Resilient Misbehaviour Detection MAC Protocol (MD-MAC) for Distributed Wireless Networks

Chaminda Alocious
School of Computer Science
University of Hertfordshire
Hatfield, United Kingdom
Email: c.alocious@herts.ac.uk

Hannan Xiao
School of Computer Science
University of Hertfordshire
Hatfield, United Kingdom
Email: h.xiao@herts.ac.uk

Bruce Christianson
School of Computer Science
University of Hertfordshire,
United Kingdom
Email: b.christianson@herts.ac.uk

***Abstract*—Wireless network security requirements are becoming more important and critical. The modern network security architectures require more attention to provide security in each network layer. This will require understanding of protocol vulnerabilities in existing protocol architectures. However, providing security requirements are not just limited to confidentiality and integrity, also availability and fairness are important security elements. IEEE 802.11 MAC protocol is one of the most common standard in modern day networks and has been designed without a consideration for providing security protection at MAC layer. IEEE 802.11 assumes all the nodes in the network are cooperative. However, nodes may purposefully misbehave in order to obtain extra bandwidth, conserve resources and disrupt network performance. This research proposes a Misbehaviour Detection MAC protocol (MD-MAC) to address the problematic scenarios of MAC layer misbehaviours, which takes a novel approach to detect misbehaviours in Mobile Adhoc Networks (MANETs). The MD-MAC modifies the CSMA/CA protocol message exchange and uses verifiable backoff value generation mechanism with an incorporated trust model which is suitable for distributed networks. The MD-MAC protocol has been implemented and evaluated in ns2, simulation results suggest that the protocol is able to detect misbehaving wireless nodes in a distributed network environment.**

***Index Terms*—Wireless Network Security, IEEE 802.11, Medium Access Control, Misbehaviours**

I. Introduction

There are many media access protocols for channel-access control in wireless networks, for example the CSMA/CA protocol for collision avoidance in wireless networks. IEEE 802.11 protocol is a most commonly used MAC protocol in current wireless networks, based on CSMA/CA access control mechanism. IEEE 802.11 MAC protocol assumes that all the nodes in the wireless network adhere to the protocol, and fully cooperate with the protocol. However, there are selfish/malicious mobile stations which do not follow the IEEE 802.11 protocol rules when sharing wireless channel. Mobile nodes in such distributed network have the motivation to be selfish and conserve their resources. Additionally, due to the vast enhancement of network device(Mobile Stations) adapters programmability, changing MAC layer protocol parameters has become easier.

This paper proposes a misbehaviour detection MAC protocol (MD-MAC) with the ability of detecting a range of MAC layer misbehaviours. Firstly, MD-MAC introduces a novel mechanism to verify CSMA/CA MAC protocol parameters (backoff/SIFS/DIFS/NAV) used in channel sharing mechanism, and prevent selfish or malicious nodes from misusing them. Such an assurance, primarily helps to build the trustworthiness between nodes working together for a better fair channel sharing operation. Secondly, this research uses wireless nodes that are in the transmission range of both sender and receiver (common neighbours) for the monitoring and trust management purpose, and eventually for the detection and diagnosis of MAC layer misbehaviours. MD-MAC could be considered as a mechanism that builds a general security platform in MAC layer for detecting and preventing misbehaviour nodes. In addition, the MD-MAC protocol does not require wireless nodes to trust each other, therefore novel trust model is suitable for distributed network architectures.

The rest of the paper is organized as below. Section II explains the background for MAC layer misbehaviours and operation of IEEE 802.11 protocol Distributed Coordination Function (DCF). Section III demonstrates the proposed MD-MAC protocol in details and section IV presents the simulation topology and the protocol implementation details. Section V presents the simulations and results analysis. Finally, section VI concludes the paper and discusses future work.

II. IEEE 802.11 based Wireless Networks MAC Layer Misbehaviours

Wireless networks are vulnerable to a vast range of MAC layer misbehaviours due to its shared channel nature, dynamic topology changes, lack of centralized authority and non-cooperativeness of the network nodes. Such features has allowed MAC layer misbehaviours to be deployed successfully on the devices running this standard protocol. In this section the paper discuses IEEE 802.11 DCF related misbehaviours in wireless networks and proposed detection/prevention mechanisms in the literature.

A. IEEE 802.11 CSMA/CA with DCF

Most of the modern MAC protocols uses Distributed Coordination Function (DCF) as the media access control mechanism. The CSMA/CA based DCF mechanism defines as, all nodes in the wireless network share a common medium, each node must wait for a randomly selected backoff value before start transmitting the data packet. DCF uses the BEB mechanism to assign backoff values to each wireless station in the network, aiming to allow each station to get a fair share of the wireless channel. Before a node transmit data, firstly it senses the channel status. If the channel is busy it waits for distributed inter frame space (DIFS) time, then the node enters the Contention Window (CW) time scale where node calculates the random backoff value within the range of (0, CW_{max}-1).

Next, if the medium becomes idle after additional DIFS time, the node starts to decrement backoff counter until the channel becomes busy or counter reaches zero. If the channel becomes busy before the counter becomes zero, then the node freezes timer. This process continues until backoff counter reaches zero. Then the node starts to send the first control packet Request to Send (RTS), the receiver then responds after a small inter frame space (SIFS) with a Clear to Send (CTS) packet. After another SIFS time the sender transmits the DATA packet. Finally, the receiver acknowledges the data by sending an ACK packet. Occasionally, two nodes can reach zero in the same time, in which case collision will happen and the node has to recalculate the backoff values in the range of [0...2 * CW_min] [1] [2].

B. IEEE 802.11 MAC Protocol Misbehaviours

MAC Layer misbehaviours can be categorized as selfish, malicious and inter layer. The selfish misbehaviours are targeted to achieve unfair advantages of the network services over the other legitimate wireless nodes. Such misbehaviours mainly consist with backoff value, Differ Timers manipulations (SIFS/DIFS/EIFS), CW cheating with altering the BEB algorithm, CTS/RTS packet scrambling, Intentional RTS/CTS packet drop, Network Allocation Vector (NAV) attacks and cross flow attacks. Malicious misbehaviour attacks targeted to disrupt the network services for legitimate nodes, which could experience severe communication delays and higher collision rates, also might not be able access the services at all. This could also involve draining the battery life of good nodes, as a result of continues retransmitting attempts. Malicious misbehaviours can be divided into link layer jamming, CTS/RTS Time-out attacks and DOS attacks. The CTS/RTS Time-outs attack based on manipulating SIFS differ timer value which could affect surrounding neighbours to wait longer by setting their NAV to a longer value. The neighbour nodes sets NAV value each time its hear an RTS/CTS/DATA/ACK packet to differ the channel busy duration in CSMA/CA mechanism. MAC layer malicious nodes could be utilize to initialize inter-layer attacks, especially in the routing layer (cross flow link breakage), where misbehaviour node using CW window cheating to create routing layer link breakage by using different CW values for varies route discovery process in protocol such as DSR and AODV.

There has been many research to analyse MAC layer misbehaviours in Wireless Networks. The research in [3] has conducted an analysis of MAC layer DoS attacks in MANETs. The authors have analysed both sender and receiver misbehaviours and concludes that misbehaving nodes can obtain a larger throughput which significantly affect the network performance, even capable of completely freeze the network at higher rate attacks. In [4] they have analysed and simulated the RTS/CTS DoS attack variants in 802.11 networks, which is one type of low rate DoS attack that capable to exploit the medium reservation mechanism of IEEE 802.11 through duration field. Also, in [5] [6] [7] has conducted an evaluation for greedy receiver misbehaviour in IEEE 802.11 Hotspots, they have identified a range of greedy receiver misbehaviours, and quantify their damage using both simulation and testbed experiments.

In literature, most of researches have focussed to solve MAC layer misbehaviour problem by modifying the existing IEEE 802.11 protocol's CSMA/CA channel sharing mechanism. One of the advantages of such detection and prevention system is detection related components could be deployed with the standard protocol. Malicious nodes could generate smaller or non-random backoff values which benefits to access the channel more frequently. The research in [8] has addressed the issue of generating smaller or non-random backoff in a centralized network environment (WLAN). Their detection and prevention method have proposed a modification to the standard IEEE 802.11 MAC protocol's DCF by allowing the receiver to generate and assigns back-off value to the sender. However, this approach only capable to detect sender misbehaviours in a trusted Access Point (AP) environment. The research done by [5] demonstrates the weakness of such detection mechanisms with the existence of malicious/misbehaving AP and discusses potential prevention strategies.

The research work in Rodosavac et al. in [9] has proposed a detection mechanism to overcome the weakness of previous proposed method in [8]. This method apply misbehaviour detection to more distributed networks and topologies. They have utilized Ensuring Randomness Algorithm (ERA), cryptographic functions to ensure the randomness of the values agreed through a public discussion between sender and receiver. The research work in [10] has presented a predictable random backoff algorithm to mitigate the effect of the smart MAC layer misbehaviours.

III. Proposed Misbehaviour Detection MAC (MD-MAC) Protocol

The proposed MD-MAC could potentially applicable for detection of MAC layer misbehaviours for higher availability and fairness in distributed networks. In distributed wireless network, nodes themselves have to monitor and control, rather by any other centralized entities. The MD-MAC protocol has diverted from standard protocol to use the common neighbours (CN) to provide more flexibility in monitoring misbehaviours. Common neighbour nodes are in the transmission range of both sender and receiver; such nodes can monitor, report, control packet communication between sender and receiver for a given data transmission at MAC layer.

The MD-MAC protocol has modified the CSMA/CA message exchange to detect misbehaving nodes which violate backoff or differ timers (SIFS, DIFS, EIFS, NAV). In MD-MAC protocol the sender generates a backoff value based on a **verifiable deterministic hash function** with CSMA/CA RTS control packet modification. In this verification procedure, the neighbour nodes and receiver have the ability to verify the sender generated backoff value used in CSMA/CA message exchange. This verification and monitoring procedure includes modification of the existing IEEE 802.11 protocol control packets, such as RTS, CTS, DATA, and ACK. These control packets have been modified to add more header fields. These additional fields provide the ability to involve common neighbours, and allow communication to be more transparent. The Fig. 1 demonstrates the MD-MAC protocol message exchange with the relevant additional header fields.

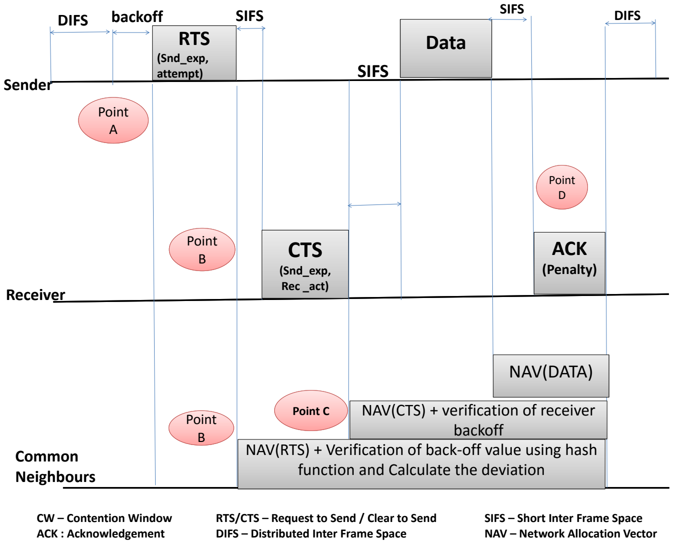
DIFS
backoff
RTS
(Snd_exp, attempt)
SIFS
Data
SIFS
SIFS
DIFS
Sender
Point A
Point D
Point B
CTS
(Snd_exp, Rec _act)
ACK
(Penalty)
Receiver
NAV(DATA)
Point C
NAV(CTS) + verification of receiver backoff
Point B
Common Neighbours
NAV(RTS) + Verification of back-off value using hash function and Calculate the deviation
CW – Contention Window
ACK : Acknowledgement
RTS/CTS – Request to Send / Clear to Send
DIFS – Distributed Inter Frame Space
SIFS – Short Inter Frame Space
NAV – Network Allocation Vector

Fig. 1. MD-MAC protocol with CSMA/CA in DCF Mode

A. MD-MAC Message Exchange with CSMA/CA Protocol

The MD-MAC protocol modifies the CSMA/CA message exchange to accommodate the required elements to detect and prevent MAC layer misbehaviours. The backoff value generated by the sender is based on the properties of the RTS packet bits header field ($pktRTS.bits$) and following steps demonstrate the message exchange sequence in MD-MAC with CSMA/CA.

1) In the Fig. 1 at point A, the sender generates the expected backoff value (Snd_{Exp}) from a deterministic backoff value generating function (f). The function f is known by receivers and neighbour nodes, which will be explained in the next section. A node to transmit data, sends the RTS control packet with incorporated (Snd_{Exp}) value and the attempt number to the receiver.

   $Sender \rightarrow Receiver$: $RTS(Snd_{Exp}, Attempt)$

2) The receiver receives the RTS packet at point B, then the receiver extract the expected backoff value (Snd_{Exp}) and also monitors the actual waiting backoff time (Rec_{Act}) using a monitoring function. The receiver also verifies the expected backoff value by calculating the backoff value (Rec_{Cal}) using RTS bits header field content and attempt number.

3) If the sender has not modified the RTS packet, then Rec_{Cal} and Snd_{Exp} are equal. Then the receiver attaches Snd_{Exp} and Rec_{Act} to the CTS packet header and sends to the sender.

   $Receiver \rightarrow Sender$: $CTS(Snd_{Exp}, Rec_{Act})$

4) Meanwhile at point B, the sender's neighbours listen to the RTS packet, they will extract the (Snd_{Exp}) value from the RTS and monitors actual waiting backoff time (Nbr_{Act}) of the sender using a monitoring function. The neighbour also verifies the accuracy of sender expected backoff (Snd_{Exp}) value by calculating the backoff value (Nbr_{Cal}) using a same deterministic function as the sender.

5) In Fig.1 at point C time, the common neighbour hears the CTS packet from the receiver and extract the Rec_{Act} and Snd_{Exp} values based on receiver's observations.

6) After successful RTS/CTS exchange the sender sends the DATA packet to the receiver

   $Sender \rightarrow Receiver$: $DATA()$

7) Finally, at point D the receiver calculates the penalty value for the sender and append it to acknowledgement packet header. The penalty value is generated in a conservative manner by consulting common neighbours trust values for a given sender. This penalty scheme will be discussed with misbehaviour prevention mechanism in the future work.

   $Reciever \rightarrow Sender$: $ACK(penalty)$

B. Verifiable Deterministic Backoff Value Generation

A verifiable hash function is used to generate backoff values, which could verify backoff values from the receiver and neighbour nodes. This deterministic hash function based backoff value generation mechanism could prevent nodes from fabricating a smaller backoff value, detect the nodes who are not doubling the CW value after a collision and also to detect sender-receiver collusion. In our research, we are using **Cyclic Redundancy Code** (CRC) of RTS control packets, bits field as a common data segment. This CRC code will be used to generate a hash value, then eventually generates the deterministic backoff value. Equation (1) shows the CRC value of the RTS packet, where $pktRTS.bits()$ is the RTS packet bit field.

$$crcRTS = CRCFunction(pktRTS.bits()) \qquad (1)$$

The CRC value helps to verify the backoff value and also to minimize the overhead of the hash function in (2). After, the CRC value generated, the equation in (2) shows how to calculate the hash value. In this case if the hash value is h, hash function uses the output of a deterministic function ”f” which is shown in equation (3). This deterministic function takes the CRC value, node id and attempt number to generate a deterministic value. The CW_{min} is the node's minimum contention window value.

$$h = Hash(f(crcRTS, nodeid, attempt)) \tag{2}$$

Where,

$$f = (ax + c) * modCW_{min} \tag{3}$$

And a =5; c = 2 * attempt+1; x = (CW * crcRTS + nodeid) mod (CW_{min} + 1)

Finally, the hash function output in (2) utilize to compute the final backoff value in equation (4).

$$backoff = h * mod2^{attempt-1} * CW_{min} \tag{4}$$

C. Trust Management with Common Neighbours/Receivers

Trust management is important in distributed networks as there is no centralized authority for network security management. Therefore, our research proposes a novel trust management mechanism which utilizes the recorded communication data in each neighbourhood. The equations (5) (6) demonstrate the trust value (Ttv) calculations by the neighbour node. Firstly, the neighbours calculates the misbehaviour factor (Mf) based on (5).

In equation (5) the Mf is the ratio between the average deviation (receiver reported deviation $(Rec_{Act} - Snd_{Exp})$ and neighbour's observed deviation $(Nbr_{Act} - Snd_{Exp}))$ backoff slots to the sender's expected backoff value (Snd_{Exp}). This will give a conservative value for calculating the final trust value which starts with 100 % maximum value for all the nodes, then consequently varies based on their behaviours in every communication. Finally, the trust table is updated by the common neighbour after every successful RTS/CTS communication. The Table I shows the four statues which a node could exist in the network based on their claimed trust value. According to each node status, the prevention policies could be applied to discourage the misbehaviours.

TABLE I
TRUST VALUE AND STATUES

$TrustValueRange$	$NodeStatus$
100 <= trust value => 80	NORMAL
79 <= trust value => 60	MISBEHAVING
59 <= trust value => 40	MISBEHAVED
39 <= trust value => 0	CRITICAL

$$Mf = \frac{((Rec_{Act} - Snd_{Exp}) + (Nbr_{Act} - Snd_{Exp}))}{2 * Snd_{Exp}} \tag{5}$$

$$Tv(\%) = Tv - Tv * Mf \tag{6}$$

IV. SIMULATION TOPOLOGY AND PROTOCOL IMPLEMENTATION

The simulation topology has been designed to simulate MD-MAC in the MANET environment with minimum required entities of the network, topology and CBR traffic configuration demonstrated in Table II. The simulated network topology consists of a sender, receiver and two CN. The Fig. 2 shows the network topology and implementation of MD-MAC entities on each node. The receiver gets CBR traffic from two sources (node 1 and node 3) at a bit rate of 2 kb/s and two CN hearing the communications between sender and receiver.

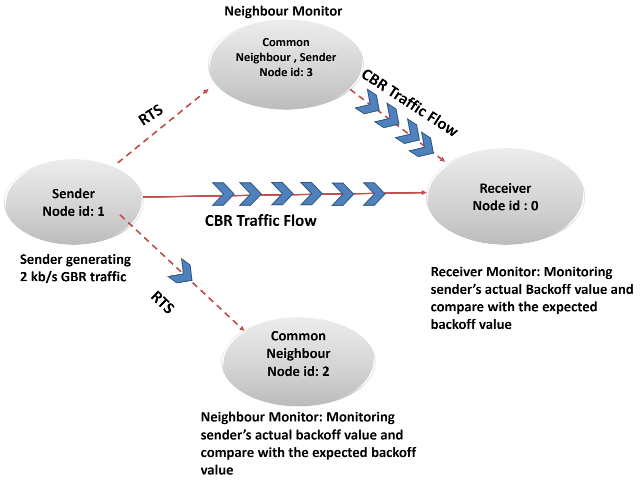
Neighbour Monitor
Common Neighbour , Sender Node id: 3
CBR Traffic Flow
RTS
Sender Node id: 1
CBR Traffic Flow
Receiver Node id : 0
Sender generating 2 kb/s GBR traffic
RTS
Receiver Monitor: Monitoring sender's actual Backoff value and compare with the expected backoff value
Common Neighbour Node id: 2
Neighbour Monitor: Monitoring sender's actual backoff value and compare with the expected backoff value

Fig. 2. MANET Network Topology in ns2

TABLE II
SIMULATION CONFIGURATION

Simulation Configuration	
Network model	ADHOC
Simulation area	1500x750 mxm
Routing protocol	DSR
Simulation time	200 s
Total nodes/Misbehaving Nodes	4 / 1
Max moving speed	10 m/s
Average moving speed	3.82 m/s
CBR Traffic Configuration	
Traffic type	CBR
Packet size	512 bytes
Packet interval	0.25 S
Max no of packets	100000

V. SIMULATIONS RESULT ANALYSIS

The MD-MAC protocol, misbehaviours nodes have been simulated in ns2 2.35 simulator. The main focus is to investigate whether the MD-MAC protocol has the ability to detect sender's backoff value manipulation. In this stage of the research, we are not enforcing the misbehaviour nodes with a penalty scheme. Therefore, MD-MAC only focuses on monitoring and detecting misbehaviour nodes.

A. Trust Model Analysis

The following communication tables are extracted from the simulation results, to demonstrate the nature of the monitoring functionality in each distributed wireless network node. There are two communication tables, the **neighbour's communication table** stores the RTS and CTS communications sent by the senders and receivers, specifically communication id (Id), sender id (Snd_{id}), receiver id (Rec_{id}), neighbour expected value (Nbr_{Exp}), neighbour observed (Nbr_{Act}) and receiver observed value (Rec_{Act}). The **receiver's communication table** consists of sender id (Snd_{id}), receiver expected backoff (Rec_{Exp}), receiver monitored actual backoff (Rec_{Act}), allowed backoff variance ($Alwd_{var}$) and actual variance (Act_{Var}). The observed actual values are measured with respect to an allowed variance, because the network condition seen by the sender, receiver and neighbour might be different.

The simulation result demonstrates in Table III provide the ability to predict misbehaving nodes in every communication. As an example the sender id 1 in table III shows continues deviation (Act_{Var}) from the actual observed backoff value (Rec_{Act}, Nbr_{Act}). Therefore, the neighbour diagnose node 1 as "Misbehaving" after monitoring for a specific period of time (MD-MAC specifies the detection monitoring window size as a protocol parameter).

TABLE III
COMMON NEIGHBOUR'S COMMUNICATION TABLE

Id	Snd_{id}	Rec_{Id}	Nbr_{Exp}	Nbr_{Act}	Rec_{Act}	$Alwd_{Var}$	Act_{Var}
1	1	0	459	681	660	45	222
2	3	0	607	593	593	60	14
3	1	0	348	435	406	34	87
4	3	0	552	572	572	55	20
5	1	0	239	321	311	23	82

TABLE IV
RECEIVER'S COMMUNICATION TABLE

Id	$Send_{Id}$	Rec_{Exp}	Rec_{Act}	$Allowed_{Var}$	$Actual_{Var}$
1	1	459	660	45	201
2	3	607	593	60	13
3	1	348	406	34	58
4	3	552	572	55	13
5	1	239	311	23	72

B. Sender Misbehaviour Detection

The misbehaving senders selects small backoff values instead of random values, or ignore to increment the attempt number after a collision. In this case node id (1) in the Fig. 2 is misbehaving by waiting a smaller backoff value than it should have. This misbehaviour has been configured as a sender misbehaviour percentage (SMP). Therefore, the sender id (1) is violating backoff value by reducing the waiting time by SMP. As an example, if the SMP is 40% then the misbehaving sender will only wait for 60% of the allocated backoff value slots. According to the Fig. 3 and Fig. 4, MD-MAC is capable of detecting sender misbehaviours with a higher accuracy by the receiver. The result shows that, good nodes have shown a small variance to the expected backoff value while the misbehaviour nodes have a higher variance between observed and actual backoff values.

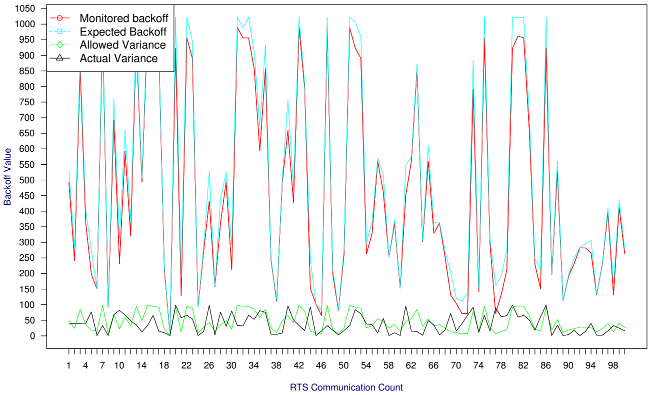

Fig. 3. Backoff values of a well-behaved sender, monitored by a good receiver

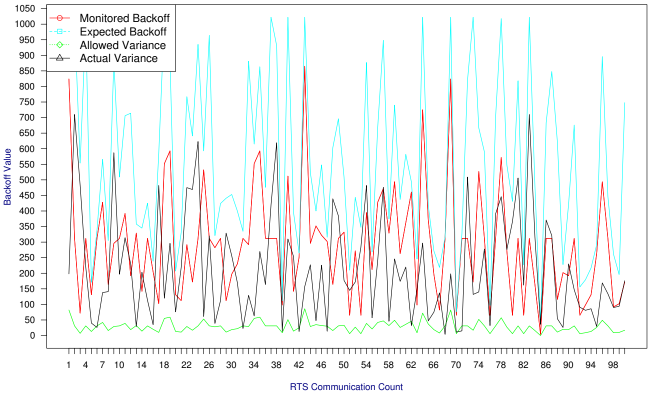

Fig. 4. Backoff values of a misbehaving sender, monitored by a good receiver

C. Receiver Misbehaviour Detection

In MANETs receivers can be non-cooperative, greedy or malicious intended nodes could misbehave by reporting wrong observation value and favour some selected senders (colluded sender-receiver). In this misbehaviour model the receiver is cheating by not reporting the correct observed backoff value of the sender. However, in MD-MAC the neighbour is still capable of detecting such misbehaviour as shown in Fig.5 and Fig.6 irrespective of receiver's collaboration. In this case the receiver trust value will be decreased for not reporting correct observed backoff value. The Fig.7 shows the trust value distribution which was calculated by a common neighbour. The trust value of a well behaved nodes is higher and maintained a good trust level in the network, however misbehaving node having a lower trust value (CRITICAL or MISBEHAVED status) throughout the monitored time period. In Fig.7 occasionally

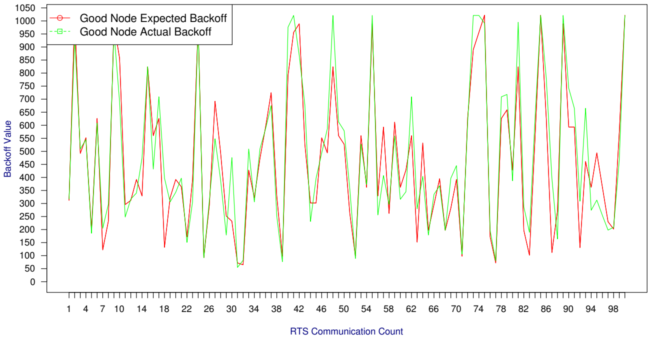

Fig. 5. Backoff values of a good sender and receiver, monitored by a common neighbour

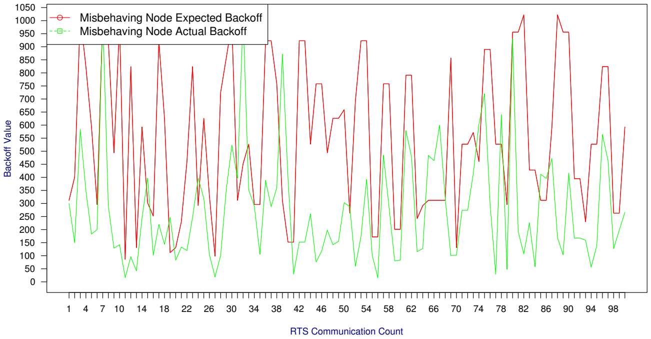

Fig. 6. Backoff values of a misbehaving sender, monitored by a common neighbour

trust value could suddenly increase to higher values from lower values due to varies network condition. However, the MD-MAC computes trust value over a period of time considering the general behaviour of the node. These trust values could be utilized in the detection and prevention mechanism or even across the layers.

VI. Conclusion and Future Work

This research proposed a novel MAC layer misbehaviour detection protocol. The MD-MAC protocol detects complex node misbehaviours in MANETs using verifiable backoff value generation mechanism with an incorporated trust model that is suitable for distributed networks. The protocol has modified CSMA/CA control packet exchange mechanism to incorporate common neighbours for monitoring. The result suggests that MD-MAC has been able to detect complicated misbehaviours in MAC layer with higher accuracy without trusting any of the communication parties. The proposed model accuracy must be compared with other detection approaches which will be carried out in future work. MD-MAC protocol can be easily adapted to detect other type of MAC layer related misbehaviours with generic and distributed model. MD-MAC could be configured to monitor any protocol parameters such as SIFS/DIFS/NAV. Also, as a future improvements MD-MAC could choose a most trusted neighbour on that node cluster, which can be used for other network layers to get information about MAC layer misbehaviours, which enhance the cross layer collaboration and allow the upper layer detection mechanisms to be more proactive.

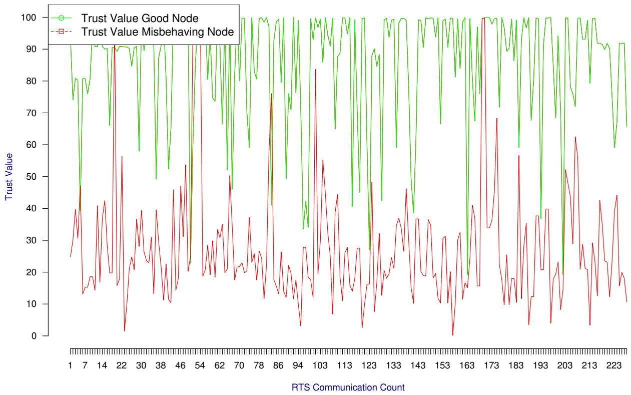

Fig. 7. Common neighbour maintained the trust value of a well-behaved and misbehaving sender node

References

[1] S. Jabbehdari, A. Sanandaji, and N. Modiri, "Evaluating and mitigating the effects of selfish MAC layer misbehavior in manets," *IEEE Wireless Communications Letters*, vol. 4, 10/2012 2012. [Online]. Available: http://www.JournalofComputing.org

[2] C.-Y. Kuo, Y.-H. Huang, and K.-C. Lin, "Performance enhancement of IEEE 802.11 dcf using novel backoff algorithm." *EURASIP J. Wireless Comm. and Networking*, vol. 2012, p. 274, 2012. [Online]. Available: http://dblp.uni-trier.de/db/journals/ejwcn/ejwcn2012.html

[3] C. Alocious, H. Xiao, and B. Christianson, "Analysis of dos attacks at mac layer in mobile adhoc networks," in *Wireless Communications and Mobile Computing Conference (IWCMC), 2015 International*, Aug 2015, pp. 811–816.

[4] P. Nagarjun, V. Kumar, C. Kumar, and A. Ravi, "Simulation and analysis of rts/cts dos attack variants in 802.11 networks," in *Pattern Recognition, Informatics and Mobile Engineering (PRIME), 2013 International Conference on*, Feb 2013, pp. 258–263.

[5] C. Alocious, H. Xiao, B. Christianson, and J. Malcolm, "Evaluation and prevention of MAC layer misbehaviours in public wireless hotspots," in *IEEE International Conference on Dependable, Autonomic and Secure Computing. DASC '13*. IEEE, 2015.

[6] H. Diwanji and J. Shah, "Effect of mac layer protocol in building trust and reputation scheme in mobile ad hoc network," in *Engineering (NUiCONE), 2013 Nirma University International Conference on*, Nov 2013, pp. 1–3.

[7] M. K. Han and L. Qiu, "Greedy receivers in IEEE 802.11 hotspots: Impacts and detection," *Dependable and Secure Computing, IEEE Transactions on*, vol. 7, no. 4, pp. 410–423, Oct 2010.

[8] P. Kyasanur and N. Vaidya, "Selfish MAC layer misbehavior in wireless networks," *Mobile Computing, IEEE Transactions on*, vol. 4, no. 5, pp. 502–516, Sept 2005.

[9] S. Radosavac, A. A. Cárdenas, J. S. Baras, and G. V. Moustakides, "Detecting IEEE 802.11 MAC layer misbehavior in ad hoc networks: Robust strategies against individual and colluding attackers," *Journal of Computer Security*, vol. 15, no. 1, pp. 103–128, 2007.

[10] L. Guang and C. Assi, "Mitigating smart selfish MAC layer misbehavior in ad hoc networks," in *Wireless and Mobile Computing, Networking and Communications, 2006. (WiMob'2006). IEEE International Conference on*, June 2006, pp. 116–123.