Journal of Theoretical and Applied Information Technology
31st August 2014. Vol. 66 No.3

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

A PRAGMATIC APPROACH TO SECURE DSR PROTOCOL FROM SINKHOLE ATTACK IN AD HOC ENVIRONMENT

[1]DEVI. P, [2]KANNAMMAL. A

[1,2] Department of Computer Applications, Coimbatore institute of Technology, Coimbatore, India

E-mail: [1]devipichaimuthu@gmail.com , [2]kannaphd@gmail.com

ABSTRACT

Mobile Ad Hoc Networks (MANETs) have the most challenging network infrastructure from the security perspective. Attacks on routing protocols, especially internal attacks will cause the damage to MANETs. Sinkhole attack is a kind of internal attack, creates fabricated route discovery request packet, convinces other normal neighboring nodes, intrudes into the network, sniffs confidential information and finally spoils the network. This routing protocol attack exhausts the network if it is not caught earlier in the stage. In this paper, we design and evaluate the Secure Efficient DSR Protocol (SEDP), which recognizes the presence of sinkhole node from packet flow information between nodes, prepares Suspected Node List, filters particular harmful node from the list and lastly isolates it from the network. Our Intrusion Detection System incorporates IDS nodes, assumed as inner layer of MANET is closely monitoring the route cache of nodes and detects compromised nodes. We have evaluated the performance of the proposed system using NS2, in terms of detection rate, detection time, routing overhead, and false positive rate. Results prove the consistency and effectiveness of our intrusion detection method.

Keywords: *Mobile Ad hoc NETwork(MANET), sinkhole attack, DSR Protocol, security and protection, malicious node detection, Intrusion Detection System (IDS), Secure Efficient DSR Protocol (SEDP)*

1. INTRODUCTION

In wireless communications, Mobile Ad hoc networks (MANET) play an important role. MANETs are collection of autonomous nodes; each node determines the topology of the network. There is no pre existing infrastructure or centralized control support. Due to its ability of dynamic infrastructure, a MANET can be widely applied for disaster rescue management, military surveillance [1], and robot networks [2]. Establishing communication between two nodes depends on other nodes existing between source and destination and relies on distributed cooperation of nodes. A MANET gains and losses many nodes simultaneously, and these nodes are pushed into the resource constraints such as bandwidth, storage and energy capacity. MANETs are thus more vulnerable to network attacks.

Attacks on MANETs can be categorized into two major groups: internal and external [3]. An internal attack is originated from a compromised node of same network. They drop, fabricate, alter, eavesdrop, or misroute data packets. External attack is not participating in the routing process but disrupt network operations like flooding, DOS, or cut-off nodes from network. Among the internal attacks, we focus on *sinkhole attacks*, an adversary node misleads routing packets not to select appropriate path between source and destination. And it diverts all routing packets to itself in order to extract network traffic information and may perform selective forwarding [4, 5, 6]. Existing routing protocols for MANETs are vulnerable to sinkhole attack [6]. Many security mechanisms like cryptographic techniques are applied to protect network. Additionally, many mathematical models are proposed to detect and prevent network attacks. But they face computation complexity, delay, and unaffordable cost.

Our work deviates from conventional method of using cryptography. The purpose of this paper is to sense the existence of malicious node, alarming other nodes, detect and isolate malicious node from the network by observing the network traffic. Our method analyzes the routing pattern, traces the communications among nodes and locates the sinkhole node. This efficient intrusion identification algorithm gives improved result on detection rate and time.

The sinkhole problem is analyzed in terms of Dynamic Source Routing (DSR) [7] protocol, a commonly practiced routing protocol in wireless scenario. We use distributed network cooperation, one of network behaviors of MANET as a key factor along with IDS nodes in order to find out the problems against routing protocol. The path between two nodes is established based on reception of route advertisements, where sinkhole node inserts its forged routing packets. Forged routing packets dominate other packets in various aspects like fake source address, fake sequence number, etc. We assume that our proposed mechanism works well as long as not more than 40% of nodes are malicious.

Proposed method was evaluated through simulations, where the reliable performance is measured. Obtained results prove the correctness, scalability and less computation complexity of the proposed system. The remainder of the paper is organized as follows. In section 2 we discuss previous works on intrusion detection techniques. Section 3 describes the sinkhole attack launched upon DSR protocol. Section 4 presents the proposed method followed by the simulation results and analysis in section 5. Lastly Section 6 draws a final conclusion and future work.

2. RELATED WORKS

An attacker intrudes into the network and either collects all data or partially forwards data or drops data. It propagates forged RREQs containing fake information about the path to other nodes. The nodes receive forged RREQs send reply if they have an entry in their route cache and forwards data to the attacker node. Thus nodes ignore the RREQs which are from genuine source. Identifying legitimate and illegitimate RREQs and isolating source of illegitimate RREQs may lead to a secured communication.

In [6], Source node requires next hop information of each node of source route. It sends Request packet to intermediate node, intermediate node sends Route Reply to source node includes all next hop information. Again source sends Request packet to next one hop node to verify the former node's reply. Source receives Reply from the one node and this process continually goes on. Thus source verifies the truthfulness of the route. [8] Requires each intermediate node to send route Confirmation REQuest (CREQ) to next hop node. Next node checks the route cache, confirms the route to destination and sends reply to source. The same procedure was followed by all intermediate nodes in the source to destination path. Above two methods has no confidence that a node sends genuine information or not. Furthermore routing misbehavior attack was not handled in these papers.

The Watchdog scheme [9] detects and mitigates malicious node attack to improve the performance of network. The two main modules watchdog and pathrater are dividing their detection process of intrusion as watchdog module listens the next-hop node transmission packets and detects misbehaving nodes accordingly, based on the result of watchdog, pathrater selects the most secured path among available paths to transmit packets. Overhearing of neighboring nodes leads this method to fails to reduce false alarms and packet loss while collision occurs. However this method is considered as premium work for intrusion detection. The work [10] suggests CONFIDANT method consists of four modules Monitor, Reputation System, Trust Manager, and Path Manager. Each node listen its neighboring node, it calculates path rate and node rate based on the packet flow and decides the transmission of alarm packets. Any routing misbehavior is detected, all four modules work together and removes malicious node. Similar to the approach [9], due to overhearing of other nodes it drops packets, enlarges routing overhead problem and these reputation-based schemes unable to meet MANETs resource constraints.

Many methods [11-15] were proposed to solve the problems posed by watchdog. Particularly in [11], each data packet is acknowledged to a node in the source route which is two hops away from the received node. Nodes use reverse route to send their acknowledgement back, and confirms the packet delivery. Each node is monitored by their acknowledgement packets instead of overhearing other nodes. It resolves collision problems and resource constraints, however it has significant routing overhead problem. The 2ACK mechanism [12] sends two-hop acknowledgement packets in the opposite direction of the source route. Furthermore, it partially acknowledges the received data packets to avoid more routing overhead. 2ACK scheme achieves good result than TWOACK since it sends only fraction of 2ACK packets. However the 2ACK scheme failed to prevent collision attack and false alarms. The method NACK [16] is also an acknowledgement based scheme, it detects routing misbehavior and collusion attack simultaneously, and furthermore it uses timestamps comparison technique to detect routing attacks. NACK considers the cases of two consecutive nodes to prevent collusion attack. The routing misbehavior prevention methods [11, 12, 16] did not handle sinkhole problems.

In method SIIS [5], two detection indicators based on sequence number of packet and route add ratio of the suspected node are proposed. This method uses predefined threshold value for the indicators, hence it produces low detection rate in case of larger number of sinkhole nodes. In [17], an agent framework over a military command structure and an agent behavioral model adopts military tactics are proposed to detect malicious nodes in the network. Furthermore, the frequency of patrols is increased if risk factor of route increases. The operational complexity makes the system to be less exercised.

3. CASE STUDY PRELIMINARIES

The Sinkhole Attack

Trait and effect

Sinkhole attack is a potential threat to MANET environment, as it grasps the dynamic property. According to [1-7], a sinkhole node channelizes all the traffic from a network through a compromised node, and exploits the information. Sinkhole achieves the above objective by keeping itself updated and attractive. Routing packets propagated by sinkhole node draw other nodes' attraction towards it. Forged and high quality data packets are sent throughout the network. So all surrounding nodes ignore original data packets and divert their traffic towards sinkhole. The sinkhole now becomes the controlling authority of the network, and drop or misroute data packets.

Sinkhole attack on DSR protocol Environment

Our proposed method is stated on the context of Dynamic Source Routing protocol (DSR). Each node in network learns about all available routes in its route cache by route discovery process before sending any data packets. If no route record is available, node finds new route by sending an RREQ packet to whole the network. The parameters included in the RREQ are source node address, destination node address and sequence number. These details give unique identification to each RREQ. Nodes receive RREQ will check their route cache for route entry to the destination. If so, nodes will send route reply RREP to source node about the available route. Otherwise intermediate nodes will append its id to the RREQ packet and forward to other nodes. Sequence number of RREQ prevents a node to listen an RREQ more than once. If a node receives an RREQ, it never attend the same RREQ again, such that avoids multiple broadcasting of same RREQ. For each rebroadcast of an RREQ, there will be an increment of sequence number, and highest value of sequence number implies more recent RREQ is only entertained by nodes.

The route discovery and cache update process is utilized by sinkhole attack. Sinkhole node eavesdrops the communication between nodes, filters the source node information, destination node information, and sequence number series from Route Request packet. Sinkhole node modifies source node id by its own id, misleads all packets in the network. Furthermore sinkhole node increases the value of sequence number, gives a new look to forged RREQs that they are latest request packets from source. Nodes in the network receive forged RREQs, ignore original packet and start communication using bogus information [5].

Sinkhole attack on DSR protocol is depicted in Figure. 1, Sinkhole node i1 broadcasts forged RREQs to all neighboring nodes with modified higher sequence number and fake source id. This false route request will be cached in the table. So the original RREQ with lower sequence number from node S will be rejected. The reply will be sent from destination node D to sinkhole node i1.

4. PREAMBLE OF PROPOSED SYSTEM

4.1 Requirements

For the feasibility of the proposed system, we assume the following requirements should be satisfied.

4.1.1 Sinkhole detection and recovery

The proposed solution should detect malicious entries of nodes into the network more accurately and concisely. Detection rate of the solution should be reasonably high with less computational over-heads. The solution will be analyzed in terms of the ratio of the true positive, false positive and false negative. Once detection process succeeds, solution should able to recover the network from damage. This may include alarming the whole network about the sinkhole node, isolate sinkhole node from other nodes, and speed up the process of recovery of data through rebuild of network.

4.1.2 Scalability

Proposed solution should be able to adopt various aspects like different topologies, network sizes, and traffic load patterns. The efficiency of the system should not be compromised when we change any parameters. Our system had been involved with various network sizes like 50 nodes, 75 nodes, and 100 nodes. Simultaneously sinkhole nodes are increased based on the size of the network. Effectiveness of the detection pattern is not sacrificed.

4.2 Assumptions

- We assume the time synchronization and bidirectional communication exist between nodes in network.
- We assume an intermediate node in a route between source and destination is reliable. Nodes are forwarding data to next hop nodes if it is not a destination. Nodes raise RERR messages in case of any disrupt.
- We assume there is no collusion among sinkhole nodes.
- Furthermore, we assume every node in the network is in pernicious attitude.

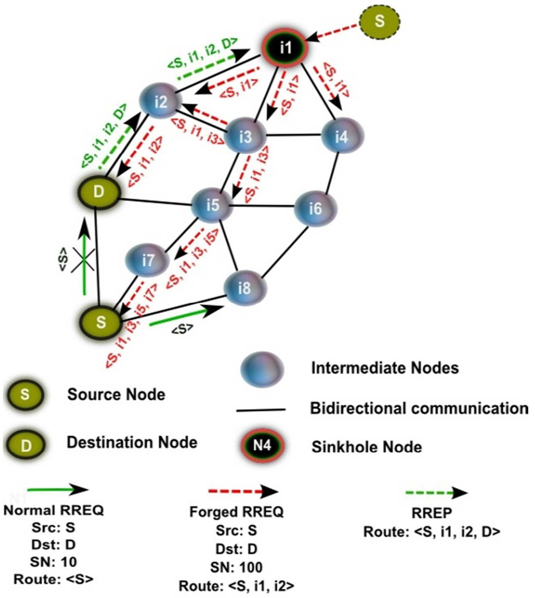


Figure. 1. Sinkhole attack on DSR Protocol

4.3 Sinkhole Detection Model

A sinkhole node will drop or partially forward the data packet if it wants to work as a middle man. These selfish nodes get involved in route discovery and route maintenance of DSR though they may misbehave or disrupt the network. Transmission of data will be affected even if reliable routes are available.

In our work, sinkhole nodes will be smelled and removed from the network. Their misbehaviors may include,

- Selective forwarding or dropping of packets.
- Masquerade as source and convince other nodes.
- Sending forged RREQs packets.
- Not to react for route confirmation packet.

4.4 Design of Proposed System

The adopted method to detect and recover from sinkhole attack is explained in this section. We introduce an efficient, distributive and co-operative detection approach by observing the communication among nodes.

4.4.1 Detecting sinkhole existence

A node receives an RREQ, it checks whether the source id of RREQ is equal to its own id. If not, node sends positive ACK packet to source and keeps forwarding it to destination. Propagated forged packets reach original source node, as its regular practice, it checks the entry of the sequence number of RREQ in its route cache. If there is no such entry and the sequence number of RREQ is greater than the sequence number of latest RREQ entry of the node, then the node smells the presence of sinkhole node. Figure 2 explains this process. Nodes involved in the route path of forged RREQ are moved to Suspected Node List (SNL). We denote S is source node and its unique id is used as fake source id by forged RREQ, i_0 to i_n are intermediate nodes, and D is the destination node. The original source node S generates the SNL as < i_0, i_1, i_2,…,i_{n-1}, i_n>, thus node list between forged source and original source node. In figure 2, S generates SNL as <i1, i3, i5, i7> and starts sinkhole detection process.

```
// Process of RREQ packet by each node from source to destination
if destination id of RREQ $\neq$ its own id
    if source id of RREQ $\neq$ its own id
        forward RREQ to next one-hop nodes;
    else if Seq_no of RREQ > Seq_no of current
                                entry in route cache
        start discovery process;
    else
        ignore RREQ packet;
    end if
else
    send positive ACK packet to source node;
end if
```

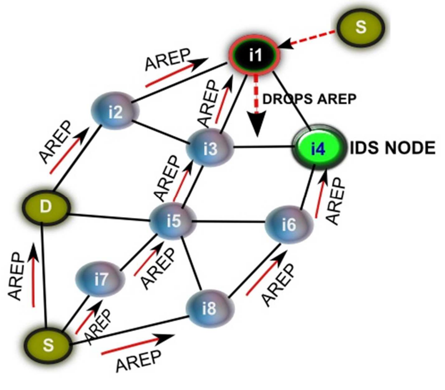
Figure 2. Finding Sinkhole Existence

4.4.2 Sinkhole node detection process

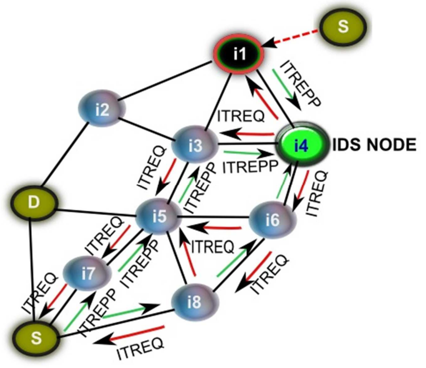
When original source node recognizes the presence of sinkhole node, immediately it broadcasts ALARM_REPLY packet (AREP) to the whole network. It consists of SNL, sequence number of forged RREQ, and other details of forged RREQ.

Intrusion Detection System (IDS) nodes are 10% of total population and activated once the AREP packet hit them. As shown in the Figure 3a, IDS nodes are informed about sinkhole nodes in the SNL by AREP packet. AREP gets an entry in the ALARM Packet table (ALAP), is maintained by IDS to closely monitor the suspected nodes. And IDS ensures all nodes in SNL have received the AREP packet. IDS sends IDS_TRIGGERING_REQUEST packet (ITREQ) to each node in SNL, directs to send the sinkhole source route, forwarding node id of forged RREQ, list of all one-hop nodes (OHN) to IDS as shown in Figure 3b. The IDS_TRIGGERING_REPLY (ITREPP) packet details will be saved in the Malicious Node Discovery table (MND), and each entry will be and only validated by IDS nodes.

Each node in the SNL < i_0, i_1, i_2,…,i_{n-1}, i_n> confirms its existence in the AREP to IDS. The last RREQ forwarding node i_n sends the source route <S, i_0, i_1, i_2,…,i_{n-1}>, i_{n-1} the node which forwards RREQ to i_n and one-hop nodes list includes i_{n-1}, S to MND. Then ITREQ asks next preceding the last node i_{n-1} to send the details. i_{n-1} sends <S, i_0, i_1, i_2,…,i_{n-2} >, i_{n-2} the packet forwarding node to i_{n-1} and one-hop nodes list includes i_{n-2}, i_n to MND. This process continues till the node i_0. The entry of MND table for the figure 1 is shown in Table 1.



3a. Propagation of AREP packet



3b. Transmission of ITREQ and ITREPP

Figure 3. Illustrations for sinkhole node detection process

The MND table entries of nodes i7, i5, i3 are validated and accepted by IDS nodes by cross checking the routing tables of nodes. But IDS finds node i1 has given wrong information to MND table.

Table 1. Malicious Node Discovery Table entry

Node_ID	Source Route	RREQ forwarded by	One-hop Nodes
i7	<S, i1, i3, i5, i7>	i5	S, i5, D, i8
i5	<S, i1, i3, i5>	i3	i7, i3, D, i2, i4, i6, i8
i3	<S, i1, i3>	i1	i5, i1, i2, i4, i6
i1	<S, i1>	S	S, i2, i3, i4

A node forwards RREQ to another node, should be in the list of one-hop nodes of later. And at least one node in the one-hop list node should exist in the sinkhole path. IDS detects S is not a forwarding node and does not exist in the one-hop node list. Finally IDS locates <S, i1> is the fabricated path, S is fake source id, is used unethically by i1 and i1 is the sinkhole node. Figure 4 shows the process of detection of sinkhole node from the network.

```
// Discovery process of sinkhole node
 1.   At Source node (S)
      if Source id of RREQ = its own id &
         Seq_no of RREQ > Seq_no of Current
entry in route cache
              Generate SNL;
              Broadcast SNL over AREP to whole
network about sinkhole intrusion;
      else
          forward RREQ to next hop nodes;
      end if
 2.   At IDS Node
      a.   AREP wake up IDS Node;
      b.   IDS node sends ITREQ to first node in
           SNL;
      c.   Step 3;
      d.   if RREQ forwarding node € {OHN
           List} &
             at least one node in OHN List €
           < Source Route >
             Removes node $i_n$ from SNL List;
           else
              Mark as malicious node;
           end if
      e.   Send IDS_ALARM packet to whole
           network about sinkhole node;
      f.   Repeat steps from b to e for all rest of
           nodes;
 3.  At each SNL node $i_n$
     Sends <S, $i_0$,$i_1$,$i_2$,…$i_{n-1}$>, forwarding node of
     RREQ $i_{n-1}$ to $i_n$ and OHN list to MND
     of IDS;
```

Figure 4 Sinkhole node detection and removal process

4.4.3 Sinkhole node removal process:

IDS nodes send IDS_ALARM packet to the whole network about the sinkhole node presence. Hence each packet of IDS is attached with Message Authentication Code (MAC) using IDS's Private Key; it cannot be modified by intermediate nodes in the network. Furthermore a nonce random number is embedded with each packet to avoid replay attacks.

Nodes receive alarm packet will stop communicating with sinkhole node. Route cache entries of nodes related to sinkhole node will be permanently removed from the table. Thus sinkhole node is automatically detached from the network and no more communication is entertained.

5. PERFORMANCE EVALUATION

In this section, we present simulation results and discuss about performance evaluation of proposed system in comparison with other existing methods Tseng's indicators, SIIS method [5] and NACK approach [16] of H.M. Sun et. al, are discussed in section 2.

5.1 Simulation Methodology and Performance Metrics

The sinkhole detection system, Secure Efficient DSR Protocol (SEDP) was implemented using NS-2 to evaluate the under mentioned features against sinkhole nodes. Area of simulation environment was 1000 x 1000 m^2 flat area, composed of 50 nodes were randomly distributed. The transmission range of each node is 250m and nodes move on according to the random-waypoint algorithm [18] with a maximum speed of 5, 10, 20 m/sec and a pause time of 0 second. Nodes in the network implement DSR protocol for data communication; UDP-Constant Bit Rate (CBR) traffic pattern was implemented. To demonstrate different traffic load 10, 20, 40 data packets per second were considered, each data packet size was fixed to 512 bytes.

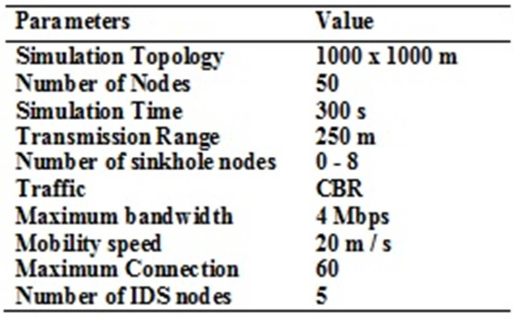
The MAC layer is based on IEEE 802.11 standard with a channel rate of 4 Mb/s. Sinkhole nodes rate ranges between 10 - 40% of normal nodes and IDS nodes rate ranges between 10 - 25% of normal nodes are randomly distributed all over the network. The total simulation time goes up to 300s. Network is vulnerable to sinkhole attack at any time. The simulation configuration details given in the Table 2, mostly adopted from [19-21], experimented for 20 times and most suitable values were adopted.

We used following metrics to evaluate the performance of proposed system.

- **Sinkhole Detection Rate, (SH_{DR}):** The percentage of sinkhole nodes exactly identified as being malicious node by our approach.
- **Sinkhole Detection Time, (SH_{DT}):** The amount of time taken by our approach to detect all sinkhole nodes.
- **Routing Overhead, (RO):** The ratio of amount of routing packets like RREQ, RREP,AREP, ITREQ, ITREPP and IDS_ALARM added by our approach to the amount of normal data packets.

- **False Positive Rate of Normal Nodes, (NN_{FPR}):** The percentage of normal nodes incorrectly identified as sinkhole node by our approach.

Table 2. Simulation parameters



Parameters	Value
Simulation Topology	1000 x 1000 m
Number of Nodes	50
Simulation Time	300 s
Transmission Range	250 m
Number of sinkhole nodes	0 - 8
Traffic	CBR
Maximum bandwidth	4 Mbps
Mobility speed	20 m / s
Maximum Connection	60
Number of IDS nodes	5

5.2 Simulation Results

5.2.1 Sinkhole detection rate (SH_{DR})

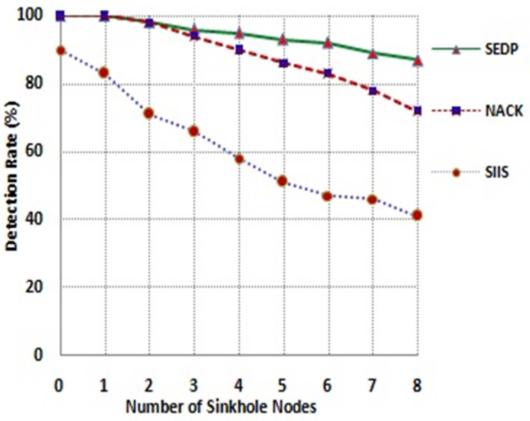


Figure 5. Sinkhole Node Detection Rate

Figure 5 shows the detection rate (SH_{DR}) of proposed method, NACK scheme, and SIIS method in the presence of 15% of sinkhole nodes and 10% IDS nodes of total population. The SH_{DR} of SIIS was approximately 61 % and NACK method was approximately 89%, while SH_{DR} of proposed method was approximately 94%, it was increased detection rate of 5% when compared to the NACK scheme.

5.2.2 Sinkhole detection time (SH_{DT})

Proposed method reacts for malicious node attacks immediately and brings suspicious nodes under control. Our method outperforms when network under threat, detects and removes spoiled nodes as soon as possible. It consumed approximately 3s as maximum time to detect and remove sinkhole nodes. NACK scheme took approximately 5.5s as maximum time and SIIS went for approximately 16s to detect and remove all sinkhole nodes. If there were approximately 8% of sinkholes then SH_{DT} of each scheme was 0.7s, 1.3s, and 3.72s respectively. Their confidence interval varies up to 0.2, 0.4, 1.2 respectively. If there were approximately 15% of sinkholes then SH_{DT} of each scheme was approximately 2.3s, 4.2s and 12.9s respectively. It is evident that the number of sinkholes increases the detection time also increases. Figure 6 depicts the average detection time of sinkhole nodes.

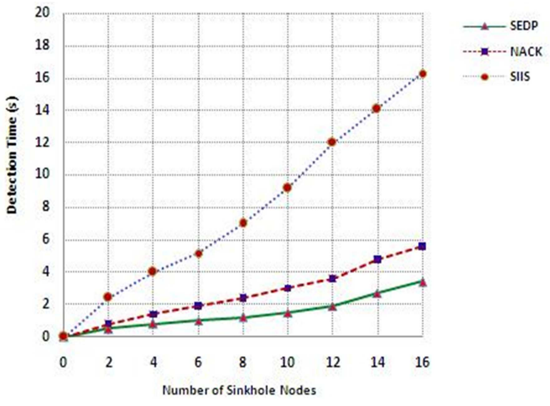


Figure 6. Sinkhole Detection Time

5.2.3 Routing overhead (RO)

It was recorded that the transmission of AREP packets, ITREQ packets and IDS_ALARM packets increased the routing overhead of proposed system. Hence extra routing packets involve, the routing overhead of proposed system was higher than NACK and SIIS methods. In figure7, the routing overhead of proposed system was approximately 34%, which is 19% more than NACK method. The routing overhead of SIIS was 11% , which was 23% less than the proposed system.

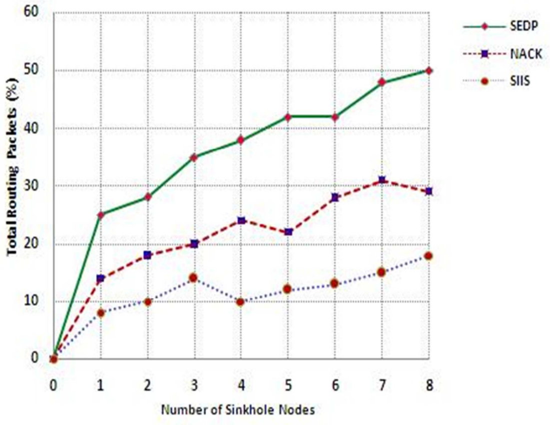


Figure. 7 Impact of routing packet overhead

5.2.4 False positive rate of normal nodes (N_{FPR})

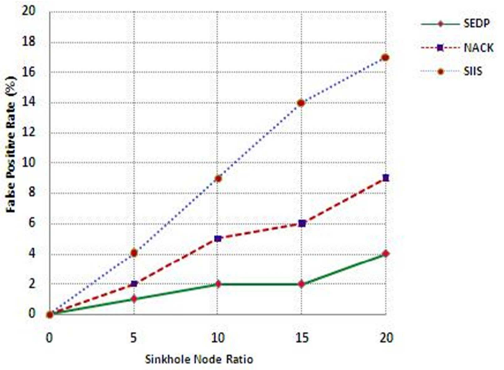


Figure. 8 False positive Rate

False positive is a false indication of normal nodes as malicious nodes. A normal node will be removed from the network if it was marked as false positive. Our proposed system tried to increase the detection rate and decrease the false positive rate of sinkholes. In figure 8, there were 20 % of sinkhole nodes, SH_{FPR} of SIIS was approximately 17% and NACK was approximately 9%. SH_{FPR} of proposed scheme was approximately 4%, decreased by 5% than NACK method. A compiled report of all the results of detection rate and false positive rate for different scenarios is tabulated in Table 3. Each value in the table is an average result of twenty simulations.

6. CONCLUSION

In this paper, we propose a distributed and collaborative effort of nodes for sinkhole detection in mobile ad-hoc networks. Sinkhole node utilizes routing information and misbehaves accordingly. Original source node realizes the existence of fake node, it propagates alert message throughout the network. Each node in the network co-operatively works together along with IDS nodes, detects sinkhole node and removes from the network. Hence this procedure consumes very little time and energy, network can survive for longer time effectively. Our system outperforms even there is 40% of intrusion. The simulation has demonstrated to proof the efficiency of proposed system, and it shows its supremacy over NACK and SIIS with respect to sinkhole detection rate and time.

Since our SEDP system is tied up with DSR protocol, in future we will focus to implement the proposed work over other routing schemes. We will evaluate and compare the results of all routing schemes, significant threats will be considered for further improvement of overall network security.

As our proposed system involves with extra simple routing packet flows, in future we will work to reduce routing overhead. We will improvise our system to meet all resource constraints mainly energy to effectively work on.

Table 3. Detection Rate and False Positive Rate results for different scenarios

Results for IDS rate of 10%						
Network Area	500x500			1000x1000		
No. of Nodes	50			75		
Sinkhole Nodes (%)	10	15	20	10	15	20
Simulation Results SH_{DR} (%)	94.66	94.39	94.12	93.92	93.67	93.4

Simulation Results SH_{FPR} (%)	3.48	3.75	4.2	4.14	4.38	4.7
Results for IDS rate of 15%						
Network Area	500x500			1000x1000		
No. of Nodes	50			75		
Sinkhole Nodes (%)	10	15	20	10	15	20
Simulation Results SH_{DR} (%)	94.91	94.7	94.43	94.22	94.02	93.68
Simulation Results SH_{FPR} (%)	3.17	3.48	3.82	3.81	4.14	4.48

Network Area	1500x1500	
No. of Nodes	100	
IDS rate	20%	25%
Sinkhole Nodes (%)	20%	40%
Simulation Results SH_{DR} (%)	93.72	93.04
Simulation Results SH_{FPR} (%)	4.66	5.23

REFERENCES:

[1] C.E. Perkins, "Ad hoc networking: an introduction", *C.E. Perkins (Ed.), Ad Hoc Networking*, Addison-Wesley, 2000

[2] B. wang, C. H. Huang, L. Y. Li, and W. Z. Yang, "Trust based minimum cost opportunistic routing for Ad Hoc networks", *Journal of Systems and Software*, vol.84, no.12, pp.2107-2122, 2011

[3] P. Papadimitratos and Z. Haas, "Securing the Internet routing infrastructure", *IEEE Communications Magazine*, vol. 40, no. 10, pp 60-68, Oct, 2002

[4] C. Karlof, and D. Wagner, "Secure routing in sensor networks: attacks and countermeasures", *Proceedings of the 1st IEEE Workshop on Sensor Network protocols and Applications*, pp. 1-15, May 2003

[5] H. C. Tseng ,B. J. Culpepper, "Sinkhole intrusion in mobile ad hoc networks: The problem and some detection indicators", *Computers & Security*, vol.24, 561-570, 2005

[6] H. Deng, W. Li, and D. P. Agrawal, "Routing security in wireless ad hoc networks", *IEEE Communications Magazine* vol. 40 no. 10, pp. 70-75, Oct. 2002

[7] Johnson, D,"The dynamic source routing protocol (DSR) for mobile ad hoc networks for IP4", 2007

[8] S. Lee, B. Han, and M. Shin, "Robust routing in wireless ad hoc networks," *Proceedings of 31st ICPP Workshops*, pp. 73–78, Aug. 2002

[9] S. Marti, T. J. Giuli, K. Lai, and M. Baker, "Mitigating routing misbehavior in mobile ad hoc networks," *Proceeding of 6th Annual International Conference on Mobile Computing and Network*, Boston, MA, pp. 255–265, 2000

[10] S. Buchegger and J.-Y. Le Boudec, "Performance Analysis of the CONFIDANT Protocol: Cooperation of Nodes, Fairness in Dynamic Ad-Hoc Networks," *Proceedings of MobiHoc*, June 2002.

[11] Balakrishnan, K., Deng, J., and Varshney, P. K, "TWOACK: Preventing selfishness in mobile ad hoc networks", *Proceedings of IEEE Wireless Communications and Networking Conference,* Vol. 4, pp. 2137–2142, 2005

[12] K. Liu, J. Deng, P. K. Varshney, and K. Balakrishnan, "An acknowledgment-based approach for the detection of routing misbehaviour in MANETs," *IEEE Transactions on Mobile Computing*, vol. 6, no. 5, pp. 536–550, May 2007

[13] J.-S. Lee, "A Petri net design of command filters for semiautonomous mobile sensor networks," *IEEE Transactions on Industrial Electronics*, vol. 55, no. 4, pp. 1835–1841, Apr. 2008

[14] T. Sheltami, A. Al-Roubaiey, E. Shakshuki, and A. Mahmoud, "Video transmission enhancement in presence of misbehaving nodes in MANETs," *International Journal of Multimedia Systems*, vol. 15, no. 5, pp. 273–282, Oct. 2009.

[15] Elhadi M. Shakshuki,Nan Kang, and Tarek R. Sheltami, "EAACK—A Secure Intrusion-Detection System for MANETs", *IEEE Transactions On Industrial Electronics*, Vol. 60, No. 3, March 2013

[16] H.-M. Sun, C.-H. Chen, Y.-F. Ku, “A Novel acknowledgement-based approach against collude attacks in MANET”, *Expert System with Applications*, vol. 39 2012, pp.7968-7975.

[17] S.K. Stafrace, N. Antonopoulos, “Military tactics in agent-based sinkhole attack detection for wireless ad hoc networks”, *Computer Communications*, 33, (2010), 619-638.

[18] T. Camp, J. Boleng, and V. Davies, “A survey of mobility models for ad hoc network research,” *Wireless Communications and Mobile Computing*, vol. 2, no. 5, pp. 483–502, Sep. 2002.

[19] Y. C. Hu, D. B. Johnson, and A. Perrig, “SEAD: secure efficient distance vector routing for mobile wireless ad hoc networks”, *Ad Hoc Networks*, vol.1, no. 1, pp. 175-192, 2003

[20] A. P. Emmanouil, N. Levon, and P. Christos, “Securing AODV against wormhole attacks in emergency MANET multimedia communications”, *Proceedings of the 5th International ICST Mobile Multimedia Communications Conference (Mobimedia ‘09)*, article 34, September 2009.

[21] I. khalil, S. Bagchi, and N. B. Shroff, “LiteWorp: detection and isolation of the wormhole attack in static multihop wireless networks”, *Computer Networks*, vol.51, no. 13, pp. 3750-3772, 2007.