CRM: A New Dynamic Cross-Layer Reputation Computation Model in Wireless Networks

Hui Lin[1], Jia Hu[2*], Jianfeng Ma[3], Li Xu[1], Li Yang[3]

[1]Key Lab of Network Security and Cryptology, Fujian Normal University, Fuzhou, China

[2]Department of Mathematics and Computer Science, Liverpool Hope University, L16 9JD, UK

[3] Shanxi Key Laboratory of Network and System Security, Xidian University, Xi'an, Shaanxi, China

Abstract: Multi-hop wireless networks (MWNs) have been widely accepted as an indispensable component of next-generation communication systems due to their broad applications and easy deployment without relying on any infrastructure. Although showing huge benefits, MWNs face many security problems, especially the internal multi-layer security threats being one of the most challenging issues. Since most security mechanisms require the cooperation of nodes, characterizing and learning actions of neighboring nodes and the evolution of these actions over time is vital to construct an efficient and robust solution for security-sensitive applications such as social networking, mobile banking, and teleconferencing. In this paper, we propose a new dynamic cross-layer reputation computation model named CRM to dynamically characterize and quantify actions of nodes. CRM couples uncertainty based conventional layered reputation computation model with cross-layer design and multi-level security technology to identify malicious nodes and preserve security against internal multi-layer threats. Simulation results and performance analyses demonstrate that CRM can provide rapid and accurate malicious node identification and management, and implement the security preservation against the internal multi-layer and bad mouthing attacks more effectively and efficiently than existing models.

Keywords: multi-hop wireless networks, network security, cross-layer design, reputation computation model

1. INTRODUCTION

Multi-hop wireless networks (MWNs), such as mobile ad hoc networks (MANETs), wireless sensor networks (WSNs), and wireless mesh networks (WMNs), are vulnerable to different security risks, especially the risks arising from the internal multi-layer attacks [1-3] due to inherent features prone to attacks such as multi-hop decentralized architecture, wireless medium, etc. All of these constrains make security protection in MWNs more complicated compared to traditional networks.

Security protection in MWNs is closely related to trust. In MWNs, trust can help characterize and learn the nodes' actions and the evolution of these actions over time, which facilitates secure cooperation and is vital to construct an efficient and robust solution for security-sensitive applications such as social networking, mobile banking, and teleconferencing. As a key scheme for managing trust, the reputation computation model (RCM) has been introduced as an effective approach to characterize and quantify nodes' behaviors for MWNs. Although a number of RCMs and reputation-based trust functions for MWNs have been proposed in the literature, all existing RCMs were based on the direct observation of layer-specifics to evaluate the node reputation, thus ignoring many key factors of reputation in other layers [4-12] such as node forwarding behaviors at the network layer, channel collisions at the MAC layer, channel quality measures at the physical layer and node access behaviors at the application layer. Moreover, they did not take into account the bad mouthing attack where attackers provide dishonest recommendations to frame up good parties and/or boost trust values of malicious peers. For example, Ben-Othman et al. [8] presented a new mechanism named Hybrid Wireless Mesh Protocol (HWMP) Watchdog at the network layer, which is a reputation model that combines the benefits of HWMP and Watchdog techniques to detect and exclude malicious nodes during the path-selection process and to protect against internal attacks. However, it only focused on the flooding and modification attacks. Luo et al. [9] proposed RFSTrust, a trust model based on fuzzy recommendation similarity to quantify and evaluate the trustworthiness of nodes at the network layer, and it only focused on the selfish node attack. Therefore, cross-layer security mechanisms need to be implemented and enforced for MWNs to resist the multi-layer and bad mouthing attacks.

With this goal in mind, in this paper, we propose a cross-layer dynamic reputation computation model named CRM for the MWNs. To the best of our knowledge, the proposed model is the first dynamic RCM considering the cross-layer design [13-14] and multi-level security technology [15] to identify and manage internal malicious nodes. The major contributions of this paper are as follows.

1) To resist the multi-layer attacks and make the malicious nodes detection more effectively and accurately, cross-layer design is introduced into node reputation computation, which incorporates network-layer node forwarding behavior observations, MAC-layer channel collision detections, physical-layer channel quality measures and application-layer node access behavior observations.

2) To further enhance the reliability and validity of the proposed model and defend against the bad mouthing attack, the node role security class relevancy and role security level relevancy (see Section 4.1.2 for details) are introduced into the recommendation reputations evaluation, which makes the recommendation reputations more reliable and credible, and the computation and update of the recommender's credibility more effectively in the presence of dishonest or unreliable referrals.

3) The node role security class and security level classification based on the reputation value make the punishment and management of malicious nodes more flexible, which improves the fault-tolerant ability and the survival ability of the proposed model in MWNs.

4) Extensive OPNET simulation experiments are conducted to validate the performance of the CRM. Simulation results show that the performance of the CRM in terms of false positive rate, packet delivery ratio and reputation update speed, are better than those of the existing SLCRM and FSLR models when multi-layer and bad mouthing attacks are present.

The remainder of this paper is organized as follows. In Section 2, some important related work is reviewed. Section 3 describes the network and adversary models. In Section 4, the proposed cross-layer dynamic RCM is presented. We verify the effectiveness of our model through extensive simulations in Section 5. Finally, the paper is concluded in Section 6.

2. RELATED WORK

RCMs have been widely studied in various fields of distributed networks to support secure and trustworthy communications and collaborations among participants [4-12].

Ben-Othman et al. [8] presented a new mechanism called HWMP-Watchdog, which is a reputation model at the network layer that combines the benefits of HWMP and Watchdog techniques to detect and exclude malicious nodes during the path-selection process and protect against internal attacks in the HWMP Routing Protocol. Luo et al. [9] proposed RFSTrust, a trust model based on fuzzy recommendation similarity to quantify and evaluate the trustworthiness of nodes at the network layer. By using fuzzy logic theory, RFSTrust provides a natural framework to deal with uncertainty and tolerance of imprecise data inputs for the subjective tasks of trust evaluation, packet forwarding review and credibility adjustment. Li et al. [4] presented a hierarchical account-aided reputation management system (ARM) at the network layer to efficiently and effectively provide cooperation incentives. The ARM built a hierarchical locality-aware dynamic hash table infrastructure for efficient and integrated operations of both reputation and price systems. Laniepce et al. [10] proposed a cross-layer reputation system which runs on the AP (Access Point) and makes use of the TCP control mechanism to evaluate the node cooperation. The system is based on the transport layer observations and evaluates node misbehaviors by deducting from the TCP control decisions. Liu et al. [11] proposed a RCM at the network layer to help nodes to recognize selfish nodes much earlier and decrease the convergence time for isolating selfish nodes by combining familiarity values with subjective opinions. The familiarity value represents a node's familiar degree with another individual node and is used to calculate the weighting factor that determines how much the node recommendation opinion impacts on the reputation computation result.

All existing RCMs are based on the direct observation of layer-specifics to evaluate the node reputation, thus ignoring many key factors of reputation in other layers such as node forwarding behaviors at the network layer, channel collisions at the MAC layer, channel quality measures at the physical layer and node access behaviors at the application layer. Also, none of them takes into account the malicious behavior of the recommendation nodes within the recommendation reputation evaluation. Therefore, the design of a cross-layer and efficient reputation computation model for MWNs to resist multi-layer and bad mouthing attacks is still an open issue, and it is also a highly challenging task for two reasons. Firstly we need to accurately capture the behavior of each layer and obtain related parameters and information from different layers, which demands careful observations of protocols at each layer. Secondly, we need to integrate all the collected information and input them to the reputation evaluation process for a precise evaluation, which needs a thorough understanding of attacking behaviors and patterns at each layer.

3. NETWORK AND ADVERSARY MODEL

3.1 Network Model

In this paper, we consider multi-hop 802.11s WMNs composed of mesh routers (or mesh nodes) and mesh clients. Mesh nodes establish a backbone to relay data from/to mesh clients. The wireless backbone is connected to the Internet through mesh portal which is a mesh router with gateway functionalities. Mesh clients connect directly to the routers or indirectly associate with mesh routers through client networks such as wireless ad hoc networks, sensor networks, cellular networks, etc. We suppose that two nodes are one-hop neighbors if they stay within the transmission range of each other and all nodes are connected through bidirectional wireless links. It is assumed that there exist a link layer protocol to manage all radios and channels through measurement procedures and a dynamic channel assignment algorithm.

3.2 Adversary Model

Multi-hop WMNs are vulnerable to both external and internal attacks [12, 16] and attacks may occur at different layers. In this paper, we consider the internal multi-layer attacks. The internal attacks are launched by an inside attacker who is a mesh node (probably as a forwarder) included in a mesh connection. Therefore, it knows its previous hop, next hop and the type of packets going through it. It can also exploit all the information encountered on the compromised node(s). For the multi-layer attacks, we consider the jamming attack at the physical layer, selfish MAC attack at the MAC layer, blackhole/grayhole attack at the network layer and malicious resource access attack at the application layer. Moreover, the bad mouthing attack is taken into account.

4. CROSS-LAYER DYNAMIC REPUTATION COMPUTATION MODEL

In this section, a novel dynamic cross-layer reputation computation model named CRM is extended from our previous work [17-18]. CRM couples uncertainty based reputation computation models [11] [19-21] with the cross-layer design [13][14] and multi-level security technology [15].

In CRM, the cross-layer design is introduced into the node reputation evaluation to resist the multi-layer attacks and make the malicious nodes detection more effectively and precisely, which incorporates the network layer node forwarding behavior observations, the MAC layer channel collision detections, the physical layer channel quality measures and the application layer node access behavior observations. Meanwhile, cross-layer interactions are modeled by a simple architecture that assumes there is a common database accessible by all layers within the protocol stack. These layers are configured to store and update the required parameters in this database, which are retrieved while needed by the reputation evaluation. Furthermore, by leveraging the multi-level security technology to classify malicious nodes and to decide whether to punish or isolate them, the management of malicious nodes becomes more flexible and the fault-tolerant ability of CRM is improved.

To further enhance the reliability and validity of the dynamic reputation computation model, the proposed CRM model also adopts a unique combination of node role level relevancy and node security level relevancy to evaluate the reliability and the credibility of the recommendation reputations that can further defend against the bad mouthing attack. The main notations and symbols used in the paper are summarized in Table 1.

Table 1. Main notations and symbols

$\omega_{x:y}$, $\omega_{x:y}^{dir}$, $\omega_{x:y}^{rec}$, $\omega_{x:y}^{final}$	*x's* reputation, direct, recommendation and final reputation towards *y*
$\omega_{t_0,x:y}^{App-dir}$, $\omega_{t_0,x:y}^{Net-dir}$, $\omega_{t_0,x:y}^{MAC-dir}$, $\omega_{t_0,x:y}^{Phy-dir}$	reputation evaluation results at the physical layer, MAC layer, network layer, and application layer at time t_0
I_s , I_t	the number of total interactions and successful interactions between nodes
P_f , P_a	the node successful forwarding probability and secure access probability
N_s , N_t	the number of packets been successfully forwarded and the total number of packets that need to be forwarded
P_{col}^{MAC} , P_{loss}^{Phy}	the packet collision probability and the packet loss probability at the MAC and physical layer respectively
$b_{x:y}$, $d_{x:y}$, $u_{x:y}$, $a_{x:y}$	x's belief, disbelief, and uncertainty towards y, x's willingness to believe y
θ_1 , θ_2	the *sc* and *sl* relevancy between *x* and *y/k*
$TH_{sl,sc}^1$, $TH_{sl,sc}^2$	the up and down thresholds of the *sl* (or *sc*)

In CRM, each node keeps a reputation value towards other nodes as a prediction of their future behaviors and a 4-tuple $\omega_{x:y} = (b_{x:y}, d_{x:y}, u_{x:y}, a_{x:y})$ is used to represent a node reputation. In the 4-tuple, $b_{x:y}$, $d_{x:y}$, and $u_{x:y}$ denote *x*'s belief, disbelief, and uncertainty towards *y* , respectively. The base rate $a_{x:y}$ denotes *x*'s willingness to believe *y*, which determines how uncertainty is viewed as belief when the reputation is used. They satisfy the following conditions:

$$\begin{cases} b_{x:y} + d_{x:y} + u_{x:y} = 1.0 \\ b_{x:y},\ d_{x:y},\ u_{x:y},\ a_{x:y} \in [0.0, 1.0] \end{cases} \tag{1}$$

When a reputation is used in a decision, it is projected onto the belief/disbelief axis through its expectation $E\left(\omega_{x:y}\right)$ that is given as follows:

$$E\left(\omega_{x:y}\right) = b_{x:y} + a_{x:y} u_{x:y} \tag{2}$$

The CRM consists of two phases: 1) Reputation Computation; 2) Malicious Node Classification and Management.

4.1 Reputation Computation

In this section, we present the process of the reputation computation. Suppose *x* and *y* are two neighboring nodes, the final reputation of *x* towards *y* at initial time t_0, $\omega_{t_0,x:y}^{final}$, includes two components. One is the direct reputation $\omega_{t_0,x:y}^{dir}$ and the other is the recommendation reputation $\omega_{t_0,x:y}^{rec}$.

4.1.1 Direct Reputation Computation

In CRM, direct reputation computation operates independently at each node that stores its opinion towards the others' reputation in the local reputation database. The proposed direct reputation computation model depends on the information from the physical, MAC, network, and application layers. For neighbor nodes *x* and *y*, *x's* direct reputation towards *y* at time $t_{0,}$ $\omega^{dir}_{t_0,x:y}$, can be given by

$$\begin{cases} \omega^{dir}_{t_0,x:y} = \alpha_1 * \omega^{App-dir}_{t_0,x:y} + \alpha_2 * \omega^{Net-dir}_{t_0,x:y} + \alpha_3 * \omega^{MAC-dir}_{t_0,x:y} + \alpha_4 * \omega^{Phy-dir}_{t_0,x:y} \\ \alpha_1 + \alpha_2 + \alpha_3 + \alpha_4 = 1 \end{cases} \tag{3}$$

where the first part considers node security level according to its resource access behavior at the application layer; the second part considers node forwarding reliability according to node forwarding behavior at the network layer; the third part considers the link quality or the collision probability at the MAC layer; and the last part considers the channel quality at the physical layer. $\alpha_i (i = 1..4)$ is the weight factor, which determines how much the direct reputation evaluation result of the specific layer affects the final direct reputation. $b^{dir}_{t_0,x:y}$, $d^{dir}_{t_0,x:y}$, $u^{dir}_{t_0,x:y}$ and $a^{dir}_{t_0,x:y}$ in $\omega^{dir}_{t_0,x:y}$ can be computed as

$$\begin{cases} u^{cr-dir}_{t_0,x:y} = \alpha_1 * u^{App-dir}_{t_0,x:y} + \alpha_2 * u^{Net-dir}_{t_0,x:y} + \alpha_3 * u^{MAC-dir}_{t_0,x:y} + \alpha_4 * u^{Phy-dir}_{t_0,x:y} \\ b^{cr-dir}_{t_0,x:y} = P_a * P_f * (1 - P^{Phy}_{loss}) * (1 - P^{MAC}_{col}) * (1 - u^{cr-dir}_{t_0,x:y}) \\ d^{cr-dir}_{t_0,x:y} = \left[1 - P_a * P_f * (1 - P^{Phy}_{loss}) * (1 - P^{MAC}_{col})\right] * (1 - u^{cr-dir}_{t_0,x:y}) \\ a^{cr-dir}_{t_0,x:y} = 0.5 \end{cases} \tag{4}$$

where $u^{App-dir}_{t_0,x:y}$, $u^{Net-dir}_{t_0,x:y}$, $u^{MAC-dir}_{t_0,x:y}$, and $u^{Phy-dir}_{t_0,x:y}$ denote *x's* uncertainty on *y* at the application, network, MAC and physical layers respectively. They can be calculated as

$$\begin{cases} u^{App-dir}_{t_0,x:y} = P_a \\ u^{Net-dir}_{t_0,x:y} = 1 - {}^{I_s}\!/\!_{I_t} \\ u^{Mac-dir}_{t_0,x:y} = P^{MAC}_{col} \\ u^{Phy-dir}_{t_0,x:y} = P^{Phy}_{loss} \end{cases} \tag{5}$$

where I_s is the number of successful interactions between nodes and I_t is the total number of interactions between nodes.

P_f is the node successful forwarding probability and P_a is the node secure access probability. They are given by

$$\begin{cases} P_f = {}^{N_s}\!/\!_{N_t} \\ P_a = {}^{\sum_{i=\varepsilon}^{n} v_i}\!/\!_{\sum_{i=1}^{n} v_i} \end{cases} \tag{6}$$

where v_i is the number of times that the node accessing behavior is confirmed as the security level *i*. N_s is the number of packets that have been successfully forwarded by the node. N_t is the total number of packets that need to be forwarded.

P^{MAC}_{col} is the packet collision probability at the MAC layer and can be obtained based on its relationship to the link busyness ratio $r_{link-bussy}$ as [22]

$$\begin{cases} r_{link-bussy} = 1 - \dfrac{f(P_{col}^{MAC}) * \lambda_{idle}}{f(P_{col}^{MAC}) * \lambda_{idle} + g(P_{col}^{MAC}) * \lambda_{suc} + h(P_{col}^{MAC}) * \lambda_{col}} \\ f(P_{col}^{MAC}) = (1 - P_{col}^{MAC})^{n/_{n-1}} \\ g(P_{col}^{MAC}) = n * (1 - P_{col}^{MAC})[1 - (1 - P_{col}^{MAC})^{1/_{n-1}}] \\ h(P_{col}^{MAC}) = 1 + (n-1)(1 - P_{col}^{MAC})^{n/_{n-1}} - n * (1 - P_{col}^{MAC}) \end{cases} \tag{7}$$

where λ_{idle} , λ_{suc} , and λ_{col} denote the length of the idle slot, the duration of a successful transmission, and the duration of a collision, respectively, which can be determined from the 802.11-based models [22-23].

P_{loss}^{Phy} is the packet loss probability depending on the wireless channel quality at the physical layer. We estimate P_{loss}^{Phy} by modeling the underlying time varying wireless channel as a Gilbert-Elliott two-state Markov error model [13, 22]. As shown in Fig. 1, the model has two states 0 (good state) and 1 (bad state) and the parameters *p* and *q* denote the transition probability from states 0 ((good state) to 1 (bad state) and vice versa.

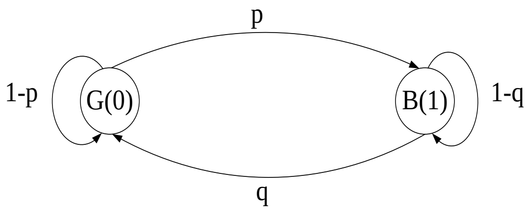

Fig. 1. Gilbert-Elliott Two-State Markov Error Model

P_{loss}^{Phy} can be expressed as

$$\begin{cases} P_{loss}^{Phy} = P_{l-g} * \dfrac{q}{(p+q)} + P_{l-b} * \dfrac{p}{(p+q)} \\ p = \sum_{i=1}^{n-1} m_i \Big/ m_0 \\ q = \sum_{i=1}^{n-1} m_i \Big/ \sum_{i=1}^{n-1} m_i * i \end{cases} \tag{8}$$

where m_0 is the number of delivered packets and m_i is the number of lost bursts with the length *i*. P_{l-g} and P_{l-b} are the loss probability in good and bad states, respectively. The details on how to estimate the parameters P_{l-g} and P_{l-b} can be found in [22-24].

4.1.2 Recommendation Reputation Computation

When there is not enough historical interaction data for *x* to evaluate the direct reputation towards *y* or the direct reputation is not enough for *x* to make a decision on *y*, *x* will start a recommendation reputation query by broadcasting a reputation query message to the neighbors to ask for their reputation opinions on *y*.

Whenever an *x's* neighbor receives the Query message, it will check its local reputation table to see whether there is a direct reputation on *y* with the uncertainty value less than 1.0. If there is one, the node will send a Reply message to *x* which contains its id, 2-tuple (*sl*, *sc*), the valid time period and its direct reputation on *y*, otherwise it simply ignores the query, where *sl* represents the security level of *x*'s role and *sc* represents the security class of *x*'s role. Security level stands for the level of security of a node, *e.g.*, high, medium, low; and security class represents a finer granularity of security at each level, *e.g.*, there may be multiple *sc* at the same *sl*, and with the same *sc*, different nodes may have various *sl*.

Let R represent the set of recommenders ($\left|R\right| = n, n > 1$). After receiving the replies, *x* will execute the recommendation reputation evaluation phase as follows.

(1) If *n*=2 and the two recommendation reputations from *y* and *k* are conflict, *x* will evaluate the reliability of two recommenders considering the *sl* and the *sc* of the recommender as (9), and then select the recommendation opinion from the more trustworthy one.

$$\begin{cases} \xi_{x:y/k} = \beta_1 * \theta_1 + \beta_2 * \theta_2 \\ \beta_1 + \beta_2 = 1 \end{cases} \tag{9}$$

where θ_1 is the *sc* relevancy between *x* and *y/k* and θ_2 is the *sl* relevancy between *x* and *y/k*, they can be computed as (10) and (11), respectively. The β_1, β_2 are the weight factors of the *sc* and *sl* respectively, which determine how much the *sc* and *sl* affect the final reliability of recommenders.

$$\theta_1 = 1 - \frac{\left|sc_{y/k} - sc_x\right|}{N_{sc}}, sc \in [1, N_{sc}] \tag{10}$$

$$\theta_2 = \begin{cases} \dfrac{sl_{y/k} - sl_x}{N_{sl}}, \ sl_{y/k} \geq sl_x \\ 1 - \dfrac{\left|sl_{y/k} - sl_x\right|}{N_{sl}}, \ sl_{y/k} < sl_x \end{cases}, sl \in [1, N_{sl}] \tag{11}$$

Considering that *y* and *k* have been defined over the same context and use the same policy vector, we say *y* is more trustworthy than *k*, if any of the following conditions holds.

$$\begin{cases} \xi_{x:y} > \xi_{x:k} \\ \xi_{x:y} = \xi_{x:k} \wedge b_y > b_k \\ \xi_{x:y} = \xi_{x:k} \wedge b_y = b_k \wedge d_y < d_k \\ \xi_{x:y} = \xi_{x:k} \wedge b_y = b_k \wedge E(\omega_y) > E(\omega_k) \end{cases} \tag{12}$$

(2) If n>2, for each recommender $i \in R$, let $R'(|R'| = \tau)$ be the new set of the recommenders, R' is defined as (13). Furthermore, we allocate an appropriate weight f_i given by Eq. (14) to each recommendation reputation and calculate $\omega_{t_0,x:y}^{rec}$ by Eq. (15).

$$R' = \left\{ i \middle| \xi_{x:i} \geq TH_{\xi} \right\} \tag{13}$$

$$f_i = \xi_{x:i} * E(\omega_{t_0,x:i}) \Big/ \sum_{k \in R'} \xi_{x:k} * E(\omega_{t_0,x:k}) \tag{14}$$

$$\begin{cases} b_{t_0,x:y}^{\mathrm{rec}} = \sum_{k=1,k\in R'}^{\tau} f_k \cdot b_{t_0,k:y}^{\mathrm{dir}} \Big/ \tau \\ d_{t_0,x:y}^{\mathrm{rec}} = \sum_{k=1,k\in R'}^{\tau} f_k \cdot d_{t_0,k:y}^{\mathrm{dir}} \Big/ \tau \\ u_{t_0,x:y}^{\mathrm{rec}} = \sum_{k=1,k\in R'}^{n} f_k \cdot u_{t_0,k:y}^{\mathrm{dir}} \Big/ \tau \\ a_{t_0,x:y}^{\mathrm{rec}} = \sum_{k=1,k\in R'}^{\tau} f_k \cdot a_{t_0,k:y}^{\mathrm{dir}} \Big/ \tau \end{cases} \tag{15}$$

4.1.3 Dynamic Final Reputation Computation

After getting the direct reputation and the recommendation reputation, the static final reputation at time *t0* $\omega_{t_0,x:y}^{final} = (b_{t_0,x:y}^{final}, d_{t_0,x:y}^{final}, u_{t_0,x:y}^{final}, a_{t_0,x:y}^{final})$ can be calculated as [11]:

$$\begin{cases} b_{t_0,x:y}^{final} = \left(b_{t_0,x:y}^{dir} \cdot u_{t_0,x:y}^{rec} + b_{t_0,x:y}^{rec} \cdot u_{t_0,x:y}^{dir} \right) \Big/ \left(u_{t_0,x:y}^{dir} + u_{t_0,x:y}^{rec} - u_{t_0,x:y}^{dir} \cdot u_{t_0,x:y}^{rec} \right) \\ d_{t_0,x:y}^{final} = \left(d_{t_0,x:y}^{dir} \cdot u_{t_0,x:y}^{rec} + d_{t_0,x:y}^{rec} \cdot u_{t_0,x:y}^{dir} \right) \Big/ \left(u_{t_0,x:y}^{dir} + u_{t_0,x:y}^{rec} - u_{t_0,x:y}^{dir} \cdot u_{t_0,x:y}^{rec} \right) \\ u_{t_0,x:y}^{final} = \left(u_{t_0,x:y}^{dir} \cdot u_{t_0,x:y}^{rec} \right) \Big/ \left(u_{t_0,x:y}^{dir} + u_{t_0,x:y}^{rec} - u_{t_0,x:y}^{dir} \cdot u_{t_0,x:y}^{rec} \right) \\ a_{t_0,x:y}^{final} = \left(a_{t_0,x:y}^{dir} \cdot u_{t_0,x:y}^{rec} + a_{t_0,x:y}^{rec} \cdot u_{t_0,x:y}^{dir} \right) \Big/ \left(u_{t_0,x:y}^{dir} + u_{t_0,x:y}^{rec} - u_{t_0,x:y}^{dir} \cdot u_{t_0,x:y}^{rec} \right) \end{cases} \tag{16}$$

However, as the belief, disbelief, uncertainty and base rate change over time, the trust relationship changes over time too. Thus, the trust relationship at present depends not only on the values of the underlying parameters but also on the decayed values of the previous trust. The time-dependent value of a trust relationship from time t_i, computed at the present time t_n, is given by:

$$\begin{cases} b_n = b_i \times e^{-((b_i)^{-1} \Delta t)^{2k}} \\ d_n = d_i \times e^{-((d_i)^{-1} \Delta t)^{2k}} \\ u_n = 1 - b_n - d_n \\ a_n = a_i \times e^{-((a_i)^{-1} \Delta t)^{2k}} \\ \Delta t = t_n - t_i \end{cases} \tag{17}$$

where k (k >= 1) is the decay rate.

The dynamic final reputation considering the trust decay at the time t_n can be given by

$$\varpi_{fin,x:y}^{dynamic} = \eta_1 \times (b_i, d_i, u_i, a_i) + \eta_2 \times (b_n, d_n, u_n, a_n) \tag{18}$$

where η_1 and η_2 are the weight factors ($\eta_1 + \eta_2 = 1, (\eta_1, \eta_2 \in [0,1])$) used to determine how much the reputation evaluation results at time t_i and t_n affect the dynamic final reputation.

4.2 Malicious Node Identification and Management

In this subsection, a novel malicious node identification and management scheme is proposed, which makes the management of the malicious nodes more flexible and improves the fault-tolerant ability and the survival ability of the CRM in multi-hop wireless networks [25, 26].

In CRM, each node is assigned a *sl* (and *sc*) according to its reputation value. Each *sl* (and *sc*) level has its thresholds $TH_{sl,sc}^1$ and $TH_{sl,sc}^2$ ($TH_{sl,sc}^1 < TH_{sl,sc}^2$, $sl\,(sc) = 1 \ldots N_{sl}\,(N_{sc})$). After getting the final reputation, *x* will compare it with the thresholds of each *sl* and *sc*, and then make the decision whether to punish or isolate the malicious node.

For example, suppose there are four nodes *p*, *q*, *r*, *u* and their final reputations are: $\varpi_{x:p}^{fin1} \in \left[TH_{2,3}^1, TH_{2,3}^2\right]$, $\varpi_{x:q}^{fin1} \in \left[TH_{2,4}^1, TH_{2,4}^2\right]$, $\varpi_{x:r}^{fin2} \in \left[TH_{4,3}^1, TH_{4,3}^2\right]$ and $\varpi_{x:u}^{fin3} \in \left[TH_{5,3}^1, TH_{5,3}^2\right]$. Since the *sl*'s values of *p* and *q* are both 2, which meet the predefined security level requirement, nodes *p* and *q* will be perceived as cooperative nodes and the service request will be taken into account. However, *y* and *k* will be considered as malicious nodes because that their security levels are lower than the predefined security level requirement. Furthermore, because *u*'s *sl* is 5, lower than the lowest security level requirement 4, it will be isolated, while *r* will be punished with the security level 4. For *p* and *q*, suppose the *sc* of *x* is 1, since the *sc* distance between *x* and *p* is 2, less than the distance between *x* and *q*, *p* will be selected as the cooperative node at last.

Many methods can be used to punish the malicious nodes, *e.g.*, (1) reduce the corresponding reputation value of the malicious nodes by leveraging the punishment factors. (2) forbid the malicious nodes from participating in the network activities for a period of time. The detailed process of the punishment is out of scope of this paper.

5. SIMULATION RESULTS AND ANALYSIS

In this section, we present simulation results on OPNET [27-28] to demonstrate the performance of the CRM. The simulated network consists of 100 wireless mesh nodes located in a rectangular space of the size 1000m x 1000m. The physical layer uses a fixed range transmission model, where two nodes can directly communicate with each other only if they are within one hop. The MAC layer protocol is the IEEE 802.11 Distributed Coordination Function (DCF) [29]. Hybrid Wireless Mesh Protocol (HWMP) is used as the underlying routing protocol. Traffic sources are constant bit-rate (CBR) and each source sends data packets of the size 1024 bytes. The other parameters used in the CRM are as follows. α_1, α_2, α_3, and α_4 are 0.2, 0.3, 0.3, and 0.2 respectively, β_1 and β_2 are 0.5 each, and η_1 and η_2 are 0.4 and 0.6 respectively. These settings make a specific scenario where there are more weights on the network and MAC layer compared to the application and physical layer, more weights on the present than the history, and the weightings of the *sc* and *sl* are the same. Simulations were performed for a duration of 100 seconds, and all simulations were repeated 50 times to obtain reliable results.

In the simulation, we consider a more practical scenario achieved by simulating multi-layer attacks, which represent the real-world attacks better than single-layer attacks, because attacks are likely to be launched at different layers in practice. Specifically, we consider the jamming attack at the physical layer, selfish MAC attack at the MAC layer, blackhole/grayhole attack at the network layer and malicious resource access attack at the application layer, respectively. In the jamming attack, attackers send a lot of data on the physical channel in a short time interval to keep the channel busy and prevent other nodes from transmitting data. In the selfish MAC attack, attackers manipulate the parameters and the rules of the MAC layer to degrade the performance of other hosts and increase their own shares of the common transmission resource. In the grayhole/blackhole attack, attackers refuse to forward certain packets and simply drop them. Specially, if attackers drop all the packets, the attack is then called a black hole attack. In the malicious resource access attack, attackers access the resource without authorization.

The computation cost of our model is $O(n)$, same as the compared models in the paper. Owing to the linear run time of our model and the strong computing power of modern devices, the network delay caused by the computation is little and is thus negligible considering the great benefits of enhanced security with the computation.

In this section, we compare the proposed CRM against the FSLR model in [11] and the SLCRM model in [18], respectively. The performance is evaluated using the following metrics:

- *False Positive Rate (FPR)*: the ratio of the number of false reports on malicious nodes to the total number of reports on malicious nodes.

- *Packet Delivery Rate (PDR):* the ratio of the number of data packets received at the destinations to the total number of data packets generated by the CBR sources.
- *Reputation Update Speed (RUS):* the rising or dropping speed of the node reputation under different attacking scenarios.

5. 1 False Positive Rate (FPR)

In the first experiment, the evaluated performance metric is the FPR. In these models, false positive means that a normal node is classified as a misbehaving one.

Fig. 2 illustrates that the FPR increases in all the three reputation computation models, when the percentage of malicious nodes working at the network layer increases, given that the link/channel quality is good and the recommendations are honest. As shown in the figure, when only attacks in a single layer are considered (*e.g.* the network layer), the FPRs of the three models are close since they all can detect the network layer attacks (*e.g.* backhole/grayhole attacks). But the update speed of reputation values of CRM and SLCRM is faster than that of the FSLR, because the CRM and SLCRM models are dynamic while the FSLR is static. Thus, the FPR of the FSLR is slightly higher than those of the CRM and SLCRM.

Figs. 3 and 4 show that the FPRs of all the three models increase by 10%-15% compared to the results in Fig. 2 under the scenarios where the link/channel quality is bad and the recommendations are honest, and the link/channel quality is good and the recommendations are dishonest, respectively. In these scenarios, the bad mouthing attack and the multi-layer attack (*e.g.* the network, MAC and physical layer attacks) are taken into account, which result in the link/channel loss and dishonest recommendations. Also, we can see that the FPRs of the three models are different and the CRM has the lowest FPR. For the FSLR, because it cannot detect the multi-layer attacks launched in other layers in addition to the network layer and the dishonest recommendations, its FPR is the highest. The FPR of the CRM is lower than that of the SLCRM because the CRM considers both the multi-layer attacks and the dishonest recommendations while the SLCRM can only detect the attacks launched at the network and MAC layers. The attacks launched at the physical layer cause channel interrupts and packet losses, and dishonest recommendation nodes always give fake recommendations, either bad mouthing or false praise towards other nodes. Both of the above-mentioned constrains pose significant impacts on the detection of malicious nodes, which cause the SLCRM to produce more false positives than the CRM.

In Fig. 5, we build the experimental environment with the bad link/channel quality and dishonest recommendations. Similar to the results in Figs. 3 and 4, the FPRs of all three models increase when the percentage of the malicious recommendation nodes rises. Since both the bad mouthing and multi-layer attacks are present, the increase in this experiment is about 10% more than the results in Figs. 3 and 4, and the fastest and the slowest growing models are the FSLR and the CRM, respectively. Because of the effective detection and defending mechanism against both bad mouthing and multi-layer attacks proposed in the CRM, its superiority is more obvious than that in Figs. 3 and 4.

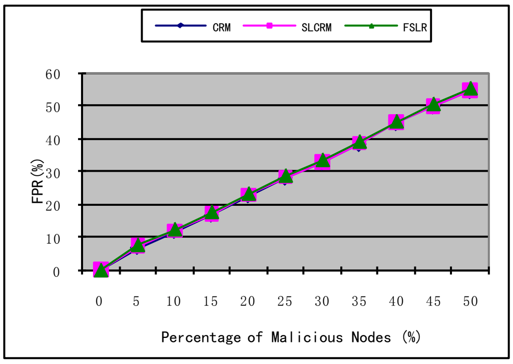

Fig. 2. Comparison of average FPR with a good link/channel quality and honest recommendations

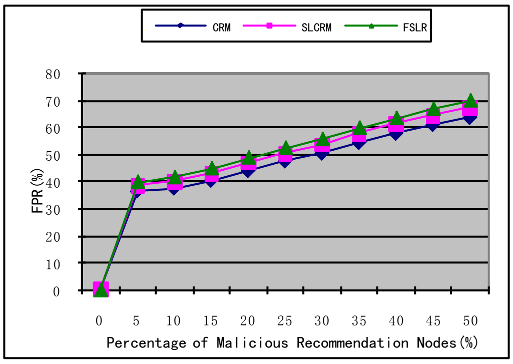

Fig. 4. Comparison of average FPR with bad mouthing attacks

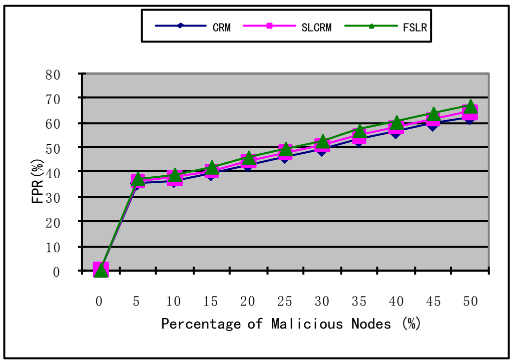

Fig. 3. Comparison of average FPR with multi-layer attacks

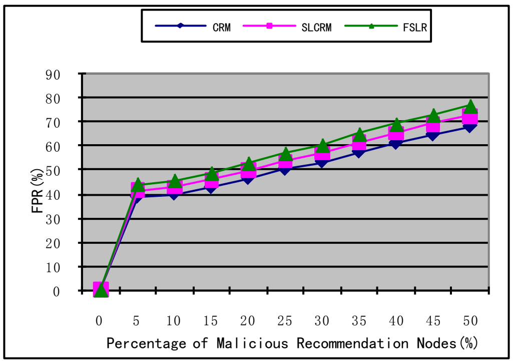

Fig. 5. Comparison of average FPR with multi-layer attacks and bad mouthing attacks

5.2 Packet Delivery Rate (PDR)

In this subsection, we compare the PDR of the CRM to those of the SLCRM and FSLR. First, the scenario is set with attacks launched at the network layer, a good link/channel quality, and honest recommendations. As shown in Fig. 6, the average PDR will be significantly degraded by malicious attackers without taking reputation computation model into account. We can also observe that by using the three reputation computation models to detect and management the malicious nodes, the average PDR can be substantially improved. As the time increases, the average PDR without reputation decreases accordingly. While for all the three models, the average PDR decreases in the initial 60 seconds, and then starts to restore by using the reputation computation model to detect and identify the less trustworthy nodes effectively and avoid them during the node selection process. Moreover, since there are only attacks at the network layer, the link/channel quality is good and recommendations are honest, the average PDR of the three models are close.

Second, we consider the scenario with the honest recommendations and bad link/channel quality. In this scenario, the link/channel loss caused by the multi-layer attacks and the normal loss are considered. From the results in Fig. 7, we can see that: (1) the average PDR of all the three models decreases when the percentage of the bad link/channel increases. (2) For FSLR, since it cannot detect the multi-layer attacks launched at other layers in addition to the network layer, it cannot distinguish the reasons of packet losses, which will result in more normal nodes being classified as misbehaving nodes and then be isolated from the network. Consequently, the average PDR of the FSLR falls at the highest speed. (3) The average PDR of the CRM drops slower than that of the SLCRM. The SLCRM can only detect the attacks launched at the network and MAC layers but ignoring the channel interrupts and packet losses caused by the attacks launched at the physical layer, which can cause a significant impact on the detection of malicious nodes and make the average PDR of the SLCRM fall faster than that of the CRM.

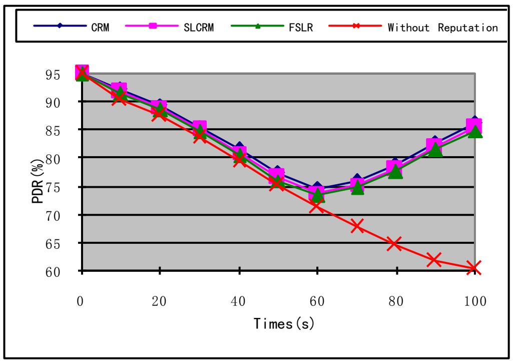

Fig. 6. Comparison of average PDR with attacks launched at the network layer

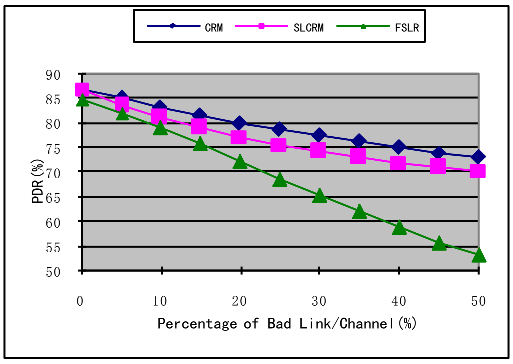

Fig. 7. Comparison of average PDR with multi-layer attacks

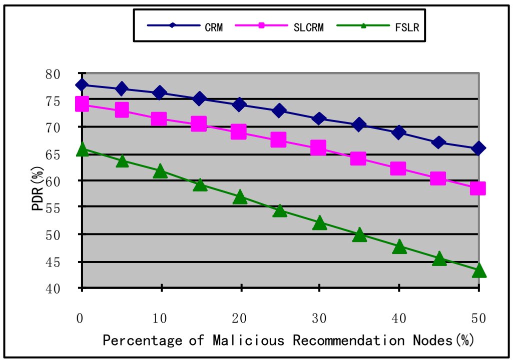

Fig. 8. Comparison of average PDR with multi-layer and bad mouthing attacks

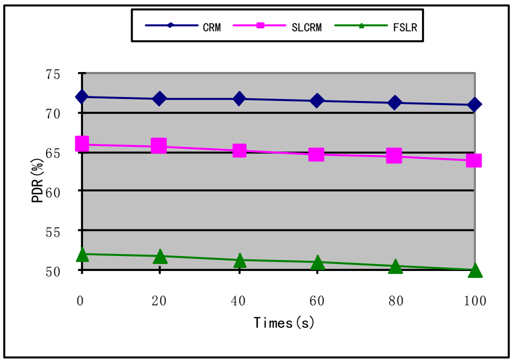

Fig. 9. Comparison of average PDR with multi-layer and bad mouthing attacks (with different damage degrees)

Third, we consider the scenario with multi-layer attacks and bad mouthing attacks present. At the beginning, the percentage of the dishonest recommendations is from 0% to 50%. Then, we assume that the percentage of the dishonest recommendations is 30% and divide the severity degree of the dishonest recommendations into three levels, $m1$, $m2$ and $m3$ ($m1>m2>m3$). The results are shown in Figs. 8 and 9, respectively.

In Fig. 8, similar to the results in Fig. 7, the average PDR of all the three models decreases when the percentage of the malicious recommendation nodes increases. Since both the bad mouthing and multi-layer attacks are present and cannot be detected effectively, the PDRs of SLCRM and FSLR decrease at percentages up to 16% and 23%, respectively. However, for the CRM, because of its effective detection and defending mechanism against the channel interrupts and packet losses caused by multi-layer attacks and the fake recommendations or false praises, its PDR only decreases by 10%, which is lower than that of the SLCRM and FSLR.

In Fig. 9, we compare the fault tolerance and flexibility of the three models. The average PDR of the CRM is 71-72% when the percentage of the dishonest recommendations increases from 30% to 35%, which is increased by 1% (in Fig. 8, it is 70% -72%). However, the average PDRs of the SLCRM and FSLR in Fig. 9 are 64-66% and 50-52%, respectively, being the same as the results in Fig. 8. The reasons of the difference between CRM, SLCRM and FSLR lie in: (1) both the SLCRM and FSLR cannot detect the dishonest recommendations attacks. (2) In the CRM, the dishonest recommendation behavior is punished according to the degree of damage it has caused, which implements the fine granularity malicious nodes identification and management. Therefore, in the CRM, only those malicious nodes causing severe damages will be eliminated from the network immediately, which can avoid network congestion and flow interruptions caused by a large number of nodes being ejected from the network, and thus improve the average PDR effectively.

5. 3 Reputation Update Speed (RUS)

In this subsection, we first compare the RUS of the CRM to those of the SLCRM and FSLR in a hostile network environment where attacks such as bad mouthing and multi-layer are present. Figs. 10 and 11 show the reputation increase speed and decrease speed, respectively.

Fig. 10 shows the performance of the reputation increase speed of the CRM, SLCRM and FSLR. For CRM, the reputation value rises to 0.9 at 70 seconds and increases to 0.95 at 90 seconds. In contrast, FSLR takes 55 and 80 seconds, and SLCRM takes 60 and 85 seconds to achieve the reputation values of 0.9 and 0.95, respectively.

In a hostile network environment, the multi-layer and bad mouthing attacks are assumed to be present. The multi-layer attacks enable the attacks to be launched at the network, MAC, physical and application layers. While the bad mouthing attack enables the dishonest recommendation nodes to increase collusion nodes' reputation rapidly by giving fake recommendations and false praises. Since the FSLR and SLCRM cannot detect the bad mouthing attack and the attacks launched at the physical layer effectively, their RUS is higher than that of the CRM. Moreover, because CRM can detect the fake recommendations and false praises, only the recommended nodes with a higher $\xi_{x:y}$ value have a possibility to be accepted, the RUS of the CRM is lower than that of the FSLR and SLCRM.

In this subsection, the situation of decreasing reputation is also considered. Fig. 11 shows that reputation decreases when the bad mouthing and multi-layer attacks are present. The value of reputation computed under the CRM decreases more rapidly than those under the SLCRM and FSLR. In the FSLR, it cannot effectively and accurately identify the malicious nodes that launch the bad mouthing and multi-layer attacks. As a result, the reputation of the malicious node decreases the slowest. In the SLCRM, it cannot detect the bad mouthing and the attacks launched at the physical layer, thus the reputation of the malicious node decreases slower than that of the CRM.

Consequently, we can conclude that the proposed reputation computation model can adapt the status of nodes, especially malicious nodes, to the environment more rapidly, effectively and accurately.

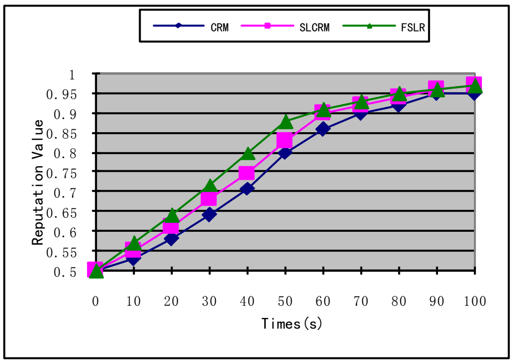

Fig. 10. Comparison of reputation increase speed

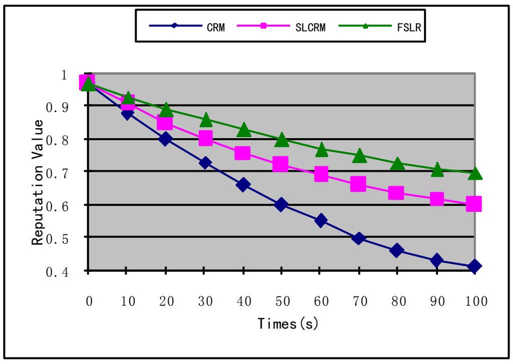

Fig. 11. Comparison of reputation decrease speed

6. CONCLUSIONS AND FUTURE WORK

In this paper, we have investigated the problems of the internal multi-layer and bad mouthing attacks in MWNs and have proposed a new dynamic cross-layer reputation computation model named CRM.

Based on the combination of the uncertainty based reputation computation model, cross-layer design, multi-level security technology and recommendation reputation evaluation, CRM can effectively defend against the internal multi-layer and bad mouthing attacks. Elaborate theoretical analyses have demonstrated that the CRM is secure and efficient. Furthermore, extensive simulation results have verified that the false positive rate, packet delivery ratio and reputation update speed of the proposed CRM are better than those of the SLCRM and FSLR models. The CRM could be applied to enhance the security and trust of various networks and systems such as social networks, distributed systems, and peer-to-peer networks.

In future work, we plan to introduce game theory into the CRM to make the reputation evaluation more accurately and effectively. In addition, we intend to extend this CRM model to incorporate the encryption and signature based privacy preserving technology into the evaluation and transmission process of reputation.

Acknowledgement

The authors would like to thank anonymous reviewers and editors for their constructive comments. This work is supported by: 1. Changjiang Scholars and Innovative Research Team in University (IRT1078), 2. the Key Program of NSFC-Guangdong Union Foundation (U1135002), 3. National Natural Science Foundation of China (61202390), 4. Fujian Natural Science Foundation：2013J01222, 5. the open research fund of Key Lab of Broadband Wireless Communication and Sensor Network Technology (Nanjing University of Posts and Telecommunications, Ministry of Education).

References

[1] Khan, S., Loo, K. K. and Din, Z. D. (2009) Cross Layer Design for Routing and Security in Multi-hop Wireless Networks. *Journal of Information Assurance and Security*, **4**, 170-173.

[2] Kamhoua, C. A., Pissinou, N. and Makki, K. (2011) Game Theoretic Modeling and Evolution of Trust in Autonomous Multi-Hop Networks. *Proceedings of ICC 2011*, Kyoto, Japan, IEEE, USA.

[3] Sun, L. and Wang, W. (2013) Understanding Blackholes in Large-Scale Cognitive Radio Networks under Generic Failures. *Proceedings of INFOCOM 2013*, Turin, Italy, 728-736, IEEE, USA.

[4] Li, Z. and Shen, H. (2011) A Hierarchical Account-aided Reputation Management System for Large-Scale MANETs. *Proceedings of INFOCOM 2011*, Shanghai, China, 909 – 917, IEEE, USA.

[5] Sicari, S., Coen-Porisini, A. and Riggio, R. (2013) DARE: Evaluating Data Accuracy Using Node Reputation. *Computer*

Networks, **57**, 3098-3111.

[6] Shen, H., Lin, Y. and Li, Z. (2013) Refining Reputation to Truly Select High-QoS Servers in Peer-to-Peer Networks. *IEEE Transactions on Parallel and Distributed Systems*, **24**, 2439-2450.

[7] Moati, N., Otrok, H. and Mourad, A. (2014) Reputation-Based Cooperative Detection Model of Selfish Nodes in Cluster-Based QoS-OLSR Protocol. *Wireless Personal Communications*, **75**, 1747-1768.

[8] Othman, J. B., Claude, J. P. and Benitez, Y. I. S. (2012) A Novel Mechanism To Secure Internal Attacks in HWMP Routing Protocol. *Proceedings of ICC 2012*, Ottawa, Canada, 162-166, IEEE, USA.

[9] Luo, J., Liu, X. and Fan, M. (2009) A Trust Model Based on Fuzzy Recommendation for Mobile Ad-Hoc Networks. *Computer Networks*, **53**, 2396-2407.

[10] Laniepce, S., Lancieri, L., Achemlal, M. and Bouabdallah, A. (2010) A Cross-Layer Reputation System for Routing Non-Cooperation Effects Mitigation Within Hybrid Ad-Hoc Networks. *Proceedings of IWCMC 2010*, Caen, France, 296-300, IEEE. USA.

[11] Liu, Y., Li, K., Jin, Y., Zhang, Y. and Qu, W. (2011) A Novel Reputation Computation Model Based on Subjective Logic for Mobile Ad Hoc Networks. *Future Generation Computer Systems*, **27**, 547-554.

[12] Long, X. and Joshi, J. (2010) BaRMS: A Bayesian Reputation Management Approach for P2P Systems. *Proceedings of the IRI 2010*, Las Vegas, Nevada, USA, 147-152, IEEE, USA.

[13] Salleh, N. M., Muhammad, M., Zakaria, M. S., Gannapathy, V. R., Suaidi, M. K., Ibrahim, I. M., AbdulAziz, M. Z. A., Johar, M. S. and Ahmad, M. R. (2007) Wireless Mesh Networks: Cross Layer Design Challenge. *Proceedings of the ASIA-PACIFIC Conference on Applied Electromagnetics 2007*, Melaka, Malaysia, IEEE, USA.

[14] Akyildiz, I. F. and Wang, X. (2008) Cross-Layer Design in Wireless Mesh Networks. *IEEE Transactions on Vehicular Technology*, **57**, 1061-1076.

[15] Lu, W. and Sundareshan, M. K. (1990) A Model for Multilevel Security in Computer Networks. *IEEE Transactions on Software Engineering*, **16**, 647–659.

[16] Khan, S., Loo, K. K., Mast, N. and Naeem, T. (2010) SRPM: Secure Routing Protocol for IEEE 802.11 Infrastructure Based Wireless Mesh Networks. *Journal of Network and Systems Management*, **18**, 190-209.

[17] Lin, H., Ma, J., Hu, J. and Yang, K. (2012) PA-SHWMP: A Privacy-Aware Secure Hybrid Wireless Mesh Protocol for IEEE 802.11s Wireless Mesh Networks. *EURASIP Journal on Wireless Communications and Networking*, **69**, 1-16.

[18] Lin, H., Ma, J. and Hu, J. (2012) SLCRM: Subject Logic Based Cross layer Reputation Mechanism for Wireless Mesh Networks. *China Communications*, **19**, 40-49.

[19] Yu, H., Shen, Z., Miao, C., Leung, C. and Niyato, D. (2010) A Survey of Trust and Reputation Management Systems in Wireless Communications. *Proceedings of the IEEE*, **98**, 1755-1772.

[20] Li, F. and Wu, J. (2010) Uncertainty Modeling and Reduction in MANETs. *IEEE Transactions on Mobile Computing*, **9**, 1035-1048.

[21] Noack, A. (2011) Trust Agreement in Wireless Mesh Networks. *Proceedings of the WISTP 2011*, Heraklion, Crete, Greece, 336–350, Springer-Verlag, Berlin.

[22] Shila, D. M., Cheng, Y. and Anjali, T. (2010) Mitigating Selective Forwarding Attacks with a Channel-Aware Approach in WMNs. *IEEE Transactions on Wireless Communications*, **9**, 1661-1675.

[23] Zhai, H., Chen, X. and Fang, Y. (2005) How well can the IEEE 802.11 wireless LAN support quality of service?. *IEEE*

Transactions on Wireless Communications, **4**, 3084-3094.

[24] Gandikota, V. R., Tamma, B. R. and Murthy, C. (2008) Adaptive FEC-Based Packet Loss Resilience Scheme for Supporting Voice Communication over Ad Hoc Wireless Networks. *IEEE Transactions on Mobile Computing*, **7**, 1184-1199.

[25] Qureshi, B., Min, G. and Kouvatsos, D. (2012) A Distributed Reputation and Trust Management Scheme for Mobile Peer-to-Peer Networks. *Computer Communications*, **35**, 608-618.

[26] Qureshi, B., Min, G. and Kouvatsos, D. (2013) Countering the Collusion Attack with A Multidimensional Decentralized Trust and Reputation Model in Disconnected MANETs. *Multimedia Tools and Applications*, **66**, 303-323.

[27] Sethi, A. S. and Hnatyshin, V. Y. (2012) The Practical OPNET User Guide for Computer Network Simulation. CRC Press, UK.

[28] Johnson, D. B. and Maltz, D. A. (1996) Dynamic Source Routing in Ad Hoc Wireless Networks. *Springer Mobile Computing*, **353**, 153-181.

[29] 802.11-2007 (2007) IEEE Standard for Information Technology - Telecommunications and Information Exchange Between Systems - Local and Metropolitan Area Networks - Specific Requirements - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications. IEEE, USA.