A Bayesian Trust Model for the MAC Layer in IEEE 802.15.4 Networks

Bernardo Machado David
Electrical Engineering Department
University of Brasilia
bernardo.david@redes.unb.br

Rafael Timoteo de Sousa Jr
Electrical Engineering Department
University of Brasilia
desousa@redes.unb.br

***Abstract*—While the IEEE 802.15.4 standard provides internal security mechanisms, it is still possible that a malicious or faulty node abuses CSMA/CA and GTS allocation to gain continuous access to media or to prevent other nodes from transmitting, resulting in MAC unfairness. In this paper we present a practical attack that takes advantage of this vulnerability and analyse how it could affect network performance and ultimately lead to Denial-of-Service attacks even if secure Ad-Hoc routing protocols are in use. Considering the usual computational resources constraints of LR-WPAN nodes, which limit the use of standard cryptography and authentication solutions, we present a novel bayesian trust model based on MAC sublayer data to mitigate unfairness and consequent Denial-of-Service attacks. The proposed model may also be used to enforce GTS allocation policies.**

I. Introduction

Low rate wireless personal networks (LR-WPANs) [1] are progressively being adopted in wireless sensor networks (WSN) setups. The ZigBee protocol stack, which is specifically based on the IEEE 802.15.4 standard, and diverse Ad-Hoc routing protocols used on top of LR-WPANs' MAC and physical layers fit the power consumption and processing performance constraints of WSN nodes [2]. In this kind of resource constrained environment, denial of service (DoS) attacks pose a serious threat as they waste power and cpu cycles. Moreover, they degrade overall network performance, causing delays that are unacceptable in real-time applications.

There are several kinds of DoS attacks in WSNs [3], which affect diverse layers. The majority of these attacks require the adversary nodes to perform non-standard operations in the network, generating traffic or noise patterns that can be identified and used to detect these nodes; therefore, they can be addressed by secure routing protocols and trust based solutions on the Network layer. However, attacks resulting from MAC unfairness do not clearly expose the attacking nodes because they rely on small modifications to CSMA/CA parameters, so that attacks appear to the other nodes as legitimate communication.

Most WSN security analysis surveys and proposed security models in current literature are focused on authentication and confidentiality issues, providing solutions mainly for the Network and Application layers. However, these models do not properly address MAC protocol attacks and also tend to increase packet size, which means longer transmission times per packet and, consequently, shorter battery life [4]. In face of the packet overhead, high cpu loads and consequent poor battery performance inherent to cryptographic solutions, trust-based security models were proposed for both WSNs and Ad-Hoc networks in general. While trust-based models provide the security required by various applications they do so without increasing transmission time and having small impact on memory and processing resources. Being focused on upper layers, these models still do not address MAC unfairness but are a promising approach to solving this problem if applied at MAC sublayer level.

We present a new attack which results in controlled MAC unfairness through arbitrary modifications of the CSMA/CA algorithm employed by LR-WPANs. This attack is effective both in beacon-enabled and non-beacon-enabled modes and may be used by malicious nodes to achieve higher medium access priority, arbitrarily allocate GTS and perform Denial of Service (DoS) attacks that disrupt legitimate communication between its surrounding nodes. In order to efficiently mitigate the presented attack we propose a novel bayesian trust model based on data regarding the MAC sublayer in which a central node (the Coordinators) infers the trust value of a node by combining previous information on its behaviour with information gathered from the other nodes. The proposed model is also suitable for enforcing GTS allocation policies and may serve as a component of a more comprehensive Multi-Layer trust model.

The rest of this paper is structured as follows: In section II, we present a brief overview of common DoS attacks focused on the Link Layer and MAC sublayer of WSNs. In section III, we thoroughly analyse how MAC unfairness can be used to achieve DoS and present our attack. In section IV, we discuss the countermeasures against the presented attack and secure protocols for WSNs in current literature. In section V, we describe the proposed trust model in detail. In section VI, we present simulation results regarding our trust model's performance in different scenarios. Finally, in section VII, we conclude with a summary of our results and directions for future research.

II. Denial of Service Attacks in WSNs

A Denial of Service attack is defined as any deliberate action that keeps the network from performing adequately, causing

long delays or even completely disrupting communications. Wireless networks use a shared medium that can be monitored and tampered by any adversary within transmission range, making them more vulnerable to such attacks. In order to save energy, WSN nodes are configured to operate in low-power mode for most of the time, activating the radio transciever only when there's data being received or queued for transmission, operations that consume much more power than regular data acquisition and processing. During a DoS attack, nodes may be forced to keep the radio in receive mode for long periods, or, in the worst case, attempt several retransmissions before finally dropping a packet. This decreases battery life and wastes processing resources, thus posing a serious threat to resource-constrained WSNs [5].

DoS attacks can be launched against several layers, exploring protocol vulnerabilities or design-level issues. In this work we focus on the Link Layer, specifically on the Media Access Control (MAC) sublayer.

A. Link Layer (and MAC sublayer)

Link layer (including MAC sublayer) is not affected by as much vulnerabilities as the Network Layer but, considering that it controls the radio power modes [1] and that its activity might go unoticed by security solutions placed in upper layers, it figures as an important target. Collision attacks are extremely energy efficient because they require the attacker to cause a collision in one byte of a transmission to completely alter a packet checksum, forcing the attacked nodes to retransmit. However, depending on the error correction codes in use, it would be necessary to induce a large number of collisions to cause packet corruption, making the attack less effective, or else, turning it into a deceptive jamming attack [6] (that consists of constantly sending valid PPDUs despite of medium access control procedures). Exhaustion attacks explore protocol characteristics to force as much retransmissions as possible. In IEEE 802.15.4 networks the number of retransmissions per frame is limited by the aMaxFrameRetries paramater, making this attack less effective. In Denial-of-Sleep attacks certain MAC protocol control messages are captured and replayed by the attacker, forcing the nodes to stay awake. Unfairness attacks (which will be further analyzed in the next section) consist of bypassing the MAC protocol priority scheme, mainly by cheating when negotiating channel access. This way a malicious node can keep the other nodes from transmitting while maintaining legitimate communication, making it difficult to identify this attack.

III. MAC Unfairness and Potential Attacks

MAC fairness is achieved when nodes have the same medium access priority. Put differently, the MAC sublayer is fair when the bandwidth is equally allocated to each contending node over similar periods of time. The fairness of a MAC protocol may be verified by observing the network on a short-term or a long-term basis [7]. Although the MAC sublayer achieves long-term fairness it might present short-term unfairness, which degrades real-time applications performance [8]. MAC unfairness happens in scenarios where a node or a group of nodes captures and monopolizes the channel for a long period. It can be achieved by malicious nodes that cheat when contending for access, subverting multiple access protocols so as to gain access before other nodes.

Attacks based on MAC unfairness are extremely effective because they do not generate any easily identifiable traffic pattern, thus looking like legitimate traffic. Furthermore, being targeted at the MAC sublayer, these attacks can not be thwarted by security solutions based on upper layers. Simulation results in [9] show that, even though packet delivery rate (PDR) is not significantly affected by MAC unfairness for moderate traffic loads, the packet delivery latency (PDL) tends to grow for any traffic load. An adversary could build on the increased latency to perform attacks against other protocols and layers, such as Network Layer adhoc routing protocols.

LR-WPANs operate in both beacon and non-beacon enabled modes, each requiring different multiple access protocols for channel access. Non-beacon-enabled mode uses CSMA-CA while beacon-enabled mode uses *slotted* CSMA-CA during the contention access period (CAP), since it provides better performance for synchronized networks. Attack methods differ from one scenario to another.

A. Non-beacon-enabled Mode

In the CSMA-CA multiple access algorithm, a node vying for access will first wait for a random backoff period of $P = random(2^{BE} - 1) * aUnitBackoffPeriod$ symbols (where $BE = macMinBE$ in the first iteration of the algorithm) and then perform the Clear Channel Assessment (CCA) procedure. If the channel is idle the node proceeds and transmits its data, whereas, if the channel is busy, it will make BE=BE+1, wait for another random period P and retry (performing again the CCA). The IEEE 802.15.4 *macMaxCSMABackoffs* parameter controls how many times this process will be repeated before CSMA-CA terminates with a CHANNEL_ACCESS_FAILURE status, which will be recieved in the MLME-COMM-STATUS.indication primitive issued by the MLME.

An adversary who wants to disrupt the network would perform a DoS attack against the PAN or local coordinators, depriving the ordinary nodes, which are reduced function devices (RFDs), of communication. In multihop networks, such an attack could completely isolate one region of the network if there aren't any alternative routes (coordinators) outside the attacker's transmission range.

A simple method to execute this attack is to perform the CCA repeatedly, without waiting for the backoff period, until the channel is found to be idle and capturing the channel as soon as possible. The attacker could then keep the channel busy by transmitting a large sequence of messages. This would prevent the other nodes that were contending for access from transmitting their data, causing delays. It is important to notice that, if this attack is repeated too frequently, it will resemble a deceptive jamming attack, becoming easier to detect, even though it is exploring a MAC unfairness vulnerability. Another

way to achieve similar results is to wait for arbitrarily small backoff times before performing the CCA. An attacker using the latter method would not always capture the channel, but it still increases PDL and makes attack detection more difficult.

B. Beacon-enabled Mode

In beacon-enabled mode two beacons delimit a superframe structure, which is divided into 16 time slots by default and further broken down into Contention Access Period (CAP), Contention Free Period (CFP) and inactive period. Slotted CSMA-CA is used during CAP and no multiple access algorithm is used during CFP, instead, nodes allocate Guaranteed Time Slots (GTS), during which they have total priority to transmit data to a coordinator. A GTS may consist of up to seven time slots and is allocated by sending a GTS request during the CAP and waiting for the coordinator's response in the next beacons.

A beacon-enabled mode DoS attack targeting the CAP is achievable by capturing the channel immediately after a beacon is received. A cheating node may simply wait for the arrival of a beacon packet and then start transmitting immediately by skipping backoff and CCA processes. The attacker can then maintain its control of the channel by transmitting successive messages as in non-beacon-enabled mode.

It is also interesting to target the CFP, causing the coordinator to waste resources and GTS dependent applications to fail. If an adversary can capture the channel during the CAP, it can issue one or more GTS request commands to allocate the possible maximum number of GTS and then keep the channel busy, so as to prevent other nodes from also allocating GTS. The coordinator would probably allocate all the CFP in the next superframe to the malicious node, that could simply send nothing or send random data, forcing the coordinator to receive and process it.

In both attacks, the other nodes will get a CHANNEL_ACCESS_FAILURE status when issuing GTS request commands or contending for channel access. When all GTSs have been allocated the GTS request commands issued by legitimate nodes will receive a MLME-GTS.confirm primitive with a status of DENIED.

IV. Countermeasures and Secure Protocols

To defeat DoS and other attacks several individual countermeasures have been suggested for WSNs . They include protocol modifications and hardening on diverse layers but it would be impractical to individually implement each correction on pre-existing networks or even in new projects. In order to solve various security problems in a more global manner, a number of security protocol suites have been suggested and implemented, including SPINS [10], a Network layer focused solution, and TinySec [11], that is aimed at the Link layer. Both security architectures provide an excelent solution for complex networks that demand strict data confidenciality, authentication and integrity assurances. However, in WSNs that need only to be secured against multiple attacks but demand no confidenciality regarding their data, these protocol suites unecessarily overwhelm the node's restricted resources.

Most of these defences rely on cryptographic funtions to provide confidenciality and authentication services. As stated earlier, in the resource constrained world of WSNs, cryptography centered solutions turn out to be inefficient in some scenarios because they consume too much of a node's limited memory and processing power. Moreover, the overhead inherent to secure protocol control data and encrypted information increases packet size, consequently increasing transmission time. The intensive use of processing resources and longer transmission (radio on) times contribute to dramatically decrease the node's battery lifetime. In scenarios where confidenciality is not a key factor in network design, it is justifiable to discard cryptographic solutions and start considering other mechanisms that are more energy efficient, such as trust models.

In any case, to the best of our knowledge, the security models for WSNs presented do not correctly address MAC unfairness and most of DoS attacks targeted at the IEEE 802.15.4 MAC protocol. It is clear that security models placed on upper layers won't completely solve MAC sublayer vulnerabilities, hence the need for a security protocol specific to this layer.

V. A trust based approach to DoS attacks mitigation in WSNs

The use of computational trust models in Ad-Hoc networks has been extensively researched, yielding interesting results. Many trust based routing protocols reliably construct and maintain routes, being resilient to failures and attacks without the cryptography and authentication overheads. By evaluating node reputation through statistical models (which are less computationally intensive than cryptographic functions) it is possible to determine a trust value related to the node's past behaviour in the network, detecting malicious nodes and enabling other security mechanisms to take actions against them.

Communication trust is defined as the trust value calculated by nodes based on their cooperation in routing messages around the network [12]. Momani et al. also define data trust, a new concept regarding the trustworthiness of sensed data, which is extremely relevant in WSNs. Applying such concepts directly to the MAC sublayer won't completely solve MAC unfairness but, at least, minimize attack feasibility and impact.

Processing MAC sublayer operational data such as CSMA-CA completion status with the Beta Reputation System [13] combined with a communication trust model enables the PAN and local coordinators of a IEEE 802.15.4 based WSN to determine if a node is unfair or malicious and take defensive actions against it. We propose a Bayesian trust model where coordinators receive MAC sub-layer operational data from sensor nodes and compare it with past information collected over time to calculate a node's trust value. By cross-referencing data collected from different nodes in different points of the network it is possible to obtain reliable information about a potential malicious node. This approach reduces the processing

load in the nodes, making the coordinators responsible for most of the calculations.

This model is able to detect MAC unfairness and the offending nodes, so that it can be used to thwart attacks where a node tries to gain control of the transmission media in order to increase its bandwidth or simply disrupt legitimate communication. However, it should be combined with a method of authentication for certain messages sent by the nodes to the coordinator (the MLME status messages that will be further discussed in the next session) to make it impossible for malicious nodes to forge such messages causing honest node's trust values to decrease. This model could also be used as a basis for a GTS allocation algorithm, providing information regarding the GTS allocation history of all nodes.

A. Protocol Modifications: MLME Status Reporting

Nodes must report the status of the MLME-COMM-STATUS.indication and MLME-GTS.confirm primitives issued by the MLME to the PAN Coordinator after each channel access and GTS requests. A node keeps two records regarding interactions, namely *Negative_Interactions* (*Neg_Int*) and *Positive_Interactions* (*Pos_Int*), that it will report to the Coordinator. This minor addition to the protocol enables the Coordinator to calculate trust values based on the behavior of all nodes in the network and also transfers processing loads from the nodes to the Coordinator, which usually has more resources. It is important to notice that these modifications may be implemented in the Application Layer through the use of SCSS Layer functionality, making it easy to adapt current networks to operate with this trust model.

In beacon-enabled mode a node increments the *Neg_Int* record if it receives MLME-COMM-STATUS.indication or MLME-GTS.confirm primitives with a status of CHANNEL_ACCESS_FAILURE or DENIED and increments the *Pos_Int* record if these primitives have any other status values. It will wait for a random period and try to report the interactions records each time a beacon is received. If the report transmission succeeds then both records are cleared whereas if the node is unable to send the report it will keep incrementing the records and will wait for the next beacon before retrying, as illustrated in Figure 1 (A).

In non-beacon-enabled mode a node increments the *Neg_Int* record if it receives MLME-COMM-STATUS.indication primitive with a status of CHANNEL_ACCESS_FAILURE and increments the *Pos_Int* record if this primitive has any other status values. In order to keep the node (loosely) synchronized with the coordinator, both will keep a timer T_L (started and adjusted by the node at the moment it joins the network) and the node will send its reports after periods of P_C time units, where P_C is a predefined constant. If the node can't send the report it will keep incrementing the records and will wait for P_C time units before retrying, as illustrated in Figure 1 (B).

The coordinator receives the reports and marks them with a *time stamp* before analyzing the information. It uses the current beacon sequence number as the *time stamp* in beacon-enabled mode and the current time (based on T_L) in non-beacon-enabled mode respectively. The reports are temporarily stored in the following format:

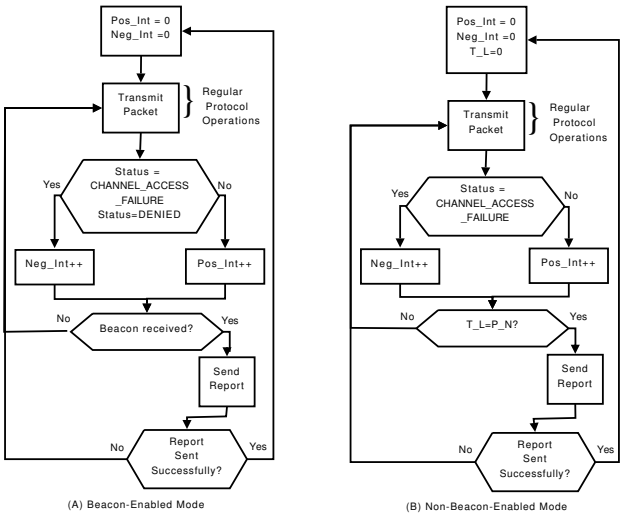

Fig. 1. MLME Status Reporting Algorithm

time stamp	*Neg_Int*	*Pos_Int*
1 Octets	2 Octets	2 Octets

TABLE I
REPORT FORMAT

B. The Proposed Trust Model for WSNs

Using the Beta Reputation System [13] and the concept of Communication Trust for WSNs presented in [12] it is possible to define a model where the reputation of node N_i maintained by the PAN Coordinator C is: $R_{Ci} = Beta(\alpha_{Ci}+1, \beta_{Ci}+1)$, where α_{Ci} and β_{Ci} are respectively the number of positive and negative transactions a node N_i had with the other nodes of the network as seen by the coordinator and $Beta(\alpha, \beta)$ is the Beta function (Euler integral of the first kind). The values α_{Ai} and β_{Ai} represent respectively the number of positive and negative transactions a node N_i had with the other nodes of the network as seen by the nodes $N_{j\neq i}$. In this context, positive transactions represent fair protocol operation while negative transactions represent potentially malicious or unfair protocol operation. Thus, the trust between the PAN coordinator and the nodes T_{Ci} is defined as:

$$T_{Ci} = E(R_{Ci}) = E[Beta(\alpha_{Ci}+1, \beta_{Ci}+1)] = \frac{\alpha_{Ci}+1}{\alpha_{Ci}+\beta_{Ci}+2} \quad (1)$$

When a new node joins the PAN, the Coordinator sets $\beta_{Ai} = \alpha_{Ai} = 0.5 = \beta_{Ci} = \alpha_{Ci}$. This means that the probability of misbehavior for the new node is equal to the probability of honest behavior, as seen by either the coordinator and the other nodes. The coordinator waits for an adjustable period of time $P_C = t_2 - t_1$ and starts the trust update process. First the Coordinator calculates $Thres^i_S$ and $Thres^i_F$ using the reports R^t_i received from each node during the period P_C

and updates the variables α_{Ai} and β_{Ai}. $Sr_i^t > Thres_S^i \rightarrow \beta_{Ai} = \beta_{Ai}+1$: If a node's success rate is bigger than $Thres_S^i$, β_{Ai} is incremented. $Fr_i^t > Thres_F^i \rightarrow \alpha_{Ai} = \alpha_{Ai} + 1$: If a node's failure rate is bigger than $Thres_F^i$, α_{Ai} is incremented. The values of α_{Ai} and β_{Ai} are not modified otherwise. It is important to notice that, if a node has a high Failure Rate, it means that medium access for this node is being granted in an unfair way in comparison to the other nodes, and that a high Success Rate implies that the node gets channel access (or GTS allocation) more often than the other nodes.

Now we define the *Success Rate*, *Failure Rate* and threshold values $Thres_S^i$ and $Thres_F^i$. Once the PAN Coordinator has collected MLME Status Reports $R_i^t = (t, S_i^t, F_i^t)$ from a node $N_i, i \in \{1, ..., n\}$, where n is the number of nodes in the network (for the sake of simplicity, the node's address is represented by i), it has access to the following information: S_i^t and F_i^t, which respectively represent the successful and failed transactions (either GTS or Channel Access requests) during the period P_C, where $t_1 \leq t \leq t_2$ ($t \in P_C = [t_1, t_2]$). In order to prevent malicious nodes from cheating, the coordinator compares the number of packets successfully received from each node during P_C with the reported values S_i^t and uses the larger one for subsequent calculations (we denote by S_i^t this larger value). Using these values it's possible to determine the *Success Rate* Sr_i^t and *Failure Rate* Fr_i^t of a node N_i during the period P_C:

$$Sr_i^t = \frac{S_i^t}{S_i^t + F_i^t}, \text{ where } S_i^t, F_i^t \neq 0 \tag{2}$$

$$Fr_i^t = \frac{F_i^t}{S_i^t + F_i^t} = 1 - Sr_i^t, \text{ where } S_i^t, F_i^t \neq 0 \tag{3}$$

The threshold values $Thres_S^i$ and $Thres_F^i$ are defined as follows:

$$Thres_S^i = \sum_{i=1}^{n} \frac{Sr_i^t}{n} + \sqrt{\sum_{i=1}^{n} \frac{(Sr_i^t - \overline{Sr_i^t})^2}{n-1}} * T_{c_i} * C,\ t \in P_C \tag{4}$$

$$Thres_F^i = \sum_{i=1}^{n} \frac{Fr_i^t}{n} - \sqrt{\sum_{i=1}^{n} \frac{(Fr_i^t - \overline{Fr_i^t})^2}{n-1}} * T_{c_i} * C,\ t \in P_C \tag{5}$$

In the equations above $C \in [0, 1]$ is a convergence factor that controls how fast T_{c_i} changes over time, which constitutes an *aging factor* similarly to the approach introduced in [14]. The terms $\sqrt{\sum_{i=1}^{n} \frac{(Sr_i^t - \overline{Sr_i^t})^2}{n-1}} * T_{c_i} * C$ and $\sqrt{\sum_{i=1}^{n} \frac{(Fr_i^t - \overline{Fr_i^t})^2}{n-1}} * T_{c_i} * C$ represent the trust value multiplied by the convergence factor and the standard deviation of Sr_i^t and Fr_i^t respectively. These threshold values are compared to the node's success and failure rates in the process of determining if a node is being unfair.

After the α_{Ai} and β_{Ai} variables are incremented the Coordinator can calculate the new trust value T_{Ci}^{new} for each node. The equations below are based on the model presented in [12] and were first given by [15]. They have been adapted to the trust model proposed in this section, where the central Coordinator maintains Trust and Reputation information about the ad-hoc nodes.

$$\alpha_{Ci}^{new} = \alpha_{Ci} + \frac{2 * \alpha_{Ci} * \alpha_{Ai}}{(\beta_{Ci} + 2) * (\alpha_{Ai} + \beta_{Ai} + 2) + (2 * \alpha_{Ci})} \tag{6}$$

$$\beta_{Ci}^{new} = \beta_{Ci} + \frac{2 * \alpha_{Ci} * \beta_{Ai}}{(\beta_{Ci} + 2) * (\alpha_{Ai} + \beta_{Ai} + 2) + (2 * \alpha_{Ci})} \tag{7}$$

$$T_{Ci}^{new} = E(R_{Ci}^{new}) = E[Beta(\alpha_{Ci}^{new} + 1, \beta_{Ci}^{new} + 1)] = \\ = \frac{\alpha_{Ci}^{new} + 1}{\alpha_{Ci}^{new} + \beta_{Ci}^{new} + 2} \tag{8}$$

The value T_{Ci}^{new} represents the trust value of the node i as seen by the coordinator. Based on the trust value obtained, the PAN Coordinator can detect malicious and unfair nodes and take actions, such as: decide wether to allocate GTSs to a certain node or not, stop routing packets from unfair nodes or warn legitimate nodes and other coordinators about misbehaving nodes. We note that this protocol may be used as part of a GTS allocation policy, serving as a tool to predict and adjust the probability of GTS allocation by a specific node or to determine if certain nodes have higher GTS allocation success rates.

Because it's designed to analyse MAC sublayer information without needing to access to central routing statistics, the proposed trust model may be implemented as a distributed system between all the WSN coordinators in a large WSN, not only the PAN Coordinator. All the coordinators would share the computational loads and maintain Reputation of nodes they are responsible for, exchanging information about reputation and trust values of nodes in different zones only when needed. This characteristic makes this trust model scalable and resilient to single points of failure.

VI. Simulation Analysis and Evaluation

Simulation experiments where conducted to verify the theoretic model proposed above. The proposed trust model, a subset of the IEEE 802.15.4 protocol Messages and a Two-Ray ground reflection radio propagation model were implemented using numerical programming methods in order to simulate GTS allocation requests and media access contention in Beacon-enabled mode. In this experiments we consider a sensor network consisting of 10 nodes equally distant from the PAN Coordinator in star topology transmitting constant bit rates during intervals smaller than the reporting period P_C. The convergence factor is set to $c = 0.5$ and the trust values T_{c_i} between the coordinator and five of the network nodes are observed against a time interval of $1000 * P_C$ in order to verify the convergence speed and behavior of the trust model. Three different attack scenarios were simulated: where all nodes are honest, where a node performs the attack during all simulation (static adversary) and where a node (or more) performs the attack for a period $400 * P_C$ and then starts behaving honestly, changing its behavior over time during the simulation (dynamic adversary). The attack performed is

the one described in Section 3, where an offending node cheats when contending for medium access, provoking MAC unfairness and resulting in a DoS attack.

First we consider the case where all nodes are normal (Figure 2). The results show that T_{c_i} starts gradually increasing and stabilizes after some periods of oscillation in the beginning of the simulation, showing that the trust between the coordinator and the honest nodes is increasing over time. We note that it is possible to control the time taken by the model to stabilize and T_{c_i} to increase by adjusting the convergence factor C.

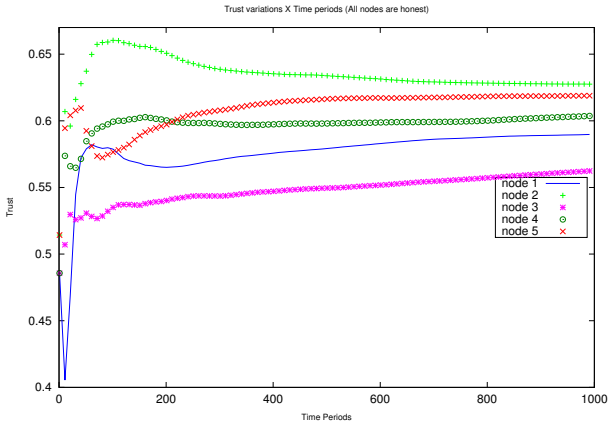

Fig. 2. Trust between the PAN Coordinator and the nodes T_{c_i} versus time periods P_C in the case where all nodes are honest

The results for the case of a static adversary are shown in Figure 3. It is showed that the behavior of T_{c_i} is similar to the first (honest) scenario for all nodes except node 5, which is the malicious node performing the attack. While $T_{c_1}, T_{c_2}, T_{c_3}, T_{c_4}$ increase gradually after a periods of stabilization, T_{c_1} decreases exponentially, making it is easy to identify the malicious node.

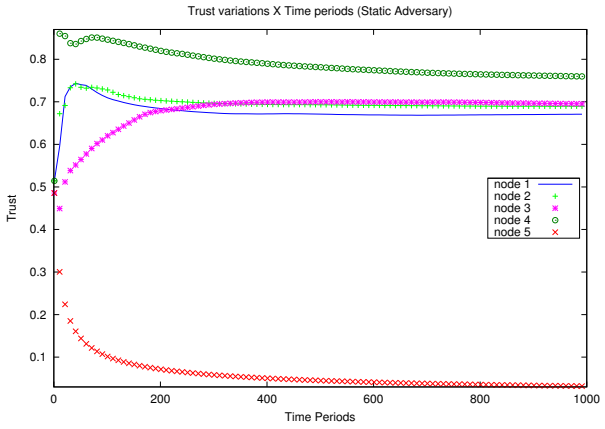

Fig. 3. Trust between the PAN Coordinator and the nodes T_{c_i} versus time periods P_C in the case of a node 5 being corrupt

The case of a dynamic adversary is shown in Figure 4. In this scenario we analyse the model's behavior in a more realistic dynamic setting, with nodes alternating between honest and malicious behavior in the course of the simulation. While nodes 2,3 and 4 behave honestly during the whole simulation, node 5 performs the attack until period 400 ($time = 400 * P_C$) and then starts behaving honestly. Conversely, node 1 behaves honestly until period 400 ($time = 400 * P_C$) and then starts performing the attack. It is possible to observe how T_{c_1} and T_{c_5} respectively increase and decrease after a reasonably large period of stabilization, and the effect of the previous reputation data collected on the convergence speed after the nodes change their behavior. After the instant $400 * P_c$, T_{c_1} decreases slower than it increased until $400 * P_c$, and so does T_{c_5}, that increases slower than it decreased before. This characteristic is inherent to bayesian statistical models, which take into account previous knowledge on the variable that is being estimated.

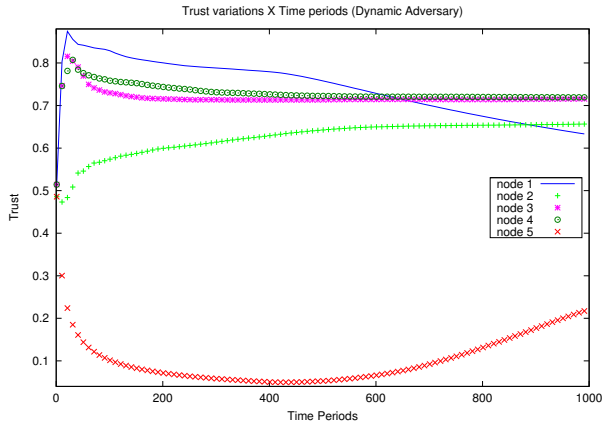

Fig. 4. Trust between the PAN coordinator and the nodes T_{c_i} versus time periods P_C in the case of dynamic adversaries that stops the attack at period 400

VII. Conclusion

Wireless sensor networks based on IEEE 802.15.4 are vulnerable to a number of DoS attacks and are subject to MAC unfairness issues. Current security solutions for WSNs are placed on upper layers and focused mainly on confidentiality and authenticity, leaving the MAC sublayer unprotected and overloading node's constrained resources with cryptographic calculations and packet overhead transmission. We introduce an attack that exploits this MAC unfairness vulnerability enabling malicious nodes to arbitrarily control the transmission medium and ultimately cause Denial-of-Service, disrupting legitimate communication. The bayesian trust model we propose is computationally efficient and mitigates MAC based DoS attacks with high probability and little resources expenses. Furthermore, it is adaptable to different scenarios and applications through simple parameters adjustment. As a future work this trust model could be generalized and adapted to other protocols and networks, also, an algorithm for intelligent adaptive adjustment of the trust model parameters and a more efficient ageing factor could be proposed. This trust model could also be adapted to serve as a component for a Multi-Layer trust model.

REFERENCES

[1] IEEE, "Ieee standard for information technology - telecommunications and information exchange between systems - local and metropolitan area networks specific requirements part 15.4: wireless medium access control (mac) and physical layer (phy) specifications for low-rate wireless personal area networks (lr-wpans)," *IEEE Std 802.15.4-2003*, pp. 1–670, 2003.

[2] J. Zheng and M. J. Lee, "A comprehensive performance study of ieee 802.15.4," *Sensor Network Operations*, pp. 218–237, 2006.

[3] A. D. Wood and J. A. Stankovic, "Denial of service in sensor networks," *Computer*, vol. 35, no. 10, pp. 54–62, 2002.

[4] A. Perrig, J. Stankovic, and D. Wagner, "Security in wireless sensor networks," *Commun. ACM*, vol. 47, no. 6, pp. 53–57, 2004.

[5] D. R. Raymond and S. F. Midkiff, "Denial-of-service in wireless sensor networks: Attacks and defenses," *IEEE Pervasive Computing*, vol. 7, no. 1, pp. 74–81, 2008.

[6] W. Xu, W. Trappe, Y. Zhang, and T. Wood, "The feasibility of launching and detecting jamming attacks in wireless networks," in *MobiHoc '05: Proceedings of the 6th ACM international symposium on Mobile ad hoc networking and computing*. New York, NY, USA: ACM, 2005, pp. 46–57.

[7] Z. Li, S. Nandi, and A. Gupta, "Achieving mac fairness in wireless ad-hoc networks using adaptive transmission control," *Computers and Communications, IEEE Symposium on*, vol. 1, pp. 176–181, 2004.

[8] C. E. Koksal, H. Kassab, and H. Balakrishnan, "An analysis of short-term fairness in wireless media access protocols (poster session)," *SIGMETRICS Perform. Eval. Rev.*, vol. 28, no. 1, pp. 118–119, 2000.

[9] J. Zheng, M. J. Lee, and M. Anshel, "Toward secure low rate wireless personal area networks," *IEEE Transactions on Mobile Computing*, vol. 5, no. 10, pp. 1361–1373, 2006.

[10] A. Perrig, R. Szewczyk, V. Wen, D. Culler, and J. D. Tygar, "Spins: Security protocols for sensor networks," in *Wireless Networks*, 2001, pp. 189–199.

[11] C. Karlof, N. Sastry, and D. Wagner, "Tinysec: a link layer security architecture for wireless sensor networks," in *SenSys '04: Proceedings of the 2nd international conference on Embedded networked sensor systems*. New York, NY, USA: ACM, 2004, pp. 162–175.

[12] M. Momani, S. Challa, and R. Alhmouz, "Can we trust trusted nodes in wireless sensor networks?" in *Computer and Communication Engineering, 2008. ICCCE 2008. International Conference on*, May 2008, pp. 1227–1232.

[13] A. Josang and R. Ismail, "The beta reputation system," in *In Proceedings of the 15th Bled Electronic Commerce Conference*, 2002.

[14] S. Ries, "Extending bayesian trust models regarding context-dependence and user friendly representation," in *SAC '09: Proceedings of the 2009 ACM symposium on Applied Computing*. New York, NY, USA: ACM, 2009, pp. 1294–1301.

[15] S. Ganeriwal and M. B. Srivastava, "Reputation-based framework for high integrity sensor networks," in *SASN '04: Proceedings of the 2nd ACM workshop on Security of ad hoc and sensor networks*. New York, NY, USA: ACM, 2004, pp. 66–77.