International Journal of Engineering Research in Electronic and Communication Engineering (IJERECE) Vol 3, Issue 5, May 2016

Secure and Reliable Routing Protocols for Heterogeneous Multi Hop Wireless Networks

[1] Mr. Sathish Kumar.P, [2] Mr. Rinald Vincent.A [3] Mr. Balamuralikrishna.N [4] Mr.Venkatesan.R
[1] Assistant professor, [2] [3] [4] Final Year,
[1][2][3][4]Department of Electronics and Communication Engineering, Vels University, Chennai, India
[2]rinaldvincent11@gmail.com, [3]bmk9867@gmail.com

***Abstract*— In this paper, we propose E-STAR for establishing stable and reliable routes in heterogeneous multihop wireless networks which combines payment and trust systems with a trust-based and energy-aware routing protocol. The payment system rewards the nodes that relay others' packets and charges those that send packets. The trust system evaluates the nodes' competence and reliability in relaying packets in terms of multi-dimensional trust values. The trust values are attached to the nodes' public-key certificates to be used in making routing decisions. E-STAR can stimulate the nodes not only to relay packets. This is because any loss of trust will result in loss of future earnings. Moreover, for the efficient implementation of the trust system, the trust values are computed by processing the payment receipts. Analytical results demonstrate that E-STAR can secure the payment and trust calculation without false accusations. Simulation results demonstrate that our routing protocols can improve the packet delivery ratio and route stability.**

I. INTRODUCTION

In multihop wireless networks, when a mobile node needs to communicate with a remote destination, it relies on the other nodes to relay the packets [1]. This multihop packet transmission can extend the network coverage area using limited power and improve area spectral efficiency. In developing and rural areas, the network can be deployed more readily and at low cost. We consider the civilian applications of multihop wireless networks, where the nodes have long relation with the network. We also consider heterogeneous multihop wireless networks (HMWNs), where the nodes' mobility level and hardware/energy resources may vary greatly. HMWNs can implement many useful applications such as data sharing and multimedia data transmission . In military and disaster-recovery applications, the nodes' behavior is highly predictable because the network is closed and the nodes are controlled by one authority. However, the nodes' behavior is unpredictable in civilian applications for different reasons. The nodes are typically autonomous and self-interested and may belong to different authorities. The nodes also have different hardware and energy capabilities and may pursue different goals. In addition, malfunctioned nodes frequently drop packets and break routes due to faulty hardware or software. Since the mobile nodes are battery driven and one of the major sources of energy consumption is radio transmission, selfish nodes are unwilling to lose their battery energy in relaying other users packets. When more nodes are cooperative in relaying packets, the routes are shorter, the network connectivity is more, and the possibility of network partition is lower. Moreover, since the nodes are equipped with different hardware capability, such as CPU speed and buffer size, the nodes having large hardware resources can perform packet relay more successfully than others.

II. SYSTEM MODELS

2.1 Network Model

The considered HMWN has mobile nodes and offline trusted party (TP) whose public key is known to all the nodes. The mobile nodes have different hardware and energy capabilities. The network is used for civilian applications, its lifetime is long, and the nodes have long relation with the network. Thus, with every interaction, there is always an expectation of future reaction. Each node has a unique identity and public/private key pair with a limited-time certificate issued by TP. Without a valid certificate, the node cannot communicate nor act as an intermediate node. TP maintains the nodes' credit accounts and trust values. Each node contacts TP to submit the payment receipts and TP updates the involved nodes' payment accounts and trust values. This contact can occur via cellular networks or Internet.

2.2 Adversary Model

The adversaries have full control on their nodes. They can change the nodes' normal operation and obtain the cryptographic credentials. They may attempt to attack

the payment system to steal credits, pay less, or communicate for free. Some adversaries may report incorrect energy capability to increase their chance to be selected by the routing protocol, e.g., to earn more credits. The adversaries may also attempt to attack the trust system to falsely augment their trust values to increase their chance to participate in routes. They may try to defame other nodes' trust values. Attackers may launch denial-of-service attacks by breaking the communication routes intentionally. When a node B receives packets from A to forward to the next node in the route, B drops the packets and keeps silent to let A believe that B is out of transmission range and the link between them is broken. These attacks may be launched by compromised, malfunctioned, or low-resource nodes. The mobile nodes are probable attackers but TP is fully secure. The nodes are autonomous and self-interested and thus motivated to misbehave, but TP is run by an operator that is interested in ensuring the network secure operation...

III. THE PROPOSED E-STAR

Fig. 1 shows that E-STAR has three main phases. In Data Transmission phase, the source node sends messages to the destination node. In Update Credit-Account and Trust Values phases, TP determines the charges and rewards of the nodes and updates the nodes' trust values. Finally, in Route Establishment phase, trust-based and energy-aware routing protocol establishes stable communication routes.

3.1 Data Transmission Phase

Let the source node S send messages to the destination node D through a route with the intermediate nodes X, Y, and Z. The route is established by the routing protocols that will be discussed in Subsection 4.3. For the ith data packet, S computes the signature ĥS(i) = {H(mi), ts, R, i)}KS+ and sends the packet < R, ts, i, mi, ĥS(i)> to the first node in the route. R, ts, and mi are the concatenation of the identities of the nodes in the route (R = IDS, IDX, IDY, IDZ, IDD in Fig. 2), the route establishment time stamp, and the ith message, respectively. H(d) is the hash value resulted from hashing the data d using the hash function H(). {d}KS+ is the signature of d with the private key of S. The purpose of the source node's signature is to ensure the message's authenticity and integrity and secure the payment by enabling TP to ensure that S has sent i messages. Each intermediate node verifies ĥS(i) and stores ĥS(i) and H(mi) for composing the receipt. It also removes the previous ones (ĥS(i-1) and H(mi-1)) because ĥS(i) is enough to prove transmitting i messages. Signing H(mi) instead of mi can reduce the receipt size because the smaller-size H(mi) is attached to the receipt instead of mi. The destination node generates a one-way hash chain by iteratively hashing a random value hS S times to obtain the hash chain {hS, hS-1,…, h1, h0}, where hi-1 = H(hi) for 1 " i " S and h0 is called the root of the hash chain. The node signs h0 and R to authenticate the hash chain and links it to the route, and sends the signature to the source node in route establishment phase. In order to acknowledge receiving the message mi, the destination node sends ACK packet containing the preimage of the last released hash chain element or hi. Each intermediate node verifies the hash chain element by making sure that hi-1 is obtained from hashing hi, and saves hi for composing the receipt and removes hi1. The underlying idea is that ĥS(i) and hi are undeniable proofs for sending and receiving i messages, respectively. Each node in the route composes a receipt and submits it when it has a connection to TP to claim the payment and update its trust values. A receipt is a proof for participating in a route and sending, relaying, or receiving a number of messages. A receipt contains R, ts, i, H(mi), h0, hi, Cm, and an undeniable cryptographic token for preventing payment manipulation. Cm is data that depends on the used routing protocol, such as the number of messages the intermediate nodes commit to relay. The cryptographic token contains the hash value of the last source node's signature and Auth_Code. Auth_Code is the authentication code that authenticates the hash chain and the intermediate nodes to hold them accountable for breaking the route. More details about Cm and Auth_Code will be given in Subsection 4.3. If i messages are delivered, the format of the receipt is <R, ts, i, H(mi), h0, hi, Cm, H(ĥS(i), Auth_Code)>. ĥS(i) and Auth_Code are hashed to reduce the receipt's size.

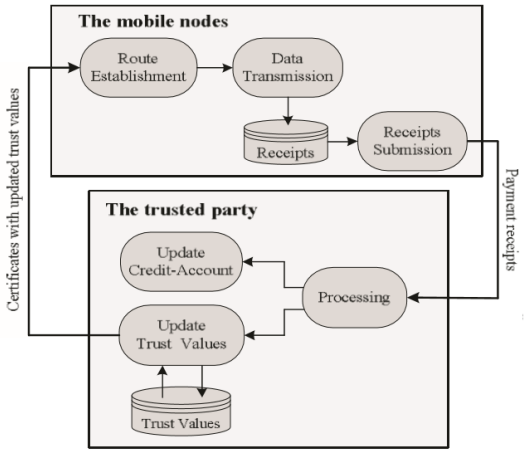


Fig. 1: The architecture of E-STAR.

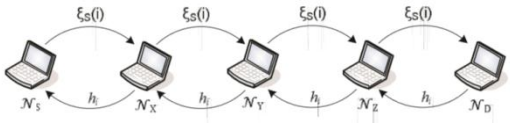
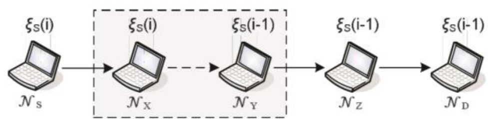


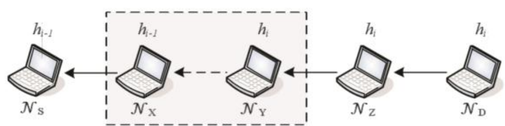
Fig. 2: The exchanged cryptographic tokens during data transmission.

3.2 Update Credit Account and Trust Values Phase

Once TP receives a receipt, it first checks if the receipt has been processed before using its unique identifier (R, ts). Then, it verifies the credibility of the receipt by computing the nodes' signatures (ȟS(i) and Auth_Code) and verifies the credibility of the receipt by computing the nodes' signatures (ȟS(i) and Auth_Code) and hashing them. The receipt is valid if the resultant hash value is identical to the receipt's cryptographic token. TP verifies the destination node's hash chain by making sure that hashing hi i times produces h0. TP clears the receipt by rewarding the intermediate nodes and debiting the source and destination nodes. The number of sent messages (i) is signed by the source node and the number of delivered messages can be computed from the number of hashing operations to obtain h0 from hi..A node can protect its trust values by not involving itself in routes with a neighbor that frequently breaks routes or has low trust values. Additionally, we are sure that the nodes that are not in a broken link did not break the route, which coincides with our objective of identifying good nodes.



a. A broken route during relaying the ith data packet.



b. A broken route during relaying the ith ACK packet.
Fig. 3: Evaluating the nodes' trust values.

Our trust system adopts multi-dimensional trust management framework in which the notion of trustworthiness is further classified into several attributes (or dimensions). Each attribute can indicate to what extent the node will conduct one specific action. We use multi-dimensional trust values instead of one trust value to precisely predict the nodes' future behavior. Since there is a stronger belief in the trust values that are computed from recent sessions, given in Eq. 4 depicts how k was recently active in participating in sessions, or to what extent the other trust values are fresh, i.e., computed from recent sessions. is the total number of sessions k participated in, in the last period ——————————————————ݐover a normalizing factor ( ) that depicts the maximum number of sessions a trusted node should participate in, in t. Note that the maximum value of is 1, and thu if < the number of sessions k participated in, t.

Since a connection to TP may not be available on a regular basis, the receipts may be submitted after some time, and thus the trust values may be updated after some delay. This is acceptable because: (1) the routing protocol is sensitive to any degradation in the trust values; and (2) the nodes' behavior is repetitive, i.e., for a normal node the probability of breaking a route is fixed.

$\mathcal{N}_S \rightarrow *$: RREQ, $D = (ID_S, ID_D, H_{max}, ts, T_r, E_r)$, $\{D\}K_{S+}$, $Cert_S$
$\mathcal{N}_X \rightarrow *$: RREQ, D, $\underline{ID_X}$, $\underline{\{\{D\}K_{S+}\}K_{X+}}$, $Cert_S$, $\underline{Cert_X}$
$\mathcal{N}_Y \rightarrow *$: RREQ, D, ID_X, $\underline{ID_Y}$, $\underline{\{\{\{D\}K_{S+}\}K_{X+}\}K_{Y+}}$, $Cert_S$, $Cert_X$, $\underline{Cert_Y}$
$\mathcal{N}_Z \rightarrow *$: RREQ, D, ID_X, ID_Y, $\underline{ID_Z}$, $\underline{Sig = (\{\{\{\{D\}K_{S+}\}K_{X+}\}K_{Y+}\}K_{Z+})}$, $Cert_S$, $Cert_X$, $Cert_Y$, $\underline{Cert_Z}$

a. The format of RREQ packet.

$\mathcal{N}_D \rightarrow \mathcal{N}_Z$: RREP, $R = (ID_S, ID_X, ID_Y, ID_Z, ID_D)$, h_0, $\{Sig, h_0\}K_{D+}$, $Cert_D$
$\mathcal{N}_Z \rightarrow \mathcal{N}_Y$: RREP, R, h_0, $\{Sig, h_0\}K_{D+}$, $Cert_D$, $\underline{Cert_Z}$
$\mathcal{N}_Y \rightarrow \mathcal{N}_X$: RREP, R, h_0, $\{Sig, h_0\}K_{D+}$, $Cert_D$, $Cert_Z$, $\underline{Cert_Y}$
$\mathcal{N}_X \rightarrow \mathcal{N}_S$: RREP, R, h_0, $\{Sig, h_0\}K_{D+}$, $Cert_D$, $Cert_Z$, $Cert_Y$, $\underline{Cert_X}$

b. The format of RREP packet.
Fig. 4: The format of RREQ and RREP packets in the SRR routing protocol.

3.3 Route Establishment Phase

In this section, we present two routing protocols called the Shortest Reliable Route (SRR) and the Best Available Route (BAR). SRR establishes the shortest route that can satisfy the source node's trust, energy, and route-length requirements, but the destination node selects the best route in the BAR protocol. The routing protocols have three processes: i) Route Request Packet (RREQ) delivery; ii) Route selection; and iii) Route Reply Packet (RREP) delivery.

3.3.1 The SRR Routing Protocol

To establish a route to the destination node D, the source node S broadcasts RREQ packet and waits for RREP packet. The source node embeds its requirements in the RREQ packet, and the nodes that can satisfy these requirements broadcast the packet. The destination node establishes the shortest route that can satisfy the source node's requirements. The rationale of the SRR protocol is that the node that satisfies the source node's requirements is trusted enough to act as a relay. The protocol is useful to establish a route that avoids the low-trusted nodes

Route Selection: If there is a route that can satisfy the source node's requirements, the destination node receives at least one RREQ packet. The destination node composes the RREP packet for the route traversed by the first received RREQ packet, and sends it to the source node. This route is the shortest one that can satisfy the source node's requirements. The source node's requirements cannot be achieved if it does not receive the RREP packet within a time period. It can initiate a second RREQ packet

but with more flexible requirements, e.g., by increasing Hmax and/or decreasing Er and Tr, or revert to the BAR protocol.

3.3.2 The BAR Routing Protocol

RREQ: As shown in Fig. 5, the RREQ packet contains IDS, IDD, ts, Hmax, the source node's certificate and signature (SigS), and the number of messages it needs to send (Er(S)). For the first received RREQ packet, an intermediate node broadcasts the packet after attaching its identity and certificate, the number of messages it commits to relay (Er(X)). Unlike the SRR protocol, Er(X) can be fewer than Er(S). also signs the concatenation of Er(X) and the signature received in the RREQ packet. Er(X) not only depends on the available battery energy in , but also on other factors such as the cooperation strategy (or the node's willingness for relaying packets) and the link quality and stability. For example if the links between and its two neighbors in the route are unstable, it can decrease its Er(X) to decrease the probability of breaking the route. The nodes are motivated to report correct energy commitments to avoid breaking the route and thus degrading their trust values.

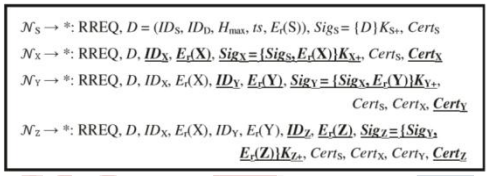


Fig. 5: The format of RREQ packet in the BAR routing protocol.

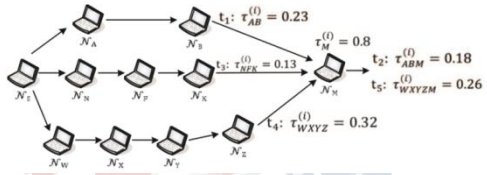


Fig. 6: Broadcasting the RREQ packet in the BAR routing protocol

Blind RREQ flooding generates few routes because each node broadcasts the packet once, which disables potential better routes. To solve this issue, BAR allows each node to broadcast the RREQ more than once if the route reliability or lifetime of the recently received packet is greater than the last broadcasted packet. The route lifetime is the minimum number of packets the intermediate nodes commit to relay, e.g., if the commitments of the intermediate nodes are Er(X) = 10, Er(Y) = 8, and Er(Z) = 17, the route lifetime is 8 packets.
Route Selection:

After receiving the first RREQ packet, the destination node waits for a while to receive more RREQ packets if there are. Then, it selects the best available route if a set of feasible routes are found. If there are multiple routes with lifetimes at least Er(S), the destination node selects the most reliable route, otherwise, it establishes multiple routes with at least total lifetime of Er(S) in such a way that reduces the routes' number and maximizes the reliability. The destination node should not select multiple routes with common node(s) (if possible) to disallow one node to break the routes.

IV. SECURITY ANALYSIS

Securing the payment and trust calculation are based on the following well known cryptographic properties: (1) forging or modifying a signature without knowing the private key is infeasible; (2) deriving the private keys from the public ones is infeasible; (3) computing the hash value of a signature without computing the signature is infeasible; and (4) computing the hash function's input from its output is infeasible. The hash function is unidirectional in the sense that it is feasible to compute H(X) from X, but it is infeasible to compute X from H(X). These cryptographic properties are used to enable TP to make sure that the source, intermediate, and destination nodes have indeed participated in a route and to verify the number of transmitted, received, and relayed messages by each node. They also enable the intermediate nodes to compose valid receipts and verify them In route establishment, the nodes that report incorrect trust values can be detected because the trust values are signed by TP. The nodes cannot manipulate their trust values because they cannot forge the TP's signature. For destination node impersonation attack, the attacker attempts to send RREP packet to let the source node believe that it communicates with the destination node. This is infeasible in E-STAR because the destination nodes sign the RREP packets to ensure that only the destination node can respond to the RREQ packet. For the RREQ flooding attack launched by internal attackers, since the source nodes sign the RREQ packets, the attackers can be identified in an undeniable way. The network nodes can ignore a node's packets when it sends a large number of RREQ packets in a short time. For route lengthening attack, in E-STAR, elongating a route by inserting non-existing nodes to the RREQ packet requires signing the packet with the private keys of these nodes. It also decreases the chance of selecting the route because the route reliability decreases, as discussed in Table 2. More security analyses are given in Appendix A

Fig. 7: E-STAR can improve the packet delivery ratio due to selecting good intermediate nodes.

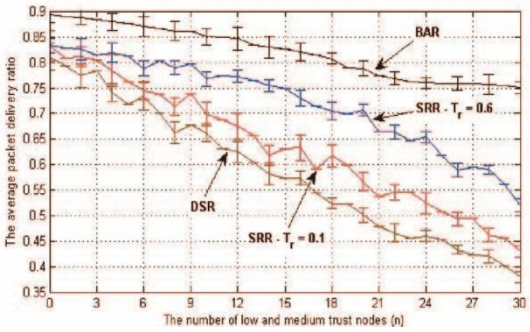


Fig. 8: SRR generates fewer RREQ broadcasts because the nodes that cannot satisfy the source node's requirements do not broadcast the packets.

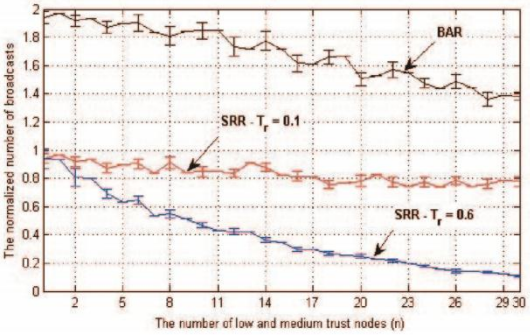
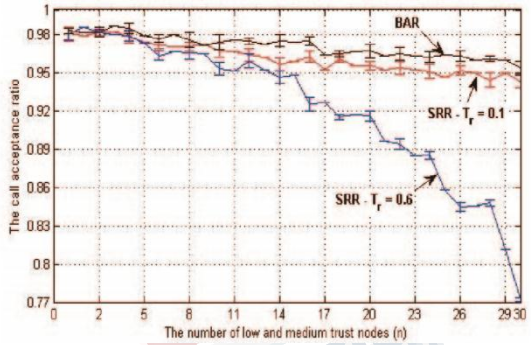


Fig. 9: Routes are not established if the source node's trust requirement is not well selected in SRR.



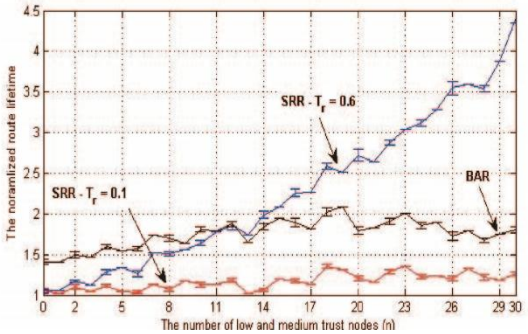
V. PERFORMANCE EVALUATIONS

We simulate a heterogeneous multihop wireless network by randomly deploying 55 nodes in an area of 1000 m × 1000 m. n is the number of nodes having low and medium trust values. The number of nodes having high trust values is 55-n and their trust values are uniformly distributed in [0.8, 1). The number of nodes having low trust values is [0.67 x n] and their trust values are uniformly distributed in [0, 0.3). The number of nodes having medium trust value is [0.33 x n]and their trust values are uniformly distributed in [0.3, 0.8). A node with a trust value of 0.6 breaks routes with the probability of 1 - 0.6 = 0.4. By this way, the trust values can be used to simulate the variety in the nodes' lack of resources and malicious actions. When we compare our protocols with DSR, we actually compare between two strategies: informed routing decisions and randomly selecting intermediate nodes. DSR randomly selects intermediate node, but our protocols make informed routing decisions by selecting the nodes that behaved well in the past and have enough energy. Therefore, improvement techniques proposed for DSR such as route recovery schemes can also be used with our protocols. We can see that the packet delivery ratio of DSR significantly degrades as the number of low-trust nodes increases due to involving these nodes in routes more frequently. For SRR, the increase of Tr can increase the packet delivery ratio due to selecting more trusted nodes, but as we will discuss later the probability of establishing routes decreases. At Tr = 0.1, the increase of n decreases PDR because more low-trust nodes participate in routes. However, the reduction in PDR at Tr = 0.6 is mainly due to the messages the source nodes could not send because they did not find routes with this trust requirement. In BAR, the increase of the low-trust nodes has little effect on PDR because it can avoid these nodes and select nodes with good trust values and sufficient energy. Moreover, we can see that at n = 0 and at few low-trust nodes, the packet delivery ratios in SRR and BAR are higher than that of DSR, because they select the nodes having sufficient energy.

10 Fig. 8 shows the number of RREQ broadcast transmissions in E-STAR to this of the DSR at different values of n. The Wait Period at each node is 20ms in BAR. We can see that the normalized number of broadcasts in SRR is always less than one because the nodes that cannot satisfy the energy or trust requirements do not broadcast the RREQ packets. At Tr = 0.6, the number of broadcasts is less because more nodes cannot satisfy the trust requirements and thus do not broadcast RREQ packets. For BAR, the normalized number of broadcasts is always above one because a node may broadcast a RREQ packet more than once, but in DSR each node broadcasts a RREQ packet at most once. In Fig. 9, the call acceptance ratio is the ratio of times a route is established after sending a RREQ packet. We can see that the call acceptance ratio in BAR nearly does not depend on n. However, the increase of n decreases the call acceptance ratio in SRR because more nodes cannot satisfy the trust requirement, and thus more routes cannot be established. At Tr = 0.6, the call acceptance ratio significantly decreases with the increase of n because more nodes cannot satisfy the trust requirement. In Fig. 10, the normalized route lifetime is the average route lifetime in E-STAR to that of DSR. The route lifetime is the number of packets sent in one route before it is broken. Route lifetime is a good measure for route stability. Since the normalized route lifetime is always more than one, E-STAR can establish more stable routes comparing to DSR. At n > 12, SRR with Tr = 0.6 may establish more stable routes but as indicated in Fig. 9,

the likelihood of establishing a route decreases as n increases. More simulation results are given in Appendix A.

Fig. 10: The route lifetime in E-STAR is more than that in DSR because of establishing more stable routes.



VI. CONCLUSION

We have proposed E-STAR that uses payment/trust systems with trust-based and energy-aware routing protocol to establish stable/reliable routes in HMWNs. E-STAR stimulates the nodes not only to relay others' packets but also to maintain the route stability. It also punishes the nodes that report incorrect energy capability by decreasing their chance to be selected by the routing protocol. We have proposed SRR and BAR routing protocols and evaluated them in terms of overhead and route stability. Our protocols can make informed routing decisions by considering multiple factors, including the route length, the route reliability based on the nodes' past behavior, and the route lifetime based on the nodes' energy capability. SRR establishes routes that can meet source nodes' trust/energy requirements. It is useful in establishing routes that avoid the low-trust nodes, e.g., malicious nodes, with low overhead. For BAR, destination nodes establish the most reliable routes but with more overhead comparing to SRR. The analytical results have demonstrated that E-STAR can secure the payment and trust calculation without false accusations. Moreover, the simulation results have demonstrated that E-STAR can improve the packet delivery ratio due to establishing stable routes.

REFERENCES

[1] G. Shen, J. Liu, D. Wang, J. Wang, and S. Jin, "Multi-hop relay for nextgeneration wireless access networks", Bell Labs Technical Journal, vol. 13, no. 4, pp. 175-193, 2009.

[2] C. Chou, D. Wei, C. Kuo, and K. Naik, "An efficient anonymous communication protocol for peer-to-peer applications over mobile ad-hoc networks", IEEE Journal on Selected Areas in Communications, vol. 25, no. 1, January 2007

[3] S. Marti, T. Giuli, K. Lai, and M. Baker, "Mitigating routing misbehavior in mobile ad hoc networks", Proc. of IEEE/ACM MobiCom'00, pp. 255–265, Boston, MA, August 6-11, 2000

[4] X. Li, Z. Li, M. Stojmenovic, V. Narasimhan, and A. Nayak, "Autoregressive trust management in wireless ad hoe networks", Ad hoc & Sensor Wireless Networks, vol. 16 Issue 1-3, pp. 229-242 2012

[5] G. Indirania, K. Selvakumara, "A swarm-based efficient distributed intrusion detection system for mobile ad hoc networks (MANET)", International Journal of Parallel, Emergent and Distributed Systems, 2013.

[6] H. Li and M. Singhal, "Trust management in distributed systems", IEEE Computers, vol. 40, no. 2, pp. 45-53, February 2007.

[7] K. Liu, J. Deng, and K. Balakrishnan "An acknowledgement-based approach for the detection of routing misbehavior in MANETs", IEEE Transaction on Mobile Computing, vol. 6, no. 5, pp 536–550, May 2007.