ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com
International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 3, Special Issue 19, April 2016

Energy and Memory Efficient Clone Detection in Wireless Sensor Networks

[1]E.Sujatha M.Tech., M.B.A., Ph.D.,, [2]R. Christy Priya, B.Tech.,, [3]N. Kanimozhi, B.Tech.,, [4]L. Joys Kiruba, B.Tech.,[5]D. Deborah, B.Tech.,

[1] Associate Professor ,[2,3,4] UG Scholars, Department of Information Technology
Kings Engineering College, Irungattukottai.Chennai,India
sanjaymohankumar@gmail.com[1],christysmarty94@gmail.com[2],kanimozhimay26@gmail.com[3],
joyskiruba1994@gmail.com[4], deborahmercyangel@gmail.com[5]

Abstract — **In this paper, we propose an energy efficient location-aware clone detection protocol in densely deployed WSNs, which can guarantee successful clone attack detection and maintain satisfactory network lifetime. Specifically, we exploit the location information of sensors and randomly select witnesses located in a ring area to verify the legitimacy of sensors and to report detected clone attacks. The ring structure facilitates energy efficient data forwarding along the path towards the witnesses and the sink. We theoretically prove that the proposed protocol can achieve 100% clone detection probability with trustful witnesses. We further extend the work by studying the clone detection performance with untruthful witnesses and show that the clone detection probability still approaches 98% when 10% of witnesses are compromised. Moreover, in most existing clone detection protocols with random witness selection scheme, the required buffer storage of sensors is usually dependent on the node density, i.e., O(n), while in our proposed protocol, the required buffer storage of sensors is independent of *n* but a function of the hop length of the network radius *h*, i.e., O(h). Extensive simulations demonstrate that our proposed protocol can achieve long network lifetime by effectively distributing the traffic load across the network.**

Index Terms — Keywords: wireless sensor networks, clone detection protocol, energy efficiency, and network lifetime

I. INTRODUCTION

Wireless sensors have been widely deployed for a variety of applications, ranging from environment monitoring to telemedicine and objects tracking, etc. [2]–[4]. For cost-effective sensor placement, sensors are usually not tamper-proof devices and are deployed in places without monitoring and protection, which makes them prone to different attack-s [5]–[9]. For example, a malicious user may compromise some sensors and acquire their private information. Then, it can duplicate the sensors and deploy clones in a wireless sensor network (WSN) to launch a variety of attacks [10],which is referred to as the clone attack [11]–[13]. As the duplicated sensors have the same information, e.g., code and cryptographic information, captured from legitimate sensors, they can easily participate in network operations and launch attacks.

Due to the low cost for sensor duplication and deployment, clone attacks have become one of the most critical security issues in WSNs. Thus, it is essential to effectively detect clone attacks in order to ensure healthy operation of WSNs.

To allow efficient clone detection, usually, a set of nodes are selected, which are called witnesses, to help certify the legitimacy of the nodes in the network. The private information of the source node, i.e., identity and the location information are shared with witnesses at the stage of witness selection. When any of the nodes in the network wants to transmit data, it first sends the request to the witnesses for legitimacy verification, and witnesses will report a detected attack if the node fails the certification. To achieve successful clone detection, witness selection and legitimacy verification should fulfil two requirements: 1) witnesses should be randomly selected; and 2) at least one of the witnesses can successfully receive all the verification message(s) for clone detection [11]. The first requirement is to make it difficult for malicious users eavesdrop the communication between the current source node and its witnesses, so that the malicious users cannot generate duplicate verification messages. The second requirement is to make sure that at least one of the witnesses can check the identity of the sensor nodes to determine whether there is a clone attack or not. To guarantee a high clone detection probability, i.e., the probability that clone attacks can be successfully detected, it is critical and challenging to fulfil these requirements in clone detection protocol design. Different from wireless terminal devices, wireless sensors are usually of smaller size and lower price, and have limited battery and memory capacity. Therefore, the design criteria of clone detection protocols for sensor networks should not only guarantee the high performance of clone detection probability but also consider the energy and memory efficiency of sensors. In the literature, some distributed clone detection protocols have been proposed, such as Randomized Efficient and Distributed protocol (RED) [10] and Line-Select Multi-cast protocol (LSM) [11]. However, most approaches mainly focus on improving clone detection probability without considering efficiency and balance of energy consumption in WSNs. With such kind of approaches, some sensors may use up their batteries due to the unbalanced energy consumption, and dead sensors may cause network partition, which may further affect the normal operation of WSNs. Christo Ananth et al. [3] discussed about a system, In this proposal, a neural network approach is proposed for energy conservation routing in a wireless sensor network. Our designed neural network system has been successfully applied to our scheme of energy conservation. Neural network is applied to predict Most Significant Node and selecting the Group Head amongst the association of sensor nodes in the network. After having a precise prediction about Most Significant Node, we would like to expand our approach in future to different WSN power management techniques and observe the results. In this proposal, we used arbitrary data for our experiment purpose; it is also expected to generate a real time data for the experiment in future and also by using adhoc networks the energy level of the node can be maximized. The selection of Group Head is proposed using neural network with feed forward learning method. And the neural network found able to select a node amongst competing nodes as Group Head.

Most existing approaches can improve the successful clone detection at the expense of energy consumption and memory storage, which may not be suitable for some sensor networks with limited energy resource and memory storage.

In this paper, besides the clone detection probability, we also consider energy consumption and memory storage in the design of clone detection protocol, i.e., an energy and memory efficient distributed clone detection protocol with random witness selection scheme in WSNs. Our protocol is applicable to general densely deployed multi-hop WSNs, where adversaries may compromise and clone sensor nodes to launch attacks. A preliminary work is presented in [1]. In that work, we proposed an energy-efficient ring based clone detection (ERCD) protocol to achieve high clone detection probability with random witness selection, while ensuring normal network operations with satisfactory network lifetime of WSNs. The ERCD protocol can be divided into two stages: witness selection and legitimacy verification. In wit-ness selection, the source node sends its private information to a set of witnesses, which are randomly selected by the mapping function. In the legitimacy verification, verification message along the private information of the source node is transmitted to its witnesses. If any of witnesses successfully receives the message, it will forward the message to its witness header for verification. Upon receive the messages, the witness header compares the aggregated verification messages with stored records. If multiple copies of verification messages are received, the clone attack is detected and a revocation procedure will be triggered. As such, to have a comprehensive study of the ERCD protocol, we extend the analytical model by evaluating the required data buffer of ERCD protocol and by including experimental results to support our theoretical analysis. First, we theoretically prove that our proposed clone detection protocol can achieve probability 1 based on trustful witnesses. Considering the scenario that witnesses can be compromised, our simulation results demonstrate that the clone detection probability can still approach 98% in WSNs with 10% cloned nodes by us-ing the ERCD protocol. Second, to evaluate the performance of network lifetime, we derive the expression of total energy consumption, and then compare our protocol with existing clone detection protocols. We find that the ERCD protocol can balance the energy consumption of sensors at different locations by distributing the witnesses all over WSNs except non-witness rings, i.e., the adjacent rings around the sink, which should not have witnesses. After that, we obtain the optimal number of non-witness rings based on the function of energy consumption. Finally, we derive the expression of the required data buffer by using ERCD protocol, and show that our proposed protocol is scalable because the required buffer storage is dependent on the ring size only. Extensive simulation results demonstrate that our proposed ERCD protocol can achieve superior performance in terms of the clone detection probability and network lifetime with reasonable data buffer capacity.

We present the remainder of this paper as follows. We summarize the previous works of clone detection protocols in Section II. In Section III, the system model and problem statement are introduced. The ERCD protocol is proposed in Section IV. Then, we analyze the performance of the ERCD protocol in terms of clone detection probability, network lifetime and data buffer storage in Section V. Experiment results are presented in Section VI, followed by the conclusion in Section VII.

II. RELATED WORK

As one of the utmost important security issues, clone attack has attracted people's attention. There are many works [14]–[16] that studies clone detection protocols in the literature, which can be classified into two different categories, i.e., centralized and distributed clone detection protocols. In centralized protocols, the sink or witnesses generally locate in the center of each region, and store the private information of sensors. When the sink or witnesses receive the private information of the source node, they can determine whether there is a clone attack by comparing the private information with its pre-stored records [17], [18]. Normally, centralized clone detection protocols have low overhead and running complexity. However, the security of sensors' private information may not be guaranteed, because the malicious users can eavesdrop the transmission between the sink node and sensors. Moreover, the network lifetime may be dramatically decreased since the sensor nodes close to the sink will deplete their energy sooner than other nodes.

Different from centralized protocols, in distributed clone detection protocols, a set of witnesses are selected to match with every sensor [10], [11], which prevents the transmission between the sink and sensors from being eavesdropped by malicious users. There are three different types of witness selection schemes in distributed clone detection protocols: i) deterministic selection, ii) random selection, and iii) semi-random selection. The deterministic witness selection based clone detection protocols like RED [10] choose same set of witnesses for all sensor nodes. By using deterministic witness selection, a low communication overhead and a high clone detection probability can be achieved. In addition, the required buffer storage capacity of such protocols is very low, which is only related to the number of witnesses without considering network scale and node density. Nevertheless, due to the deterministic characteristic, the mapping function can be easily obtained and a variety of attacks may be launched by malicious users. To enhance the network security, the distributed clone detection protocols with random witness selection [11], [12] like LSM are proposed, which are closely related to our work. In random witness selection, it is difficult for malicious users to acquire the information of witnesses since the witnesses of each sensor are randomly generated. However, the randomness of mapping function also increases the difficulty for the source node to reach its witnesses, which makes it challenging to achieve a high clone detection probability. To ensure the clone detection probability, LSM lets all the nodes in the route between source and witnesses store the private information of the source node, which leads to a high requirement of data buffer and energy consumption. Thus, it is essential to guarantee the clone detection probability with low energy consumption and required buffer storage in clone detection protocols with random witness selection approach. Other distributed clone detection protocols, such as Parallel Multiple Probabilistic Cells (P-MPC), proposed semi-random witness selection approach [13], [19], trying to combine the advantages of both random and deterministic witness selection approaches. In this kind of witness selection scheme, a deterministic region is generated for the source node according to the mapping function, and then witnesses of the source node will be randomly selected from the sensors in this region. However, the two-phases witness selection and randomness of the witnesses for each sensor leads to a high overhead and time complexity. The energy consumption and the required buffer storage of such protocols are lower than the random witness selection approach but higher than the deterministic ones. Overall, most previous works aim at maximizing the clone detection probability without considering the impact of proposed clone detection protocol on the network lifetime and required data buffer storage. In this paper, we carefully design a distributed clone detection protocol with random witness selection by jointly considering the clone detection probability, network lifetime and data buffer capacity.

III. SYSTEM MODEL AND PROBLEM STATEMENT

In this work, we consider a network region with one base station (BS) and an enormous number of wireless sensor nodes randomly distributed in the network. We use the sink node as the origin of the system coordinator. Based on the location of the BS, the network region is virtually separated into adjacent rings, where the width of each ring is the same as the transmission range of sensor nodes. The network is a densely deployed WSN, i.e., i) for each node, there exist sensor nodes located in each neighbouring ring, and ii) for each ring, in each ring, there are enough sensor nodes to construct a routing path along the ring. The network model can be simply extended into the case of multiple BSs, where different BSs use orthogonal frequency-division multiple access (OFDMA) to communication with its sensor nodes. For each sensor, it has to accomplish the tasks of data collection as well as clone detection. In every data collecting cycle, sensors send the collected data to the sink node through multi-hop paths. To be capable of conducting legitimacy verification, every sensor has the same buffer storage capacity to store the information. Buffer storage capacity should be sufficient to store the private information of source nodes, such that any node can be selected as a witness. When the buffer storage of the sensor node is full, the oldest information will be dropped to accept the latest incoming information.

In our network, the link level security can be guaranteed by employing a conventional bootstrapping cryptography scheme, and the sink node uses a powerful cryptography scheme, which cannot be compromised by malicious users. A key pair (a; b) is assigned to each node, where a and b are the node ID and the node secret key, respectively. All nodes share their ID information with other nodes in the network. If either side of the link is compromised by malicious users, the link key is compromised. Each sensor node knows the physical information and the relative locations of its neighbours, where the relative location refers to the hop distance between a sensor node and the sink, and the hop distance can be obtained by a breadth-first search. At first, the sink node broadcasts the message, which notifies the receivers that the message comes from index 0. All nodes, which receive the message, will update their ring index to 1 and rebroadcast the message to their neighbours. Each node will update the ring index only when the message has a lower ring index than that it received in previous transmissions. The above procedure repeats until all the nodes broadcast the message and record their ring indexes. A malicious user has the capability to compromise a set of sensor nodes located at arbitrary locations. Utilizing the private information of compromised nodes, a large number of cloned nodes can be generated and deployed into the network by the malicious user [10], [11]. However, we suppose that malicious users cannot compromise the majority of sensor nodes, since no protocol can successfully detect the clone attack with little legitimate sensor nodes [10], [11], [20].

In this paper, we focus on designing a distributed clone detection protocol with random witness selection by jointly considering clone detection probability, network lifetime and data buffer storage. Initially, a small set of nodes are compromised by the malicious users. Utilizing the clone detection protocol, we aim at maximizing the clone detection probability, i.e., the probability that cloned node can be successfully detected, to ensure the security of WSNs; mean-while, the sufficient energy and buffer storage capacity for data collection and operating clone detection protocol should be guaranteed, which means that the network lifetime, i.e., the period from the start of network operation until the first outage occurs [21], [22], should not be impacted by the proposed clone detection protocol with sensors' buffer storage. Overall, our objective is to propose a distributed clone detection protocol with random witness selection in order to maximize the clone detection probability while the negative impact of network lifetime and the requirement of data buffer storage should be minimized.

IV. ERCD PROTOCOL

In this section, we introduce our distributed clone detection protocol, namely ERCD protocol, which can achieve a high clone detection probability with little negative impact on network lifetime and limited requirement of buffer storage capacity. The ERCD protocol consists of two stages: witness selection and legitimacy verification. In witness selection, a random mapping function is employed to help each source node randomly select its witnesses. In the legitimacy verification, a verification request is sent from the source node to its witnesses, which contains the private information of the source node. If witnesses receive the verification messages, all the messages will be forwarded to the witness header for legitimacy verification, where witness headers are nodes responsible for determining whether the source node is legitimacy or not by comparing the messages collected from all witnesses. If the received messages are different from existing record or the messages are expired, the witness header will report a clone attack to the sink to trigger a revocation procedure.

Initially, the network region is virtually divided into h adjacent rings, where each ring has a sufficiently large number of sensor nodes to forward along the ring and the width of each ring is r. To simplify the description we use hop length to represent the minimal number of hops in the paper. Since we consider a densely deployed WSN, hop length of the network is the quotient of the distance from the sink to the sensor at the border of network region over the transmission range of each sensor, i.e., the distance of each hop refers to the transmission range of sensor nodes. TABLE I shows the mathematical symbols utilized in this section.

The ERCD protocol starts with a breadth-first search by the sink node to initiate the ring index, and all neighbouring sensors periodically exchange the relative location and ID information [23], [24]. After that, whenever a sensor node establishes a data transmission to others, it has to run the ERCD protocol, i.e., witness selection and legitimacy verification, to verify its legitimacy.

In witness selection, a ring index is randomly selected by the mapping function as the witness ring of node a. To help relieve the traffic load in hot spot, the area around the sink cannot be selected by the mapping function. After that, node a sends its private information to the node located in witness ring, and then the node forward the information along the witness ring to form a ring structure. In the legitimacy verification, a verification message of the source node is forwarded to its witnesses. The ring index of node a, denoted O_a, is compared with its witness ring index O_a^w to determine the next forwarding node. If $O_a^w > O_a$, the message will be forwarded to any node located in ring $O_a + 1$; otherwise, the message will be forwarded to any node in ring O_a 1. This step can forward the message toward the witness ring of node a. The ERCD protocol repeats above operations until a node, denoted b, located in the witness ring O_a^w is reached. Node b stores the private information of node a and forwards the message to any node located in ring O_a^w within its transmission range, denoted as c. Then, node c stores the information and forwards the message to the node d, where link (c,d) has longest projection on the extension line of the directional link from b to c. The procedure will be repeated until node b reappears in the transmission range. Therefore, the witnesses of node a have a ring structure, consisting of b,c,..b as shown in Fig. 1.

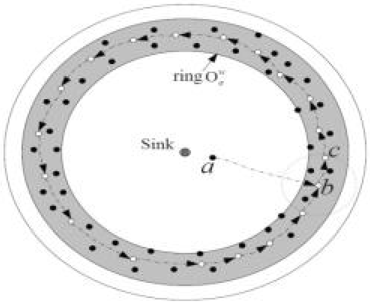


Fig.1 Ring structure of witness

In the legitimacy verification, node a sends a verification message including its private information following the same path towards the witness ring as in witness selection. To enhance the probability that witnesses can successfully receive the verification message for clone detection, the message will be broadcast when it is very close to the witness ring, namely three-ring broadcasts, i.e., the message will be broadcast in O_a^w 1, O_a^w and O_a^w +1 as shown in Fig 2.

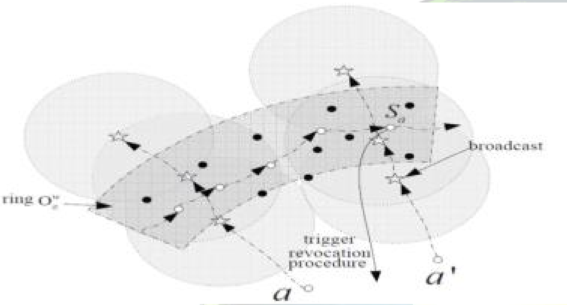


Fig.2 Legitimacy verification

In Theorem 1, we prove that the three ring broadcasts can ensure the network security, i.e., the clone detection probability is one, under the assumption that all witnesses are trustful. To determine whether there exists a clone attack or not, all the verification messages received by witnesses are forwarded to the witness header along the same route in witness selection. The sensors nodes in the transmission route but not located in the witness ring are called the transmitters. The witness header of the source node a, denoted by S_a, is a sensor located in witness ring O_a^w, meanwhile it is also in the communication range of the transmitter located in ring index O_a^w 1 or O_a^w + 1. The witness header S_a is randomly selected by the transmitter in the neighbouring witness ring, i.e., the ring of O_a^w 1 or O_a^w + 1. If more than one copy or incorrect copies or expired copies are received by the witness header, the ERCD protocol will trigger a revocation procedure; if no copy is received from the source node due to packet loss or silent cloned node, transmissions from the source node will not be permitted.

An example is shown in Fig. 2. Let a and a^0 denote the source node and one cloned node. The verification messages of both a and a^0 are broadcast in ring O_a^w 1, O_a^w and O_a^w+1. After that, both messages are received by the witness header S_a, and a revocation procedure is triggered. We describe the detail of the ERCD protocol in Algorithm 1.

In addition to the normal operations, the recovery mechanism is very easy to be established based on ERCD protocol. For the case when the clone detection fails due to outage or clone attack, another clone detection cycle will be initiated and the source node will randomly choose a new route and forward the message en route to a new witness header.

V. AODV PROTOCOL

A wireless ad-hoc network, also known as IBSS Independent Basic Service Set, is a computer network in which the communication links are wireless. The network is ad-hoc because each node is willing to forward data for other nodes, and so the determination of which nodes forward data is made dynamically based on the network connectivity. This is in contrast to older network technologies in which some designated nodes, usually with custom hardware and variously known as routers, switches, hubs, and firewalls, perform the task of forwarding the data. Minimal configuration and quick deployment make ad hoc networks suitable for emergency situations like natural or human-induced disasters, military conflicts.

A major limitation with mobile nodes is that they have high mobility, causing links to be frequently broken and re established. Moreover, the bandwidth of a wireless channel is also limited, and nodes operate on limited battery power, which will eventually be exhausted. Therefore, the design of a mobile ad hoc network is highly challenging, but this technology has high prospects to be able to manage communication protocols of the future. The cross-layer design deviates from the traditional network design approach in which each layer of the stack would be made to operate independently. The modified transmission power will help that node to dynamically vary its propagation range at the physical layer. This is because the propagation distance is always directionally proportional to transmission power. This information is passed from the physical layer to the network layer so that it can take optimal decisions in routing protocols. A major advantage of this protocol is that it allows access of information between physical layer and top layers (MAC and network layer).

As in a fix net nodes maintain routing tables. Distance vector protocols are based on calculating the direction and distance to any link in a network. "Direction" usually means the next hop address and the exit interface. "Distance" is a measure of the cost to reach a certain node. The least cost route between any two nodes is the route with minimum distance. Each node maintains a vector (table) of minimum distance to every node. The cost of reaching a destination is calculated using various route metrics. RIP uses the hop count of the destination whereas IGRP takes into account other information such as node delay and available bandwidth.

One key problem in wireless ad hoc networks is foreseeing the variety of possible situations that can occur. As a result, Modelling and Simulation (M&S) using extensive parameter sweeping and what-if analysis becomes an extremely important paradigm for use in ad hoc networks. Traditional M&S tools include NS2 (and recently NS3), OPNET Modeller, and NetSim.

However, these tools focus primarily on the simulation of the entire protocol stack of the system. Although this can be important in the proof-of-concept implementations of systems, the need for a more advanced simulation methodology is always there. Agent-based modelling and simulation offers such a paradigm. Not to be confused with multi-agent systems and intelligent agents, agent-based modelling originated from social sciences, where the goal was to evaluate and view large-scale systems with numerous interacting "AGENT" or components in a wide variety of random situations to observe global phenomena. Unlike traditional AI systems with intelligent agents, agent-based modelling is similar to the real world. Agent-based models are thus effective in modelling bio-inspired and nature-inspired systems. In these systems, the basic interactions of the components of the system, also called a complex adaptive system, are simple but result in advanced global phenomena such as emergence The Path Discovery process is initiated whenever a source node needs to communicate with another node for which it has no routing information in its table. Every node maintains two separate counters: a node sequence number and a broadcast id. The source node initiates path discovery by broadcasting a route request (RREQ) packet to its neighbours. Each neighbour either satisfies the RREQ by sending a route reply (RREP) back to the source), or rebroadcasts the RREQ to its own neighbours after increasing the hop_cnt. Notice that a node may receive multiple copies of the same route broadcast packet from various neighbours. When an intermediate node receives a RREQ, if it has already received a RREQ with the same broadcast id and source address, it drops the redundant RREQ and does not rebroadcast it. If a node cannot satisfy the RREQ, it keeps track of the following information in order to implement the reverse path setup, as well as the forward path setup that will accompany the transmission of the eventual RREP:

- Destination IP Address
- Source IP Address
- Broadcast_id
- Expiration time for reverse path route entry
- Source node's sequence number

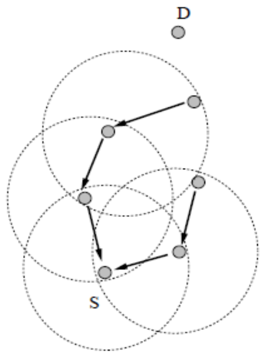


Fig.3 Reverse path Formation

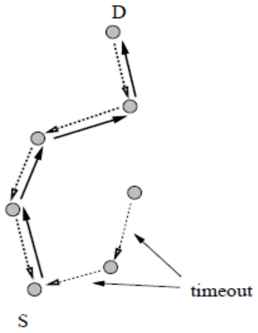


Fig.4 Forward path Formation

Reverse Path Setup

There are two sequence numbers (in addition to the broadcast id) included in a RREQ: the source sequence number and the last destination sequence number known to the source. The source sequence number is used to maintain freshness information about the re-verse route to the source, and the destination sequence number specifies how fresh a route to the destination must be before it can be accepted by the source.

As the RREQ travels from a source to various destinations, it automatically sets up the reverse path from all nodes back to the source [4], as illustrated in Figure 1. To set up a reverse path, a node records the address of the neighbour from which it received the _rst copy of the RREQ. These reverse path route entries are maintained for at least enough time for the RREQ to traverse the network and produce a reply to the sender.

Forward Path Setup

Eventually, a RREQ will arrive at a node (possibly the destination itself) that possesses a current route to the destination. The receiving node _rst checks that the RREQ was received over a bi-directional link. If an intermediate node has a route entry for the desired destination, it determines whether the route is current by comparing the destination sequence number in its own route entry to the destination sequence number in the RREQ. If the RREQ's sequence number for the destination is greater than that recorded by the inter-mediate node, the intermediate node must not use its recorded route to respond to the RREQ. Instead, the intermediate node rebroadcasts the RREQ. The intermediate node can reply only when it has a route with a sequence number that is greater than or equal to that contained in the RREQ. If it does have a current route to the destination, and if the RREQ has not been processed previously, the node then unicasts a route reply packet

(RREP) back to its neighbour from which it received the RREQ

Route Table Management

In addition to the source and destination sequence numbers, other useful information is also stored in the route table entries, and is called the soft-state associated with the entry. Associated with reverse path routing entries is a timer, called the route request expiration timer. The purpose of this timer is to purge reverse path routing entries from those nodes that do not lie on the path from the source to the destination. The expiration time depends upon the size of the ad-hoc network. Another important parameter associated with routing entries is the route caching timeout, or the time after which the route is considered to be invalid.

In each routing table entry, the address of active neighbours through which packets for the given destination are received is also maintained. A neighbour is considered active (for that destination) if it originates or relays at least one packet for that destination within the most recent active timeout period. This in-formation is maintained so that all active source nodes can be notified when a link along a path to the destination breaks. A route entry is considered active if it is in use by any active neighbours. The path from a source to a destination, which is followed by pack-ets along active route entries, is called an active path. Note that, as with DSDV, all routes in the route table are tagged with destination sequence numbers, which guarantee that no routing loops can form, even under extreme conditions of out-of-order packet delivery and high node mobility (see Appendix A).

A mobile node maintains a route table entry for each destination of interest. Each route table entry contains the following information:

- Destination
- Next Hop
- Number of hops (metric)
- Sequence number for the destination
- Active neighbours for this route
- Expiration time for the route table entry

Each time a route entry is used to transmit data from a source toward a destination, the timeout for the entry is reset to the current time plus active route timeout.

If a new route is offered to a mobile node, the mobile node compares the destination sequence number of the new route to the destination sequence number for the current route. The route with the greater sequence number is chosen. If the sequence numbers are the same, then the new route is selected only if it has a smaller metric (fewer numbers to the destination.

VI. CLONE DETECTION

Sensor nodes that are deployed in hostile environments are vulnerable to capture and compromise. An adversary may obtain private information from these sensors, clone and intelligently deploy them in the network to launch a variety of insider attacks. This attack process is broadly termed as a clone attack. Currently, the defenses against clone attacks are not only very few, but also suffer from selective interruption of detection and high overhead (computation and memory). In this paper, we propose a new effective and efficient scheme, called SET, to detect such clone attacks. The key idea of SET is to detect clones by computing set operations (intersection and union) of exclusive subsets in the network.

First, SET securely forms exclusive unit subsets among one-hop neighbours in the network in a distributed way. This secure subset formation also provides the authentication of nodes' subset membership. SET then employs a tree structure to compute non-overlapped set operations and integrates interleaved authentication to prevent unauthorized falsification of subset information during forwarding. Randomization is used to further make the exclusive subset and tree formation unpredictable to an adversary. We show the reliability and resilience of SET by analyzing the probability that an adversary may effectively obstruct the set operations. Performance analysis and simulations also demonstrate that the proposed scheme is more efficient than existing schemes from both communication and memory cost standpoints.

VII. PERFORMANCE ANALYSIS

In this section, the performance of the ERCD protocol is evaluated in terms of clone detection probability, power consumption, network lifetime, and data buffer capacity. At first, we prove that the clone detection probability of the ERCD protocol can almost surely achieve probability 1 under the scenario that witnesses are trustful in Subsection V-A. Then, we derive the expression of energy consumption and network lifetime by using ERCD protocol, and obtain the ratio of network lifetime by using ERCD protocol over RED or LSM protocol in Subsection V-B. Finally, the required data buffer of the ERCD protocol is derived in Subsection V-C.

A. Probability of Clone Detection

In distributed clone detection protocol with random witness selection, the clone detection probability generally refers to whether witnesses can successfully receive the verification message from the source node or not. Thus, the clone detection probability of ERCD protocol is the probability that the verification message can be successfully transmitted from the source node to its witnesses. In ERCD protocol, the verification message is broadcast when it is near the witness ring, i.e., in the rings of O_a^w 1, O_a^w and O_a^w +1, to guarantee the network security. With such kind of method and assumption of trustful witnesses, we can prove that at least one of the witnesses can receive the message, i.e., the clone attack can be detected with probability one. To simplify the analysis, the transmission ranges of all sensor nodes, r, are the same.

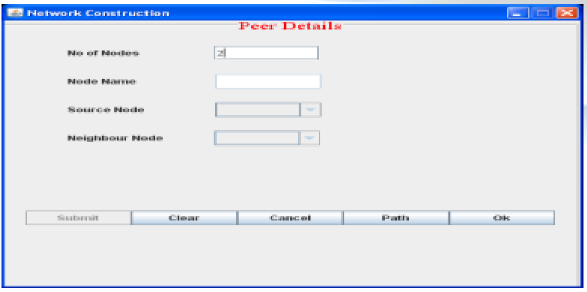
B. Energy Consumption and Network Lifetime

In WSNs, since wireless sensor nodes are usually powered by batteries, it is critical to evaluate the energy consumption of sensor nodes and to ensure that normal network operations will not be broken down by node outage. Therefore, we define the network lifetime as the period from the start of network operation until any node outage occurs to evaluate the performance of the ERCD Protocol. We only considered the transmission power consumption, as the reception power consumption occupies little percentage of total power consumption. Since witness sets in our ERCD Protocol are generated based on ring structure, sensor nodes in the same ring as similar task. To simplify the analysis, we suppose that all sensor nodes in the same ring have same traffic load. Our analysis in this work is generic, which can be applied to various energy model.

C. Data buffer capacity

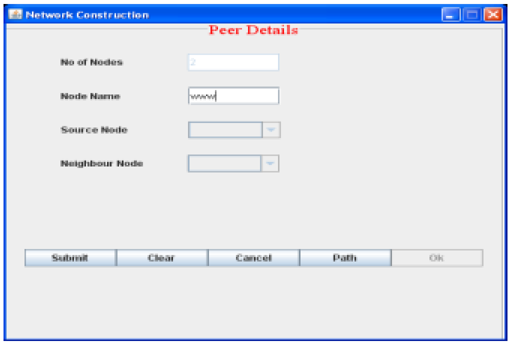
Usually, sensors are of small size and have very limited capacity of both data buffer and energy battery. In this subsection, we analyze the required data buffer capacity, also referred to as data buffer of sensors to evaluate the performance of the proposed ERCD protocol. Let denote the required packet storage size for being a witness of a sensor node.

VIII. SCREEN SHOTS



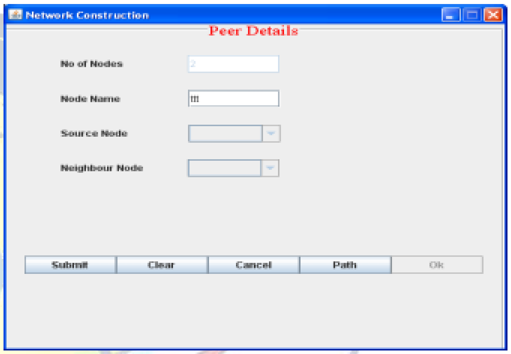
No of Nodes 2
Node Name
Source Node
Neighbour Node

Fig.5 Specifies the no of Nodes



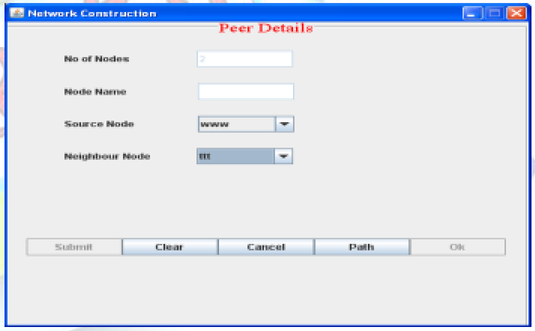
No of Nodes 2
Node Name www
Source Node
Neighbour Node

Fig.6 First Node Creation



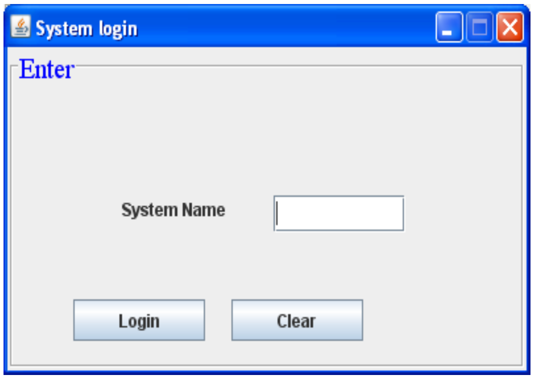
No of Nodes 2
Node Name ttt
Source Node
Neighbour Node

Fig.7 Second Node Creation



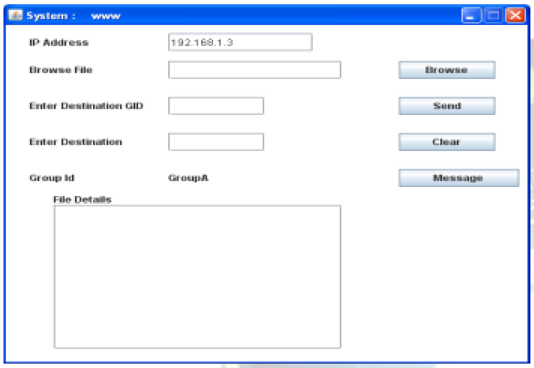
No of Nodes 2
Node Name
Source Node www
Neighbour Node ttt

Fig.8 Path Setting for Data Transfer



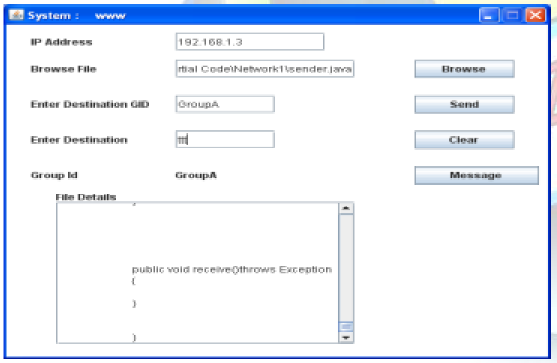
System Name

Fig.9 Specifies the Source Node



IP Address 192.168.1.3
Browse File
Enter Destination GID
Enter Destination
Group Id GroupA
File Details

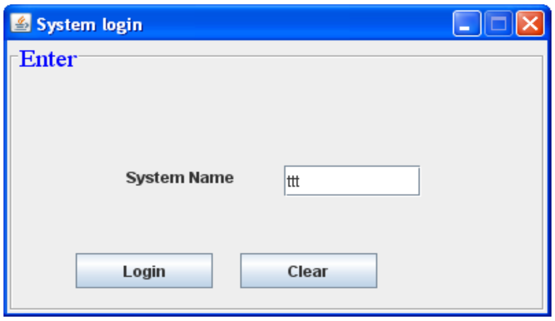
Fig.10 Source Home login form



IP Address 192.168.1.3
Browse File rtial Code\Network1\sender.java
Enter Destination GID GroupA
Enter Destination ttt
Group Id GroupA
File Details

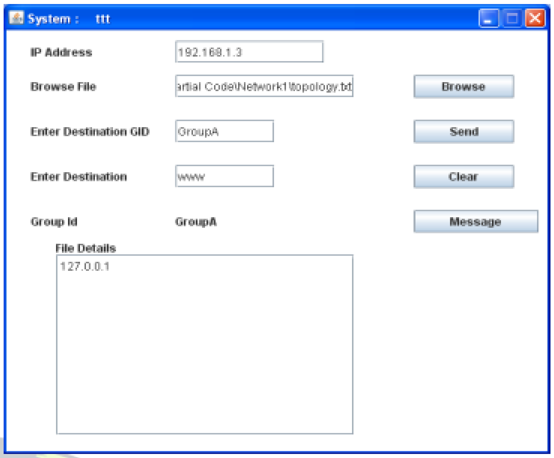
```
public void receive()throws Exception
{

}


}
```

Fig.11 Select the Data to transfer



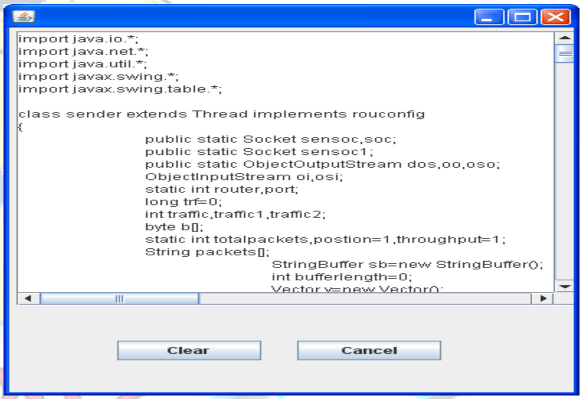
System Name ttt

Fig.12 Specifies the destination form



IP Address 192.168.1.3
Browse File artial Code\Network1\topology.txt
Enter Destination GID GroupA
Enter Destination www
Group Id GroupA
File Details
127.0.0.1

Fig.13 Destination Home login form



```
import java.io.*;
import java.net.*;
import java.util.*;
import javax.swing.*;
import javax.swing.table.*;

class sender extends Thread implements rouconfig
{
                    public static Socket sensoc,soc;
                    public static Socket sensoc1;
                    public static ObjectOutputStream dos,oo,oso;
                    ObjectInputStream oi,osi;
                    static int router,port;
                    long trf=0;
                    int traffic,traffic1,traffic2;
                    byte b[];
                    static int totalpackets,postion=1,throughput=1;
                    String packets[];
                                        StringBuffer sb=new StringBuffer();
                                        int bufferlength=0;
                                        Vector v=new Vector();
```

Fig.14 Data Transferred

IX. CONCLUSION

In this paper, we have proposed distributed energy efficient clone detection protocol with random witness selection. Specifically, we have proposed the ERCD protocol, which includes the witness selection and legitimacy verification stages. Both of our theoretical analysis and simulation results have demonstrated that our protocol can detect the clone attack with almost probability 1, since the witnesses of each sensor node is distributed in a ring structure which makes it easy be achieved by verification message. In addition, our protocol can achieve better network lifetime and total energy consumption with reasonable storage capacity of data buffer. This is because we take advantage of the location information by distributing the traffic load all over WSNs, such that the energy consumption and memory storage of the sensor nodes around the sink node can be relieved and the network lifetime can be extended. In our future work, we will consider different mobility patterns under various network scenarios.

REFERENCES

[1] Z. Zheng, A. Liu, L. X. Cai, Z. Chen, and X. Shen, "ERCD: An energy-efficient clone detection protocol in wsns," in Proc. IEEE INFOCOM, Turin, IT, Apr. 14-19 2013, pp. 2436–2444.

[2] R. Lu, X. Li, X. Liang, X. Shen, and X. Lin, "GRS: The green, reliability, and security of emerging machine to machine communications," IEEE Communications Magazine, vol. 49, no. 4, pp. 28–35, Apr. 2011.

[3] Christo Ananth, A.Nasrin Banu, M.Manju, S.Nilofer, S.Mageshwari, A.Peratchi Selvi, "Efficient Energy Management Routing in WSN", International Journal of Advanced Research in Management, Architecture, Technology and Engineering (IJARMATE), Volume 1, Issue 1, August 2015,pp:16-19

[4] Liu, J. Ren, X. Li, Z. Chen, and X. Shen, "Design principles and improvement of cost function based energy aware routing algorithms for wireless sensor networks," Computer Networks, vol. 56, no. 7, pp. 1951–1967, May. 2012.

[5] T. Shu, M. Krunz, and S. Liu, "Secure data collection in wireless sen-sor networks using randomized dispersive routes," IEEE Transactions on Mobile Computing, vol. 9, no. 7, pp. 941–954, Jul. 2010.

[6] P. Papadimitratos, J. Luo, and J. P. Hubaux, "A randomized coun-termeasure against parasitic adversaries in wireless sensor networks,"

[7] IEEE Journal on Selected Areas in Communications, vol. 28, no. 7, pp. 1036–1045, Sep. 2010.

[8] R. Lu, X. Lin, T. H. Luan, X. Liang, and X. Shen, "Pseudonym changing at social spots: An effective strategy for location privacy in VANETs," IEEE Transactions on Vehicular Technology, vol. 61, no. 1, pp. 86–96, Jan. 2012.

[9] Z. M. Fadlullah, M. Fouda, N. Kato, X. Shen, and Y. Nozaki, "An early warning system against malicious activities for smart grid communications," IEEE Network, vol. 25, no. 5, pp. 50–55, May. 2011.

[10] R. Lu, X. Lin, X. Liang, and X. Shen, "A dynamic privacy-preserving key management scheme for location based services in VANETs," IEEE Transactions on Intelligent Transportation Systems, vol. 13, no. 1, pp. 127–139, Jan. 2012.

[11] M. Conti, R. D. 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, Sep.-Oct. 2011.

[12] Parno, A. Perrig, and V. Gligor, "Distributed detection of node replication attacks in sensor networks," in Proc. IEEE Symposium on Security and Privacy, Oakland, CA, USA, May. 8-11 2005, pp. 49–63.

[13] 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. 28, pp. 677–691, Jun. 2010.

[14] 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, Jul. 2010.

[15] Y. Xuan, Y. Shen, N. P. Nguyen, and M. T. Thai, "A trigger identification service for defending reactive jammers in WSN," IEEE Transactions on Mobile Computing, vol. 11, no. 5, pp. 793–806, May. 2012.

[16] R. Lu, X. Lin, H. Zhu, X. Liang, and X. Shen., "BECAN: A bandwidth-efficient cooperative authentication scheme for filtering injected false data in wireless sensor networks," IEEE Transactions on Parallel and Distributed Systems, vol. 23, no. 1, pp. 32–43, Jan. 2012.

[17] J. Li, J. Chen, and T. H. Lai, "Energy-efficient intrusion detection with a barrier of probabilistic sensors," in Proc. IEEE INFOCOM, Orlando, FL, USA, Mar. 25-30 2012, pp. 118–126.

[18] 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, vol. 37, no. 6, pp. 1246–1258, Nov. 2007.

[19] W. Naruephiphat, Y. Ji, and C. Charnsripinyo, "An area-based ap-proach for node replica detection in wireless sensor networks," in Proc. IEEE TrustCom, Liverpool, UK, Jun. 25-27 2012, pp. 745–750.

[20] M. Zhang, V. Khanapure, S. Chen, and X. Xiao, "Memory efficient protocols for detecting node replication attacks in wireless sensor networks," in Proc. IEEE ICNP, Princeton, NJ, USA, Oct. 13-16 2009, pp. 284–293.

[21] T. Bonaci, P. Lee, L. Bushnell, and R. Poovendran, "Distributed clone detection in wireless sensor networks: An optimization approach," in Proc. IEEE WoWMoM, Lucca, IT, Jun. 20-23 2011, pp. 1–6.

[22] Liu, P. Zhang, and Z. Chen, "Theoretical analysis of the lifetime and energy hole in cluster based wireless sensor networks," Journal of Parallel and Distributed Computing, vol. 71, no. 10, pp. 1327–1355, Oct. 2011.

[22]Q. Chen, S. S. Kanhere, and M. Hassan, "Analysis of per-node traffic load in multi-hop wireless sensor networks," IEEE Transactions on Wireless

Communications, vol. 8, no. 2, pp. 958–967, Feb. 2009.

[23] H. Chan, A. Perrig, and D. Song, “Random key predistribution schemes for sensor networks,” in Proc. IEEE Symposium on Security and Privacy, Berkeley, CA, USA, May 11-14 2003, pp. 197–213.

[24] Ok, S. Lee, P. Mitra, and S. Kumara, “Distributed routing in wireless sensor networks using energy welfare metric,” Information Sciences, vol. 180, no. 9, pp. 1656–1670, May 2010.

[25] OMNET++ network simulation framework: http://www.omnetpp.org/.