drawback of this solution is that it has neglected to ensure that the participating clones report their keys honestly to the base station.

Choi et al. [23] proposed another centralized detection technique named SET in 2007 which was an attempt to reduce the detection overhead by computing set operations. But the message authentication codes used for additional security resulted in even higher detection cost in terms of computation and communication. Moreover, SET protocol is highly complex due to its complicated components, and unexpectedly an adversary can misuse the detection protocol to revoke honest nodes.

Another centralized approach was proposed in 2008 by Xing et al. [34] which used social fingerprint for the detection of clones, but it was purely based on fixed WSNs, and thus neither node addition nor disappearance can be handled. Furthermore, besides all the common limitations of centralized solutions, it cannot handle a sophisticated replica which can cleverly compute by itself a fingerprint consistent with its neighborhood in order to flee the detection at the sensor side. A more intelligent replica can dodge and avoid the detection at the base station simply by not communicating with the base station.

The most recent solution for the detection of node replication attack or clones is a centralized technique given

Table 1: Asymptotic performance of centralized schemes.

Type of scheme	Technique/scheme	Communication cost	Memory cost
Key usage based	Brooks et al. scheme [32]	$O(n \log n)$	—
Base station based	SET [23]	$O(n)$	$O(d)$
	CSI [38]	$O(n \log n)$	—
Neighborhood social signature based	Xing et al. scheme [34]	$C \cdot (1 + \text{ratio})$	$O(d) + \min(M, \omega \cdot \log_2 M)$
Cluster head-based techniques	Znaidi et al. scheme [36]	$O(t^2)$	$O(t)$

n: no. of nodes in the network, ω: the column weight in the superimposed s-disjunct code, C: message generated by sensor node, d: degree of neighboring nodes, M: the number of rows in the superimposed s-disjunct code, ratio = $\log_2 M / L_{\text{packet}} \times 100\%$, and L_{packet}: the bit-length of a regular message.

by Yu et al. [38] in 2012. They have used a novel concept of compressed sensing for the identification of clones in the sensor network. This technique has the lowest communication overhead, but it suffers from all the common drawbacks of centralized techniques as BS is responsible for the aggregation of the result (decision) about the identification of clones in the network.

Considering the limitations of centralized detection schemes, the researchers move to a distributed solution for detecting clones, and the first naïve solution that was proposed was called node-to-network broadcasting (N2NB). Although the scheme was simple it also suffered from high memory and communication cost for large sensor networks.

3.2. Distributed Techniques. We have investigated a dozen distributed detection protocols by asymptotically comparing their communication and memory costs, and they are shown in Table 2. As all the proposed solutions use different motivations and assumptions and thus have their respective strengths and weaknesses, we cannot make any general or definite remarks that which solution is the best one.

Distributed techniques for the detection of clone node attack are categorized into three main classes, namely, witness node-based, neighbor-based, and generation-based or group-based techniques. All the three categories have their own pros and cons. For neighbor-based technique [48], the neighboring nodes should be static and any addition or removal of nodes is not possible throughout the detection process because in doing so the detection process is affected severely. For the generation- or group-based techniques [40, 41, 54, 55] all the nodes are deployed in groups, and no new node can be added in a particular group. Also, nodes should have location or network information before node deployment. These techniques only prevent the node replication attack but are unable to detect the clone nodes.

The witness node-based techniques use a framework called claimer-reporter-witness framework in which a node referred to as claimer locally broadcasts it, location claim to its neighbors. Each neighbor serves as a reporter and employs a function to map the claimer ID, to a witness. The neighbor forwards the claim to the witness and if it receives two different location claims for the same nodded id then it means that the adversary has replicated a node. The adversary can also employ a function to know about the witness for the given claimer ID and may also locate and compromise the witness node before she inserts the replicas into the wireless sensor networks in order to evade the detection.

A relatively more mature distributed detection scheme was proposed in 2007 by Parno et al. [28] known as deterministic multicast (DM) which was the first to use a framework called claimer-reporter-witness framework. Although its design goal was to reduce communication cost, it was treated as an unfavorable protocol because of its several drawbacks. Firstly, it does not provide much security as an adversary only needs to compromise all the g witnesses for a given claimer id deploying as many replicas as she desires without activating an alarm. Secondly, it does not work for large g as both the network communication and the node storage are proportional to g, and with very small g, an adversary can produce unlimited replicas. Considering DM as unappealing due to its deterministic property, Parno et al. [28] have proposed and developed two more techniques as improvements of DM protocol, namely, randomized multicast (RM) and line selected multicast (LSM). The security was improved but at the price of increased communication/memory costs. In both of these protocols the problem lies in the selection of witness nodes (i.e., Probabilities) and also it is not always true that location claims of clone nodes are received to the same witness node. Moreover, both RM and LSM are unable to detect masked replication attack. To decrease the communication cost of RM protocol, LSM was developed as a less expensive version of RM, but it suffers from uneven distribution of witnesses nodes. As majority of witness nodes are selected from the center of the network, thus the energy of these nodes is depleted soon, and also they become the point of interest for the adversary.

Zhu et al. [44, 45] proposed two techniques called single deterministic cell (SDC) and parallel multiple probabilistic cells (P-MPC) in 2007 as the variations of DM. Practically, both of these techniques depend upon the careful selection of a cell size (s) because if the cell size is too large, they incur high communication cost like N2NB, and if s is too small, it will be very easy for an adversary to trounce them by compromising all nodes in the g deterministic tiny cells. An important problem with SDC is that in order to reduce the broadcast overhead, it requires to execute the flooding only when the first copy of a node location claim arrives at the cell, and the following copies are ignored. In doing this, the node in the cell that first receives the location claim is unable to distinguish between claims of original node and replica node.

Another attempt to detect clones was made by Conti et al. [42, 43] in 2007 who have proposed a randomized, efficient, and distributed protocol named RED by combining

Table 2: Asymptotic performance of distributed schemes.

Type of scheme	Technique/scheme	Communication cost	Memory cost
Node-to-network broadcasting	N2NB [28]	$O(n^2)$	$O(1)$
Witness node	DM [28]	$O(g \log \sqrt{n}/d)$	$O(g)$
	RM [28]	$O(n^2)$	$O(\sqrt{n})$
	LSM [28]	$O(n\sqrt{n})$	$O(\sqrt{n})$
	RED [42, 43]	$O(g \cdot p \cdot dn\sqrt{n})$	$O(g \cdot p \cdot d)$
	SDC [44, 45]	$O(r \cdot \sqrt{n}) + O(s)$	$O(\omega)$
	P-MPC [44, 45]	$O(r \cdot \sqrt{n}) + O(s)$	$O(\omega)$
	B-MEM [3]	$O(k \cdot n \cdot \sqrt{n})$	$O(tk + t'k\sqrt{n})$
	BC-MEM [3]	—	$O(tk + t'k\sqrt{n'})$
	C-MEM [3]	—	$O(t + t'\sqrt{n})$
	CC-MEM [3]	—	$O(t + t'\sqrt{n'})$
	Melchor et al. [51]	$O(\sqrt{n})$	$O(d)$
	RDE [52]	$O(d \cdot n \cdot \sqrt{n})$	$O(d)$
	RAWL [4]	$O(\sqrt{n} \log n)$	$O(\sqrt{n} \log n)$
	TRAWL [4]	$O(\sqrt{n} \log n)$	$O(1)^2$
	Kim et al. [56]	$O(\sqrt{n})$	$O(\sqrt{n})$
Generation or group based	Bekara and Laurent-Maknavicius [40, 41]	$O(\sqrt{n})$	$O(1)$
	Basic scheme [54]	$O(m)$	$O(m)$
	Location claim base scheme [54]	$O(m + d)$	$O(d + 2m)$
	Multigroup base scheme [54]	$3 * O(m + d)$	$O(d + 2 * m\,(1 + D_{\max}))$
	Sei and Honiden [55]	$O(r)$	$O(r \cdot \sqrt{n})$
—	Ho [49]	$O(n\sqrt{n})$	$O(n)$
Neighborhood based	NBDS [48]	$O(r \cdot \sqrt{n})$	$O(r)$

n: no. of nodes in the network, d: degree of neighboring nodes, g: no. of witness nodes, r: communication radius, s: the number of sensors in a cell, p: probability that neighboring node will forward the location claim, ω: the column weight in the superimposed s-disjunct code, and ξ: distinct IDs from set of nodes as monitor.

the benefits of both DM and RM. This protocol is considered to be the most promising detection protocol which has solved the crowded center problem as the selection of witness nodes is random and fully distributed. Also, RED [4] is such an "area oblivious" protocol that associates sensor nodes with almost even responsibility, and the selection of witness nodes is pseudorandom which leads to a uniform witness distribution. Besides these advantages, the only drawback of RED is the deterministic selection of witness nodes and that the infrastructure for distributing RED's random seed may not always be available. RED is also unable to detect masked replication attack.

Bekara et al. [40, 41] in 2007 proposed a solution for preventing WSN from node replication attack which exploits the fact that excluding new nodes from joining the network can prevent replication attacks. The main drawback of this scheme is that the sensor nodes are bound to their groups and geographic locations.

In 2009, Zhang et al. [3] have proposed four memory efficient multicast protocols for the detection of replicated nodes, namely, Bloom filter MEM, Bloom filters and cell forwarding MEM, cross forwarding MEM, and last is cross forwarding and cell forwarding MEM. B-MEM is an extension of LSM, but it incurs additional memory consumption per node, and it may also lower the detection rate of LSM due to false verifications (false positives of Bloom filters). BC-MEM requires highly accurate localization due to its cell division and anchor node selection which may not be affordable for current generation of WSNs. Also, an adversary can elude BC-MEM by compromising certain deterministic anchor nodes. In case of both C-MEM and CC-MEM, cross forwarding achieves high detection probability for convex deployment field (particularly for rectangle-shaped deployment field), but for other irregular topologies considered by LSM (like thin cross and large H), these two schemes may work poorly by dropping the detection rate significantly.

A simplified version of N2NB was proposed by Zhang et al. [3] in 2009 known as randomly directed exploration (RDE). Its network communication overhead is reduced, but storage cost remains the same with N2NB. The detection rate is also decreased and may not be very significant even for a convex deployment field concluding that RDE appears to be feasible only for an ideal network model.

Another work in this area is done by Zeng et al. [4] in 2010 who have proposed two detection protocols, namely, RAndom WaLk (RAWL) and Table-assisted RAndom WaLk (TRAWL) for the detection of node replication attack. Both of these protocols are an extension of LSM and thus suffer from the same drawbacks. Although they have much higher detection probability than LSM, both RAWL and TRAWL require more than twice the communication overhead of LSM.

For an inclusive survey, we have also analyzed some other distributed techniques which are neither very popular nor have promising results in detecting node replication attack. These techniques include Ho et al. [54] proposed in 2009, Kim et al. [56] proposed in 2009, and Meng et al. [53] proposed in 2010.

4. Detection Techniques for Mobile WSNs

Mobility has become an important area of research for WSN community. In mobile WSNs, mobility plays a key role in the execution of the application as the introduction of mobile entities can resolve some problems and offer many advantages over the static WSNs. The node replica detection techniques developed for static WSNs, do not work when the nodes are expected to move as in mobile WSNs, and thus they have turned out to be ineffective for mobile WSNs. As a result some techniques (still not mature enough) have also been developed for mobile WSNs to detect the replica or clone nodes. These techniques are classified into two main classes as centralized and distributed and are described below.

4.1. Centralized Techniques

4.1.1. Fast Detection of Replica Node Attack in Mobile Sensor

Networks Using Sequential Analysis. Ho et al. [58, 59] have proposed a mobile replica detection scheme based on the sequential probability ratio test (SPRT) [50]. Their protocol is based on the fact that an uncompromised mobile node should never move at speeds in excess of the system-configured maximum speed. As a result, an uncompromised (original) mobile sensor node measured speed will appear to be at most the system-configured maximum speed as long as speed measurement system with low error rate is employed. On the other hand, replica nodes will appear to move much faster than original nodes, and thus their measured speeds will likely be over the system-configured maximum speed because they need to be at two (or more) different places at once. Accordingly, if it is observed that a mobile node measured speed is over the system-configured maximum speed, it is then highly likely that at least two nodes with the same identity are present in the network. By leveraging this intuition, the SPRT is performed on every mobile node using a null hypothesis that the mobile node has not been replicated and an alternate hypothesis that it has been replicated. In using the SPRT, the occurrence of a speed that either lessens or exceeds the system-configured maximum speed will lead to acceptance of the null and alternate hypotheses, respectively. Once the alternate hypothesis is accepted, the replica nodes will be revoked from the network.

4.1.2. A New Protocol for the Detection of Node Replication

Attacks in Mobile Wireless Sensor Networks. Deng and Xiong [60] have proposed a new protocol to detect the replicas in mobile WSNs. They have used the idea of polynomial-based pair-wise key pre-distribution and Bloom Filters which insure that the replicas can never lie about their real identifiers and collect the number of pair-wise keys established by each sensor node. Replicas are detected by looking at whether the number of pair-wise keys established by them exceeds the threshold. The protocol works in three steps, node initialization, pair-wise establishment, and detection. In node initialization, before nodes are deployed, the key server randomly generates a bivariate symmetric polynomial over a finite field. After deployment between nodes, pairwise keys are established. Each node periodically constructs a report, which includes its ID and counting Bloom filter (or compressed counting Bloom filter), and sends it to the base station. At base station, counting bloom filters collect the number of pairwise keys established by each node. Nodes whose number of pair-wise keys exceeds the threshold value are considered to be the clones.

4.2. Distributed Techniques

4.2.1. Mobile Sensor Networks Resilient against Node

Replication Attacks. Chia et al. [61] proposed a novel protocol, called extremely efficient detection (XED), against node replication attack in mobile sensor networks. The idea behind XED is motivated from the observation that for the networks without replicas, if a sensor node s_i meets the other sensor node s_j at earlier time and s_i sends a random number r to s_j at that time, then when s_i and s_j meet again, s_i can ascertain whether this is the node s_j met before by requesting the random number r. Based on this observation, a "remember and challenge strategy" is proposed. Once two sensor nodes, s_i and s_j, are within the communication ranges of each other, they first, respectively, generate random numbers $rs_i \rightarrow s_j$ and $rs_j \rightarrow s_i$ of b bits, and then they exchange their generated random numbers. They also use a table to record the node ID, the generated random number, and the received random number in their respective memory. In case the pair of two nodes met before, the above procedure is also performed such that the random number stored in the memory is replaced by the newly received random number. Consider the example shown in Figure 4, in which the sensor node s_i meets another sensor node s_j. If s_i never meets s_j before, they exchange random numbers. Otherwise, the sensor node s_i requests the sensor node s_j for the random number $rs_i \rightarrow s_j$ exchanged at easier time. For the sensor node s_i, if the sensor node s_j cannot replies or reply a number which does not match the number in s_i memory, s_i announces the detection of a replica. When the replicas meet the genuine nodes, the replicas can always pretend that they meet for the first time. However, if the genuine nodes have a record showing that they ever met at earlier time, the replicas are also detected.

4.2.2. Efficient and Distributed Detection of Node Replication

Attacks in Mobile Sensor Networks. Chia et al. [62] proposed an efficient and distributed detection (EDD) scheme and its variant, storage-efficient EDD (SEDD) scheme to detect the node replication attack. The idea behind EDD and SEDD is motivated from the following observations. For a network without replicas, the number of times, $\mu 1$, in which the node u encounters a specific node v, should be limited in a given time

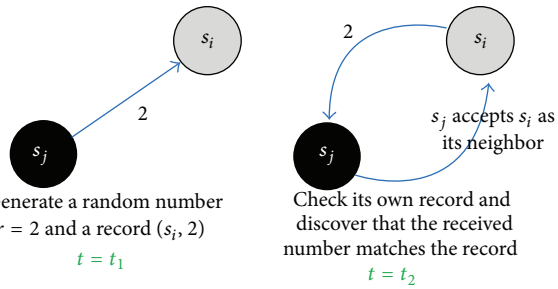

FIGURE 4: The operations between two genuine nodes in XED at time t_1 and t_2 (gray and black nodes are genuine) [61].

interval of length T with high probability. For a network with two replicas v, the number of times, $\mu 2$, in which u encounters the replicas with the same ID v, should be larger than a threshold within the time interval of length T. According to these observations, if each node can discriminate between these two cases, each node has the ability to identify the replicas. The EDD scheme is composed of two steps: offline step and online step. The offline step is performed by the network planner before the sensor deployment. The goal is to calculate the parameters, including the length T of the time interval and the threshold ψ used for discrimination between the genuine nodes and the replicas. On the other hand, the online step will be performed by each node per move. Each node checks whether the encountered nodes are replicas by comparing ψ with the number of encounters at the end of a time interval. It can be observed from EDD that each node should maintain a list L, leading to $O(n)$ storage overhead. A storage-efficient EDD (SEDD) scheme is proposed based on the tradeoff between storage overhead and time interval length. The basic idea behind SEDD is that instead of monitoring all nodes, each node only monitors a subset of nodes, called monitor set, in a specific time interval. When the cardinality of the monitor set is selected as ξ, the simplest way for each node to select the nodes to be monitored at the beginning of a time interval is to randomly pick ξ distinct IDs from $\{1, \ldots, n\}$. Since the storage overhead is equal to the number of nodes being monitored, the storage overhead is reduced to the cardinality of the monitor set, $O(\xi)$, in the SEDD scheme.

4.2.3. Patrol Detection for Replica Attacks on Wireless Sensor Networks.

Wang and Shi [63] have employed mobile nodes as patrollers to detect replicas distributed in different zones in a network, in which a basic patrol detection protocol and two detection algorithms for stationary and mobile modes are presented. The detection of replicas in stationary sensors is based on the assumptions that if two or more sensors in different locations have the same ID, then all the nodes with the ID will be regarded as compromised nodes or its replicas. Also, for mobile sensors (patroller), if a mobile node moves with a speed higher than the denoted maximum speed, it will be regarded as a replica attack. In the replica detection of static sensor nodes, when a mobile patrol node moves to a new zone, it first discovers its location and then broadcasts its patrol claim. Each node will be patrolled by at least two mobile nodes. After receiving the location messages, the stationary node takes the mobile nodes who patrolled him as the anchor nodes and will send the patrol node its location claim. After collecting the answer message, patrol node will check the location of node, and if the distance is larger than the signal range, it ignores the wrong message. Otherwise, it will check the ID of the answer message by using the security assumption "A legitimate ID only has one location." Then, it saves the answer from the original node (benign node) in a whitelist, saves the replica node ID in a blacklist, and revokes the replica ID by refusing to distribute secret material and broadcasting its two answer messages to other mobiles nodes. Then, patrol node will move to another location to send his patrol claim in another interval. After a round, it collects all the saved information of the white- and blacklists to the user when collecting the sensing data. If the replicas are deployed in a zone where a patrol node collects their answer message in a patrol interval, then the patroller can revoke them immediately after he receives the second answer and the distance between the two locations exceeds. Else if the replicas' answers are collected by different patrol nodes, then they will be found by the base station or by exchange messages of patrollers after a round. If the adversary compromises and replicates the patrol node, firstly, an original mobile patroller will wait for the answer message after he reaches a new position and sends his claim in time T, so there is a static period interval after the patrol broadcasts his claim. Accordingly, if the patroller node moves and changes its position in time $(T, T + \text{interval})$, then it is highly likely that at least two nodes with the same identity are present in the network. Further, the mobile patroller should never move faster than the system-configured maximum speed $V_{\max}$.

4.2.4. Single-Hop Detection of Node Clone Attacks in Mobile Wireless Sensor Networks.

Lou et al. [64] have proposed a node clone attack detection protocol, namely, the single hop detection (SHP) for mobile wireless sensor networks. The SHD protocol exploits the fact that at any time, a physical node (or equivalently, its node ID and private key) cannot appear at different neighborhood community; otherwise, there must be replicas in the network. The neighborhood community of a node is characterized by its one-hop neighbor node list, which is readily available in a typical WSN since sensor nodes need to know their neighbors in order to communicate with each other. The SHD protocol consists of two phases, the fingerprint claim and the fingerprint verification phases. In the fingerprint claim phase each node is required to sign its neighbor node list. The signed neighbor node list is a fingerprint of its current neighborhood community, hereafter referred to as fingerprint claim. The fingerprint claim is broadcasted in one-hop neighborhood. Upon reception of a fingerprint claim from a neighboring claim node, the receiver node will decide whether to become a witness node of the claim node. When it decides to become a witness node, the node will then verify the fingerprint claim and finally store the fingerprint claims of the witnessed nodes locally if the claim passed the verification process. In the fingerprint

verification phase, when two nodes meet with each other, they exchange their witnessed node lists, and this can be done by piggybacking the witnessed node list in the two nodes and then checking for a possible fingerprint claim conflict with received claims. In a fingerprint claim conflict, there are two fingerprint claims with the same ID and private key claiming two different neighborhood communities, which implies two detected replicas.

4.2.5. Detecting Node Replication Attacks in Mobile Sensor

Networks: Theory and Approaches. Zhu et al. [65] have proposed two replica detection algorithms for mobile sensor networks. First algorithm is a token-based authentication scheme proposed for the detection of replication attack in which the replicas do not cooperate (nonconspiring case). For the case in which the replicas cooperate by communicating with each other in an efficient manner, a detection method is proposed which is based on statistics and the random encounters between physical nodes. In the first algorithm, the base station periodically broadcasts to the entire sensing region a timestamp protected by a broadcast authentication protocol. The broadcast announces the beginning of a detection round. Upon hearing the timestamp, a genuine mobile node randomly selects a secret seed $s_i \in \{0,1\}^l$, where l is a common security parameter, and empties its local storage of the previously received tokens. The detection consists of a token exchange phase and a mutual authentication phase When a mobile node first meets with another mobile node in the detection round, they will exchange a token with each other and will record the tokens in their memories. When these mobile nodes meet again in the same detection round, each will ask the other for the previously exchanged token. Upon receiving the correct reply, each believes that the other is authenticated. Otherwise, in case of replica, when genuine node asks a replica node (to whom it met before) about the token they have exchanged in their first meeting, the replica node will reply in no or with a wrong token which will mark him as replica.

The second algorithm is a statistics-based detection scheme for detecting replicas that cooperate with each others. This idea is partially inspired by [66] whose detection principle is that if a node is not "seen again" by others, it is likely that the node has been captured. Similarly, herein, the principle is that if in a certain detection round a node is "seen again" too many times by others, it is likely that the node is a replica. Every genuine node contains a step counter "T" and also its "acquaintance list" consisting of n Boolean variables. Each time a mobile node meets another mobile node, it increases the counter T by 1. If this is its first meeting with any mobile node, it treats it as an acquaintance and sets the corresponding bit in the list to 1. Once the acquaintance list contains all 1's, the statistics stops. In the nondetection stage each node reports its numbers of meetings with others when dropping by the base station. The base station is employed for centralized analysis. Finally, the node with more encounters is detected as replica, and base station finally broadcasts the entire network for replicated IDs.

4.2.6. Emergent Properties: Detection of the Node Capture

Attack in Mobile Wireless Sensor Networks. Conti et al. [66] have proposed two algorithms for the detection of node capture attack in mobile wireless sensor networks. Their first algorithm is simple distributed detection (SDD) in which the attack is detected using only information local to the nodes. The second algorithm is called cooperative distributed detection (CDD) which exploits node collaboration to improve the detection performance. Both of the proposed algorithms are based on the simple observations that if node a will not remeet node b within a certain period of time, then it is possible that node b has been captured. Hence, node a can autonomously know the probability that a "not yet remet" node has been actually captured by the adversary. The SDD follows the above simple observation that each node a is given the task of tracking a specific set T_a of other nodes. For each node $b \in T_a$ that a gets into the communication range of, a set the corresponding meeting time to the value of its internal clock and start the corresponding timeout, that will expire after λ seconds. If the time-out expires (i.e., a and b did not remeet), the network is flooded with an alarm triggered by node a to revoke node b. In CDD, network mobility and node cooperation are leveraged to improve node capture detection. When two nodes a and b exchange information about the nodes (if any) that are tracked by both a and b, that is, the nodes in T_aT_b, the node exchanges information only when cooperating nodes are in the same communication radius. This shared information is further used for node capture.

4.2.7. Mobility-Assisted Detection of the Replication in Mobile

Wireless Sensor Networks. Deng et al. [67] have proposed two schemes for the detection of node replication attack in mobile wireless sensor networks. The first is called unary time location storage and exchange (UTLSE), and, second is called multitime location storage and diffusion (MTLSD). In both protocols, after receiving the time-location claims, witnesses carry these claims around the network instead of transmitting them. That means that data are forwarded only when appropriate witnesses encounter each other. Only if two nodes encounter each other, they exchange their time-location claims, that is, if a tracer receives a time location claim from its tracked neighbor node, it does not immediately transmit this time-location claim to the witness if the witness is not currently within its communication range but stores that location claim until encountering the witness. UTLSE detects the replicas by each of the two encountered witnesses which stores only one time-location claim. On the other hand, MTLSD stores more time-location claims for each tracked node and introduces time-location claims diffusion among witnesses. The detection probability of the MTLSD protocol is greater than the probability of protocol UTLSE.

5. Comparison of Node Replica Detection Schemes for Mobile WSNs

Mobile wireless sensor networks (MWSNs) are still in their infancy, and there are many challenges in MWSNs that are

still needed to be resolved. These challenges include deployment, localization, self-organization, navigation and control, coverage, energy, maintenance, and data process [26]. In case of localization, node position can be determined once during initialization when sensor nodes are deployed statically [68]. However, when sensor nodes are mobile, they must continuously obtain their positions as they navigate through the whole sensing region. As a result, in mobile WSNs, localization requires additional time and energy and also the availability of a rapid localization service. Due to the dynamic network topology of mobile WSNs, they cannot rely on routing tables or recent route histories as static WSNs do for passing messages through the network because table data become outdated quickly; thus, route discovery data must repeatedly be performed extensively in terms of power, time, and bandwidth.

Ho et al. [58, 59] have proposed a centralized detection scheme for mobile WSNs in which accurate measurement is a prerequisite for acceptable false-negative and -positive rates. In result, it requires dynamic and precise localization system and a tight time synchronization which are both nontrivial tasks. Also, better and accurate sampling entails even much more expensive equipment (GPS) and thus may not be affordable for the current generation of WSNs. Another centralized detection technique is proposed by Deng and Xiong [60] in which there is no way to ensure, the participating clone node will report their keys honestly to the base station. It is possible that an original node number of pairwise keys exceed the threshold value due to its communication. Also as the effectiveness of both the above centralized detection techniques relies on the involvement of the base station, this easily incurs the problems of single-point failure and fast energy depletion of the sensor nodes around the base station.

Yu et al. [61] have proposed distributed detection technique called extremely efficient detection technique (XED) in which the authors have assumed that the replicas cannot communicate and collaborate (or cooperate) with each other which is the weakness of this scheme because in case when the replicas cooperate with each other, they can establish secret channels among each other, and then they can easily deceive the detection technique. Efficient and distributed detection (EDD) is another distributed detection technique for mobile WSNs proposed by Yu et al. [62] which is inapplicable due to high storage overhead for large-scale WSNs.

Zhu et al. [65] have proposed a token-based detection technique which fails when a smart attacker establishes secret channels among replicas as by doing this, replicas can share the tokens and make the protocol exist in name only.

Conti et al. [66] have proposed two solutions, namely, SDD and CDD for the detection of node capture. Their approach is based on a simple observation which completely assumes that there is no membership change in the network; for example, at least no nodes die out (meaning run out of power) which is not the case in reality. Also, it is assumed implicitly that any senor node is able to flood the entire mobile WSN with a broadcast message which is also not possible in reality.

An asymptotic comparison of all the detection schemes for mobile WSNs is shown in Table 3 where their communication and memory costs are compared. As all the proposed solutions use different motivations and assumptions and thus have their respective strengths and weaknesses, we cannot make any general or definite remarks that which solution is the best one.

6. Discussion

Node replication attack or clone attack is one of the most harmful and dangerous threat to an unattended wireless sensor network because in this attack an adversary not only compromises the sensor nodes but can also carry out a large class of internal attacks for instance DoS attack, Sybil attack, and Black hole, and wormhole attack, by surreptitiously inserting arbitrary number of replicas at strategic positions of the network. Furthermore this is more niggling and troublesome because these replicated nodes, under the control of an adversary, having all the keying materials, pretend as authorized users in the network and thus deceiving the network into accepting them as legitimate nodes. It is difficult to identify replicas because of two major reasons. First, since a clone or replica is considered to be completely honest by its neighbors, the legitimate nodes cannot be aware of the fact that they have a clone among them. Voting mechanisms [33, 69] remain unsuccessful to detect clone nodes that are not within the same neighborhood as a voting mechanism is used to detect misbehaving nodes and clones within the neighborhood to agree on the legitimacy of a given node. Thus, there is a need for global countermeasure that can detect clones on the global level. Second, the general purpose security protocols for secure sensor network communication would allow replica nodes to create pair-wise shared keys with other nodes and the base station, and thus in doing so, the replica nodes are able to encrypt, decrypt, and authenticate all of their communications as if they were the original captured nodes.

The process or stages of node replication attack can be described in the form of a flow chart as shown in Figure 5. The flow chart concisely describes the instigation of node replication attack and its detection, from physical node capture, extraction of secret credentials, cloning and redeployment and finally the detection and prevention of node replication attack. 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. 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. At 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 them

Table 3: Asymptotic performance of schemes against clone node attack in mobile sensor networks.

Nature of scheme	Type of scheme	Technique/scheme	Communication cost	Memory cost
Centralized	Node speed based	Ho et al. scheme [58, 59]	$O(n\sqrt{n})$	$O(n)$
	Key usage based	Deng and Xiong scheme [60]	$O(n \log n)$	—
Distributed	Information exchange based	XED [61]	$O(1)$	$O(4 \cdot d \cdot E[X])$
	Node meeting based	EDD [62]	$O(1)$	$O(n)$
		SEDD [62]	$O(n)$	$O(\xi)$
	Mobility assisted based	Wang and Shi scheme with Base Station [63]	$O(n)$	—
		Wang and Shi scheme with out Base Station [63]	$O(n * \sqrt{k})$	—
		UTLSE [67]	$O(n)$	$O(\sqrt{n})$
		MTLSD [67]	$O(n)$	$O(\sqrt{n})$

n: no. of nodes in the network, ξ: distinct IDs from set of nodes as monitor, d: degree of neighboring nodes, and k: total number of zones.

at strategic positions of the network for further insider attacks at Stage 5. Finally these replicas or clones can be detected by using various detection schemes.

Since clone nodes carry all the cryptographic and keying materials, all the traditional authentication and intrusion detection techniques are ineffective to discover and detect these clones or replicas in the network. Keeping this in mind many techniques have been proposed for the detection of node replication attack and recall that these are broadly categorized into centralized and distributed techniques. Some fairly serious limitations of centralized technique like the base station introduces a single point of failure, and any compromise of the base station will make the solution useless thus making distributed solutions a necessity. One important class of distributed techniques is witness node-based techniques which are considered to the most favorable techniques yet for detecting clone nodes. But according to Zeng et al. [4], replica detection protocols must be non-deterministic and fully distributed in order to circumvent the existing drawbacks of witness-based strategies. The witness node-based strategies ought to fulfill three requirements to have a high probability of detecting clones or replicas. Firstly, the selection of witness-nodes should be nondeterministic as it is more difficult for an adversary to launch clone attacks in nondeterministic protocols successfully because the witnesses of node are not known and are different in each execution of the protocol. Secondly, for any given node, all the nodes should have an equal probability to be the witnesses of that node during the lifetime of the network. Thirdly, the witness-nodes should be selected from all over the network randomly and not from particular area of the network every time meaning that the witness distribution should be uniform throughout the entire network.

There are two types of attacks which are the variations of node replication attack and can be launched by an adversary against witness node-based schemes. These are named as smart attack and masked replication attack. Smart attack is a special witness compromising attack, and in this attack an adversary avariciously chooses which sensor to corrupt in order to maximize its chance for its replicas to go undetected. The adversary finds out the witness nodes which are used to detect replicas and only compromises these witness nodes to avoid detection. The witness node-based techniques use a framework called claimer-reporter-witness framework in which a node referred to as claimer, locally broadcasts it location claim to its neighbors. Each neighbor serves as a reporter and employs a function to map the claimer ID to a witness. The neighbor forwards the claim to the witness and if it receives two different location claims for the same node ID then it means that the adversary has replicated a node. The adversary can also employ a function to know about the witness for the given claimer ID, and may also locate and compromise the witness node before she inserts the replicas into the wireless sensor network in order to evade the detection. In masked replication attack, the adversary may turn to compromise all the neighbors of a replica so as to prevent a location claim from propagating to any witness thus eliminating the reporters at all. This attack makes it possible for such a replica, whose neighbors have all been compromised, to lie about its physical position. So far, all the witness node-based techniques have assumed a static WSN, and are seemed to be the most promising schemes till yet to detect replicas or clones in static WSN, but alas these witness node-based schemes and location-based replication detection schemes are unable to detect and counter these types of replication attacks.

Nowadays, mobility has become an important area of research for WSN community. In mobile WSNs, mobility plays a key role in the execution of the application [68] as the integration of mobility in WSN can improve the coverage and utility of the sensor network deployment and enables more versatile sensing applications as well. However, besides that the introduction of mobile entities (which freely roam in the network and are autonomous as being able to reposition and organize themselves in the network) can resolve some problems by offering many advantages over the static WSNs the unique properties of mobile WSNs and the dynamic mobile network topology pose many new challenges in the security of mobile WSNs. The idea of detecting clone nodes in static WSNs is extensively based on the elitism of the node location meaning that a sensor node should be allied to a unique deployment position, and if one logical node id is found to be associated with two or more physical locations, the node replication is detected. But noticeably this is not

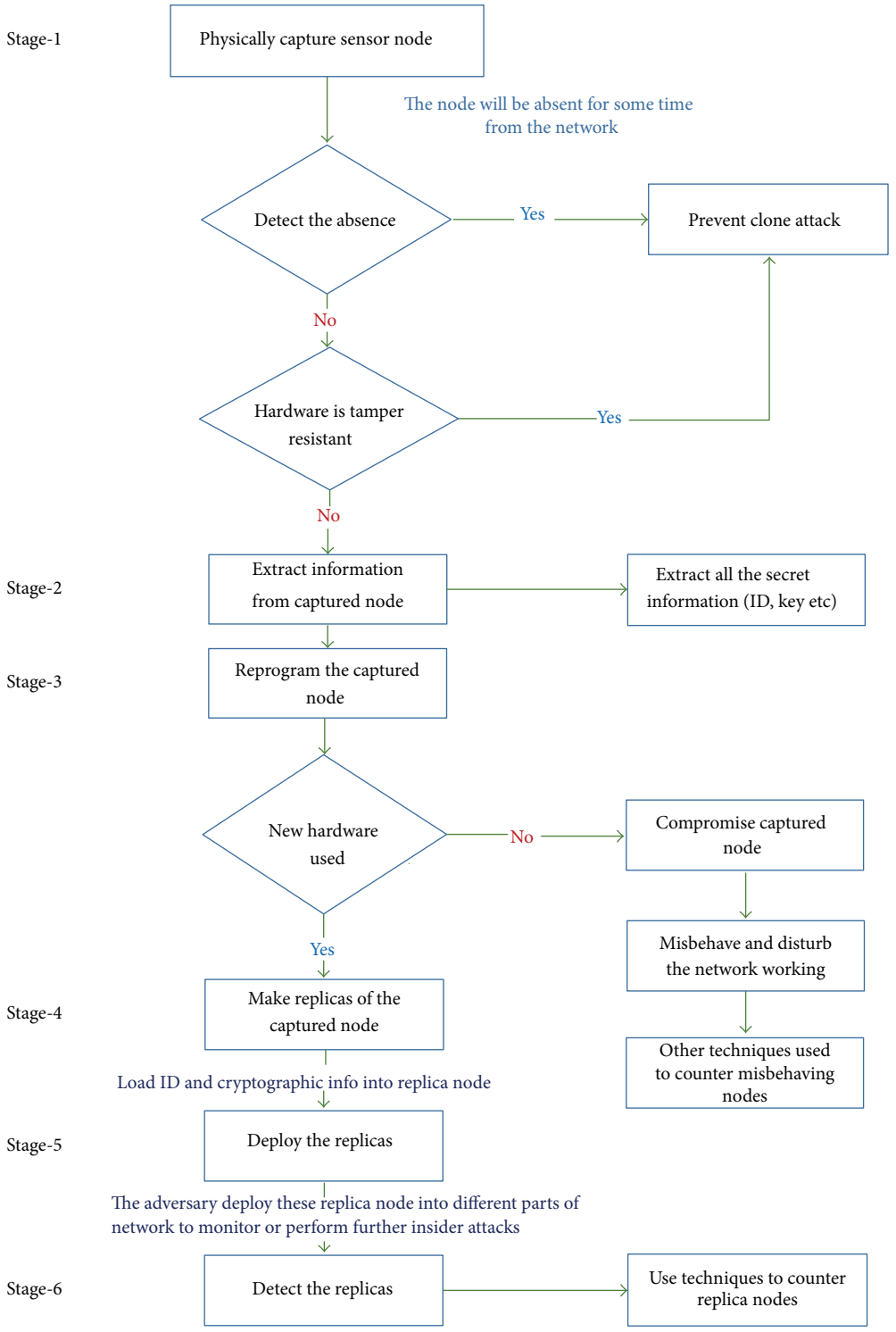

Figure 5: Stages of node replication attack in wireless sensor networks.

applicable to the emerging mobile WSNs where the sensor nodes are moving freely all the time in the network. Thus, a little work (which includes significantly different scenarios and techniques) has been done so far to deal with replicas or clones in mobile WSNs.

In mobile WSNs, the adversary is also mobile. In the literature, the assumed scenario of mobile WSN is that the sensors are unable to transmit sensed data at their will because the sink is not always present. Thus, the data accumulated in their memories become targets of many adversaries. In

[30], a mobile adversary model is proposed in which mobile adversary visits and travels around the network trying to compromise a subset of sensors within the time interval when sinks are not present in the network. The time taken by a mobile adversary to compromise a set of sensors is much shorter than the time between two successive data collections of a sink. Thus, it is much difficult to snatch mobile compromised nodes as well as mobile clones.

Another challenge arises in mobile WSNs when a mobile adversary adopts a more sophisticated strategy named "group mobility strategy." In this stratagem, the replicas form a physically close group which always moves together, but only a representative of them communicates with the genuine nodes, whereas the rest of the replicas remain inactive as "silent learners" so that they can learn (from encounters with genuine nodes) about any received token or corresponding meeting instant. Once the replicas have met all the "n" genuine nodes (and thus acquired all the necessary knowledge to pass later authentications), they can scatter in the sensing region, and each behaves actively and independently, until the next detection round starts.

Also, when the mobile replicas communicate and collaborate with each other and share their keys or random numbers, they can make the detection technique fails to thwart them easily. Thus, mobile WSNs offer much more challenges in detecting mobile replicas, and it is highly needed to overcome these challenges by developing some new, different and more efficient detection techniques for detecting mobile replicas or clones.

7. Conclusion

This paper reviewed the state-of-the-art schemes for detection of node replication attack also called clone attack. The existing techniques are broadly categorized into two classes distributed and centralized. Both classes of schemes are proficient in detecting and preventing clone attacks, but both schemes also have some noteworthy drawbacks. However, to sum up, the current study highlights the fact that there are still a lot of challenges and issues in clone detection schemes that need to be resolved to become more applicable to real-life situations and also to become accepted by the resource constrained sensor node.

Acknowledgment

The authors wish to acknowledge the anonymous reviewers for their valuable comments for the improvement of this paper.

References

[1] T. Bonaci, P. Lee, L. Bushnell, and R. Poovendran, "Distributed clone detection in wireless sensor networks: an optimization approach," in *Proceedings of the 2nd IEEE International Workshop on Data Security and Privacy in Wireless Networks (WoWMoM '11)*, Lucca, Italy, June 2011.

[2] W. T. Zhu, J. Zhou, R. H. Deng, and F. Bao, "Detecting node replication attacks in wireless sensor networks: a survey," *Journal of Network and Computer Applications*, vol. 35, no. 3, pp. 1022–1034, 2012.

[3] M. Zhang, V. Khanapure, S. Chen, and X. 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.

[4] Y. Zeng, J. Cao, S. Zhang, S. Guo, and L. 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] R. Di Pietro, L. V. Mancini, C. Soriente, A. Spognardi, and G. Tsudik, "Data security in unattended wireless sensor networks," *IEEE Transactions on Computers*, vol. 58, no. 11, pp. 1500–1511, 2009.

[6] C. Karlof and D. Wagner, "Secure routing in wireless sensor networks: attacks and countermeasures," in *Proceedings of the 1st IEEE International Workshop on Sensor Network Protocols and Applications*, May 2003.

[7] B. Parno, M. Luk, E. Gaustad, and A. Perrig, "Secure sensor network routing: a cleanslate approach," in *Proceedings of the ACM CoNEXT Conference (CoNEXT '06)*, December 2006.

[8] F. Ye, H. Luo, S. Lu, and L. Zhang, "Statistical en-route filtering of injected false data in sensor networks," in *Proceedings of the IEEE INFOCOM*, 2004.

[9] L. Yu and J. Li, "Grouping based resilient statistical en-route filtering for sensor networks," in *Proceedings of the IEEE INFOCOM*, 2009.

[10] S. Zhu, S. Setia, S. Jajodia, and P. Ning, "An interleaved hop-by-hop authentication scheme for filtering of injected false data in sensor networks," in *Proceedings of the IEEE Symposium on Security and Privacy*, pp. 259–271, May 2004.

[11] H. Chan, A. Perrig, and D. Song, "Secure hierarchical in-network aggregation in sensor networks," in *Proceedings of the13th ACM Conference on Computer and Communications Security (CCS '06)*, pp. 278–287, November 2006.

[12] J. Deng, R. Han, and S. Mishra, "Security support for in network processing in wireless sensor networks," in *Proceedings of the ACM Workshop on Security in Ad Hoc and Sensor Networks (SASN '03)*, pp. 83–93, 2003.

[13] B. Przydatek, D. Song, and A. Perrig, "SIA: secure information aggregation in sensor networks," in *Proceedings of the 1st International Conference on Embedded Networked Sensor Systems (SenSys '03)*, pp. 255–265, November 2003.

[14] Y. Yang, X. Wang, S. Zhu, and G. Cao, "SDAP: a secure hop-by-hop data aggregation protocol for sensor networks," in *Proceedings of the 7th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MOBIHOC '06)*, pp. 356–367, May 2006.

[15] S. Capkun and J. P. Hubaux, "Secure positioning in wireless networks," *IEEE Journal on Selected Areas in Communications*, vol. 24, no. 2, pp. 221–232, 2006.

[16] S. Ganeriwal, S. Čapkun, C. C. Han, and M. B. Srivastava, "Secure time synchronization service for sensor networks," in *Proceedings of the ACM Workshop on Wireless Security (WiSe '05)*, pp. 97–106, September 2005.

[17] X. Hu, T. Park, and K. G. Shin, "Attack tolerant time synchronization in wireless sensor networks," in *Proceedings of the 27th IEEE Conference on Computer Communications (INFOCOM '08)*, pp. 41–45, Phoenix, Ariz, USA, April 2008.

[18] Z. Li, W. Trappe, Y. Zhang, and B. Nath, "Robust statistical methods for securing wireless localization in sensor networks,"

in *Proceedings of the 4th International Symposium on Information Processing in Sensor Networks (IPSN '05)*, pp. 91–98, April 2005.

[19] D. Liu, P. Ning, and W. Du, "Attack-resistant location estimation in sensor networks," in *Proceedings of the 4th International Symposium on Information Processing in Sensor Networks (IPSN '05)*, pp. 99–106, April 2005.

[20] H. Song, S. Zhu, and G. Cao, "Attack resilient time synchronization for wireless sensor networks," *Ad Hoc Networks*, vol. 5, no. 1, pp. 112–125, 2007.

[21] K. Sun, P. Ning, C. Wang, A. Liu, and Y. Zhou, "TinySeRSync: secure and resilient time synchronization in wireless sensor networks," in *Proceedings of the 13th ACM Conference on Computer and Communications Security (CCS '06)*, pp. 264–277, 2006.

[22] C. Hartung, J. Balasalle, and R. Han, "Node compromise in sensor networks: the need for secure systems," Tech. Rep. CU-CS-988-04, Department of Computer Science, University of Colorado at Boulder, 2004.

[23] H. Choi, S. Zhu, and T. F. L. Porta, "SET: detecting node clones in sensor networks," in *Proceedings of the 3rd International Conference on Security and Privacy in Communication Networks (SecureComm '07)*, pp. 341–350, September 2007.

[24] S. Gautam Thakur, "CINORA: cell based identification of node replication attack in wireless sensor networks," in *Proceedings of the IEEE International Conference on Communications Systems (ICCS '08)*, 2008.

[25] S. Hussain and M. S. Rahman, "Using received signal strength indicator to detect node replacement and replication attacks in wireless sensor networks," in *Data Mining, Intrusion Detection, Information Security and Assurance, and Data Networks Security 2009*, vol. 7344 of *Proceedings of SPIE*, April 2009.

[26] J. Yick, B. Mukherjee, and D. Ghosal, "Wireless sensor network survey," *Computer Networks*, vol. 52, no. 12, pp. 2292–2330, 2008.

[27] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless sensor networks: a survey," *International Journal of Computer and Telecommunications Networking*, vol. 38, no. 4, pp. 393–422, 2002.

[28] B. Parno, A. Perrig, and V. Gligor, "Distributed detection of node replication attacks in sensor networks," in *Proceedings of the IEEE Symposium on Security and Privacy (IEEE S and P '05)*, pp. 49–63, May 2005.

[29] A. Seshadri, A. Perrig, L. van Doorn, and P. Khosla, "SWATT: softWare-based attestation for embedded devices," in *Proceedings of the IEEE Symposium on Security and Privacy (IEEE S and P '04)*, pp. 272–282, May 2004.

[30] R. D. Pietro, L. V. Mancini, C. Soriente, A. Spognardi, and G. Tsudik, "Catch me (If you can): data survival in unattended sensor networks," in *Proceedings of the 6th Annual IEEE International Conference on Pervasive Computing and Communications (PerCom '08)*, pp. 185–194, March 2008.

[31] F. Hu and N. K. Sharma, "Security considerations in ad hoc sensor networks," *Ad Hoc Networks*, vol. 3, no. 1, pp. 69–89, 2005.

[32] R. Brooks, P. Y. Govindaraju, M. Pirretti, N. Vijaykrishnan, and M. T. Kandemir, "On the detection of clones in sensor networks using random key predistribution," *IEEE Transactions on Systems, Man and Cybernetics C*, vol. 37, no. 6, pp. 1246–1258, 2007.

[33] L. Eschenauer and V. D. Gligor, "A key-management scheme for distributed sensor networks," in *Proceedings of the 9th ACM Conference on Computer and Communications Security*, pp. 41–47, Washington, DC, USA, November 2002.

[34] K. Xing, X. Cheng, F. Liu, and D. H. C. Du, "Real-time detection of clone attacks in wireless sensor networks," in *Proceedings of the 28th International Conference on Distributed Computing Systems (ICDCS '08)*, pp. 3–10, Beijing, China, July 2008.

[35] K. Xing, X. Cheng, L. Ma, and Q. Liang, "Superimposed code based channel assignment in multi-radio multi-channel wireless mesh networks," in *Proceedings of the 13th Annual ACM International Conference on Mobile Computing and Networking (MobiCom '07)*, pp. 15–26, September 2007.

[36] W. Znaidi, M. Minier, and S. Ubeda, "Hierarchical node replication attacks detection in wireless sensors networks," in *Proceedings of the 20th IEEE Personal, Indoor and Mobile Radio Communications Symposium (PIMRC '09)*, pp. 82–86, Tokyo, Japan, September 2009.

[37] D. Xia and N. Vlajic, "Near-optimal node clustering in wireless sensor networks for environment monitoring," in *Proceedings of the 21st International Conference on Advanced Networking and Applications (AINA '07)*, pp. 632–641, IEEE Computer Society, Washington, DC, USA, 2007.

[38] C. M. Yu, C. S. Lu, and S. Y. Kuo, "CSI: compressed sensing-based clone identification in sensor networks," in *Proceedings of the IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOM Workshops '12)*, pp. 290–295, Lugano, Switzerland, March 2012.

[39] A. J. Menezes, S. A. Vanstone, and P. C. V. Orschot, *Handbook of Applied Cryptography*, CRC Press, New York, NY, USA, 1996.

[40] C. Bekara and M. Laurent-Maknavicius, "A new protocol for securing wireless sensor networks against nodes replication attacks," in *Proceedings of the 3rd IEEE International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob '07)*, White Plains, NY, USA, October 2007.

[41] C. Bekara and M. Laurent-Maknavicius, "Defending against nodes replication attacks on wireless sensor networks," 2012, http://www-public.it-sudparis.eu/ lauren_m/articles/bekara-SARSSI07.pdf.

[42] M. Conti, R. Di Pietro, L. V. Mancini, and A. Mei, "A randomized, efficient, and distributed protocol for the detection of node replication attacks in wireless sensor networks," in *Proceedings of the 8th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc '07)*, pp. 80–89, September 2007.

[43] M. Conti, R. Di Pietro, L. Mancini, and A. Mei, "Distributed detection of clone attacks in wireless sensor networks," *IEEE Transactions on Dependable and Secure Computing*, vol. 8, no. 5, pp. 685–698, 2011.

[44] B. Zhu, V. G. K. Addada, S. Setia, S. Jajodia, and S. Roy, "Efficient distributed detection of node replication attacks in sensor networks," in *Proceedings of the 23rd Annual Computer Security Applications Conference (ACSAC '07)*, pp. 257–266, Miami Beach, Fla, USA, December 2007.

[45] B. Zhu, S. Setia, S. Jajodia, S. Roy, and L. Wang, "Localized multicast: efficient and distributed replica detection in large-scale sensor networks," *IEEE Transactions on Mobile Computing*, vol. 9, no. 7, pp. 913–926, 2010.

[46] S. Ratnasamy, B. Karp, L. Yin et al., "GHT: a geographic hash table for data-centric storage," in *Proceedings of the 1st ACM International Workshop on Wireless Sensor Networks and Applications (WSNA '02)*, pp. 78–87, September 2002.

[47] F. Fei, L. Jing, and Y. Xianglan, "Space-time related pairwise key predistribution scheme for wireless seneor networks," in

Proceedings of the International Conference on Wireless Communications, Networking and Mobile Computing (WiCOM '07), pp. 2692–2696, Shanghai, China, September 2007.

[48] L. C. Ko, H. Y. Chen, and G. R. Lin, "A neighbor-based detection scheme for wireless sensor networks against node replication attacks," in *Proceedings of the International Conference on Ultra Modern Telecommunications and Workshops (ICUMT '09)*, pp. 1–6, St. Petersburg, Russia, October 2009.

[49] J. W. Ho, "Distributed detection of node capture attacks in wireless sensor networks," in *Smart Wireless Sensor Networks*, H. D. Chunch and Y. K. Tan, Eds., pp. 345–360, InTech, Rijeka, Croatia, 2010.

[50] A. Wald, *Sequential Analysis*, Dover, New York, NY, USA, 2004.

[51] C. A. Melchor, B. Ait-Salem, P. Gaborit, and k. Tamine, "Active detection of node replication attacks," *International Journal of Computer Science and Network Security*, vol. 9, no. 2, pp. 13–21, 2009.

[52] Z. Li and G. Gong, "Randomly directed exploration: an efficient node clone detection protocol in wireless sensor networks," in *Proceedings of the 6th IEEE International Conference on Mobile Adhoc and Sensor Systems (MASS '09)*, pp. 1030–1035, Macau, China, October 2009.

[53] X. Meng, K. Lin, and K. Li, "Note based randomized and distributed protocol for detecting node replication attack," in *Algorithms and Architectures for Parallel Processing*, vol. 6081 of *Lecture Notes in Computer Science*, pp. 559–570, 2010.

[54] J. W. Ho, D. Liu, M. Wright, and S. K. Das, "Distributed detection of replica node attacks with group deployment knowledge in wireless sensor networks," *Ad Hoc Networks*, vol. 7, no. 8, pp. 1476–1488, 2009.

[55] Y. Sei and S. Honiden, "Distributed detection of node replication attacks resilient to many compromised nodes in wireless sensor networks," in *Proceedings of the 4th Annual International Conference on Wireless Internet (WICON '08)*, 2008.

[56] C. Kim, S. Shin, C. Park, and H. Yoon, "A resilient and efficient replication attack detection scheme for wireless sensor networks," *IEICE Transactions on Information and Systems*, vol. 92, no. 7, pp. 1479–1483, 2009.

[57] B. Dutertre, S. Cheung, and J. Levy, "Lightweight key management in wireless sensor networks by leveraging initial trust," SDL Technical Report SRI-SDL-04-02, 2004.

[58] J. W. Ho, M. Wright, and S. K. Das, "Fast detection of mobile replica node attacks in wireless sensor networks using sequential hypothesis testing," *IEEE Transactions on Mobile Computing*, vol. 10, no. 6, pp. 767–782, 2011.

[59] J. W. Ho, M. Wright, and S. K. Das, "Fast detection of replica node attacks in mobile sensor networks using sequential analysis," in *Proceedings of the IEEE INFOCOM*, pp. 1773–1781, Rio de Janeiro, Brazil, April 2009.

[60] X. M. Deng and Y. Xiong, "A new protocol for the detection of node replication attacks in mobile wireless sensor networks," *Journal of Computer Science and Technology*, vol. 26, no. 4, pp. 732–743, 2011.

[61] C. M. Yu, C. S. Lu, and S. Y. Kuo, "Mobile sensor network resilient against node replication attacks," in *Proceedings of the 5th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks (SECON '08)*, pp. 597–599, June 2008.

[62] C. M. Yu, C. S. Lu, and S. Y. Kuo, "Efficient and Distributed Detection of Node Replication Attacks in Mobile Sensor Networks," in *Proceedings of the 70th IEEE Vehicular Technology Conference (VTC Fall '09)*, pp. 20–23, Anchorage, Alaska, USA, September 2009.

[63] L. M. Wang and Y. Shi, "Patrol detection for replica attacks on wireless sensor networks," *Sensors*, vol. 11, no. 3, pp. 2496–2504, 2011.

[64] Y. Lou, Y. Zhang, and S. Liu, "Single hop detection of node clone attacks in mobile wireless sensor networks," in *Proceedings of the International Workshop on Information and Electronics Engineering (IWIEE)*, 2012.

[65] W. T. Zhu, J. Zhou, R. Deng, and F. Bao, "Detecting node replication attacks in mobile sensor networks: theory and approaches," *Security and Communication Networks*, vol. 5, no. 5, pp. 496–507, 2012.

[66] M. Conti, R. Di Pietro, L. V. MAncini, and A. Mei, "Emergent properties: detection of the node-capture attack in mobile wireless sensor networks," in *Proceedings of the 1st ACM Conference on Wireless Network Security (WiSec '08)*, pp. 214–219, Alexandria, Va, USA, 2008.

[67] X. Deng, Y. Xiong, and D. Chen, "Mobility-assisted detection of the replication attacks in mobile wireless sensor networks," in *Proceedings of the 6th Annual IEEE International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob '2010)*, pp. 225–232, October 2010.

[68] I. Amundson and X. D. Koutsoukos, "A survey on localization for mobile wireless sensor networks," in *Proceedings of the 2nd International Conference on Mobile Entity Localization and Tracking in GPS-Less Environments (MELT '09)*, vol. 5801 of *Lecture Notes in Computer Science*, pp. 235–254, 2009.

[69] H. Chan, A. Perrig, and D. Song, "Random key predistribution schemes for sensor networks," in *Proceedings of the IEEE Symposium on Security And Privacy (IEEE S and P '03)*, pp. 197–213, May 2003.