Secure opinion sharing for reputation-based systems in mobile ad hoc networks

Abdur Rashid Sangi1, Jianwei Liu2, Mohammed S Alkatheiri3 and Satish Anamalamudi4

Abstract
Due to the basic nature of mobile ad hoc networks, that is, infrastructure-less, it is prone to individual or collective misbehaviors by participating node(s). Participating nodes could act selfishly and does cause massive loss to network performance because of limited resources or belonging to a different administrative domain. Reputation-based solutions are widely used to mitigate selfishness. These solutions are to some extent depend on the feedback from participating nodes for any given node which required its secure exchange in an adverse environment. This paper introduces a secure opinion sharing based on network coding to ensure the effectiveness of any reputation against selfishness in an adverse environment. The proposed scheme addresses the threat to opinion exchange in any reputation-based solution with minor changes. In addition, it can be used to exchange secure data in an adverse environment, for example, virtual currency and feedback exchange for credit payment and game theory–based solutions, respectively. Simulation results proved that this scheme achieves excellent opinion exchange ratio, moderate delay, and affordable per cycle overhead.

Keywords
Selfishness, reputation-based solutions, secure opinion sharing, network coding, network simulator–2, MANETs

Date received: 27 July 2019; accepted: 13 December 2019

Introduction
A wireless network without any centralized access point and additionally does not depend upon pre-existing infrastructure is called ad hoc networks. Wireless with mobility-based ad hoc network is created with set of mobile nodes that are connected through a radio frequency (RF) of infrared interface and have a capability of communicating with one another by creating connections through a decentralized manner. All the mobile nodes within the mobile ad hoc network (MANET) have equal importance. In other words, any node within the network will work as a host or router and might communicate by transmitting the data on to any node or device on the network. Hence, MANETs face enormous challenges due to its basic peer to peer–based architecture. As it is an infrastructure-less network and rely on node participation to carry out network operations, it is easy to disrupt by anyone or group of participating nodes. Although these networks are easy to install, support mobility does not require fixed infrastructure and are suitable in emergency situations but it becomes more difficult if the participating nodes belong to different administrative body, that is, several organizations in the context of a commercialized network.

Even for MANETs in the same administrative domain, there exists the possibility that malicious nodes could disrupt this network due to any mutual cause. Apart from other limitations, each node is resources constraint and to prolong its life, a node may act selfishly by consuming its resources only for its own operations. Selfishness mitigation remains an interesting research topic, and so far various solutions1–3 are proposed that fall under either reputation, credit payment, game theory, load balancing–based category, or a combination of these techniques.

1Faculty of Computer and Software Engineering, Huaiyin Institute of Technology, Huaiian, China
2School of Cyber Science and Technology, Beihang University, Beijing, China
3College of Computer Science and Engineering, Department of Cybersecurity, University of Jeddah, Jeddah, Saudi Arabia
4Department of Computer Science Engineering, SRM University–AP, Amaravati, India

Corresponding author:
Abdur Rashid Sangi, Faculty of Computer and Software Engineering, Huaiyin Institute of Technology, Huaiian 223003, China. Email: sangi_bahrian@yahoo.com

Creative Commons CC BY: This article is distributed under the terms of the Creative Commons Attribution 4.0 License (https://creativecommons.org/licenses/by/4.0/) which permits any use, reproduction and distribution of the work without further permission provided the original work is attributed as specified on the SAGE and Open Access pages (https://us.sagepub.com/en-us/nam/open-access-at-sage).
Many researchers proposed different reputation-based solutions require opinion sharing by all or few participating nodes to decide the status of any particular node. In an adverse environment, non-cooperative nodes could alter this valuable information and deceive the requesting node to perceive a legitimate node, selfish or not. The origin of network coding is from the concurrent nonlinear theory and linear theory. To date, network coding is having wide applications to many established fields including coding theory, computer networks, computer science, distributed data storage, information security, information theory, optimization theory, peer-to-peer (P2P) content delivery, switching theory, and wireless/satellite communications. As shown in Figure 1(a) and (b), it creates a multicast network as a finite acyclic-directed multi-graph with a single source node. An edge within the directed multi-graph represents a noiseless communication channel of unit capacity. The source or head node generates a message per unit time for the transmission to receiver nodes (members). Figure 1(a) shows a simple example of network coding in a cluster formation where both member nodes are subscribed to the same cluster, and Figure 1(b) shows a situation when both communicating nodes are a member of two different clusters. To exploit these features of network coding and use message authentication code (MAC) to ensure the integrity of shared opinion. This secure opinion could be used with any existing or new reputation-based solution to increase its efficiency, reliability, and robustness.

This paper, a network coding–based opinion sharing technique that ensures integrity and confidentiality of valuable feedback to conclude correct perception about any given node was proposed. Along with maximizing network throughput, network coding could be used to provide data confidentiality. The rest of this paper structured as follows. The second section review the previous work related to reputation-based solutions and network coding. The third section introduces the detail steps of the proposed secure opinion sharing scheme. Finally, the simulation implementation details including the data structure and transitional chart are explained and concluded in detail.

**Related work**

The functionality of the MANET is different from the fixed IP network and the infrastructure-based wireless network, which leads to new challenges to guarantee its security. This is because the distributed wireless communication manner makes the threat of being attacked, thus suffering from a series of security issues like black hole, wormhole, and so on. In addition, the self-organization of mobile nodes blurs the boundary of network. Due to this, mobile node in the MANET is prone to physical security fragile. To contain the limited resources, a participating node in MANETs could act selfishly by denying entertaining the interest of other nodes. Any degree of non-cooperation by a node or group of nodes would result in a decline in overall network performance. The chances would be more in a situation where all participating nodes do not belong to the same administrative domain. Selfishness is widely mitigated by using reputation systems. These reputation systems rely on the observation feedback that is provided by other nodes for any specific node. We call this feedback “opinion.” Under a famous reputation-based solution, the participating nodes exchange their opinion in a non-secure fashion and do not ensure its integrity. This valuable information is the only source to conclude the status of any observed node and if this information has tampered then it could make a selfish node, non-selfish, or vice versa.

Applications of network coding are no more limited only to enhance network throughput, but the features of network coding also made it a candidate to devise network security solutions. These security solutions include an efficient scheme based on XOR network coding to combat pollution attack, an algorithm to resist byzantine attacks, an efficient protocol for wireless sensor networks based on network coding, and so on. The nature of MANETs, that is, ever-changing network topology, multi-hop, decentralized, and self-organizing properties, introduces more serious security challenges than those in static networks. An extremely important problem is how to distribute and update an opinion in a secure way and with its integrity intact. We did not find any similar technique.
to address this issue. The opinion is the only input to a reputation system that helps to conclude the behavior of any given node to a selfish node or legitimate. With this consideration, we have proposed a secure opinion sharing mechanism that could be used with any reputation system.

The proposed scheme

This scheme has three reasonable requirements, cluster-oriented topology to enable network coding strategy, offline trusted third party (TTP) to distribute required information at the beginning, and sufficient memory space at each participating node. Table 1 provides abbreviation details, and our scheme is explained below in three parts: initializing, opinion exchange, and opinion updating part.

Initializing part

The offline TTP in the network generates a secret key $K_i \in P$, where $P$ is the complete key pool generated by TTP, and the corresponding identifiers $ID_i$, $i \in \{0, \ldots, N-1\}$ for each of the ad hoc node. TTP stores an entire encrypted list of the other node’s keys $K_i \oplus a_j$, $j = 0, \ldots, i-1, i+1, \ldots, N-1$ (notice that $a_i = a_j$) into node $i$ alone with all corresponding identifiers of all ad hoc nodes. Then TTP chooses a secure hash function $h(x)$.

Note that, after a complete initialization phase, each node only knows its own secret key and the encrypted version of other nodes. This will reduce the possibility of secret key leakage when one node is captured and compromised.

Opinion exchange part

After ad-hoc node deployment is complete, there are two distinguishable cases that have already been mentioned above.

First case: Both the given ad-hoc nodes belong to the same clusterhead. Figure 2 shows the case.

First verification and computation. Source member node $A$ computes $opinion_A \oplus \{opinion_A \oplus opinion_B\} = opinion_B$, and $MAC_A \oplus \{MAC_A \oplus MAC_B\} = MAC_B$, and then computes the MAC of destination node that it received, $MAC_B = h(opinion_B||K_A \oplus (K_B \oplus a_{AB}))$.

Destination node $B$ computes $opinion_B \oplus \{opinion_A \oplus opinion_B\} = opinion_A$, and $MAC_B \oplus \{MAC_A \oplus MAC_B\} = MAC_A$, and then computes the MAC of source node that it received, $MAC_A = h(opinion_A||K_B \oplus (K_A \oplus a_{AB}))$.

Source member node $A$ verifies to confirm if $MAC_B = MAC_B$ and destination member node $B$ verifies to confirm if $MAC_A = MAC_A$. If they are identical, then both member node $A$ and member node $B$ will perceive that the opinion was shared securely and without losing its integrity.

Second case: Both ad-hoc nodes are subscribed to two different clusterheads. Figure 3 shows more details of the protocol.

Step 1. Node $member_A$ sends its opinion $opinion_A$, a message authentication code $MAC_A = h(opinion_A||K_A \oplus (K_B \oplus a_{AB}))$, addresses of $member_A$, $member_B$, and $member_z$ (for which the opinion is shared) to its
clusterhead \( Head_i \), \( i \in \{1, \ldots, N\} \), where \( N \) is the maximum number of clusterheads in the ad-hoc network at present.

Step 2. Upon receiving the message from \( member_A \), \( Head_i \) first checks if \( member_A \) and \( member_B \) are associated with it. If not, \( Head_i \) records \( member_A \), \( member_B \), \( member_z \), \( opinion_A \), \( MAC_A \) and broadcasts a seven-tuple \( member_A, member_B, member_z, opinion_A, MAC_A, r_j, Head_i \) to the other clusterheads, where \( r_j \) is the new random challenge generated by \( Head_i \).

Step 3. As soon as the clusterhead \( Head_j \) receives the seven-tuple \( member_A, member_B, member_z, opinion_A, MAC_A, r_j, Head_i \) broadcasted from \( Head_i \), \( Head_j \) knows that \( member_A \) want to share an opinion with \( member_B \) that is associated as its member. Then \( Head_j \) saves the seven-tuple \( member_A, member_B, member_z, opinion_A, MAC_A, r_j, Head_i \) and broadcasts \( member_A, member_B, member_z \) to \( member_B \).

Step 4. Upon receiving \( member_A, member_B, member_z \), the \( member_B \) comes to know that \( member_A \) wants to communicate with it. \( member_B \) then sends its opinion regarding \( member_z \) and \( MAC_B = h(opinion_B || K_A \oplus (K_B \oplus a_{AB})) \) to \( Head_j \).

Step 5. Upon receiving \( opinion_B, MAC_B \), \( Head_j \) first performs a simple table search and generates a random challenge \( r_j \) and sends a five-tuple \( opinion_B, MAC_B, Head_j, r_j, MAC_A \) to \( Head_i \).

Step 6. Upon receiving the 5-tuple, \( Head_i \) performs a simple table search and computes \( MAC_A = h(r_j || K_j \oplus (K_i \oplus a_{ij})) \). Then \( Head_i \) checks if \( MAC_A = MAC_i \). If the two values are the same, \( Head_i \) authenticates \( Head_j \). Then \( Head_i \) computes the value \( opinion_A \oplus opinion_B \oplus MAC_A \oplus MAC_B \) and broadcasts the value to destination member \( member_B \).

Step 7. On receipt, \( Head_i \) computes the value \( opinion_A \oplus opinion_B \oplus MAC_A \oplus MAC_B \) and uses network coding paradigm to broadcast the value to source member \( member_A \). Upon receiving \( Head_i, MAC_A, Head_i \) performs a simple table look-up and computes \( MAC_A = h(r_j || K_i \oplus (K_j \oplus a_{ij})). \) Then \( Head_i \) checks if \( MAC_A = MAC_i \). If the two values are equal, then \( Head_i \) authenticates \( Head_j \). Then \( Head_i \) performs a simple table look-up and computes \( MAC_A = h(r_j || K_i \oplus (K_j \oplus a_{ij})). \) Then \( Head_i \) unicasts \( Head_j, MAC_A \) to \( Head_i \).

Second verification and computation. Source member node \( A \) computes \( opinion_A \oplus \{ opinion_A \oplus opinion_B \} = opinion_A \oplus \{ MAC_A \oplus MAC_B \} = MAC_B \), and then calculates the MAC of destination member node that it received, \( MAC_B = h(opinion_B || K_A \oplus (K_B \oplus a_{AB})). \)

Destination node \( B \) computes \( opinion_B \oplus \{ opinion_A \oplus opinion_B \} = opinion_B \) and \( MAC_B \oplus \{ MAC_A \oplus MAC_B \} = MAC_B \), and then computes the MAC of source node that it received, \( MAC_A = h(opinion_A || K_B \oplus (K_A \oplus a_{AB})). \)

Source member node \( A \) verifies to confirm if \( MAC_B = MAC_B \), and destination member node \( B \) verifies to confirm if \( MAC_A = MAC_A \). If they are same,
then both member node $A$ and member node $B$ will perceive that the opinion was shared securely and without losing its integrity.

**Opinion updating part**

As soon as the degree or value of opinion would be changed, the existing opinion would be changed accordingly. In this way, a previously shared opinion could be updated with a new opinion received at the end of each opinion exchange cycle.

**Implementing in network simulator–2**

Computer simulation acts as a primary step to employ and study each aspect of the proposed technique before implementing it in a real-world application/system. The famous network simulator–2 (NS-2) is used in this paper to design the simulation for this scheme. As mentioned above, our scheme requires a cluster topology so we used CBRP (Cluster Based Routing Protocol) to form dynamic clusters and used its route acquisition mechanism to acquire routes between two nodes. OpenSSL, an open-source library, was used to compute MAC in NS-2. The route to any destination was discovered by source cluster head as soon as an opinion exchange packet (OEP) was received from any requesting member node. To ensure the effectiveness of the scheme, five different packets were designed and used to exchange an opinion. Figure 4 shows the format of these OEPs 1–5.

With reference to Figure 2 as well as Figure 3, OEPs 1, 3, 5 and OEPs 1, 2, 4 were exchanged within and outside the cluster head, respectively. As soon as the opinion sharing cycle is initiated by requesting a member node, the OEP-1 is forwarded to the relevant cluster head. The source cluster head is responsible to find out the subscription of destination member node. OEP-1 would be replayed to the destination node in case if both the communicating nodes belong to the same cluster head node otherwise the source cluster head would arrange the route and send OEP-2 toward the destination member node. In case both member nodes are subscribed to same cluster head, OEP-1 is sent back to cluster head, which then broadcast the final information in OEP-5 for both source and destination member node. In case both member nodes are subscribed to different cluster head when the OEP-2 reaches destination cluster head then OEP-3 is sent to the required destination member to prepare relevant information for opinion exchange. OEP-1 is sent back to destination cluster head from destination member node, which leads to cluster head authentication step by sending and receiving OEP-4 and OEP-1, respectively. At this time both source and destination cluster heads would have the final information to send OEP-5 to their relevant member node. Mentioned in section “The proposed Scheme,” each node has its own responsibility to compute information before sending or forwarding OEP.

Figure 5 shows the transition diagram for OEPs that were transmitted between the source and destination member node. After receiving OEP-5, both member

---

**Table 2.** Parameters used during the simulation.

| Parameters                  | Value                                      |
|-----------------------------|--------------------------------------------|
| Simulation time             | 1500 s                                     |
| Space                       | $1000 \times 1000$                         |
| Number of nodes             | 50                                         |
| Traffic type                | Constant opinion exchange packets          |
| Mobility                    | Moderate mobility                          |
| Traffic load                | 2, 4, 6, 8, 10 random connections          |
| Radio range                 | 250 m                                      |
| Node speed                  | 10 m/s                                     |
| MAC protocol                | 802.11                                     |
| Mobility model              | Random waypoint                            |
| Opinion exchange rate       | 10, 20, & 30 s                             |

MAC: message authentication code.
nodes compute the received information to make sure the integrity and confidentiality of opinion shared.

**Results and discussions**

The proposed scheme validated by three different performance metric, that is, overall throughput, delay and OEP packet overhead. A more detailed explanation of these matrixes is as follows:

- **Opinion exchange ratio**: It is the ratio of total opinion exchange cycles that were initiated to the total opinion exchange cycles that were successfully exchanged.
Average exchange delay: The average delay that all successful opinion exchange cycles faced is calculated in these criteria.

Opinion exchange packet overhead: Average number of OEPs that were used to exchange all opinions successfully is calculated in these criteria.

Except for opinion exchange packet overhead, other criteria were calculated while simulating traffic between member nodes that were subscribed to the same or different clusters. Opinion exchange packet overhead remains the same, that is, four packets while exchanging opinion between both member nodes in the same cluster but would vary if an opinion is exchanged between member nodes in different clusters. Table 2 shows the details of the parameter used and its given values.

Figure 6 describes about the opinion exchange ratio (within same cluster and different clusters) with respect to the opinion exchange ratio. Network traffic load was varied from 2 to 10 random connections to record the opinion exchange ratio. Within same cluster, the achievable opinion exchange ratio is higher for increased traffic load. The success rate of opinion exchange ratio remained excellent while both member nodes were in the same cluster as shown in Figure 6. On the contrary, while exchanging opinion between two nodes in different clusters, it changed to high, moderate, and low with respect to opinion exchange rate 10, 20, and 30 s, respectively.

Figure 7 shows the average delay for all successful opinion shared in both cases, during entire simulations. The slight and insignificant difference was noticed while both communicating nodes were subscribed to the same cluster and when each opinion exchange sending rate, respectively. Overall, the delay was increased as a number of communicating pairs increased.

While exchanging opinion between nodes that were subscribed to two different clusters, a varied amount of delay was observed. In this second case, an unpredictable number of intermediate clusters and member nodes are involved to pass back and forth these OEPs until successful completion of each cycle. Delay was measured diverse because of an unpredictable number of intermediate nodes. There was an increase in delay as the traffic was increased, and it was concluded that delay is directly proportional to the amount of packet exchange cycle.

Unlike opinion exchanged between nodes in different clusters, per cycle overhead is four packets in the first case where both communicating nodes were subscribed to the same cluster. Figure 7 shows only per cycle overhead in the second case, that is, opinion exchange between two member nodes in a different cluster. Overhead was low during fewer amounts of opinion exchange cycles, and it increased with the increased volume of opinion exchange cycle. This overhead includes the packet forwarding by an unpredictable number of intermediate nodes on the way from the source to the destination clusters.

Conclusion

The Opinion is the only input that a reputation-based selfishness mitigation solution uses to conclude the legitimacy of any given node. In an adverse MANET environment, this opinion is prone to modification and results in malfunctioning of the entire reputation solution. The proposed secure scheme to exchange this valuable opinion ensures confidentiality and integrity to opinions that were exchanged. Simulation results proved that the proposed scheme achieved higher success ratio, moderate delay, and affordable overhead. Opinion exchange cycles were more efficiently completed while both member nodes were in the same cluster but a noticeable decline in performance was observed when both member nodes were subscribed in
different clusters. The main reason for this decline is the unpredictable number of intermediate node and distance between two clusters. The proposed scheme can be further enhanced by considering and facilitating any solution that requires secure information exchange between two nodes in any peer-to-peer network, for example, virtual currency exchange in credit payment-based solutions, feedback exchange in game theory-based solutions, and so on against selfishness.

Declaration of conflicting interests
The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Funding
The author(s) disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: This research is supported in part by Natural Science Foundation of Jiangsu Province under contracts BK20161302 and State Grid Foundation of Jiangsu Corporation J2017123.

ORCID iD
Abdur Rashid Sangi https://orcid.org/0000-0002-7806-6996

References
1. Yoo Y and Agrawal DP. Why does it pay to be selfish in a MANET? IEEE Wirel Comm Mag 2006; 13(6): 87–97.
2. Ojetunde B, Shibata N and Gao J. Secure payment system utilizing MANET for disaster areas. IEEE Trans Syst Man Cybern 2017; 49: 2631–2663.
3. Aydos M, Vural Y and Tekerek A. Assessing risks and threats with layered approach to Internet of Things security. Meas Control 2019; 52: 338–353.
4. Marti S, Giali TJ, Lai K, et al. Mitigating routing misbehavior in mobile ad hoc networks. In: Proceedings of the 6th annual international conference on Mobile computing and networking, Boston, MA, 6–11 August 2000, pp. 255–265. New York: ACM.
5. Sadeghi M and Yahya S. Analysis of wormhole attack on MANETs using different MANET routing protocols. In: Fourth international conference on ubiquitous and future networks (ICUFN), Phuket, Thailand, 4–6 July 2012, pp. 301–305. New York: IEEE.
6. Mayol AR and Gozalvez J. Improving selfishness detection in reputation protocols for cooperative mobile ad-hoc networks. In: Proceedings of the 21st annual IEEE international symposium on personal, indoor and mobile radio communications, Instanbul, 26–30 September 2010.
7. Paul K and Westhoff D. Context aware detection of selfish nodes in DSR based ad-hoc networks. In: Proceedings of the global telecommunications conference, Taipei, Taiwan, 17–21 November, pp. 178–82. New York: IEEE.
8. Buchegger S and Boudic J-YI. Performance analysis of the CONFIDANT protocol (Cooperation of Nodes: fairness In Dynamic Ad-hoc NeTworks). In: Proceedings of the 3rd ACM international symposium on Mobile ad hoc networking & computing, Lausanne, Switzerland, 9–11 June 2002, pp. 226–36. New York: ACM.
9. Michiardi P and Molva R. Core: a collaborative reputation mechanism to enforce node cooperation in mobile ad hoc networks. In: Proceedings of 6th IFIP communication and multimedia security conference, Protoroz, Solvenia, 26–27 September 2002.
10. Yau P-W and Mitchell CJ. Reputation methods for routing security for mobile ad hoc networks. In: Symposium on trends in communications and multimedia security conference, Bratislava, 28 October 2003, pp. 130–37. New York: IEEE.
11. Miranda H and Rodrigues L. Friends and foes: preventing selfishness in open mobile ad hoc networks. In: Proceedings of 6th 23rd international conference on distributed computing systems workshops, Providence, RI, 19–22 May 2003, pp. 440–445. New York: IEEE.
12. Rath PS and Rao CM. An enhanced threshold based cryptography with secrete sharing and particle swarm optimization for data sending in MANET. In: Proceedings of 3rd Asia-Pacific conference on intelligent robot systems (ACIRS), Singapore, 21–23 July 2018, pp. 87–91. New York: IEEE.
13. Balakrishnan K, Deng J and Varshney PK. TWOACK: preventing selfishness in mobile ad hoc networks. In: Proceedings of IEEE wireless communications and networking conference, New Orleans, LA, 13–17 March 2005.
14. Adams WJ, Hadjiichristofi GC and Davis NJ IV. Calculating a node’s reputation in a mobile ad hoc network. In: Proceedings of IEEE international performance, computing, and communications conference, Phoenix, AZ, 7–9 April 2005, pp. 303–307. New York: IEEE.
15. Li S-YR, Yeung RW and Cai N. Linear network coding. IEEE Trans Inform Theory 2003; 49: 371–381.
16. Koetter R and Medard M. An algebraic approach to network coding. IEEE ACM Trans Netw 2003; 11: 782–795.
17. Senthil Kumar T and Prabakaran S. Security and privacy enforced wireless mobile communication using PI-MAKA protocol design. Meas Control 2019; 52: 788–793.
18. Ho T, Koetter R, Medard M, et al. A random linear network coding approach to multicast. IEEE Trans Inform Theory 2006; 52: 4413–4430.
19. Katti S, Gollakota S and Katabi D. Embracing wireless interference: analog network coding. In: Proceedings of the 2007 conference on applications, technologies, architectures, and protocols for computer communications, Kyoto, Japan, 27–31 August 2007.

20. Liu J and Du R. A key distribution scheme using network coding for mobile ad hoc network. In: Proceedings of the 5th international ICST conference on communications and networking in China (CHINACOM), Beijing, China, 25–27 August 2010.

21. Yu Z, Wei Y, Ramkumar B, et al. An efficient scheme for securing XOR network coding against pollution attacks. In: Proceedings of the 28th IEEE conference on computer communications (INFOCOM), Rio de Janeiro, Brazil, 19–25 April 2009.

22. Jaggi S, Langberg M, Katti S, et al. Resilient network coding in the presence of byzantine adversaries. In: Proceedings of the 26th IEEE international conference on computer communications, Barcelona, 6–12 May 2007.

23. Wu B, Chen J, Wu J, et al. A survey on attacks and countermeasures in mobile ad hoc networks. In: Xiao Y, Shen X and Du D-Z. (eds) Wireless/mobile network security. Boston, MA: Springer, 2007, pp. 103–135.

24. Oliveira PF, Costa RA and Barros J. Mobile secret key distribution with network coding. In: Proceedings of the international conference on security and cryptography (SECRYPT’07), Barcelona, 13–28 July 2007.

25. Network Simulator-2, 2011, www.isi.edu/nsnam/ns

26. Jiang M, Li J and Tay Y-C. Cluster Based Routing Protocol (CBRP), functional specification (Internet-draft). Mobile Ad-hoc Network (MANET) Working Group, IETF, Fremont, CA, 1998.

27. Rashid Sangi A, Alkatheiri MS, Anamalamudi S, et al. Cognitive AODV routing protocol with novel channel-route failure detection. Multimed Tools App. Epub ahead of print 22 February 2019. DOI: 10.1007/s11042-019-7352-7.

28. Anamalamudi S, Rashid Sangi A, Alkatheiri B, et al. AODV routing protocol for cognitive radio access based Internet of Things (IoT). Future Gener Comp Syst 2018; 83: 228–238.

29. Viega J, Messier M and Chandra P. Network security with OpenSSL. 1st ed. Cambridge, MA: O’ Reilly, 2002.

30. Rashid Sangi A, Liu J and Liu Z. Performance comparison of single and multi-path routing protocol in MANET with selfish behaviors. In: Proceedings of world academy of science, engineering and technology (WASET) 65, Tokyo, Japan, 28–33 May 2010.

31. Rashid Sangi A, Liu Z and Liu J. Route information poisoning in MANETs: analysis and defenses. In: Proceedings of fourth IITA conference, Qinhuangdao, China, 5–7 November 2010.