International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 2 Issue 6, June - 2013

An Unobservable Secure Path Tracing Routing Protocol For Mobile Adhoc Networks

M. Viswanathan
Senior Software Engineer, Systole consultancy pvt ltd., Chennai, Tamil Nadu, India

Abstract

The focus of this paper is to preserve privacy & protection against wormhole attack in routing in MANET. MANET is a open medium & it does not have central authority, so it is liable to numerous routing attacks. Privacy preserving routing is important in MANET , however till now number of schemes have been proposed but none of them offer complete unlinkability & content unobservability. One of the serious routing attack is wormhole attack which leads to the high probability of packet dropping. In this paper we define a routing protocol with strong privacy requirements & Path Tracing(PT) algorithm which provides the detection & prevention of wormhole attack. USPTR which is the novel combination of group signature, ID based encryption for route discovery & Path Tracing algorithm. PT algorithm run during the route discovery process. It is based on per hop distance & frequency of the link appearing in route . We implement USPTR on ns2 & simulation results are compared with USOR & AODV. The simulation results proves that USPTR provides strong privacy & detects the wormhole effectively.

Keywords – Routing protocols, security, privacy, routing attacks, wormhole attack.

1. Introduction

MANET is a mobile wireless network, capable of autonomous operation & it operates without base station infrastructure , centralized administration. Privacy protection in routing & maintaining attack less MANET is a challenging problem. Privacy related notions in communication network that are anonymity, unlinkabillity, unobservability.

- Anonymity is the state of being not identifiable within a set of subjects, the anonymity set.
- Unlinkability of two or more IOIs(Item Of Interest) means these IOIs are no more or no less related from the attacker's view.
- Unobservability of an IOI (Item Of Interest) is the state that whether it exists or not is indistinguishable to all unrelated subjects, and subjects related to this IOI are anonymous to all other related subjects.

Existing anonymous routing protocols that provides anonymity & partial unlinkability among the data packets. Complete unlinkability and unobservability are not guaranteed due to incomplete content protection. Existing schemes that does not protect all the content of packet such as sequence no from attackers. This information that relate the two packets which breaks unlinkability & source traceback attacks. It is extremely difficult to hide information on packet type and node identity. Another drawback of most previous schemes is that they rely heavily on public key cryptography that leads to a very high computation overhead.

Unobservability is the strongest one in that it implies not only anonymity but also unlinkability. Unobservability into two types: 1) *Content Unobservability*, refers to no useful information can be obtained from content of any message; 2) *Traffic Pattern Unobservability*, refers to no useful information can be obtained from frequency, length, and source-destination patterns of message traffic.

Wormhole attack is a severe attack in wireless ad hoc network in which the adversary builds a tunnel between two end points which are usually multi-hops away. This tunnel between two malicious nodes is called wormhole. The message recorded at one end point is relayed to the other end and re-broadcasted into the network. The detection of wormhole attack is difficult because they don't compromise any node and don't expose their original identities.

In this paper we define a routing protocol with strong privacy preserving & prevention against wormhole attack. USPTR that has two phases. 1) Anonymous key establishment, 2) Privacy preserving route discovery. In Anonymous key establishment , construction of secret session keys takes place. In Privacy preserving route discovery , find a

route to destination takes place. The Path Tracing algorithm that runs along with route discovery. USPTR is to protect *all* parts of a packet's content, and it is independent of solutions on traffic pattern unobservability.

The paper is organized as follows. Section II that describes related work on anonymous , unobservable routing schemes & wormhole attack detection & prevention schemes on mobile adhoc networks. Section III that describes the proposed system. Section IV that describes the implementation & results. Finally conclusions and future work are given in section V.

2. Related Work

An ad hoc network is more vulnerable to wireless attacks. Ad hoc nodes are wireless in nature that makes it prone to attacks including eavesdropping, black hole, wormhole, denial of service etc. A secured MANET system can be achieved only by preventing routing protocol attacks [2]. The number of schemes have been proposed . Most of the schemes that rely on public key cryptosystems to achieve unlinkability & anonymity , but that leads to complex computation & increases overhead [1].

ANDOR is the first anonymous routing protocol that provides unlikability & anonymity in routing . It uses onion routing for route discovery & private/public key for obtaining anonymity. But it fails to provide unobservability since the packets are publicly labeled [3]. ASR [4], ARM [5], AnonDSR [6] and ODAR [7] also uses one-time public/private key pairs to achieve anonymity and unlinkability.ASR that gives strong location privacy & it discards onion routing [4]. ARM that reduces inconvenience in public/private key generation [5]. ODAR also uses bloom filter to establish multiple routes in MANET [7].

ALARM that uses public key cryptography & group signature. The group signature that provides privacy preserving and it is verified by everyone but the signed person cannot be identified. But it does not protect network topology, location [8].

Many methods for wormhole attack detection & prevention have been proposed . Distance & location based techniques that has the limitation of GPS technology [2]. Secure neighbour discovery & monitoring approach in which MOBIWORP, LITEWORP, Wormhole Attack Prevention (WAP) is categorized. MOBIWORP is based on observation schemes and it has central authority to detect wormhole nodes, but its detection rate decreases when mobility increases [9]. LITEWORP that uses local traffic monitoring scheme , but it induces other attacks in network [10].

Wormhole Attack Prevention (WAP) that identifies false route & provide the preventive measures in route discovery process. The source node that monitors the neighbour & its capable of detecting wormhole affected route. The disadvantage of WAP is that it is not possible to detect the wormhole attack when the packet content gets changed [11].

To summarize, public key cryptosystems have significant computation overhead. Existing schemes that fail to provide unobservability since it does not protect the packet content. They considered only partial unlinkability & anonymity. The existing schemes of wormhole detection that depends upon centralized authority and monitoring schemes which may lead to other attacks & decreases rate or fails to detect the wormhole node.

3.Proposed System

In this protocol, the control & data packets are not distinguishable by outsiders. Valid nodes only can able to distinguish it by symmetric decryption. Each node that can establish a key with its neighbor , with that key only it encrypt the packet and send it to neighbor. It uses group key & pair wise key to support broadcast & unicast. USPTR that consists of two phases, they are Anonymous key establishment & Privacy preserving route discovery. USPTR that provides anonymity , Unlinkablity & Unobservability.

TABLE I

NOTATIONS

A	A node in the ad hoc network, and its real identity
s	The master secret key owned by the key server
q	A 170-bit prime number
P	Generator of the elliptic curve group G_1
$H_i(*)$	Secure one-way hash functions, $i = 1, 2, 3$
gsk_A	Node A's private group signature key
gpk	The public group signature verification key
K_A	Node A's private ID-based key which is $s \cdot H_1(A)$
$E_A(*)$	ID-based encryption using A's public key
$\bar{k}_{A*}$	A local broadcast key within A's neighborhood
k_AX	A pairwise session key shared between A and X
Nym_A	The pseudonym only valid within A's neighborhood
Nym_{AX}	The pseudonym shared between A and X
T_{rep}	Time when the first bit of RREP is received

T_{req}	Time when the last bit of RREQ is broadcasted
IPD	Intra nodal processing delay
D_{AB}	Per hop distance between A and B
ν	Speed of light
R_{Th}	Maximum threshold range for per hop distance
FA_{count}	Frequency appearance count
FA_{Th}	Frequency appearance threshold

In Anonymous key establishment , construction of session keys with neighbors takes place. In privacy preserving route discovery, finds the route to destination & path tracing algorithm also run during the route discovery in order to detect the wormhole node. Notations used in this routing scheme is described in TABLE I.

A. Anonymous Key Establishment

Each node that can communicate with its neighbor in its range. If a node S , then it has the private signing key gsk_S and a private ID-based key K_S . The procedure for anonymous key establishment is follows.

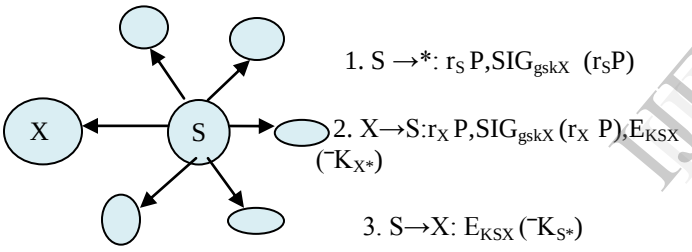


Figure 1 Anonymous key establishment. S broadcasts with its neighbors.

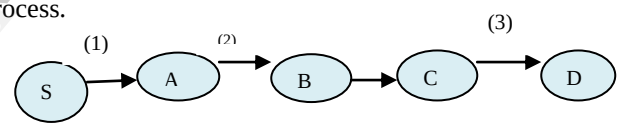
1) The node *S* first generates a random number $r_S \in Z_q$ *and then computes $r_S P$, where *P* is the generator of G1. Next it will computes a signature of $r_S P$ then with its private signing key gsk_S, able to obtain $SIG_{gskS}(r_S P)$. This signature can be verified by group public key *gpk*. Next it broadcast $(r_S P, SIG_{gskS}(r_S P))$ within its neighborhood.
2) *X* is one of the neighbor of *S* receives the message from *S* and verifies the signature in that message. If the verification is successful, then *X* chooses a random number $r_X \in Z_{q*}$ and computes $r_X P$. *X* also computes a signature $SIG_{gskX}(r_S P|r_X P)$ using its own signing key gsk_X. X then computes the session key between S & X , $k_{SX} = H_2(r_S r_X P)$, and replies to S with message $(r_X P, SIG_{gskX}(r_S P|r_X P), E_{kSX}(\bar{}k_{X*}|r_S P|r_X P))$, where $\bar{}k_{X*}$ is X's local broadcast key.
3) On receiving the reply from X, first S verifies the signature inside the message. If the signature is valid, then S proceeds to compute the session key between X and itself as $k_{SX} = H_2(r_S r_X P)$. Then S also generates a local broadcast key $\bar{}k_{S*}$, and sends the following $E_{kSX}(\bar{}k_{S*}|\bar{}k_{X*}|r_S P|r_X P)$ to its neighbor X to inform X about its local broadcast key.
4) X receives the message from S and computes the same session key , then decrypts the message with the session key to get the local broadcast key $\bar{}k_{S*}$.

Fig. 1 that describes anonymous key establishment process. The messages exchanged during this phase are not unobservable but it did not trickle any privacy information. In this phase secret session key like k_{SX} is established & local broadcast key also constructed. These keys are used in route discovery process. It prevents replay attack & session key disclosure attack.

B. Privacy Preserving route discovery

This phase is based on the keys established in the previous phase & it encloses the route request & route reply. The route request messages that overflows in entire network. The route reply is sent to source node only.

Consider source node as S, destination node as D then S has to find a route to D. Let as assume there three intermediary nodes between S & D. Fig. 2 that describes the route discovery process.

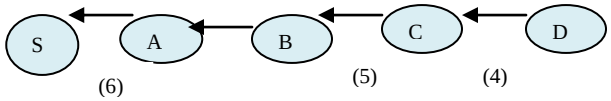


Route Request

(1):$Nonce_{s,}\ Nym_s, E_{\bar{}KS*}(RREQ, N_S, E_D(S,D,r_S P), seqno)$

(2): $Nonce_A, Nym_A, E_{\bar{}KA*}(RREQ, N_A, E_D(S,D,r_S P), seqno)$

(3): $Nonce_C, Nym_C, E_{\bar{}KC*}(RREQ, N_C, E_D(S,D,r_S P), seqno)$

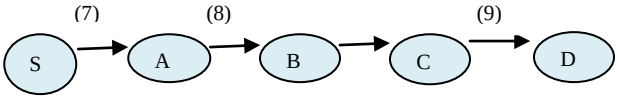


Route Reply

(4):$Nonce_{D,} Nym_{CD}, E_{KCD}(RREQ, N_C, E_S(D,S,D,r_S P, r_D P), seqno)$

(5):$Nonce_{C,} Nym_{BC}, E_{KBC}(RREQ, N_B, E_S(D,S,D,r_S P, r_D P), seqno)$

(6):$Nonce_A . Nym_{SA} . E_{K<SA}(RREQ, N_S, E_S(D,S.D.r<P. r_D P), seqno)$



Data

(7): $Nonce_{S,} Nym_{SA}, E_{KSA}(DATA, N_S, seqno, E_{KSD}(payload))$

(8): $Nonce_{A,} Nym_{AB}, E_{KAB}(DATA, N_A, seqno, E_{KSD}(payload))$

(9): $Nonce_{C,}$ Nym_{CD},E_{KCD}(DATA,N_C,seqno,E_{KSD}(payload))

Fig. 2 Privacy Preserving route discovery

1) Route Request:

First S chooses the random number r_S & uses the node D's identity to encrypt the trapdoor information which can be opened with D's private ID based key . Then it selects the sequence number for this route request & also selects N_S as route pseudonym which is the index to specific route entry .To accomplish unobservability S chooses $Nonce_S$ & calculate pseudonym $Nym_S = H_3(\bar{}k_S*|Nonce_S)$. After S encrypts these items with its local broadcast key. Then S broadcasts the following,

$$Nonce_S, Nym_S, E_{\bar{}kS*}(RREQ, N_S, E_D(S, D, r_S P), seqno) \quad (1)$$

Each node also maintains a temporary entry in its routing table (*seqno, Prev_RNym, Next_ RNym, Prev_hop,Next _hop)*, where *seqno* is the route request sequence number, *Prev _RNym* denotes the route pseudonym of previous hop, *Next_ RNym* is the route pseudonym of next hop, *Prev_hop* is the upstream node and *Next_hop* is the downstream node along the route.

Upon receiving this route request, an intermediary node A that tries with all his session keys shared with neighbors to find Nym_S .Then A tries to decrypt $E_D(S,D, r_SP)$ using his private ID based key to see whether he is the destination node. If it is an intermediary node, then it generates nonce $Nonce_A$ and a new route pseudonym N_A for this route, then calculates a pseudonym $Nym_A = H_3(\bar{}k_A*|Nonce_A)$ & broadcast (2) to its neighbors. It has the routing table entry (seqno,$N_{S,}N_{A,}$S,-) & process is repeated till it reach the destination D. After the destination D is reached , it can decrypt the trapdoor information with its private ID based key.

The duplicate request packets that can be identified by using its cache. Since cache that saves sequence number & route pseudonym.

2) Route Reply:

After D finds that he is the destination, reply messages are sent in unicast format. First *D* chooses a random number r_D and then it computes a ciphertext $E_S(D, S, r_SP, r_DP)$ showing that he is the valid destination. A session key $k_{SD} = H_2(r_S r_D P|S|D)$ is computed for data protection. Then he generates a new pairwise pseudonym $Nym_{CD} = H_3(k_{CD}|Nonce_D)$ between *C* and him. Then D computes the pairwise session key k_{CD}, using it, sends the following message to *C*:

$$Nonce_D, Nym_{CD}, E_{kCD}(RREP, N_C, E_S(D, S, r_SP, r_DP), seqno) \quad (4)$$

On receiving this message C understands who is the sender by evaluating Nym_{CD} & with the pairwise session key it decrypts the message. It identifies the corresponding route request by using seqno & N_C. Then it modifies the routing table entry as (seqno,N_B,$N_{C,}$B,D). Then it generate $Nonce_C$ & calculate Nym_{BC} & sends (5) to B. Likewise all the intermediary nodes perform the above steps till it reaches source node S.

On reaching the source node S decrypts the message with K_{SA} & computes the session key K_{SD}. Now the S finds a route to D & it modifies the routing table entry as (seqno,-,N_S,-,A). The routing table entry at each node is listed in TABLE II.

TABLE II

Routing table for all nodes

	seqno	*Prev_ RNym*	*Next_ RNym*	*Prev_ hop*	*Next_ hop*
S	seqno	-----	N_S	------	K_{A*}
A	seqno	N_S	N_A	K_{S*}	K_{B*}
B	seqno	N_A	N_B	K_{A*}	K_{C*}
C	seqno	N_B	N_C	K_{B*}	K_{D*}
D	seqno	N_C	-----	K_{C*}	-------

3) Path Tracing Algorithm:

In order to detect wormhole , we optimize USOR [1] protocol with extra fields in the packet such as prior per hop distance field, per hop distance field & timestamp is included in each packet. The difference between per hop distance & prior per hop distance is calculated. If this difference is too large when compared with the threshold value , then the wormhole node is detected. Every node in the MANET that performs this operation.

The timestamp field is initialized to the time at which the first bit of RREQ is sent & it is not altered by other nodes. Per hop distance is altered by intermediate nodes. Per hop distance is based on the Round Trip Time(RTT) that is the time between the last bit of RREQ is sent & the first bit of RREP is received.

$\Delta T = RTT = T_{rep} - T_{req}$-IPD

Per hop distance between A & B D_{AB} is given with the consideration the signals travel with the speed of light 'v'.

$D_{AB} = (v/2)*\Delta T$.

The node c is considered then compute D_{BC} then find the difference between D_{BC}-$D_{AB.}$ If it is greater than R_{Th} , then wormhole node is detected.

In order to detect the wormhole accurately, link frequent appearance is analyzed. It is given by

FA_{count} = Maximum number of times that Lj participates in a path/Total number of available links in a path $= N_j/N$.

If this FA_{count} is greater than FA_{Th}, then the wormhole node is detected 7 warning messages sent through out the network.

Steps to detect the wormhole attacks.

Step 1: Nodes in a path computes RTT values based on the time between the RREQ sent and RREP received. The RTT computation is based on its own clock.
Step 2: Compute per hop distance value using RTT value. The computed per hop distance value and timestamp are stored in each packet header.
Step 3: These informations are stored to identify the wormhole link. Every node in a path computes per hop distance with its neighbor and compares it with the prior per hop distance. If the per hop distance exceeds the maximum threshold range, R_{Th}, go to step 4.
Step 4: Check for the maximum count a link takes part in the path. If $FA_{count} > FA_{Th}$, then the link is wormhole.
Step 5: Mark the link as wormhole and the corresponding node informs other nodes to alert the network. These wormhole nodes are then isolated from the network. Fig. 3 that describes the path tracing algorithm

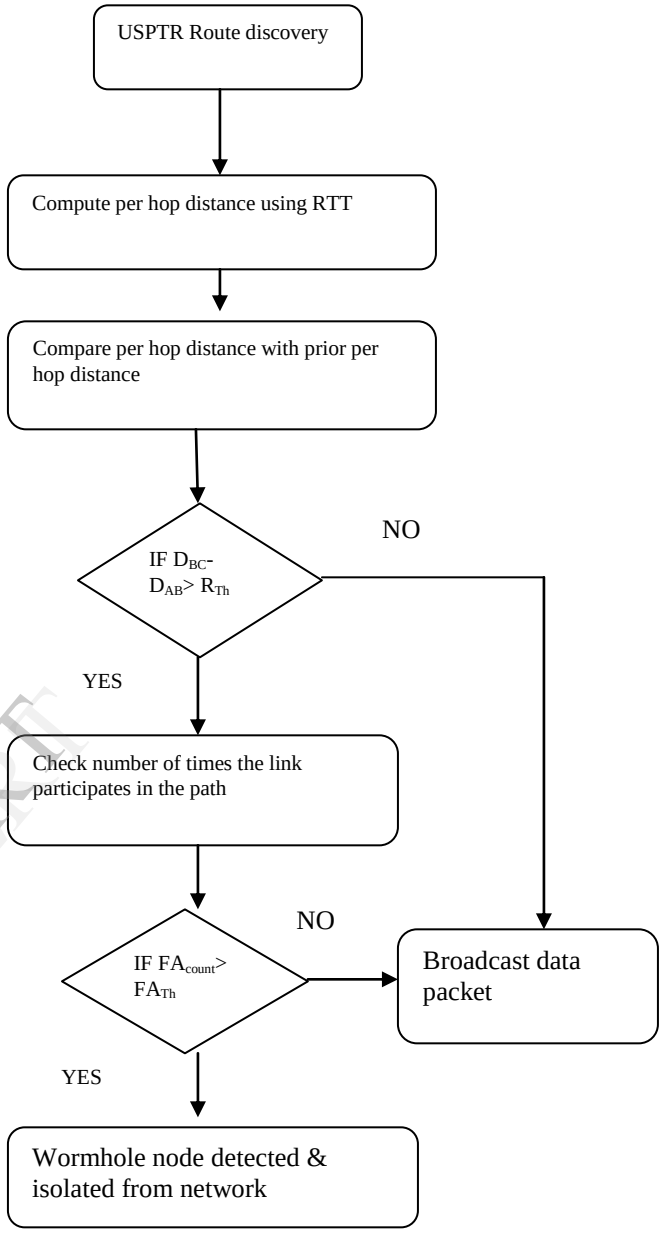


Figure 3. Path Tracing algorithm

4. Implementation And Results

We implement USPTR protocol in ns2 simulator. Then we compare its performance against USOR & AODV routing protocol.

In the simulation, 50 nodes are randomly distributed within a network field of size 1500mx300m as such a rectangle field can make the number of hops between two nodes larger. Mobile nodes are moving in the field according to the random way point model, and we adopt the speed ranges used in so that the average speeds range from 0 to 10m/s. Two different

CBR traffic loads are generated for each of the 20 pairs selected from the 50 nodes: 2 packets/s as the light load and 4 packets/s as the heavy traffic load. The local session keys are updated every 40 seconds in the simulation, and each update involves a complete anonymous key establishment procedure. To simulate cryptographic operations on each node, we force each node to delay for some time. The period a node needs to wait is determined by cryptographic operations the node performs.

We evaluate the performance of USPTR in terms of packet delivery ratio, packet delivery latency, and normalized control bytes. Fig. 4 that describes the packet delivery ratio.

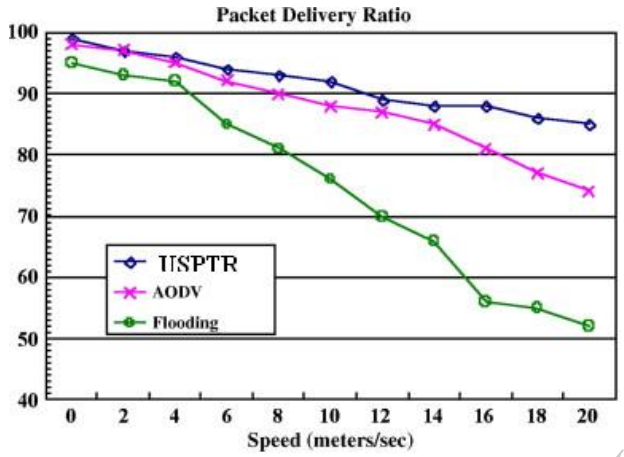


Figure 4. Packet delivery ratio

The biggest difference between USPTR and AODV on packet delivery ratio is less than 10%. The performance drop of both protocols when node speed goes up due to more frequent route disruption at higher speeds. Lower packer delivery ratio of USPTR is due to the following reasons: 1) In USPTR only trusted neighbors will forward route packets for each other, otherwise packets are simply dropped, 2) Local key update and node mobility lead to trust lost between one and its neighbors. Before neighboring nodes establish shared local keys, no traffic can be passed between them, which results in transmission delay in USPTR; 3) Route repair in AODV is not applicable in the protocol for the sake of privacy protection, as route repair requires identity information about the destination; 4) In AODV, intermediate nodes can reply to a route request if they know a route to the requested destination, while USPTR cannot do this as any intermediate node is not supposed to know either the source node or the destination node. USPTR that requires more control packets to send before sending data packets. USPTR still achieves satisfactory performance: more than 85% delivery success.

5. Conclusion

The proposed protocol that provides the strong privacy protection & also defending against wormhole attack. It offers complete Unlikability & content Unobservability. The wormhole node is detected accurately & isolated from the network. It prevents the replay attacks in other parts of network. Our future work will aim at improving the efficiency of USPTR in the terms of route changes. One possible extension is to provide the functionality of repairing broken routes locally without compromising anonymity and security.

REFERENCES

[1] Zhiguo Wan, Kui Ren, and Ming Gu, "USOR: An Unobservable Ondemand Routing protocol for Mobile Ad hoc network", 2012 ieee transactions on wireless communications, vol. 11, no. 5, may

[2] T. Sakthivel, R. M. Chandrasekaran, "Detection and Prevention of Wormhole Attacks inMANETs using Path Tracing Approach," European Journal of Scientific Research ISSN 1450-216X Vol.76 No.2 (2012), pp.240-252

[3] J. Kong and X. Hong, "ANODR: aonymous on demand routing with untraceable routes for mobile ad-hoc networks," in *Proc. ACM MOBIHOC' 03*, pp. 291–302.

[4] B. Zhu, Z. Wan, F. Bao, R. H. Deng, and M. KankanHalli, "Anonymous secure routing in mobile ad-hoc networks," in *Proc. 2004 IEEE Conference on Local Computer Networks*, pp. 102–108.

[5] S. Seys and B. Preneel, "ARM: anonymous routing protocol for mobile ad hoc networks," in *Proc. 2006 IEEE International Conference on Advanced Information Networking and Applications*, pp. 133–137.

[6] L. Song, L. Korba, and G. Yee, "AnonDSR: efficient anonymous dynamic source routing for mobile ad-hoc networks," in *Proc. 2005 ACM Workshop on Security of Ad Hoc and Sensor Networks*, pp. 33– 42.

[7] D. Sy, R. Chen, and L. Bao, "ODAR: on-demand anonymous routing in ad hoc networks," in *2006 IEEE Conference on Mobile Ad-hoc and Sensor Systems*.

[8] K. E. Defrawy and G. Tsudik, "ALARM: anonymous location-aided routing in suspicious MANETs," *IEEE Trans. Mobile Comput.*, vol. 10, no. 9, pp. 1345–1358, 2011.

[9] Issa Khalil, Saurabh Bagchi, and Ness B. Shroff, 2008 "MOBIWORP: Mitigation of the Wormhole Attack in Mobile Multihop Wireless Networks" A*d Hoc Networks*, Volume 6, Issue 3, pp. 344-362.

[10] Issa Khalil, Saurabh Bagchi, and Ness B. Shroff, 2005 "LITEWORP: A Lightweight Countermeasure for the Wormhole Attack in Multihop Wireless Networks" *IEEE International Conference on Dependable Systems and Networks*, pp. 612-621.

[11] Sun Choi, Doo-young Kim, Do-hyeon Lee, Jae-il Jung, 2008 "WAP: Wormhole Attack Prevention Algorithm in Mobile Ad Hoc Networks", *SUTC' 08 IEEE International Conference on Sensor Networks, Ubiquitous and Trustworthy Computing*, pp. 343-348.