Detecting MAC Misbehavior Switching Attacks in Wireless Mesh Networks

Divya Bansal
Assistant Professor, CSE
PEC University of Technology
Chandigarh

Dr Sanjeev Sofat
Professor, CSE
PEC University of Technology
Chandigarh

Piyush Pathak
CSE, PEC University of Technology
Chandigarh

Sanchita Bhoot
CSE, PEC University of Technology
Chandigarh

ABSTRACT

To provide internet connectivity in the areas where it is difficult to lay down Ethernet cables wireless mesh technology emerges as a promising technology. Wireless mesh networks make use of two different radios; one radio is used to provide internet access to the stations and another radio is used to provide wireless connectivity among mesh points for data forwarding. The radio which is used by stations to connect with the MAPs is vulnerable to many greedy MAC misbehaviors. To detect greedy MAC misbehavior many IDS have been designed and implemented for traditional Wireless LANs and MANETs. However our approach is a novel approach as the IDS is implemented on MP thus increasing its detection range. Moreover, this IDS is designed to detect hybrid attacks as well as fast switching attacks which otherwise are quite difficult to detect.

Keywords

Wireless Mesh Network, Intrusion detection system, Single layer, Smart Attacks.

1. INTRODUCTION

Wireless mesh networks consists of mesh routers and mesh clients, where mesh routers have minimum mobility and form the backbone of WMNs. They provide network access for both mesh and conventional clients. The integration of WMNs with other networks such as the Internet, cellular, IEEE 802.11, IEEE 802.15, IEEE 802.16, and sensor networks etc can be accomplished through the gateway and bridging functions in the mesh routers [1]. The radio that provides wireless access to clients is vulnerable to many security threats as the clients can use the vulnerabilities in MAC layer protocols to gain frequent access to the channel and to acquire more share of bandwidth.

In this paper, we have presented different greedy MAC misbehavior and an approach to prevent them. The detection mechanism presented is able to detect different greedy MAC misbehaviors which are launched by Mesh clients to gain an unfair access to channel and resources. The advantage of proposed scheme is that it has been developed for mesh networks unlike previous approaches which were mainly designed for traditional wireless networks or MANETs. Another advantage of this approach is that it can detect if an attacker node is switching between different attacks and it can also detect misbehavior if it do not fall under any specific attack.

The previous work done in this direction is DOMINO [2], which is a very popular approach and detects different greedy MAC misbehavior attacks. It has considered both the uplink and downlink traffic for detection of any misbehavior. However there are some problems associated with DOMINO; a hidden terminal may have a negative impact on DOMINO as a station may appear to be malicious but actually it is working normally according to the channel conditions prevalent near the node. Another problem that DOMINO faces is that it is unable to detect, if an attacker node is switching between attacks i.e. when it keeps on switching between different attacks the IDS would be unable to collect data to detect misbehavior.

2. RELATED WORK

The previous work done in this direction is DOMINO[3], which is a very popular approach and detects different greedy MAC misbehavior attacks. It has considered both the uplink and downlink traffic for detection of any misbehavior. However there are some problems associated with DOMINO; a hidden terminal may have a negative impact on DOMINO as a station may appear to be malicious but actually the station is behaving normally according to the channel conditions prevalent near the node. Another problem that DOMINO faces is that it is unable to detect if an attacker node is switching between attacks i.e. it keeps on switching between different attacks the IDS would be unable to collect data to detect misbehavior.

Current research in the area of security and management in WMN is still in early stages. The technology has been implemented mostly in the form of the experimental testbeds with little production use. Most of the IDS which act as an important line of defense have been proposed and implemented for 802.11 WLAN, MANETs or WSNs. Watchers[7] have been proposed for distributed environments. They can detect network traffic anomalies and misbehavior attacks but have huge memory requirements increasing cost. Watchdog & Pathraters [8, 9] detect intrusions for mainly network layer and are based on DSR routing. CONFIDANT [10] is based on reputation based approaches and is efficient mostly for packet dropping attacks. It cannot detect protocol deviation based attacks. Several research efforts have been made in developing approaches based on Cross layer designs [11,12]. However the IDS is host based and does not exploit the advantages offered by WMN.

3. PROPOSED SCHEME

In our proposed scheme, we have designed and implemented mechanism to detect MAC layer greedy misbehavior. There are different attacks through which the greedy nodes try to get more bandwidth than their actual share. Also by using these attack mechanisms the nodes gain frequent access to network as compared to legitimate nodes. There are two MAC misbehaviors which has been discussed in this paper, they are oversized NAV attack and reduced backoff attack. The advantage of proposed scheme is that along with the detection it can also determine if the node has performed switching between the attacks i.e. instead of launching only one attack it has performed more than one attack side by side.

2.1 Attack simulated

2.1.1 Oversized NAV Attack

NAV or Net Allocation Vector as we have seen is a part of virtual carrier sensing mechanism. NAV is set by all the stations that are in the communication range of two communicating entities, so that they do not send any packet till the ongoing transmission is completed and thus preventing collision. This NAV is used by the attacker i.e. the setting of the NAV by the nodes after receiving the RTS. As illustrated in Figure 1, the attacker sets a larger duration field in its RTS and hence the well behaved clients are unable to use the channel. Since a legitimate station always replies RTS by CTS, the attacker can make colluding partners to increase the magnitude of the attack. [3]

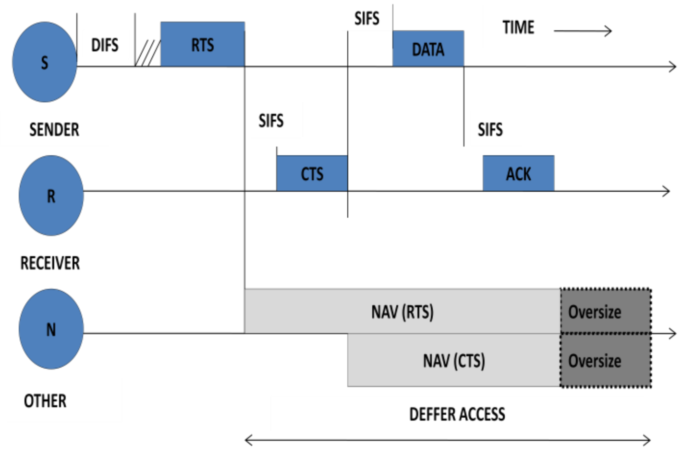

Figure 1: Oversized Network Allocation Vector Attack

Increase in average throughput of attacker node: Figure 2 shows that the attacker node occupies the channel for a longer duration of time; thereby defer the access to legitimate nodes. As a result the throughput of legitimate node falls down because they try to access the channel simultaneously and hence there is an increase in the contention of the network as the attacker node keeps the channel occupied for majority of time and thus able to send the data without any contention.

Decrease in average end to end delay: Figure 3 denotes a decrease in average end to end delay for the attacker node pair. One of the major components of average end to end delay is the queuing delay which is the amount of time the packet stays in the queue before transmission. Since the duration field of the attacker node is higher it does not wait for a longer time before transmitting the packets while the legitimate nodes have to wait for a longer duration for transmitting their packets. This result in increase in average end to end delay of legitimates nodes.

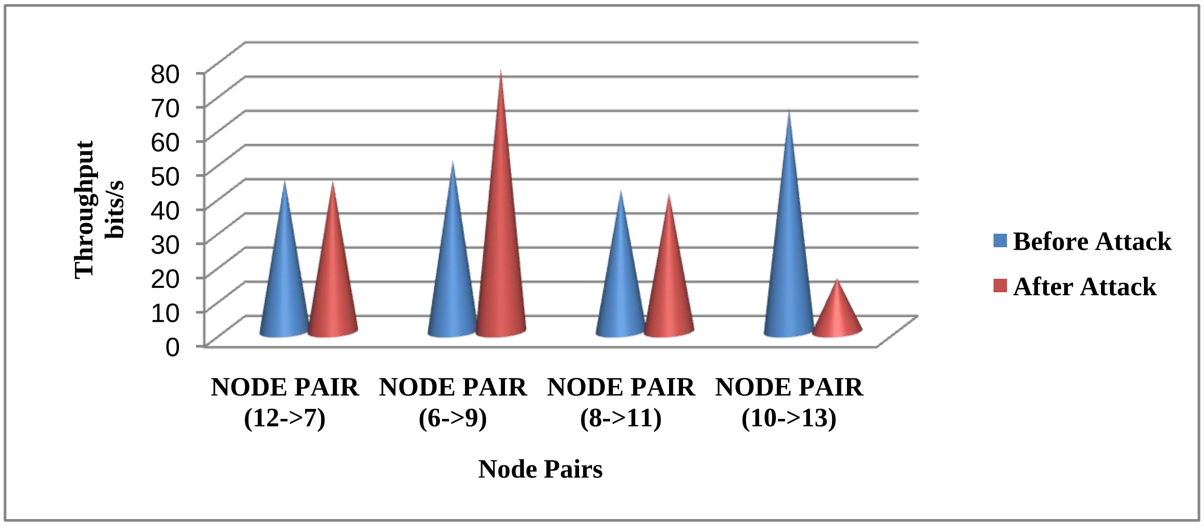

Figure 2: Change in average throughput due to attack

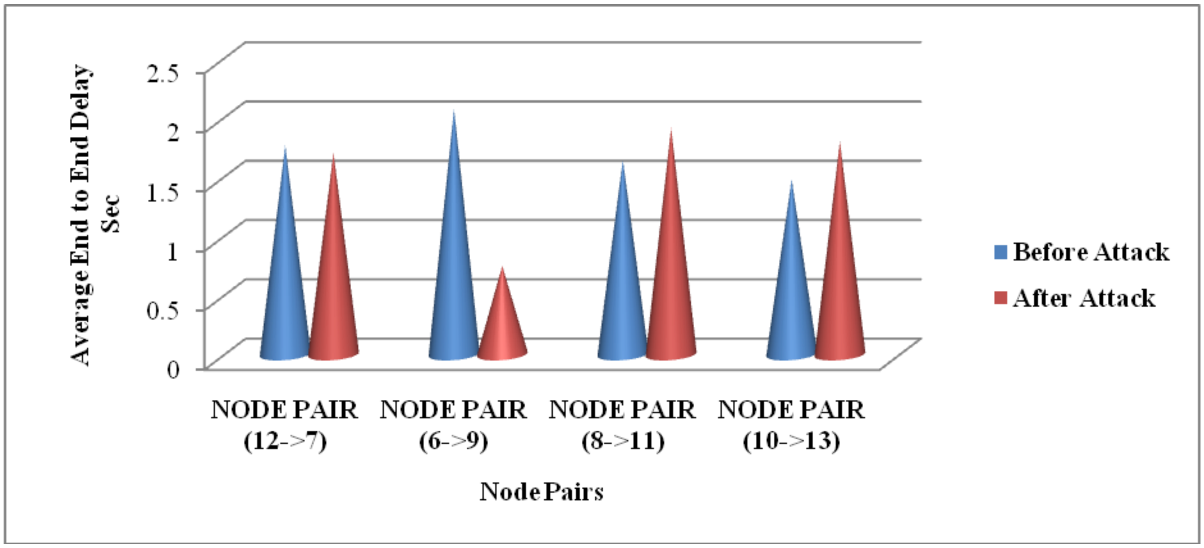

Figure 3: Change in average end to end delay due to attack

2.1.2 Reduced Backoff Attack

WMN uses CSMA/CA to access the medium. In CSMA/CA station senses the medium before sending any data to the network. If the medium is free it will wait for DIFS amount of time. After completion of DIFS if the medium is still free, station starts its backoff timer and after the completion of backoff time, it transmits the data. There is a contention window generally of size {0, CW} from which each node selects its backoff time and in case of collision doubles the backoff time before next transmission. A malicious node (as shown in fig. 4) instead of using this contention window may choose its backoff time from a comparatively smaller contention window say {0, CW/4}. It helps the misbehaving node to access the channel more frequently because the size of contention window is much smaller as compared to contention window used by other nodes. Also in case of collision the backoff time only increases to a very small amount. [4].

Increase in Total Number of Packets: Figure 5 shows an increase in the total number of packets by attacker node pair. Due to the smaller contention window of attacker node as compared to the legitimate nodes, it gets frequent access to the network which benefits the attacker node to send larger number of packets.

Decrease in average end to end delay: Figure 6 shows a decrease in average end to end delay for the attacker node pair. Since the attacker node is able to access channel more quickly rather than waiting for its turn due to smaller contention window. It results in less amount of time delay between its two consecutive packets; also the queuing delay goes low as the packets wait for lesser time in the queue of attacker node.

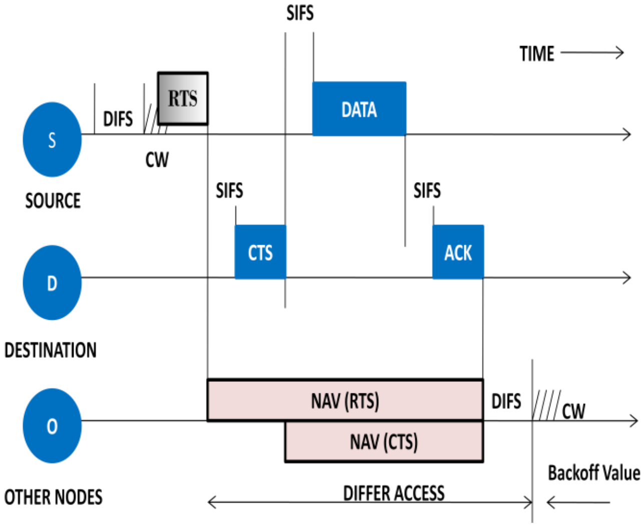

Figure 4: Reduced Backoff Attack

In case of legitimate nodes, they follow the normal contention window and also due to frequent access by attacker node the channel remains busy and hence legitimate node need to wait for longer duration for channel access which results in increased average end to end delay.

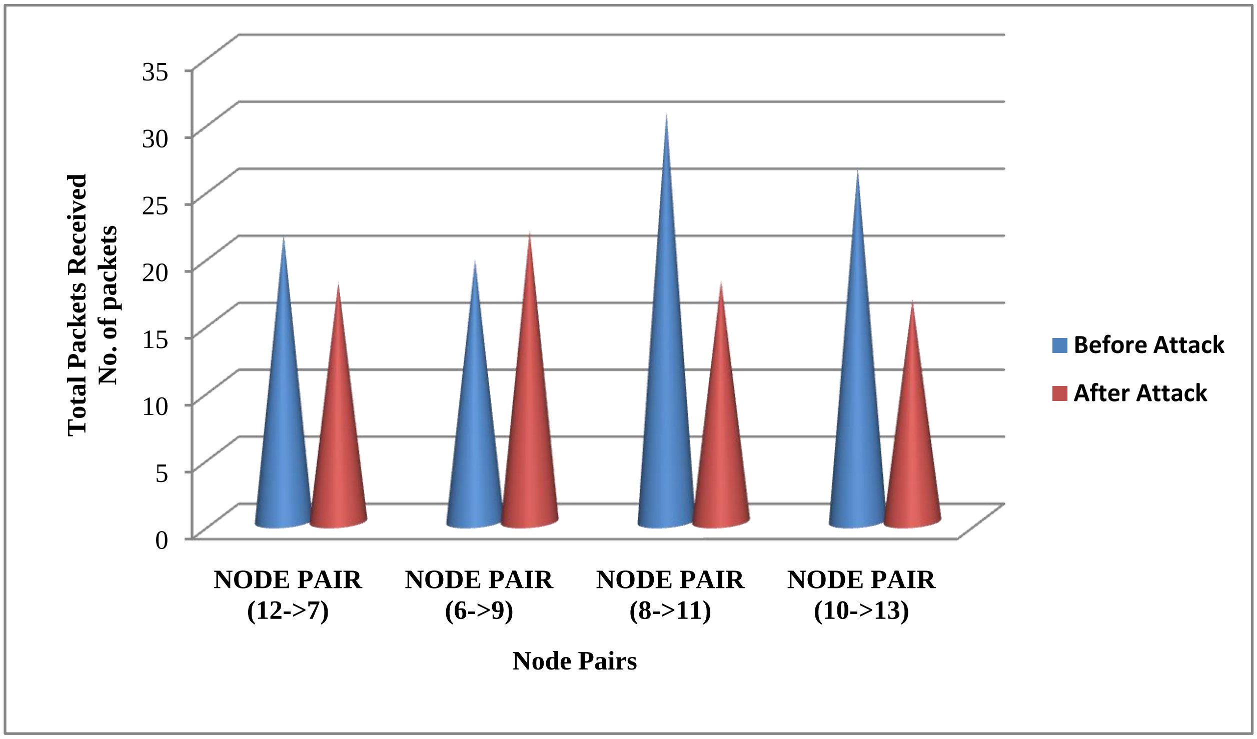

Figure 5: Change in total no. of packets received due to attack

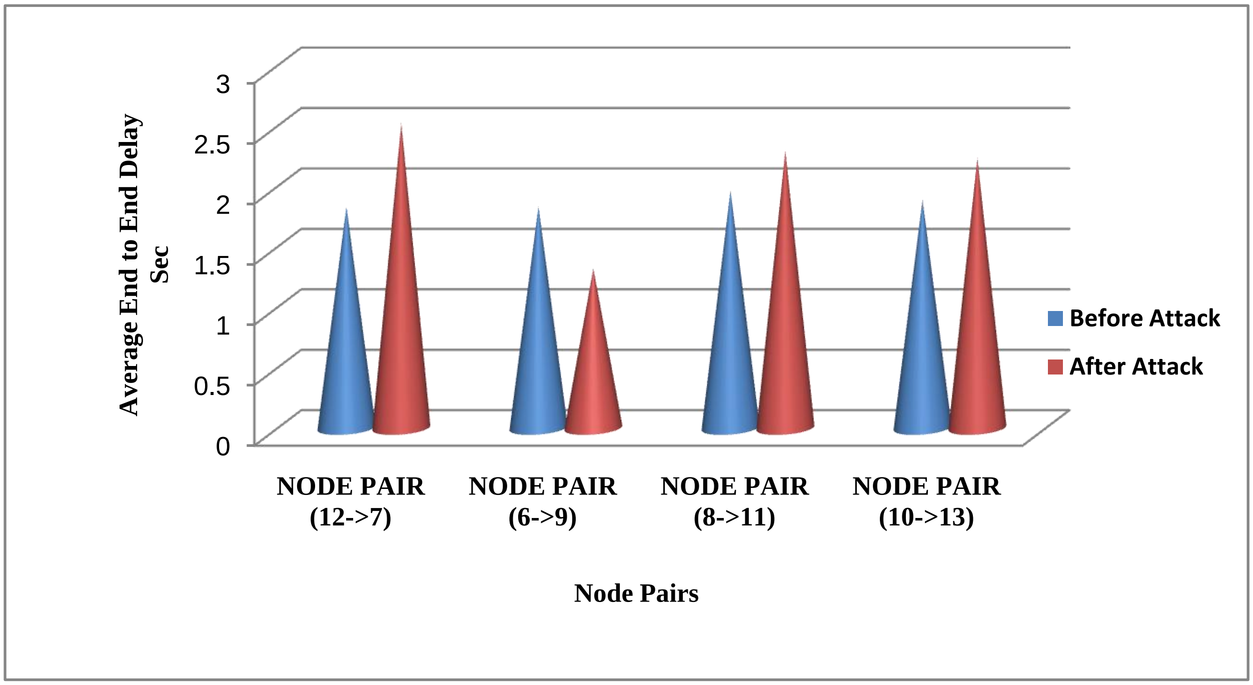

Figure 6: Change in average end to end delay due to attack

4. ATTACK DETECTION MECHANISM

To detect the simulated attacks two different algorithms are used which are implemented on the monitoring node. On receiving the RTS the monitoring node stores the required parameters and forwards the parameters to detection algorithms which in turn determine whether an attack has been launched or not.

Oversized NAV attack detection algorithm

```
1    Check the duration field of received RTS.
```

```
2   Determine the RTS belong to which node.
3   If (attacking_flag is not set)
    I.  If (Duration field > 3030).
        a.  Increase the selfish factor.
        b. Update the type of attack, node id and time stamp
           parameters in database named “switching.csv”
           for predicting switching.
    II. If (selfish factor > tolerance value).
        a. The node will be declared as malicious
        b. Time stamp is added in database when attack is
           declared.
        c. Set the attacking_flag
        d. Return (node id*10 + attacking_flag*1)
4   If (attacking_flag is set)
        a. Return (node id*10 + attacking_flag*1)
```

For the detection of ONAV attack, the monitoring node on receiving the RTS frame will check the duration field. For that node it checks whether attacking_flag value is set or not. If the attacking_flag value is not set then it checks whether the value of the duration field is greater than 3030. If it is greater than that there will be an increase in the selfish factor of the corresponding node. Every time selfish factor is increased corresponding parameters i.e. type of attack, node id and time stamp are updated in the database. Selfish factor is compared to the tolerance_value, if selfish factor is greater then

```
6   If (attacking_flag is set)
    I  Return (node id*10 + attacking_flag*1)
```

For the detection of reduced backoff attack, the monitoring node determines the time of last RTS and latest RTS received. If the sender of both RTS is same then the time difference between both RTS is determined, for the particular sender node. The attacking_flag value is checked. If the attacking_flag value is not set and calculated difference is below allowable_time_difference then increase the selfish factor for that node. Update the database with parameters i.e. node id, type of attack and time stamp. The selfish factor is compared with tolerance_value, if it is greater than the tolerance_value node will be declared as malicious node. Attacking_flag value is set and value is returned to show that attack is declared.

If attacking_flag value is set then a value is returned to show that attack is declared already for that particular node.

Detection module

A trusted mesh point is configured to monitor the nodes. A mesh point can monitor stations as well as other mesh points and thus can create a record of the activities performed by them. A mesh point is configured as monitoring node because it can listen to RTS and CTS travelling in the network.

While monitoring the stations it creates a file which contains the fields such as duration field (embedded in the RTS), sender ID (who is the sender of the RTS) and timestamp (which indicates the time at which RTS has been sent) for each participating station of a network. A file containing all these fields is

tolerance_value, node is declared as malicious. Time stamp is added in the database, attacking_flag value is set and a value is returned to show that attack has been declared.

If attacking_flag value is set then a value is returned to show that attack is already declared.

Reduced backoff attack detection algorithm

```
1   Determine the time of last RTS received
2   Determine the time of latest RTS sent
3   Determine the both RTS belong to which node
4   For particular node id
    I.  Determine t = (time of latest RTS received– time of
        last RTS received)
5   If (attacking_flag is not set)
    I.  If (t < allowable_time_difference)
        a. Increase the selfish factor
        b. Update the type of attack, node id and time stamp
           parameters in database named “switching.csv”
           for predicting switching.
    II. If (selfish factor > tolerance value).
        a. The node will be declared as malicious
        b. Time stamp is added in database when  attack is
           declared.
        c. Set the attacking_flag
        d. Return (node id*10 + attacking_flag*1)
```

maintained for each station and processed by the monitoring node for detection of the attacks.

Monitoring MP triggers the detection algorithm in order to detect the attacks whenever it listens a new packet from a node. A value is returned by detection algorithm under the heading RBOFF malicious value and ONAV malicious value, which shows whether attack is performed by that particular node or not. If an attack is detected it is entered in the result file. It also saves the time stamp that denotes the time when the attack has been detected. If a node is declared as malicious any new entry for the same node is not processed by the detection algorithm, thus saving system resources.

After the processing of the monitoring engine is finished all the data is transferred to the inference engine where the final processing takes place.

5. SWITCHING DETECTION MODULE

In this approach switching detection module or the inference engine is used to detect switching between two attacks presented in the paper. Inference engine analyzes the data collected by the monitoring node during its monitoring period for the node performing the malicious activities. The parameters collected by the monitoring node are node id, time stamp (when an attack was detected) and attack id. Switching module compares the recently collected node id with last analyzed node id, if both of the node ids are same it will next compare the attack type id. If the attack type id is different, time stamp is compared and if that is also different then switching factor is incremented by one.

If switching count for a node reaches a value which is greater than average threshold values for two declared attacks, than that node is declared as malicious and performs switching as well. It can be possible that a node is not declared as malicious under any particular detection module but may be launched the attack by switching. Such smart attacker can also be detected by proposed technique.

Algorithm for switching detection

```
1  Open the database named “switching.csv” file
   Read each row until end of file and extract parameters
   i.e.  (Type of attack, node id and time stamp)
   I. If (node id is already discovered)
      A. If (for node id recent time stamp and last time
         stamp read is different)
         i. If (recent type of attack is different from last type
            of attack)
            a. Increase switching count.
            b. Update time stamp with recent time stamp.
            c. Update type of attack by recent type of attack.
         ii. Set updated_flag
   II. If (updated_flag is not set)
         i. Add node id.
         ii. Add type of attack.
         iii. Add time stamp.
         iv. Increase switching count.
2  Close “switching.csv”
3  Print the total switching done by each node.
4  If (total switching done by a node> average thresholds of
   all the attacks)
   I.  Node is declared as attacker under switching
       environment.
```

6. CONCLUSION AND FUTURE WORK

In the given paper we have compared the proposed detection approach with the approaches designed previously for detecting selfish MAC misbehavior. We have compared our IDS with DOMINO and other proposed schemes, and it can be clearly stated that the proposed approach is able to solve the problem of switching or detection of smart attacks which DOMINO cannot detect, moreover proposed approach has positive points over other schemes. We have compared our detection approaches with the approaches that were designed earlier in the Table 1. As can be concluded from the paper, hidden terminal was a major problem with almost all the previously existing approaches which is solved by proposed mechanism. Also, the previously proposed schemes were mainly designed for MANETs while current approach works successfully on wireless mesh networks.

As the future work, this approach can be enhanced to detect new selfish MAC misbehaviors.

Table 1. Comparison of different schemes for Oversized NAV

Comparison of Oversized NAV				
Comparison	**Existing approach – I [2]**	**Existing approach – II [6]**	**Existing approach – III [5]**	**Proposed Detection Method**
Methodology	– Comparison of declared and actual duration field proposed in [2].	– Accepted duration value is limited. – Any packet that contains a larger duration field is truncated to the maximum allowable value.	– Assumed that the nodes follow a uniform distribution and depending upon that distribution mean and standard deviations are calculated. – A threshold value is set for each node. – In case of attack, the mean and standard deviation varies largely and compared with threshold value.	– Comparison of declared and Normal duration field. – In our scheme the detection method can be implemented on Mesh points which can make it much more cost effective.
Problems with existing approaches	– Hidden terminal could be a problem.	– Hidden terminal could be a problem.	– Authors assumed that the receiver node is well behaved and hence the data is collected and processed by receiving node. – If the receiver node is misbehaving as well then it is difficult to detect the misbehaving node.	– As the proposed scheme implements the detection mechanism on Mesh Point there is no need to rely on receiving nodes and hence no problem of colluding partner.

Table 2. Comparison of different schemes for Reduced BackOff Attack

Comparison of Reduced Backoff			
Parameters Selected	**Existing approach – I [2]**	**Existing approach – II [4]**	**Proposed Detection Method**
Methodology	– Actual Backoff Consecutive Backoff	– The receiver sends the backoff value in the CTS and ACK packet instead of sender selecting a backoff value. – The receiver then observes the number of ideal slots between consecutive transmissions of sender. – if the number of ideal slots is less than the assigned backoff the sender is deviating from the protocol. – It uses a penalty approach to punish the misbehaving sender.	– Difference of consecutive RTS time stamp is compared to allowable tolerance value. – In our scheme the detection method can be implemented on Mesh points, cost effective
Problems with existing approaches	– The Reduced Backoff check can be tricked, if a cheater succeed in making the monitor observe in every sample at least one Backoff value larger then or equal to the threshold. – Channel condition can also yield the similar result and thus makes the check fail.	– Assumes that there is no collusion between sender and receiver and hence this approach fails in case of colluding nodes. – Also this approach results in the modification of 802.11 protocol. – Hidden terminal can be a problem in the given approach.	– In our scheme monitoring node observes every packet and hence the cheater cannot trick the monitoring node.

Table 3. Comparison of different schemes for Switching Attack

Comparison of switching attacks			
	Existing approach – I [2]	**Existing approach – II [4]**	**Proposed Detection Method**
Methodology	– Cannot be detected. – Other approaches defined above are meant to detect only one attack and hence switching cannot be detected.	– Can be detected.	– In our prototype design inference engine is used to detecting switching between two considered attacks.

7. ACKNOWLEDGEMENT

This work is done in Cyber Security Research Center located at PEC University of Technology. The author would like to thank Government of India, Ministry of Communications and Information Technology, Department of Information Technology, New Delhi, for funding the project “Design and Development of Dependable, Secure and Efficient Protocol for Wireless Mesh Network”, under whom this work has been done.

8. REFERENCES

[1] Ian F. Akyildiz, Xudong Wang, Weilin Wang, “Wireless Mesh Networks: a Survey”, Elsevier, Computer networks (47) 2005 445-487.

[2] S.Savage, J. Bellardo, “802.11 Denial of Service Attacks: Real Vulnerabilities and Practical Solutions” USENIX Security Symposium, June 2003.

[3] I. Aad, M. Raya, J. Hubaux, “DOMINO: A System to Detect Greedy behavior in IEEE 802.11 Hotspots”, MobiSys, pages 84-97, 2004.

[4] P.Kyasanur, N. Vaidya, “Detecting and Handling of MAC Layer Misbehavior in Wireless Networks”, Dependable Systems and Networks, June 2003

[5] K. Sugantha, S. Shanmugavel, “A Statistical Approach to Detect NAV Attacks at MAC Layer”, unpublished

[6] IEEE Standard for Wireless LAN- Medium Access Control and Physical Layer Specifications, ANSI/IEEE Standard 802.11, 1999 Ediiton (2003)

[7] Chen M., Kuo S., Li P., and Zhu M., Intrusion Detection in Wireless Mesh Networks, CRC Press, 2007.

[8] Caballero J., “Vulnerabilities of Intrusion Detection Systems in Mobile Ad-hoc Networks: The Routing Problem,” in Proceedings of TKK T-110.5290 Seminar on Network Security, Japan,pp. 1-2, 2006.

[9] Rafsanjani K., Movaghar A., and Koroupi F., “Investigating Intrusion Detection Systems in MANET and Comparing Idss for Detecting Misbehaving Nodes,” in proceedings of World Academy of Science, Engineering and Technology, Canada, pp. 123-128, 2008.

[10] S. Buchegger, J-Y. Le Boudec, ”Performance analysis of the CONFIDANT protocol (Cooperation of Nodes: Fairness in Dynamic Ad-hoc Networks),” 3rd ACM Int. Symp. on Mobile Ad Hoc Networks and Computing, Switzerland, 2002, pp. 226-236.

[11] Geethapriya Thamilarasu, Ramalingam Sridhar, "CIDS: cross-layer intrusion detection system for mobile ad hoc networks", International Journal of Mobile Network Design and Innovation 2009 - Vol. 3, No.1 pp. 10 - 20

[12] Jim Parker, Anand Patwardhan, Anupam Joshi Cross-layer Analysis for Detecting Wireless Misbehavior Consumer Communications and Networking Conference, 2006. CCNC 2006. 3rd IEEE, Vol. 1, Jan. 2006, pp. 6 – 9