ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 2, Issue 7, July 2014

A New Approach of Node Clone Detection Protocols in Wireless Sensor Networks

P Swathi[1] , S Vasu[2]

[1]M.Tech Student, Department of Computer Science, SVCET, Chittoor, India

[2]Associate Professor, Department of Computer Science, SVCET, Chittoor, India

ABSTRACT: Nodes in Mobile Ad Hoc Networks (MANETs) are limited battery powered. That's why energy efficient routing has become an important optimization criterion in MANETs. The conventional routing protocols do not consider energy of the nodes while selecting routes which leads to early exhaustion of nodes and partitioning of the network. This paper attempts to provide an energy aware routing algorithm. The proposed algorithm finds the transmission energy between the nodes relative to the distance and the performance of the algorithm is analyzed between two metrics Total Transmission energy of a route and Maximum Number of Hops. The proposed algorithm shows efficient energy utilization and increased network lifetime with total transmission energy metric.

KEYWORDS: energy efficient algorithm; Manets; total transmission energy; maximum number of hops; network lifetime

I. INTRODUCTION

WIRELESS sensor networks (WSNs) have gained a great deal of attention in the past decade due to their wide range of application areas and formidable design challenges. Generally wireless sensor networks consist of hundreds and thousands of low cost, resource constrained, distributed sensor nodes, scatter usually in the surveillance area randomly, without working attendance. When the operation environment is hostile, mechanisms under security against adversaries should be taken into consideration. With many physical attacks to sensor networks, the node under clone is a serious and dangerous one [1]. With a production expense limitation, node sensors are generally short of tamper resistance hardware components; thus, an enhanced advisory can capture a few nodes, code extract and most secret credentials, and with use those materials to clone many nodes out of off-the-shelf sensor hardware. Cloned nodes that seems to legitimate can freely join the sensor network and then significantly enlarge the adversary's capacities to manipulate the network maliciously. With example those vicious nodes occupy strategic positions and cooperatively corrupt the collected information. Under a large number of cloned nodes under command, advisory even gain control of the whole network. The node clone will exacerbate most of inside attacks against sensor networks furthermore. disadvantages of existing system

- Among many physical attacks to sensor networks, the node clone is a serious and dangerous one.
- Insufficient storage consumption performance in the existing system and low security level.

In this paper, practical node clone detection protocols with different tradeoffs on network conditions and performance is presented. *distributed hash table* (DHT) [2] method is the first one, which is fully decentralized, key based caching and checking system is constructed to catch cloned nodes. The developed protocol's performance on memory consumption and a critical security metric are theoretically deducted through a probabilistic model, and the resulting equations, by necessary adjustment for real application, are supported by the simulations. According with our analysis, the simulation results show that the DHT-based protocol can detect node clone with high security level and holds strong resistance against adversary's attacks. second protocol, named *randomly directed exploration*, is intended to provide highly efficient communication performance with adequate detection probability for dense sensor networks.

In the proposed protocol, initially nodes send claiming messages containing its neighbor list along with a maximum hop limit to randomly selected neighbors; then, the sequent message transmission is regulated by a *probabilistic directed* technique to approximately maintain a line property through the network as well as to incur sufficient randomness for better performance on communication and resilience against adversary. In advance, border determination mechanism is employed to further reduce communication payload. While forwarding, intermediate nodes explore claiming messages for node clone detection. By proposal, this protocol consumes almost minimal memory, and the proposed method simulations show that it outperforms all other detection protocols in terms of communication performance, while the detection probability is satisfactory.

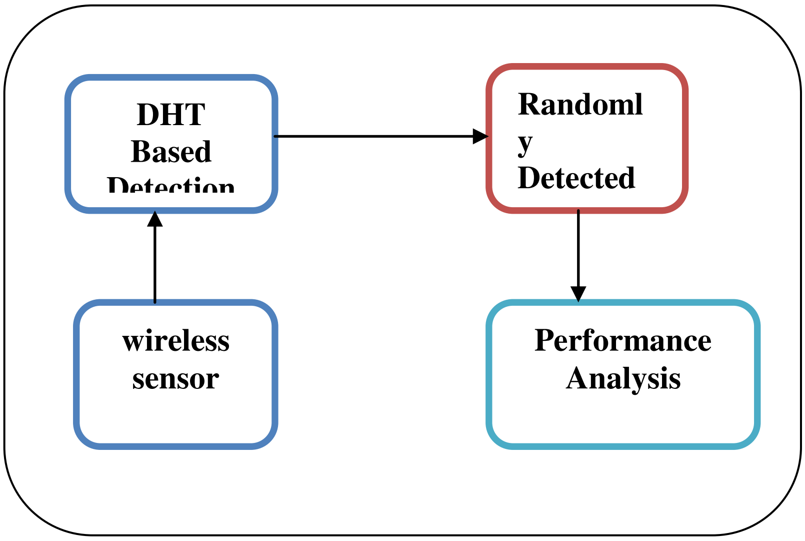


Fig 1: BLOCK DIAGRAM

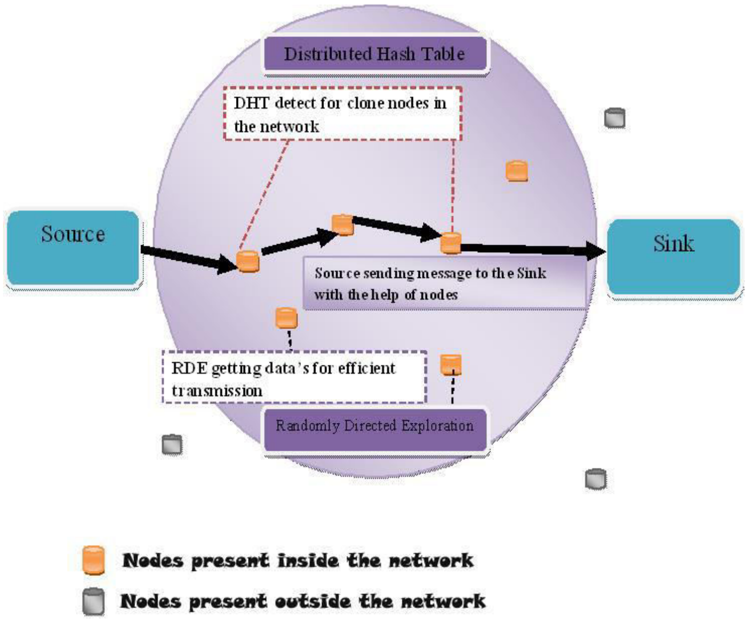
II. DETECTION PROTOCOLS

Based on the detection methodologies, we classify two novel node clone detection protocols.
1. Distributed hash table (DHT)
2. Randomly directed exploration (RDE)

A. Distributed hash table (DHT)

Distributed hash table (DHT), by which a fully decentralized, key-based caching and checking system is constructed to catch cloned nodes. The protocol's performance on memory consumption and a critical security metric are theoretically deducted through a probability model, and the resulting equations, with necessary adjustment for real application, are supported by the simulations. In accordance with our analysis, the comprehensive simulation results show that the DHT-based protocol can detect node clone with high security level and holds strong resistance against adversary's attacks.

Distributed Hash Table is a decentralized distributed system which provides a key based look up service. (Key, record) pairs are stored in the table any active node can store and retrieve records associated with specific keys. Thus distributed hash table maintain mapping from keys to records among nodes. Chord is used and choose chord as a distributed hash table implementation to demonstrate protocol. Massive virtual ring is formed by chord in which every node is located at one point, and owning a segment of the periphery. Hash function is used to achieve pseudo randomness on output by mapping an arbitrary input into a b-bit space (in the ring).Chord coordinate is assigned for each node and can join the network. Here a node's Chord point's coordinate is the hash value of the node's MAC address [1].one segment that ends at the node's Chord point is related to every node, and all records whose keys fall into that segment will be transmitted to and stored in that node[5].Every node maintains a finger table of size t= O (log n) to facilitate a binary-tree search. The finger table for a node with responsible for holding the t keys.


Distributed Hash Table
DHT detect for clone nodes in the network
Source
Sink
Source sending message to the Sink with the help of nodes
RDE getting data's for efficient transmission
Randomly Directed Exploration
Nodes present inside the network
Nodes present outside the network

- Figure 2. System architecture.

TABLE I

DISTRIBUTED DETECTION PROTOCOLS COMPARISON, WHERE n IS NETWORK SIZE, d NODE DEGREE

Protocols	Nodes requirements	Communication cost	Memory cost	Detection Cost
Node to network broadcasting	Neighbors information	O (n)	O(d)	Strong
Randomized multicast	All nodes data	O (n)	O(d√n)	Acceptable
Line selected	All nodes data	O (√n)	O(d√n)	Acceptable
RED	Knowledge of network geography	O (√n)	O(d√n)	Strong
DHT	DHT nodes information	O (log n √n)	O(d)	Strong
RDE	Neighbors information	O (√n)	O(d)	Good

between 10 and 20. The DHT enable sensor nodes to construct a chord overlay network. Cloned node may not participate in this overlay network construction [1]. And this overlay network construction is independent of node clone detection. Nodes possess the information of their direct predecessor and successor in the Chord ring and also caches information of its consecutive successors in its *successors table[6]*. The communication cost is thus reduced by this cache mechanism and it enhances systems robustness. Selection of inspectors is done using the facility of the successors table.

Detecting Rounding Stages

(i)The initial stage of detection round is done by activating all nodes by releasing an action message by initiator MACT=nonce, seed, time, {nonce||seed||time} k^{-1}initiator During each rounds the value of nonce increases monotonously and it intended to prevent the DoS attacks.

(ii) By receiving the action message each node verifies the value of nonce with previous values and verifies the signature of the message. If both are valid node will updates the nonce and stores the seed. The node act as observer to generate claiming message for each neighbor at the designated action time and transmits the message through the overlay network with respect to the claiming probability pc.

Mα4β=idβ, Lβ, idα, Lα, { idβ ||Lβ ||idα ||Lα||nonce} $k^{-1}α$. where, Lα, Lβ, are locations of α and β , respectively.

(iii) Chord intermediate nodes will forwards claiming message to its destination node. Only the source node, Chord intermediate nodes, and the destination node need to process a message, whereas other nodes along the path simply route the message to temporary targets. Algorithm 1 for handling a message and If the algorithm returns NIL, then the message has arrived at its destination. Else the message will forwarded to the next node with the ID that is returned by Algorithm[1].

Algorithm 1:

dht_ handle message(Mα4β) handle a message in the DHT-based detection, where y is the current node's Chord coordinate, finger[i] is the first node on the ring that succeeds key(($y+2^{b-I}$ mod 2^{b}),I £ [1,t] ,successors [j] is the next j^{th} successor j £[1,g][1].

Output: NIL if the message arrives at its destination; otherwise, it is the ID of the next node that receives the message in the Chord overlay network[1].

```
key<=H (seed||idβ)
if key £ [predecessor] then {has reached destination}
    inspect Mα4β {act as an inspector, see Algorithm 2}
    return NIL
for i=1 to g do
    if key £(y, successors [i]) then {destination is in the next Chord hop}
        inspect Mα4β {act as an inspector, see Algorithm 2}
        return successors [i]
for j= 1 to t do {for normal DHT routing process}
    if key £ [($y+2^{b-I}$ mod $2^{b, y}$)], then
        return finger [j]
return successor [g]
```

Message for node clone detection is examined by Algorithm 1 and Algorithm 2 compares the message with previous inspected messages that are buffered in the cache table[1]. All records in the cache table should have different examinee ID. If there exist two messages Mα4β and Mα'4β' satisfying idβ = idβ' and Lβ ≠Lβ shows that exists clone and then the witness node broadcasts the evidence to notify the whole network. All integrity nodes verify the evidence message and stop communicating with the cloned nodes. The witness does not need to sign the evidence message.

B. Randomly directed exploration (RDE)

This is intended to provide highly efficient communication performance with adequate detection probability for dense sensor networks. In the protocol, initially nodes send claiming messages containing a neighbor-list along with a maximum hop limit to randomly selected neighbors; then, the subsequent message transmission is regulated by a

Probabilistic directed technique to approximately maintain a line property through the network as well as to incur sufficient randomness for better Performance on communication and resilience against adversary. In addition, border determination mechanism is employed to further reduce communication payload. During forwarding, intermediate nodes explore claiming messages for node clone detection. By design, this protocol consumes almost minimal memory, and the simulations show that it outperforms all other detection protocols in terms of communication cost, while the detection probability is satisfactory.

Detection round

Initially the node clone detection round is activated by the initiator. At the right mentioned action time, each node creates its own neighbor list (ID of neighbor and location). Then that node act as an observer for all its neighbors and starts to generate claiming messages. The claiming message involves node ID, location and its neighbor list[6]. The claiming message by node is constructed by Mα=ttl, idα, Lα, neighbor list where ttl is time to live.

The problems associated with the dht are it incurs more communication cost because of the hard overlay network and thus it is sensitive to energy and storage consumption. To overcome these problems a new node clone detection protocol introduced namely randomly directed exploration. Here the each node only needs to know and buffer a neighbor list having all neighbors ID and ocations. During detection round each node constructs claiming message with signed version of neighbor list and then deliver message to others which will compares with its own neighbor list to detect node clone. If there exists any node clone, one witness node successfully catches the clone and notifies the entire network by broadcasting. The efficient way to achieve randomly directed exploration needs some mechanisms and routing protocols. First the claiming message needs to provide maximum hop limit and it is sent to random neighbors. Then the further message transmission will maintain a line and this transmission line property enables a message to go through a network as fast as possible[6]. The communication cost of this protocol is low and it is limited by the border determination mechanism. And the assumption made here is that each node knows about its neighbors locations.

Algorithm 2:

RDE-process message Mα: An intermediate node processes a message

```
verify the signature of Mα
compare its own neighbor list with the neighbor-list in Mα
if found clone then
    broadcast the evidence;
ttl<=ttl-1
if ttl ≤ 0 then
    discard Mα
else
    next node<=get next node (Mα) {See Algorithm 4}
    if next node =NIL then
        discard Mα
    else
        forward Mα to next node[6]
```

V. MODULES

* Setting up Network Model
* Initialization Process
* Claiming Neighbor's information
* Processing Claiming Message
* Sink Module
* Performance Analysis

VI.MODULES DESCRIPTION

A. Setting up Network Model

Our first module is setting up the network model. here We consider a large-scale, homogeneous sensor network consisting of resource constrained sensor nodes. compared to previous distributed detection approaches; we assume that an identity-based public-key cryptography facility is available in the sensor network. with the deployment, each legitimate node is allocated a unique ID and a corresponding private key by a trusted third party. in this method The public key of a node is its ID, which is the essence of an identity-based cryptosystem. Consequently, no node can lie to others about its identity. Moreover, anyone is able to verify messages signed by a node using the identity-based key. The source nodes in our problem formulation serve as storage points which cache the data gathered by other nodes and periodically transmit to the sink, in response to user queries. Such network architecture is consistent with the design of storage centric sensor networks.

B. Initialization Process

To activate all nodes starting a new round of node clone detection, the initiator uses a broadcast authentication scheme to release an action message including a monotonously increasing nonce, a random round seed, and an action time. The nonce is intended to prevent adversaries from launching a DoS attack by repeating broadcasting action messages.

C. Claiming neighbor's information

Upon receiving an action message, a node verifies if the message nonce is greater than last nonce and if the message signature is valid. If both pass, the node updates the nonce and stores the seed. At the designated action time, the node operates as an observer that generates a claiming message for each neighbor (examinee) and transmits the message through the overlay network with respect to the claiming probability. Nodes can start transmitting claiming messages at the same time, but then huge traffic may cause serious interference and degrade the network capacity. To relieve this problem, we may specify a Sending period, during which nodes randomly pick up a transmission time for every claiming message.

D. Processing claiming messages

A claiming message will be forwarded to its destination node via several Chord intermediate nodes. Only those nodes in the overlay Intermediate nodes and the destination node) need to process a message, whereas other nodes along the path simply route the message to temporary targets. Algorithm 1 for handling a message is the kernel of our DHT-based detection protocol. If the algorithm returns NIL, then the message has arrived at its destination. Otherwise, the message will be subsequently forwarded to the next node with the ID that is returned.

E. Sink Module

The sink is the point of contact for users of the sensor network. Each time the sink receives a question from a user, it first translates the question into multiple queries and then disseminates the queries to the corresponding mobile relay, which process the queries based on their data and return the query results to the sink. The sink unifies the query results from multiple storage nodes into the final answer and sends it back to the user.

F. Performance Analysis

- For the DHT-based detection protocol, the following specific measurements are used to evaluate its performance:
- Average number of transmitted messages, representing the protocol's communication cost;
- Average size of node cache tables, standing for the protocol's storage consumption;
- Average number of witnesses, serving as the protocol's security level because the detection protocol is deterministic and symmetric.

VII.CONCLUSION

Sensor nodes lack tamper-resistant hardware and are subject to the node clone attack. particularly In this paper, we present two distributed detection protocols: One is based on a distributed hash table, which creates a Chord overlay network and provides the key-based routing, caching, and verifying facilities for clone detection, and the second uses probabilistic directed technique to achieve efficient communication overhead for satisfactory detection probability. While the DHT-based protocol provides high security level for all kinds of sensor networks by one Deterministic witness and additional memory-efficient, probabilistic witnesses, the randomly directed exploration presents outstanding communication performance and minimal storage Consumption for dense sensor networks.

REFERENCES

[1] B. Parno, A. Perrig, and V. Gligor, “Distributed detection of node replication attacks in sensor networks,” in Proc. IEEE Symp. Security Privacy, 2005, pp. 49–63.
[2] H. Balakrishnan, M. F. Kaashoek, D. Karger, R. Morris, and I. Stoica, “Looking up data in P2P systems,” Commun. ACM, vol. 46, no. 2, pp.43–48, 2003.
[3] Y. Zhang, W. Liu, W. Lou, and Y. Fang, “Location-based compromise tolerant security mechanisms for wireless sensor networks,” IEEE J.Sel. Areas Commun. vol. 24, no. 2, pp. 247–260, Feb. 2006.
[4] S. Zhu, S. Setia, and S. Jajodia, “LEAP: Efficient security mechanisms for large-scale distributed sensor networks,” in Proc. 10th ACM CCS, Washington, DC, 2003, pp. 62–72.
[5] R. Anderson, H. Chan, and A. Perrig, “Key infection: Smart trust for smart dust,” in Proc. 12th IEEE ICNP, 2004, pp. 206–215.
[6] M. Conti, R. D. 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 Proc. 8thACMMobiHoc,Montreal,QC, Canada, 2007, pp. 80–89.
[7] B. Zhu, V. G. K. Addada, S. Setia, S. Jajodia, and S. Roy, “Efficient distributed detection of node replication attacks in sensor networks,” in Proc. 23rd ACSAC, 2007, pp. 257–267.
[8] H. Choi, S. Zhu, and T. F. La Porta, “SET: Detecting node clones insensor networks,” in Proc. 3rd Secure Comm, 2007, pp. 341–350.
[9] 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 Trans. Syst.s, Man, Cybern. C, Appl. Rev., vol. 37, no. 6, pp. 1246–1258, Nov. 2007.
[10] L. Eschenauer and V. D. Gligor, “A key-management scheme for distributed sensor networks,” in Proc. 9th ACM Conf. Comput. Commun. Security, Washington, DC, 2002, pp. 41–47.