Security Threats and Intrusion Detection at the MAC Layer in 802.15.4 Sensor Networks

by

Jobaida Begum

A thesis submitted to the Faculty of Graduate Studies of
The University of Manitoba
in partial fulfillment of the requirements for the degree of

Master of Science

Department of Computer Science
The University of Manitoba
Winnipeg, Manitoba, Canada

March 2007

Copyright © 2007 by Jobaida Begum

THE UNIVERSITY OF MANITOBA

FACULTY OF GRADUATE STUDIES

*****

COPYRIGHT PERMISSION

SECURITY THREATS AND INTRUSION DETECTION AT THE MAC LAYER IN 802.15.4 SENSOR NETWORKS

BY

Jobaida Begum

A Thesis/Practicum submitted to the Faculty of Graduate Studies of The University of Manitoba in partial fulfillment of the requirement of the degree

Master Of Science

Jobaida Begum © 2007

Permission has been granted to the Library of the University of Manitoba to lend or sell copies of this thesis/practicum, to the National Library of Canada to microfilm this thesis and to lend or sell copies of the film, and to University Microfilms Inc. to publish an abstract of this thesis/practicum.

This reproduction or copy of this thesis has been made available by authority of the copyright owner solely for the purpose of private study and research, and may only be reproduced and copied as permitted by copyright laws or with express written authorization from the copyright owner.

Thesis advisor
Dr. Vojislav B. Mišić

Author
Jobaida Begum

Security Threats and Intrusion Detection at the MAC Layer in 802.15.4 Sensor Networks

Abstract

Security is a key concern in wireless sensor networks (WSN), despite the many challenges posed by the use of the wireless medium, battery power, and devices with limited computing and communication capabilities. In particular, the MAC layer of WSNs is vulnerable to security threats that cause denial of service in the network. In this thesis, we address the security problems in the MAC layer of WSNs. We investigate possible MAC layer attacks and study the impact of those attacks on the recently introduced IEEE 802.15.4 communication technology. We investigate the feasibility of a simple, traffic based intrusion detection method for detecting the anomalous activities in the MAC layer. In order to filter out some of the inherent variability of traffic arrival patterns, we use simple exponential averaging that minimizes storage and computational requirements. We propose measures to improve the accuracy of the detection algorithm, and investigate the performance of the algorithm under different traffic patterns of regular devices. Our simulation results show that the proposed intrusion detection method implemented in this manner operates quickly and efficiently, and thus provides improved security to WSN users.

Acknowledgements

I would like to begin by expressing my cordial appreciation to my research supervisor, Dr. Vojislav B. Mišić, for his continuous support in completing my thesis tasks and improving my research skills. I got enormous enthusiasm to build an insight and patience in any situation from Dr. Vojislav B. Mišić.

I would like to thank the Graduate Studies Committee in the Department of Computer Science for their constructive comments and suggestions on my thesis proposal. I would also like to thank the Department of Computer Science for providing me a teaching assistantship that helped me financially along the way of my study.

I would like to express my special thanks to my thesis committee members. I am also thankful to Dr. Ellen Liu. I got valuable knowledge about network simulation and the analysis of output while I was doing a Computer Network course 74.781 to fulfil my course requirements.

Finally, I could not finish my thesis without the blessing and the mercy of the Merciful and Almighty Allah (God) and He is worthy of all the praises.

JOBAIDA BEGUM

The University of Manitoba
March 2007

Dedication

This thesis is dedicated to my parents and my husband Tariqul Hoque.

Table of Contents

List of Figures

Chapter 1

Introduction

Sensor networks are composed of a large number of battery operated devices or nodes, which collaborate on one or more tasks defined by the sensing application [15]. Sensing applications are intended to provide periodic or event-based coverage of relevant physical phenomena or events sensed by the sensor nodes deployed within a defined area of interest. Sensor nodes collect, process, and possibly aggregate the desired information and send it to a collecting node, commonly known as the network sink or the base station. The sink analyzes the information it receives, makes appropriate decisions, and informs the application accordingly.

In many sensing tasks, sensor nodes are expected to perform for prolonged periods of time with little or no human intervention. Typically, sensor nodes are not easily replaceable or rechargeable, although much research work is done in these areas. Low cost, low power, multi-functional sensor devices are usually characterized by limited sensing, computation and wireless communication capabilities. Moreover, the sensor network should operate with little or no infrastructure, which is why relevant

protocols and algorithms must have the self-organizing capabilities as well. These protocols and algorithms need to effectively deal with the unique resource constraints and application requirements of the sensor networks [3, 15].

Wireless sensor networks (WSNs) offer a promising and wide spectrum of network applications that need viable economic solutions as well as large scale, real time data processing. WSNs are utilized in a wide range of military and civil applications such as protecting and monitoring safety, security of buildings, factory instrumentation and domestic infrastructures, pollution sensing, structural integrity monitoring, and battlefield surveillance [3, 8, 25, 31].

1.0.1 MAC protocols

Wireless sensor networks by nature are broadcast networks since nodes use a common wireless radio channel to communicate with each other. Since the access to the wireless medium is open to all, and any node from any location can transmit at any moment, the shared channel can suffer from problems such as media contention, bandwidth underutilization, and unfairness [3]. Therefore, appropriate rules, protocols, and algorithms, collectively namely the medium access control (MAC) mechanism, are required to ensure fair, effective, and efficient sharing of the radio channel among the network nodes. The MAC layer is normally considered as a sublayer of the data link layer in the network protocol stack [32].

The MAC protocols for wireless network could be broadly categorized into two groups, depending on the mechanism for avoiding collisions in the shared channel: scheduled and contention-based MAC protocols. Time-division multiple access

(TDMA), frequency-division multiple access (FDMA), and code-division multiple access (CDMA) are examples of scheduled MAC protocols. The basic purpose behind the scheduled MAC protocols is to provide contention-free access to the channel by assigning different sub-channels to individual nodes prior to actual transmission. The sub-channels can be distinguished by time, frequency or orthogonal codes. While schedule-based protocols avoid congestion, they also rely on the presence of a central scheduling authority, which is typically absent from sensor networks.

The other class of MAC protocols are contention based MAC protocols, in which individual nodes compete for access with one another using a predefined set of rules that aim to ensure fairness and maximize throughput. Some among the multitude of such protocols include carrier sense multiple access (CSMA) [17] and ALOHA [1], CSMA with collision avoidance (CSMA-CA) [11], bandwidth reservation with request-to-send (RTS)/clear-to-send (CTS) handshake mechanism, and their many variants. Typically, any node that wants to transmit a packet must first make sure that the medium is not busy. Since wireless devices cannot, in general, listen to their own transmissions (like wired devices can), collisions are detected indirectly, through the absence of appropriate acknowledgments received immediately upon packet transmission. While the contention-based protocols do not require a central scheduling authority, unique characteristics of sensor devices make it unlikely that traditional MAC protocols can be used for wireless sensor networks without appropriate modifications.

1.1 Security Concerns in Sensor Networks

Successful deployment of sensor network necessitates a thorough understanding of many facets of their design and operation, not the least important of which is security [22]. In many sensor network applications such as building security, public safety, monitoring critical infrastructure and mission critical military applications, security is essential to provide the expected functionality [29]. However, the resource constraints of sensing hardware, as well as the shared wireless medium used for communication, pose unique challenges in providing the desired level of security. Limited computational and communication capabilities mean that security must be based on simple and time efficient algorithms, which limits the applicability of the most secure algorithms for encryption, authentication, and access control . The openness of the wireless communication medium means that potential adversaries may easily gain access to network communication and compromise its confidentiality, availability, and other properties [28, 30]. Therefore, prevention-based security services, usually considered as the first line of defense against the attacker, are not always sufficient to guarantee security, and the ability to deal with security attacks and possible breaches must be present to ensure the functionality of a WSN even in the presence of adversaries.

Unlike wire-line networks, the sensor network environment poses many challenges in applying traditional security techniques [26, 31]. First, use of wireless links among the sensor devices makes the network susceptible to various attacks such as eavesdropping, replay attacks, active impersonation or jamming attack or useless packet injection. Second, as a sensor field may consist of hundreds or even thousands of

sensor nodes, it is difficult to provide end-to-end security mechanisms. Besides, intrusion and disruptions are more difficult to detect in WSN [22]. Third, sensor devices typically have limited energy at their disposal and the chips' computational power is limited, which restricts the choice of cryptographic techniques used to attain the desired level of privacy and integrity. Finally, in some environments, sensor devices may be subject to damage, or even physical capture, by a capable adversary. Also, to guarantee functional long term operation of the sensor devices in WSN, any security algorithm should be designed so that minimum memory and battery power is needed [30]. Therefore, the security issues in WSN need much attention.

In the sensor network environment, defending against the class of attacks conducted through the MAC layer is a particularly interesting task. Specially, the MAC layer protocol that rely on cooperative schemes of carrier sensing are vulnerable to different security threats. Most of the current MAC protocols for WSN concentrate on fair sharing of the single broadcast channel, which can only be gained when all the nodes in the network exactly conform to the MAC protocol [26, 29, 31]. However, an adversary may unfairly capture the channel for a significant amount of time, which may reduce the available resources for the regular nodes. Therefore, the MAC protocol may be especially exploited to cause disruptions in accessing the limited (in case of WSN) network services. Most attacks that occur through the MAC layer [29, 31] are attacks on availability of the shared channel resources. Conventional security mechanisms such as encryption, authentication and digital signature [15] do not suffice to make WSN resilient against the MAC layer attacks where an adversary captures the channel illegally or induces different malicious activities by being reprogrammed or

subverted [26]. In such cases, we should be able to detect the anomalous behavior through some detection or monitoring techniques. A typical detection technique monitors the network traffic to compare the current state to the normal, "baseline" data and detect any anomalous activity [29, 30] so that appropriate measures can be taken to reduce the impact of the attack or, in some cases, change network parameters to sidestep it.

1.2 Research Goals

Achieving an appropriate level of security in wireless sensor networks requires a careful analysis of the security requirements of the intended application. Among all the security threats, the attacks launched through the MAC layer for hindering availability of the shared wireless medium are critical in all networks [31]. These attacks disrupt the performance of a network, possibly to the point that the network collapses because of those attacks. Hence, it is important to identify all the possible MAC layer attacks on WSNs as effective and secure MAC layer protocol ensures the efficient and fair use of the shared wireless medium. As such, we need to investigate the current MAC protocols for sensor networks and then design the necessary steps to make them secured. It is also important to detect these attacks as soon as possible, so that appropriate countermeasures can be taken to mitigate the attacks or to ensure graceful degradation of network performance. A network traffic monitoring or intrusion detection technique using the MAC layer feature set could be helpful to maintain the desired level of channel access and minimize the waste of bandwidth and power resources. The features of MAC layer is minimal compare to the other

upper layer's features and hence, it gives us an advantage of designing an algorithm that helps to take into account of the energy limitation in sensor network.

In this research, we try to identify possible MAC layer attacks by a thorough analysis of an existing MAC layer protocol for WSN and propose a possible MAC layer intrusion detection technique. We have considered the IEEE 802.15.4 MAC protocol as an example protocol to identify the possible MAC layer attacks. Recently, IEEE has adopted the 802.15.4 standard for low bit-rate short-range Wireless Personal Area Networks (WPAN) built using small, cheap, energy-efficient, less complex, portable, or mobile devices [12] that require little or no infrastructure to operate. The industrial standard IEEE 802.15.4 includes many features for enabling low power consumption and low cost implementation, which is the basic requirement for wireless sensor network as well. Moreover, the standard offers a contention based CSMA-CA MAC protocol suitable for the WSN environment [26]. Therefore, this standard appears to be one of the potential MAC mechanism for WSNs [7, 10]. Hence, such a MAC protocol should be investigated and analyzed with reference to network operation, performance and more importantly, security. The analysis of the protocol in the context of security threats exhibits the possible MAC layer vulnerabilities in a 802.15.4 compliant WSN.

The type of network for which we have designed our security solution is a sensor network where the sensing application such as monitoring temperature, humidity, pressure, vehicular movement, etc. is mostly event-based. The IEEE 802.15.4 standard is aimed for a low rate WPAN (where maximum data rate is less than 250 kbps) with a very short transmission range (approximately 10 meters) and it can easily

cater to the needs of such applications.

The focus of my thesis are the MAC layer security threats of an 802.15.4 WSN with two major objectives:

1. Identify and classify possible attacks at the MAC layer, and examine the impact of those attacks on the performance of an IEEE 802.15.4 compliant WSN.

2. Investigate techniques for intrusion detection, develop a suitable technique to detect anomalous activities, and suggest possible defenses against such activities.

1.3 Thesis Organization

This thesis is organized as follows.

Chapter 2 provides the background knowledge on the operations, network association and security services of the IEEE 802.15.4 standard. In this chapter, we also point out some inconsistencies and omissions that may affect security aspects of the networks that adhere to the standard.

In Chapter 3, we describe the related work on the current MAC layer security in WSN and the intrusion detection method suggested for providing the MAC layer security in WSN.

Chapter 4 describes and classifies the MAC layer attacks on the IEEE 802.15.4 MAC protocol.

In Chapter 5, we present the impact of different MAC layer attacks on the performance of an IEEE 802.15.4 compliant network.

In Chapter 6, we discuss possible intrusion detection techniques and present a novel intrusion detection technique based on monitoring and averaging individual node packet arrival rates. We discuss the model and its assumptions and discuss the choice of parameter values needed to optimize the performance of the proposed approach. Performance is assessed through common variables such as the probability of false positives and false negatives, and the delay in detecting an attack; but also in more practical descriptors such as mean time to detect an attack, mean time between false alarms, and mean time to recover after detecting an attack.

Finally, we conclude the thesis with our research findings in Chapter 7.

Chapter 2

Overview of IEEE 802.15.4 Standard MAC and Its Security Services

In this chapter, we describe the basic operations and security services of the IEEE 802.15.4 MAC protocol [12]. The standard offers two types of channel access mechanisms, the beacon-enabled slotted CSMA-CA and the non-beacon enabled un-slotted CSMA-CA. In this section, we only describe the beacon-enabled slotted CSMA-CA as we have considered this channel access mechanism for our research. The un-slotted CSMA-CA mechanism is very similar to the IEEE 802.11 standard and it is simpler than the slotted CSMA-CA.

In December 2000, IEEE 802 working group 15 along with the two standards groups namely Zigbee and a HomeRF spinoff began the development of a low-rate wireless personal area network (LR-WPAN) standard, which is now called the 802.15.4 standard. This standard is considered to be a potential MAC protocol for enhancing wireless sensor network as WSNs are composed of small, low cost, low power and portable moving devices with the aim to provide a mechanism that will enable a low-data-rate wireless connectivity among such devices [6].

The IEEE 802.15.4 standard specifies the physical (PHY) and medium access control (MAC) sub layers of the network protocol stack. Different applications may use different higher layer protocols according to their requirements [6]. According to the standard, a central controller device known as the PAN coordinator is able to build a WPAN along with the other devices within a small physical operating space (POS) [12]. In IEEE 802.15.4 compliant WPAN, a device may be a full function device (FFD) or a reduced function device (RFD). An FFD includes all the details of the MAC services whereas a RFD is a very simple device having a reduced set of MAC services and it is allowed to talk to a single FFD at a time. The PAN coordinator of a WPAN must be an FFD. As the FFD forms a network with a new PAN identifier, it becomes the PAN coordinator. Once the network is built, any other RFD and FFD may join (association) or leave (disassociation) the network. During the association process, the coordinator assigns a logical network specific address, which will be used later in all communications with the PAN coordinator [12]. An 802.15.4 compliant network can be of two different types such as star topology network and peer-to-peer topology network. In star network, all communications among the devices must be routed through the PAN coordinator. On the contrary, in peer-to-peer network, two devices within the same POS can communicate with each other directly. However, the PAN coordinator must be present to synchronize the communications.

Moreover, as mentioned earlier, in each of these network topologies, there can be two types of channel access mechanisms such as beacon-enabled slotted CSMA-CA and non-beacon enabled un-slotted CSMA-CA. In beacon-enabled slotted CSMA-CA, the PAN coordinator sends a beacon frame (packet) in predetermined intervals.

Here, the channel time is divided into superframes where each superframe starts with each transmission of a beacon frame from the PAN coordinator. The superframe may consist of an active and an inactive portion as shown in Figure 2.1. All communications among the devices occur during the active portion whereas during the inactive portions, the devices may enter a sleep mode. The active portion is divided into equally sized time slots and composed of three consecutive segments: Beacon, Contention Access Period (CAP), and Contention Free Period (CFP) [12]. Each time slot is again divided into a number of backoff periods (three in the basic configuration), the basic time units for synchronizing the data transmissions in the network. The actual duration of the backoff period unit depends on the frequency band the standard chooses to operate.

Now, during the CAP period that starts right after the beacon (transmitted at the beginning of slot 0) each device has to use the slotted CSMA-CA algorithm to access the channel for sending packets. This is because the channel access is contention based throughout the CAP period – in other words, transmissions will be done only when the medium is clear and deferred otherwise. All contention based transmissions of any devices must be completed within this CAP period of the current superframe. On the other hand, the CFP is composed of one or more guaranteed time slots (GTS) that the PAN coordinator may reserve for some devices that need unimpeded access and cannot be left to compete with other devices during the CAP.

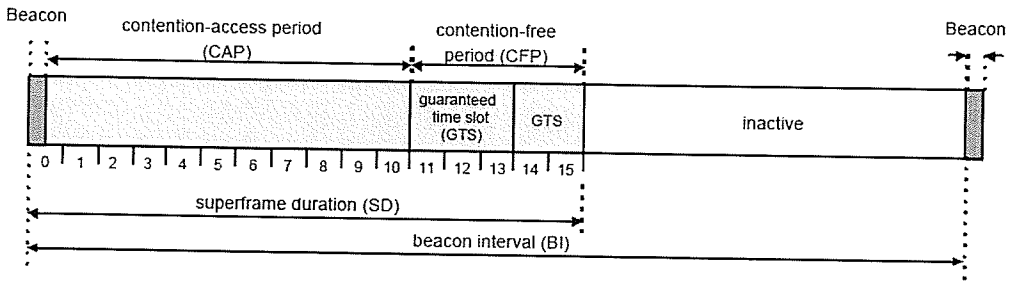


Figure 2.1: The composition of the superframe in IEEE 802.15.4 (adapted from [12])

2.1 The Slotted CSMA-CA algorithm

A device starts executing the IEEE 802.15.4 CSMA-CA MAC algorithm when a packet is ready to be transmitted. Three counters are maintained in the algorithm to keep track of the random countdown and the number of channel assessment. The flowchart of the slotted CSMA-CA algorithm is shown in Figure 2.2. As we can see, the three variables NB, CW and BE should be maintained for each attempted packet transmission.

- NB represents the index of the backoff attempt, which is required when the medium is sensed to be busy. The initial value of NB is 0. This variable takes value from 0 to *macMaxCSMABackoff-1* where *macMaxCSMABackoff* is a constant defined by the standard.

- CW stands for "contention window" and in the algorithm, it represents the index (the number of backoff periods) of the clear channel assessment (CCA) process. The CCA is the process of listening to the channel to find out whether any other device is using the channel or not. According to the standard, any device has to perform two CCAs before sending a packet. This counter is

initialized to 2 and it takes values from 0,1 or 2.

- BE is used to hold the index of the number of backoff periods that a device has

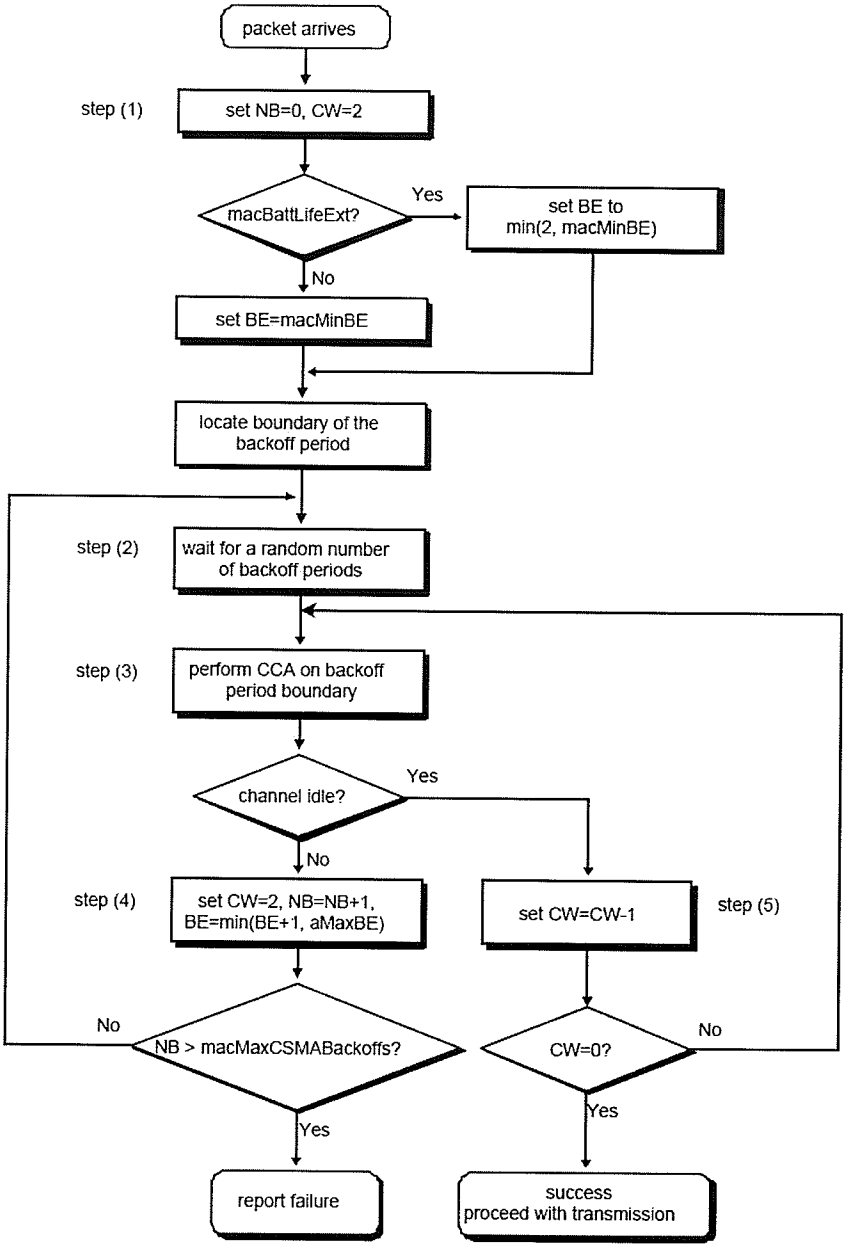


Figure 2.2: Operation of the beacon-enabled slotted CSMA-CA algorithm (adapted from [12])

to wait before performing the CCAs. As we can see from Figure 2.2, the initial value of this parameter depends on whether the device is operating on battery power or not. The standard uses the constant *macBattLifeExt* to indicate that the device operates on battery power. If so, the initial value of BE is set to the value of 2 or *macMinBE*, whichever is smaller. Otherwise, the value of BE is set to *macMinBE* and its default value is 3.

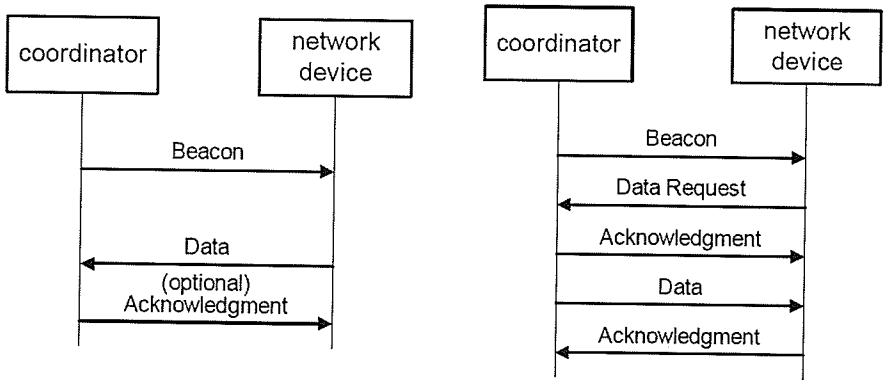
All the initializations of NB, CW and BE occur in step (1) of the algorithm. After all the initialization, the algorithm locates the next backoff boundary as all the operations of this algorithm must be synchronized to the backoff time units. In step (2), the algorithm generates a random number in the range of $0..2^{BE}$ - 1 backoff periods. A device that has a packet ready to transmit would wait for this number of random backoff periods before assessing the channel. In each subsequent backoff period, the counter that holds this random number will be decremented. As soon as it reaches zero, step (3), the device performs a CCA (listening to the channel) in next backoff period to make sure that the medium is clear that is no device is currently transmitting. Now, after the first CCA in step (3), if the channel is found busy the algorithm goes to step (4), which we describe in the next paragraph. Otherwise, the algorithm goes to step (5), the value of CW is decremented by 1 if the medium is found to be idle. The algorithm goes back to step (3) if CW is not equal to 0. Otherwise, it performs another CCA to see again if the channel is clear. If it is clear, CW is again decremented by 1. When CW reaches to 0, it means both the CCAs report an idle channel and so the device can begin its transmission at the next backoff period.

If the channel is found busy at the first CCA, then the algorithm goes to step (4) where the values of NB, BE are increased by 1, and the value of CW is set to 2 again. This step leads to repeat all the steps beginning from step (2). The difference now is the random time that the device has to wait before the CCA steps, is larger as the range $0..2^{BE}$ - 1 is larger because BE is increased. Also, any device can retry to access the channel if NB (the number of retries) is less the or equal to *macMaxCSMABackoffs* (the default value of which is 5). If NB is greater than the *macMaxCSMABackoffs*, then the algorithm terminates and reports failure status to the upper layer protocols.

As we mentioned before in this section that all the activities that includes the two CCAs, data packet transmission, waiting for the acknowledgement and getting the acknowledgement for the recently sent data packet must be completed with in the remaining CAP period of the current superframe. To ensure this, as soon as a device finishes the random countdown in step (2), it also calculates the remaining backoff periods within the current CAP period of the current superframe and check if it can accommodate all the activities. If the remaining time is not sufficient then the device pauses its activity until the next superframe begins and resume from step (3) right after the beacon frame.

2.2 The Case of Uplink and Downlink Transmission

As outlined above in the IEEE 802.15.4 beacon enable slotted CSMA-CA compliant network, any communication must go through the PAN coordinator. However,



(a) Uplink transmission (b) Downlink transmission

Figure 2.3: Uplink and downlink data transfers in beacon-enabled PAN [19]

the protocol used for data transfers in the uplink direction (i.e., from a node to the PAN coordinator) differs from the protocol for the transfers in the downlink (i.e., from the PAN coordinator to a node) direction. According to the standard, uplink communications, shown in Figure 2.3 (a), always follow the CSMA-CA algorithm, and the entire transfer, including an optional acknowledgment packet, must fit within the active portion of a single superframe.

In a downlink transmission, the PAN coordinator first announces the list of nodes that have pending downlink data (data from the coordinator or another device) by appending the list to the beacon frame. A device that learns through this list of the presence of a pending downlink data packet, must explicitly request this data frame by sending a data request frame to the coordinator. This transmission is done using CSMA-CA algorithm. Upon receiving the data request, the PAN coordinator sends an acknowledgement packet, and commences its own CSMA-CA algorithm in order

to send the downlink packet to the destination node. The message exchange in the downlink communication is shown in Figure 2.3 (b). After receiving the acknowledgement, the corresponding device will listen for the actual data packet for a fixed period of time which is defined by a MAC constant *aMaxFrameResponseTime*. The PAN coordinator uses the CSMA-CA MAC algorithm to get an access to the channel to send the packet to the device. However, the standard also suggest that the data frame could be sent by attaching it after the request acknowledgement packet, i.e., without using CSMA-CA. It will be possible only when the PAN coordinator is able to start the transmission of the data packet at the backoff period boundary of one backoff period. Also, there should be enough time remaining in the current superframe to accommodate the data, appropriate inter-frame spacing, and acknowledgement. According to the standard, the acknowledgements are sent only if it is explicitly requested by the transmitter and the transmitter must receive the acknowledgement within the prescribed time period. Otherwise, the whole process starting from the beacon announcement by the PAN coordinator has to be repeated. Moreover, a device would also wait for an acknowledgement for a specific amount of time and if it does not get an acknowledgement by this time then it will initiate retransmission of the same packet.

The authors in [21] identify the definite states that the PAN coordinator node and an arbitrary (non-coordinator) node will be in when we consider both uplink and downlink communications and this helps to better understand the activities in both uplink and downlink communication of such network. According to [21], the PAN coordinator may be in one of the following states:

- Transmitting the beacon.
- Listening to its nodes and receiving data or request packets.
- Transmitting the downlink data packet as a result of previously received MCRF packet. As soon as downlink transmission is finished coordinator switches to the listening mode.

On the other hand, an ordinary node in the network may be in one of the following states:

- Transmitting an uplink data packet.
- Transmitting an uplink request packet.
- Waiting for a downlink packet.
- Idle state without any downlink or uplink transmission pending or in progress.

2.3 Network Association

The manner in which IEEE 802.15.4-compliant networks are formed is another part of the specification that has farreaching security implications. Regardless of the topology, network formation is initiated by the upper layers of the protocol stack typically the network layer management entity [12]. A device may be an FFD which is able to establish its own network and become the PAN coordinator, or a RFD, which may only be a member of a PAN but not the coordinator. The FFD starts the network formation process by locating an appropriate channel to operate and choosing

a unique PAN identifier that is not currently being used by any other network. The coordinator then begins sending beacon frames to announce its presence to other devices. Upon scanning the available channels and identifying available PANs, a device, be it a RFD or FFD, may request to be join a PAN; this process is known as network association. If there are sufficient resources to accommodate the new device, the PAN coordinator may grant the request and assign a logical network specific address to the device. This address will be used in all subsequent communications between the device and the PAN coordinator [12]. The coordinator may request the device to disassociate from the network by sending the appropriate notification, which needs to be extracted as a regular downlink frame (described in Section 2.2). The device itself may disassociate from the PAN by sending the appropriate notification to the coordinator. Acknowledgment is not necessary in either case. The standard also defines the procedure to re-establish synchronization between the device and the PAN coordinator, should loss of synchronization occur over time. However, the standard contains no provisions to set up periodic checks for the presence of a device in the PAN (or absence thereof); such functions must be implemented through the application executing on the PAN coordinator.

2.4 Security Services and Suites

IEEE 802.15.4 standard specifies set of security suites that contains a set of operations to perform on the MAC frame that may or may not be able to protect against different level of attacks. Followings are a brief description of the security services specified by the standard [12]:

- Any device can maintain an Access Control List (ACL) of devices from which it wishes to receive data. Each entry in the ACL list corresponds to a trusted device. This mechanism is intended to filter out unauthorized communications.

- Data Encryption service helps a device encrypt a MAC frame payload using the key shared between two peers, or among a group of peers. If the key is to be shared between two peers, it is stored with each entry in the ACL list; otherwise, the key is stored as the default key. (The MAC layer provides the symmetric encryption security systems using application-provided key, or keys.) Thus, the device can make sure that the data cannot be read by devices that do not possess the corresponding key. However, device addresses are always transmitted in the clear (i.e., unencrypted), which makes attacks that rely on device identity somewhat easier to launch.

- Frame Integrity service ensures that a frame cannot be modified by a receiver device that does not share a key with the sender, by appending a message integrity code (MIC) generated from blocks of encrypted message text.

- Sequential freshness service uses the frame counter and key sequence counter that helps a device to make sure that the received frame is not replayed by a malicious node.

The use of the services listed above is optional [22]. A device can choose to operate in un-secured mode, secured mode, and ACL mode. In un-secure mode, none of the services mentioned above are available. In secured mode, the device may use one of the security suits supported by the standard [12], all of which use the Data

Encryption service explained above. A device operating in ACL mode can maintain a list of trusted devices from which it expects to receive packets, but the only security service available is access control service which enables the receiver to filter received frames according to the source address listed in the frame [22]. However, since no encryption is used in this mode, it is not possible to authenticate the true source of the data packet, or to ascertain that the packet payload has not been modified in any way. Also, we note that the procedures for key management, device authentication, and freshness protection are not specified by the standard; they are left to be implemented by the applications running on 802.15.4 devices.

2.5 Summary

In this Chapter, we have described the basic operations of the IEEE 802.15.4 MAC protocol [12]. We only describe the beacon-enabled slotted CSMA-CA mechanism as we have considered this channel access mechanism for our research. We also briefly describe the security services specified by the standard. In the next Chapter, we review the published literature that would help us to define the vision and the purpose of our work.

Chapter 3

Related Work

In this chapter, we start by discussing some background studies on the classification of different attacks on WSN. WSNs are vulnerable to quite a few security threats. Security threats of wireless network may be characterized by different attacks that affect the basic security requirements described in Section 1.1. However, these types of attacks may occur at different layers of the ISO/OSI model [22]. For example, denial of service (DoS) attack that refers to any activities that destroy a network's ability to perform its normal function [15, 26] might occur at multiple layers. Hence, we describe different attacks that may occur in WSN using a layered approach. We also describe different work that points out different countermeasures against different possible vulnerabilities in a WSN in Section 3.2. Also, in Section 3.2.2 we describe two intrusion detection algorithms that have been proposed for WSN's MAC layer security. Finally, in section 3.2.3 we describe some recent work that talks about the flaws of the security services provided by IEEE 802.15.4 standard.

3.1 Background: Attacks in Different Layers

Attacks categorized by the layer would help us to understand how a packet, as moving through different layers, is disposed to various security threats. In the following, typical wireless sensor network layers with their corresponding vulnerabilities are described.

- Physical layer attacks are typically known as the jamming attacks (also called DoS at physical layer). Also, the attacker may send radio frequency signals to interfere with an ongoing communication. The jamming attack created by sending enormous useless packets may drain the battery power of a node [2, 25].

- MAC layer attacks typically focus on disrupting channel access for regular nodes, thus disrupting the information flow both to and from the sensor node. The MAC layer provides a set of rules to access the wireless medium fairly. The adversaries can induce various attacks (by not following or by partially following the set of rules) to disturb channel access so that the regular nodes who follow the rules to access the medium cannot get the channel free. The adversary may create collision, may exhaust the battery life of a node or unfairly share the channel [25, 31]. Also, an adversary may send large size packets with useless information to create jam in the channel. All these attacks leads to different form of DoS attacks in the MAC layer. Through the sybil attack, a sybil node may capture a large fraction of the channel using multiple identities [23]. Using all the identities, the sybil node can send a large number of packets to keep a busy medium and hence the regular devices may not find the channel free for

their own use. The sybil node may or may not follow the MAC rules to access the shared medium. These attacks might also cause power resources depletion and bandwidth wastage [26] and on the whole, may diminish or eliminate a network's functionality or destroy a specific node.

- Routing or network layer attacks include spoofed, altered or replayed routing information spread by an adversary, selective forwarding of packets, sinkhole attacks which is an attack that attracts traffic from a specific area through compromised nodes, acknowledgement spoofing, sybil attacks in which a malicious node illegally takes more than one identities and therefore, this sybil node (a node that induces sybil attacks) can attract many network traffic to pass through the same node having different identities, and hence the sybil node can do selective forwarding of the packets, [15]. Attackers can inject malicious packets to create DoS attack in the network layer and hence might disrupt the routing function [25, 26]. Also, other forms of DoS attacks are possible as a node can arbitrarily neglect a routing information or forward messages along wrong paths, etc. [31].

- Finally, flooding attack and desynchronization are common attacks in transport layer [25, 31]. An adversary induces flooding attack by sending a stream of connection establishment requests to a victim node and thereby exhausts the resources of that victim node. In the desynchronization attack, a malicious node may disrupt an ongoing communication by repeatedly falsifying or forging messages or forcing retransmission from another node (or nodes).

Since the focus of this thesis is the study of security at the MAC layer, we will devote most of subsequent discussions to this particular area.

3.2 Security at the MAC Layer

Security at the MAC Layer has been mostly studied in the context of 802.11 MAC layer [9, 14, 26, 34] but sometimes also in the more general context of different types of attacks [5, 23, 31]. Some research in this area focuses on multi-layer security issues while some other tries to identify security vulnerabilities only in MAC layer or physical layer. Also, some research looks into a specific type of attack and analyze its effect in different layers.

WSN's security threats at different layers and possible countermeasures against these threats are described in [29]. The authors describe various attacks ranging from passive eavesdropping to active impersonation. To provide security solutions for the link layer or MAC layer jamming attacks that limits the network availability, the authors propose to use error correcting codes, rate limitation and small frames. The authors also mention that in case of node compromise, the network should be able to achieve graceful degradation by detecting the anomalous nodes as soon as possible and then isolate the malicious nodes (e.g., by revoking their network-wide secret keys) from all network communication.

Different types of DoS attacks at different layers are discussed in [31], using a particular sensor network protocol known as the real-time location-based protocol (RAP). RAP is a real-time communication architecture where the goal is to scale well with the large number of sensor nodes and hops as well as ensuring minimum

communication and processing overhead for the sensor nodes [18]. RAP specifies a set of convenient, high-level query and event services to real-time distributed micro-sensing applications. The authors in [18] proposed to use the IEEE 802.11 MAC protocol with some modification as the MAC layer protocol in their architecture. The authors in [31] investigate this protocol and find out that an adversary can induce different types of DoS attacks in a network where RAP protocol is used for communication. It is mentioned earlier that DoS attacks can also be induced by many sybil nodes in sensor network. Newsome et al. [23] analyze sybil attack and its defenses on WSN. They have shown how an attacker can use different types of sybil attacks to disrupt sensor network functionality. They propose a number of defense mechanisms such as radio resource testing, key validation, sensor node's position verification and registration against these attacks.

MAC layer security in mobile ad hoc networks (MANET) has been studied extensively, esp. when compared to the corresponding security issues in WSNs. For example, [9] studied both local and global effects of various types of DoS attacks on IEEE 802.11 MAC protocol. Also, Zhou et al. [34] discuss two different types of DoS attacks: namely, single adversary attack (SAA) and colluding adversary attack (CAA), and propose defenses against these attacks while they are introduced in IEEE 802.11 MAC layer protocol.

3.2.1 Intrusion Detection at the MAC Layer

The MAC layer of WSN are susceptible to many forms of intrusion. Although prevention based security solutions such as encryption and authentication are the

first line of defence against outsider attacks such as eavesdropping, packet replay attacks and modification or spoofing of packets, due to environment we cannot always guarantee the availability of WSN service and resources with the help of these security solutions. For example, when some normal (good) nodes are captured and reprogrammed by the adversary, they could access the network legally. Traditional security mechanisms such as cryptography do not cater to those attacks, as the subverted devices may still be in possession of the corresponding cryptographic keys, and thus may be able to continue their presence without alerting other nodes. Most of the times, these kinds of attacks are launched through the MAC layer (as discussed in Section 3.1) as link layer attacks exploits the medium access control protocol employed in the network [29]. Therefore, in addition to the security solutions as mentioned in Section 1.1, we need to rely on some form of intrusion detection technique to guarantee that potential adversaries will be detected as soon as possible, and appropriate measures taken to deal with the threat. Also, an intrusion detection system or an anomaly monitoring mechanism would help a system to achieve graceful degradation in the presence of such adversaries.

An intrusion detection technique discovers violations of confidentiality, integrity, and availability of information and resources within the network. Intrusion detection demands as much information as the computing resources can possibly collect and store. Therefore, by an intrusion detection system we want to maintain functionality of the network whereas the attackers want to disturb normal operation of the network.

Intrusion detection systems could be classified depending on their approach to detection; this functionality is almost the same in both wired and wireless network

[13]. The two possible approaches are commonly referred to as anomaly detection and signature detection.

- In an anomaly detection system, statistical data on regular traffic patterns is collected over time; intrusions are detected when one or more devices suddenly exhibit significant deviations from the established behavior. The implicit assumption in this kind of system is that the regular devices in the network generally exhibit stationary traffic patterns.

- Signature detection system monitor networks and host for known attacks patterns. Here, the detection system has to maintain a database of known attacks, which has to be updated frequently. This class of system is well suited to networks of highly dynamic nature.

Key performance measures of an intrusion detection technique include detection accuracy and responsiveness. An ideal technique should exhibit high accuracy, which leads to two separate and often conflicting goals: high probability of detecting real attacks, and low probability of detecting false attacks. There is an inherent trade-off between these two issues, and we have to choose an optimum balance between the two in order to ensure an efficient technique.

Little research has been done in the area of intrusion detection system for WSN compared to that of the wired network or even to that of the MANET. Anderson was the first to introduce the idea of doing anomaly detection by creating profiles of normal use and detecting deviations from those profiles [4].

The detection based security idea for MANET was first brought up by [33] which proposed statistical anomaly detection techniques. It also gives an overview of the

architectural differences of the intrusion detection system between wireline networks and the ad-hoc networks.

Since then, many algorithms have been proposed for traditional wireless ad hoc networks, both static and mobile. Unfortunately, most of these algorithms are not well suited to the unique features—application requirements and device limitations—inherent to sensor networks. For example, looking for anomalies is often expensive in WSN in terms of memory, energy consumption, and/or bandwidth. Therefore, in a WSN, an intrusion detection algorithm has to work with small amounts of data as well as low complexity computation because of resource limitations of sensor devices.

3.2.2 Approaches to Intrusion Detection

As a result, there is not much published work on general intrusion detection techniques for wireless sensor networks. Most authors have focused on detecting a specific kind of attacks such as wormhole attacks, routing holes, DoS attacks, or to particular operations, like routing, localization, etc. However, since this thesis focuses on MAC layer security in wireless sensor networks, we only mention the work that focuses on the intrusion detection method designed only for the security of the MAC layer of WSN. In the following, we describe two intrusion detection algorithms that we have chosen as the starting point for designing our intrusion detection algorithm.

Ren *et al.* analyze different MAC layer security problems such as collision attack, unfairness attack and exhaustion attack, using RTS/CTS-based MAC protocols [26]. In RTS/CTS-based MAC protocols, a node first transmits a RTS packet to request the channel when it has a packet ready to send. The receiver replies with a CTS

packet if the channel is available. Upon receiving the CTS packet successfully, the sender sends the actual data. The nodes that hear the CTS packet will defer their transmissions for a sufficiently long period of time so that the receiver can receive the entire data packet without interference. The intrusion detection system proposed in [26] uses fuzzy logic rules based on different traffic factors such as collision ratio, average waiting time, and average RTS arrival ratio. An intrusion is detected upon a violation of these rules. The authors also propose a defense module which will be triggered after an intrusion is detected. The defense mechanism forces the nodes to switch to sleep mode for some period when it finds intruders.

A real-time node-based anomaly detection technique for WSN is proposed in [24]. The technique is based on observing the arrival processes experienced by a sensor node. They develop a scheme to detect anomalous changes in the arrival process of a sensor node, using a packet-based sliding windows. At every node, two buffers (named as the intrusion buffer and the receive buffer) are employed to maintain short and long term statistics, respectively, of the arrival history of the neighbors of each node. The short term statistics keeps the averages of the N_1 number of recent packets whereas the long term statistics keeps the averages of the last N_2 (where $N_2 > N_1$) number of packets. When a packet arrives at a node, it enters into the short term (intrusion) buffer, from which the oldest packet is dropped. A new statistic about the updated buffer is calculated and compared with the statistics of the long term (receive) buffer.The comparison is done as follows:

$$\mid mean(receivebuffer) - mean(intrusionbuffer) \mid > K * standarddeviation(receivebuffer)$$

In other words, the absolute value of the difference of mean interarrival times at the long and short term buffers should be less than K standard deviation of the interarrival times in the receive buffer, and hence, an intrusion is detected. Since the receive buffer is longer, the changes in the behavior of neighbors due to anomalous activity are more salient in the intrusion buffer.

If the comparison does not reveal any anomalous activity, the packet is considered as normal and entered into the receive buffer (the oldest packet is dropped from this buffer), and new statistics are calculated.

The choice of the proper values for N_1, N_2 and K, all of which depend on the security vulnerabilities and available resources will ultimately affect the performance of the technique: both success probability and the probability of false alarms, but also the detection time. For example, small K values lead to shorter detection time and high detection probabilities, but also to high probability of false alarms; and longer window in the intrusion buffer (i.e., higher values of N_1) lead to a decrease in false alarm rate.

3.2.3 IEEE 802.15.4-specific work

Since this is a very new standard not much work has been done on its security aspects. Sastry *et al.* concentrate on the cryptographic key management, initialization vector (IV) management and integrity protection technique of the security suites [28]. The authors mention that there are significant pitfalls and vulnerabilities if we apply symmetric key cryptography using the suggested security suites such as AES-CTR, AES-CBC-MAC or AES-CCM. They also mention that the shared network keys are

not suitable for replay attack protection. Also, pairwise shared key is not well supported because of the limited number of ACL entries that a device can store. The authors suggest not to use the AES-CTR security suite for any kind of encryption as this suite does not offer an option for authentication, which may create DoS. Also, some work has been done on performance implications of a ZigBee-prescribed key exchange mechanism [16].

3.3 Discussion

In WSN, the security techniques, whether it is an intrusion detection algorithm or a protocol has to be designed for specific applications, and they interact closely with their physical environments. As we know, that security mechanisms used for wired networks do not transfer directly to sensor networks since there is not a person controlling each of the nodes, and even more importantly, energy is a scarce resource as well as batteries have a short lifetime and cannot be easily replaced on deployed sensor nodes. Now, the questions are how should we start to secure a WSN? What type of intrusion monitoring technique can be used to detect the intruder in a shared wireless medium? Do all the potential current WSN protocols have been studied to address the security concerns in such network?

With the insight on the current research on security of WSN, we have identified some potential approaches to proceed to provide security in a WSN application while we are trying to answer the above questions. Firstly, we learn that to observe the effect in the WSN activities caused by various attacks at different layers, an existing suitable protocol needs to be considered. The implementation of a protocol and proper

experiments demonstrate the destructive impact of the various attacks in WSN. Since in WSN the media is open to all and it is more easily prone to the attacks that could be launched through the MAC layer compare to the other layers, we choose to analyze a MAC layer WSN protocol (we already mentioned our goal of the research in Section 1.2). Moreover, MAC layer is close to the bottom of the network protocol stack and therefore, most node misbehavior will have a direct impact on its operation. Wood and Stankovic in [31] describes how a sensor network protocol namely RAP could be exploited by an adversary to induce different kinds of DoS attacks. However, the protocol itself does not consider any power consumption aspects of sensor devices as well as it does not suggest a CSMA-CA mechanism as a MAC layer protocol. Hence, Wood and Stankovic while investigate the RAP protocol with respect to security, they do not actually investigate a possible MAC layer protocol in sensor network. Secondly, we have to detect these attacks at multiple layers using only the existing system information. Fortunately, the information used at different layers are independent of each other and this gives us a freedom of choosing different combinations depending upon need and resource availability. Although, the security in sensor networks has been studied in recent research, few of these studies have focused on any particular WSN protocol for ensuring various security services. Moreover, most of the research in WSN focuses on overall security threats that may arise when using a wireless medium, rather than studying security threats in a specific layer such as the MAC layer. Here, we shortly summarize the related work (that we already described in the previous section) on the intrusion detection techniques designed only for resource scarce WSNs. Ren et al. in [26] proposes an intrusion detection technique to

detect different attacks in the MAC layer of WSN, it is only designed for a RTS/CTS based MAC protocol. The paper did not mention any circumstances that may arise if we use it for a CSMA-CA based MAC protocol, which we chose as our example MAC protocol. [24] proposed a real-time node-based anomaly detection technique for WSN. The technique is able to identify an intruder impersonating a legitimate neighbor by keeping a relatively small number of arrival statistics and hence suitable to be implemented in resource constrained sensor nodes. The algorithm avoids any computationally expensive calculations for finding the intrusions. However, the algorithm uses different length packet buffers to calculate the statistics, which is also a matter while considering a WSN application with limited memory capabilities. We have to find some options where we can use the least of the limited memory or resources while giving an optimum security solutions.

Finally, we should try to come up with possible countermeasures once we detect the attacker or the attacks so that the network ends up with a graceful degradation in some cases or recover totally in some other cases.

For our research, we choose the recent protocol IEEE 802.15.4 proposed for short range wireless personal area network built from small, energy-efficient devices operating on battery power and hence, it is mainly designed to enhance wireless sensor networks. We analyze the IEEE 802.15.4 MAC layer protocol to identify the possible MAC layer attacks based on the classification of the attacks that is discussed in Section 3.1. To the best our knowledge, the recent standard IEEE 802.15.4 has not been analyzed in general security context. Rather, Sastry et al. analyzes the 802.15.4 protocol for verifying the robustness of the cryptographic security solutions that the

protocol suggests [28]. The observations made by these authors on the 802.15.4 protocol are described at the end of Section 3.2.

For our intrusion detection method, we investigate the use of MAC layer traffic data to characterize the normal behaviors in the neighborhood of a sensor node and to detect misbehavior through the MAC layer anomalies. Also, we adapted the arrival process based statistical framework that is suggested in [24] as the algorithm suggested an arrival statistics calculation that is simple enough to be implemented in the network built using the IEEE 802.15.4 standard. However, we do not use the buffer feature comparison that the author suggested to use in [24]. Instead, we design a Exponential Weighted Moving Average based comparison between different arrival statistics at different backoff period for the all the network nodes in the network. The detail of this is described in Chapter 6. Using EWMA based intrusion detection technique we want to make sure that we use the least amount of memory consumption, which is often a limitation for a sensor nodes. We also want to mention here that the IEEE 802.15.4 standard has its own security specifications that is able to give the first line of defense to ensure message integrity, message authenticity and confidentiality. However, since we are talking about an wireless medium , it is not possible to ensure availability (another security criteria) using these security specifications when an intruder establishes itself as a legitimate node in the network and hence, captures the channel by sending huge packets or by propagating false alarms. We analyze the standard along with the given security specifications to find out the security vulnerabilities even if the security is enabled in the existing network. We discuss the attacks that could be launched by an attacker that follows or does not

follow or partially follows the IEEE 802.15.4 MAC protocol in the next chapter.

Chapter 4

Security Threats in IEEE 802.15.4 Compliant Network

Attacks in wireless networks can be broadly classified in two categories. First, the attacker can follow the rules of the 802.15.4 slotted CSMA-CA MAC protocol, either fully or only to a certain extent, and launch an attack. Second, the attacker can choose not to follow the 802.15.4 protocol, which results in potentially more dangerous attacks; unfortunately, defence against them is much more difficult, as might be expected. In the latter case, the attacker can use a separate 802.15.4-compliant device, possibly modified to loosed the adherence to the MAC protocol; alternatively, an existing 802.15.4 device may be captured and subverted so as to be used for malicious purposes.

In this Chapter, we present a classification of the attacks that can be launched against an IEEE 802.15.4 beacon-enabled, slotted CSMA-CA based network. Note that a preliminary version of these findings was published in [22].

4.1 Attacks that follow the MAC protocol

A number of attacks may be conducted by an adversary which follows the IEEE 802.15.4 slotted CSMA-CA protocol to the letter. Of course, the adversary must act as a legitimate member of the PAN.

A simple attack against network availability is to flood the network by simply transmitting a large number of packets. Packets should be large in size, perhaps the largest size allowed by the standard. In this manner, an adversary may degrade the network performance and drastically reduce throughput; A previous work indicates that the performance of an 802.15.4 network can be seriously affected by high packet arrival rates or by nodes operating in saturation regime [19].

An adversary may target different destination devices with unnecessary packets, possibly in other PANs, regardless of whether the destination PAN and/or device actually exist or not. If the goal of the attack is the depletion of the power source for a specific node (and the PAN coordinator), all injected packets may target that node. Since the downlink packets have to be explicitly requested from the PAN coordinator, this will keep the both the PAN coordinator and the chosen destination device busy and eventually exhaust their respective power sources.

A malicious node can simply pretend to run in battery life extension mode, by setting the *aMacBattLifeExt* variable to true. In that case, the CSMA-CA algorithm will choose the initial value of the backoff exponent as 2 instead of 3, as explained in Section 2.1, and the random number for the backoff countdown will be in the range 0..3 rather than in the range 0...7. Shorter backoff countdown means that the probability to access the medium is much higher than for a regular node. On top of

that, a regular node would have to wait for the malicious one to finish its transmission, and waste power in the process.

Note also that the node that succeeds in getting access to the medium will not increase its backoff exponent for the next transmission, while the unsuccessful one will increase it by one. Therefore, if the first attempt succeeded, the second one is even more likely to do so, which again clearly favors malicious nodes.

It should be noted that power consumption during packet reception is about one-half to two-thirds of the corresponding power consumption required for packet transmission [10]. Therefore, while transmitting does consume lots of energy (relatively speaking), receiving is not terribly efficient either, and the best way to conserve power is to turn off the radio subsystem whenever possible.

4.2 Attacks that use a modified MAC protocol

The attacks mentioned above can be conducted using a completely functional 802.15.4 device that entirely follows the protocol and it suffices for the malicious node to control the application that executes on the sensor device. However, a number of additional attacks may be launched by simply modifying or disregarding certain features of the protocol. This can be accomplished either through dedicated hardware or by controlling an otherwise fully compliant 802.15.4 hardware device, as follows.

We have already mentioned the possibility to decrease the backoff exponent and shorten the random backoff countdown, by falsely reporting that battery life extension is enabled. Similar effects may be achieved by not incrementing the backoff exponent after an unsuccessful transmission attempt.

The random number generator can be modified to give preference to shorter back-off countdowns. Again, this allows the malicious node to capture the channel in a disproportionately high number of cases, and gives it an unfair advantage over regular nodes.

The number of required CCA attempts can be reduced to one instead of two, which would give the malicious node an unfair advantage over the regular nodes.

The CCA check can be omitted altogether, in which case the node will start transmitting immediately after finishing the random backoff countdown. This case, in fact, corresponds to a node that follows the non-beacon enabled version of the 802.15.4 MAC protocol [12].

Even worse, the node can omit the random backoff countdown itself. In this manner, the malicious node can transmit as soon as they are generated, and certainly more often than a regular one. While not all of the messages will be sent successfully–there will be collisions in many cases–the malicious node probably doesn't even care, as long as the transmissions from regular nodes end up garbled and thus have to be repeated. Moreover, some of the attackers transmissions may collide with acknowledgment packets and/or uplink data request packets. Again, this wastes the bandwidth of the entire network as well as the power of the devices affected, and thus shortens the lifetime of the network whilst reducing its efficiency.

In case the acknowledgment is requested by the data frame or the beacon frame, a malicious node may simply refuse to send it. The PAN coordinator will retry transmission (up to a maximum of aMaxFrameRetries) and thus waste power and bandwidth.

4.3 Attacks in which the attacker does not follow the MAC protocol

Finally, an adversary with appropriate resources might develop and use dedicated hardware which is compatible but not compliant with the 802.15.4 standard. In other words, the attacker would need a dedicated radio subsystem that is compatible with the PHY (and, to a certain extent, MAC) layer operation, yet not fully adherent to the rules of the 802.15.4 protocol. The availability of such hardware allows for a number of different attacks at varying level of sophistication, with or without the ability to impersonate legitimate nodes.

As explained in Section 2.2, the addresses of the nodes that have pending downlink packets are announced within the beacon frame which all nodes receive. As this list is not encrypted, the adversary may learn which nodes have downlink packets, and it may send a data request packet posing as a legitimate node. With slight alteration of the MAC algorithm, as outlined in the previous subsection, the adversary may almost always succeed in sending the request before the legitimate destination node. The PAN coordinator, upon receiving the request, will proceed to send the downlink packet, which the adversary may even acknowledge; the coordinator will consider this transmission successful and delete the packet from its buffer. The legitimate destination device will not listen for the downlink packet before getting its request acknowledged by the coordinator, and it may miss the actual transmission while attempting in vain to send the data request packet. Therefore, the damage, in this case, includes not only wasted bandwidth and power, but also loss of information.

An adversary may try to destroy legitimate traffic by injecting packets of its own

with the objective to garble the legitimate packets and thus either destroy information or cause retransmission.

Since the packet headers are always transmitted in the clear and encryption protects the packet payload only, the attacker may figure out whether an acknowledgment is requested. If so, the jamming packet may be transmitted simultaneously with the regular acknowledgment packet. The absence of acknowledgment will cause the sender to repeat the transmission, which means that the original packet transmission is wasted; again, the long term target of such attacks is the power supply of the regular nodes. We note that the efficiency of such attacks is better since the acknowledgment packets are shorter than the regular data packets, and the attacker can achieve the desired objective (i.e., cause retransmission) with smaller energy expenditure.

On the other hand, if the acknowledgment is not requested, the attacker must start its transmission immediately. In this case, the attacker needs a fast radio subsystem, since switching from reception to transmission takes time (we assume that the attacking device has only one such subsystem). Even when equipped with a fast radio, the success of this kind of attack depends on the length of the legitimate data packet. Shorter data packets may actually finish before the attacker can start a transmission of its own, and thus are more resilient to this kind of attack.

A particularly attractive target in beacon-enabled PANs is the beacon frame itself, which is periodically sent by the PAN coordinator for synchronization and other purposes. A jamming packet sent at the precise time can collide with the beacon and thus disrupt the normal operation of the WPAN for prolonged periods of time.

4.4 Summary

In this Chapter, we have seen the possible MAC layer attacks that might be induced by an attacker while the IEEE 802.15.4 CSMA-CA MAC protocol is executed for accessing the open shared channel. In the next Chapter, we present the effects of some of these simulated attacks, which is depicted for the regular devices in the network.

Chapter 5

Impact of Different Attacks in IEEE 802.15.4 Compliant Network

In this Chapter, we investigate the impact of the attacks outlined in the previous Chapter on the performance of 802.15.4 networks.

To that end, we have built a simulator of the IEEE 802.15.4 compliant network using the simulation engine called Artifex, which is based on the well-known Petri Net formalism [27]. Artifex is a comprehensive tool set for modeling and analyzing discrete event systems. It provides a graphical language environment with a rich set of facilities for building, testing, and executing simulation models of networks and other discrete event systems. Furthermore, developers can use their own code written in C or C++ to augment the functionality of the simulation model.

5.1 The Network Model

We have implemented a star topology network with a single PAN coordinator and 50 regular devices. The 802.15.4 network operates in a star topology with the network coordinator; it uses the beacon enabled, slotted CSMA-CA medium access mode [12].

This configuration appears better suited to sensor networks, where all ordinary nodes will report the sensed data to the cluster coordinator, to be delivered to the network sink. We have implemented the uplink data transmission only, where data transfers occur in the direction from ordinary nodes to the PAN coordinator. For simplicity, we focus on PANS that operate in the Industrial, Scientific, and Medical (ISM) band at 2.4GHz with raw data rate 250kpbs, since the other two bands offer only modest data rates of 20 and 40kbps, respectively, which may be inadequate for many sensing applications. We assume that the network formation phase has been conducted successfully: the PAN coordinator knows the number of devices surrounding it and their packet arrival rates.

In order to conduct a realistic simulation test, we assume that all the devices—ordinary nodes and the PAN coordinator alike—are equipped with small, finite buffers for temporary storage of packets that need to be transmitted. Newly arrived packets that find the buffer full (which may occur if the arrival rate is too high or there are several transmission retries due to excessive congestion, possibly as a result of malicious activity) will simply be discarded.

We note here that the 802.15.4 cluster is likely to operate under low to moderate loads, which will keep the congestion within acceptable limits and maximize the efficiency of data transmissions. We assume that all the devices in the network generated Poisson traffic with average arrival rate λ of 120 packets per minute. Some other parameters we define according to the standard are as follows and they all are expressed as a multiple of unit backoff periods (basic time unit in the standard):

- Packet Size for each device is 3 backoff unit.
- Length of the device buffer is 3 packets.

- Duration of an individual slot is 3 backoff unit.
- Basic superframe length is 16 individual slots, or 48 units of backoff periods.

The parameters of the CSMA-CA algorithm such as *BE*, *NB* and *CW*, are defined according to the standard (see Chapter 2 for details). As we mentioned earlier, in our network, we have 50 regular devices with individual source IDs and one PAN coordinator with a distinct coordinator ID.

5.2 Introducing the Attacker Node

In order to investigate the impact of malicious behavior, we introduce two attacker devices in the network with two distinct source IDs and we introduce attacks by these two devices. The intruder devices follow the MAC protocol, possibly only to a limited degree, and try to launch a denial of service attack in the uplink direction to the cluster coordinator. Such a scenario might occur if a malicious attacker deploys a number of sensor devices of its own within the area covered by a legitimate sensor network. Another possibility is for the attacker to capture and subvert a number of regular devices, which may occur in surveillance or military applications.

We assume that the packet arrivals for the attacker devices also use the Poisson traffic to generate variable number of packets per minute. Among all of the found attacks that we describe in Chapter 4, we implement only those attacks that inhibit the regular nodes from getting the free channel. That means in my simulator, I implement one of each of the MAC layer attacks that cause channel blocking by sending huge number of packets, sending large packet sizes, omitting CCA's, and omitting random countdown while executing the CSMA-CA algorithm. We have

described in the previous Chapter how these attacks harm the network (i.e the other nodes in the network). Now, to observe the impact of these attacks we calculate the following performance issues:

- Probability that the medium is idle on the first CCA (denoted as α): To calculate this, we have to find the mean number of busy backoff periods within the superframe; and then this number is divided into the total number of backoff periods in the superframe wherein the first CCA can occur. From this, we calculate the total number of idle first CCA's, which is divided by the total number of first CCAs to give us the α.

- Probability that the medium is idle on the second CCA (denoted as β): This is calculate in the same way as for the first CCA and the only difference is now that the time in the superframe is wherein the second CCA occur.

- Probability that a packet did not collide with other packet(s) that had successful first and second CCA's (denoted as γ; also called as the total success probability): This can be calculated as the probability that there are no accesses to the medium by the other nodes or the PAN coordinator during the period of one complete packet transmission time.

- End-to-end delay that include the queuing delay and service time of a packet.

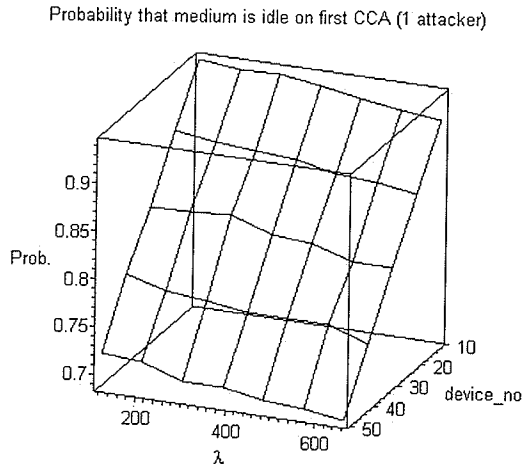
5.3 The Impact on the regular devices

In this Section, we show how the regular devices get affected due to the channel blocking caused by the various attacks induced by the two attackers in our simulated

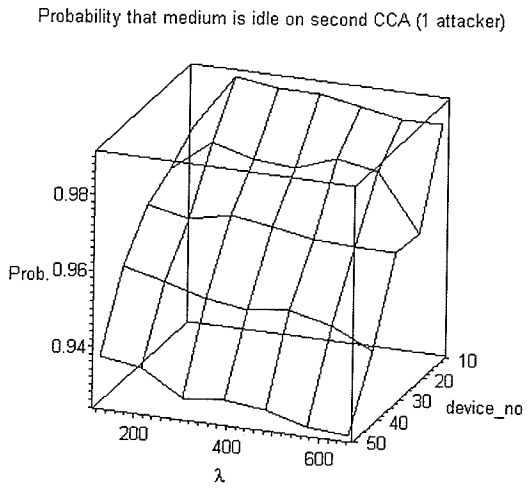
network. We have conducted different sets of experiments where the following attacks are induced by the two attackers:

- The attackers induces attacks by generating high and variable packet arrival rate whereas the other regular 50 devices in the network generated 120 packets per minute.
- The attackers sends larger sized packets with variable packet arrival rate whereas the regular devices send the packet with the size of 3 backoff period that is the minimum packet size the standard supports.
- The attacker devices omit the one CCA that are the crucial steps while executing the CSMA-CA MAC protocol to access the channel for a fair sharing of the open medium, which would give the malicious node an unfair advantage over the regular nodes.
- The attacker devices omit both the CCAs.
- The attacker devices omit the random backoff countdown.

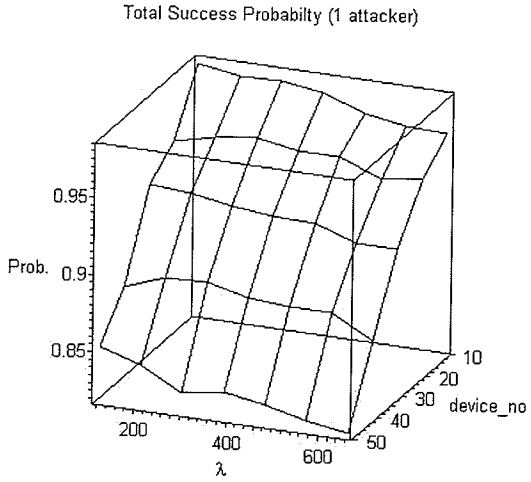
We would like to mention here that all the experiments that include the measurement of times are expressed in backoff periods (we mentioned it earlier in Chapter 2), the duration of which is prescribed by the IEEE 802.15.4 standard [12] to be 312.5μsec. Figure 5.1 shows the MAC performance in terms of probabilities that the medium is idle on first CCA, second CCA, the probability of success of overall transmission experienced by the 50 regular devices when there is only one attacker inducing this attack and the Figure is presented as function of the packet arrival rate λ and number



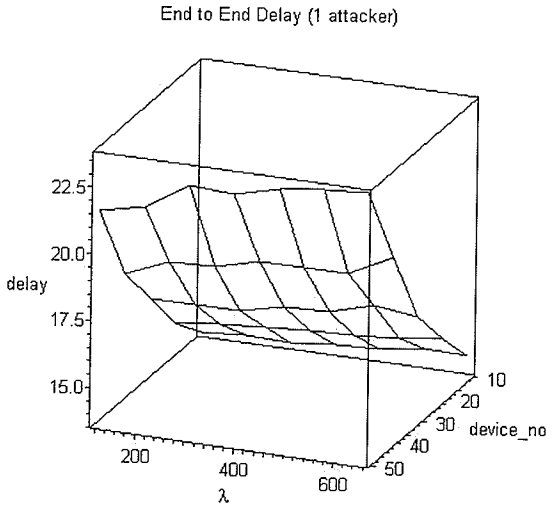
(a) α



(b) β

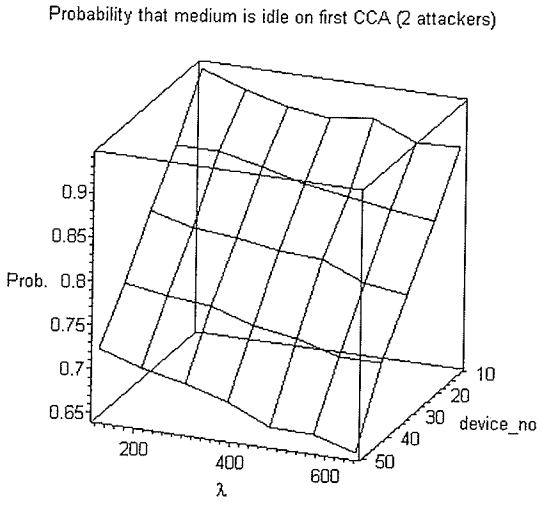


(c) γ

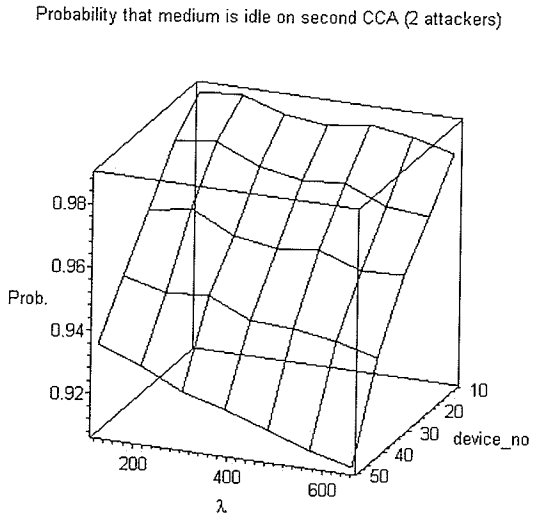


(d) End-to-end delay

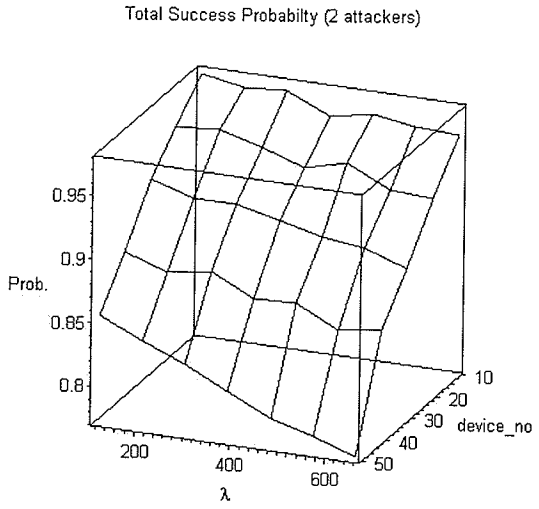
Figure 5.1: α, β, γ and incurred delay of the regular nodes when 1 attacker induces attacks by sending huge number of packets



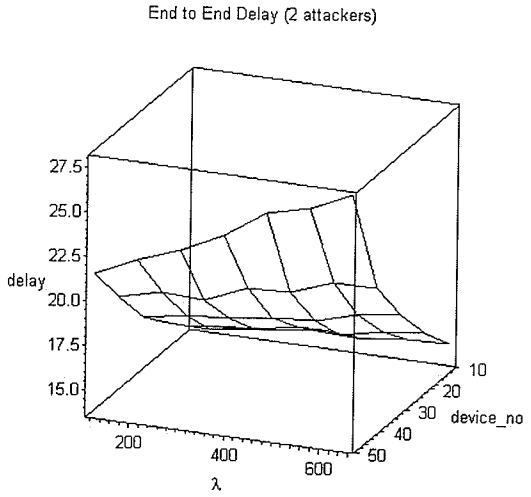
(a) α



(b) β

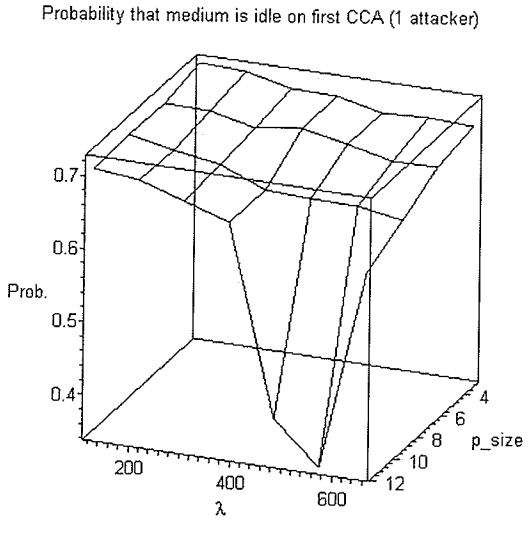


(c) γ

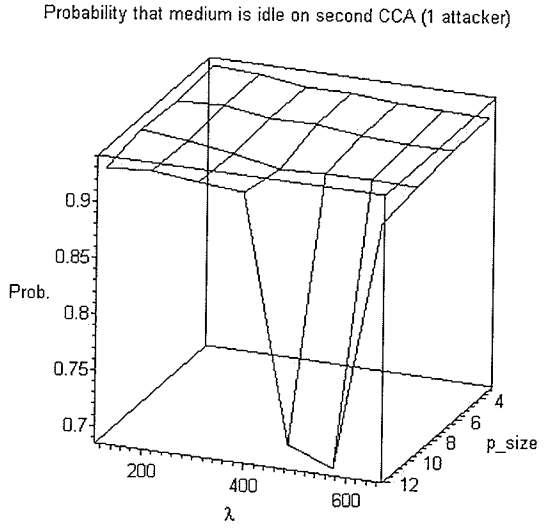


(d) End-to-end delay

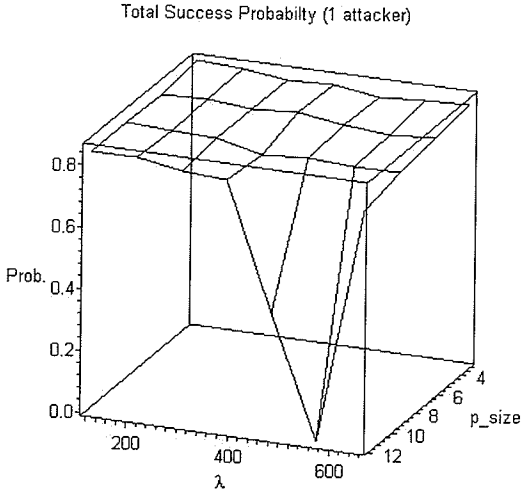
Figure 5.2: α, β, γ and incurred delay of the regular nodes when 2 attackers induce attacks by sending huge number of packets



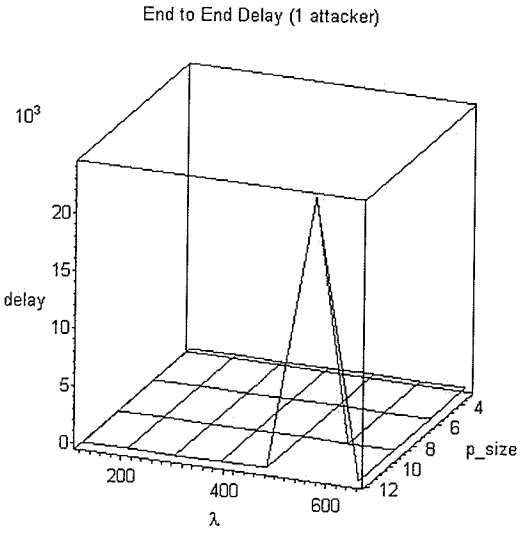
(a) α



(b) β



(c) γ



(d) End-to-end delay

Figure 5.3: α, β, γ and incurred delay of the regular nodes when 1 attacker sends variable larger sized packets (larger than the regular devices) with variable packet arrival rate

of regular devices in the network. As can be seen, success probabilities α, β, and γ are reduced as the number of regular devices increases in the network and the arrival rate of the attacker devices increases. It may be noted that, even with a single attacker in the network, the probability to that a regular device will sense an idle medium on the first CCA is around 68% as the arrival rate of the attacker increases and number of devices in the network is 50. What this reduction means in practice is, that ordinary devices will still gain access to the medium but the mean time to do so increases. As power consumption mainly depends on whether the radio is turned off or on, reduced success probability translates into increased energy expenditure for each packet transmission, and the ensuing lifetime reduction. Also, Figure 5.1(d) shows that the overall packet delay increases as the attacker's arrival rate and the number of devices are increasing.

From Figure 5.2(a) and 5.2(c), where we have two attackers in the network, we may notice the gradual decrease in the probabilities of getting a free channel on the first CCA as well as the total success probability. In Figure 5.2(d), it is obvious that in the presence of two attackers we have higher delays than there is only one attacker in the network. From both the Figure 5.1(b) and 5.2(b) We may also notice that β that is the probabilities of getting a free channel on the second CCA is mostly above or around 90%. The attacks could not effect this much as the interval between the first CCA and the second CCA is just one backoff period. That means only if a device finds a channel free on the first CCA, then it will go for the second CCA on the next backoff period and within that one backoff time period any other device might not block the channel and hence, the probabilities for getting the free channel on the

second CCA might not be affected by the ongoing attacks and it would be as higher as it is during a normal network operation when there is no attack. However, we will see a different situation in our next experiments. The author in [20] claim that the performance of an 802.15.4 cluster is fairly sensitive to high traffic loads, even moderate increases in packet arrival rate are likely to lead to substantial reductions in throughput originating from the legitimate nodes.

In our next experiment, we simulate an attack where the attacker device send larger sized packets with variable packet arrival rate. Again here, we first measure the same MAC performance in the presence of one attacker and then in the presence of two attackers. Figure 5.3 again shows the α, β, γ and the end-to-end delay that emphasize the effect of such attacks on the regular devices on the network. As can be seen, α, β, and γ reaches as low as 34%, 69% and 0.5% (which we did not notice in the previous experiments) with the attacker packet size 12 and attacker packet arrival rate of 570 packets per minute. The delay in Figure 5.3(d) also shows a sudden rise to a high value with same network size and parameters. Now when there are two attackers who induces the same attacks as above, we could notice from Figure 5.4 that even if there are no such sudden high or low peak in the α, β, γ and the delay, the overall performance degradation occurs as the packet size and the packet arrival rate of the attackers increases.

Our next experimental results show the same performance issues namely the α, β, γ and the end-to-end delay when the attacker devices omit one CCA or both of the CCAs that are required while executing the CSMA-CA algorithm to make sure that any packets are not collided. The attacker devices might reduce the number of

required CCA attempts which would give them an unfair advantage over the regular nodes. Here, we consider two attackers (instead of one attacker first and two attackers next) and compute the α, β, γ and incurred delay when the attacks are induced. The experimental setup is same as the experimental setup that is depicted in Figure 5.2.

Figure 5.5 shows how much resources the attacker devices can capture by just not executing one CCA, which only take one backoff period and also how much resources the regular devices are not getting due to the attack. We can see a slight increase on the end to end delay, as well as a slight decrease on the α, β, γ when the attacker devices skip both of the CCAs and these are depicted in Figure 5.6.

We also implemented the attack where the attacker devices skip the random countdown that we describe in Section 4.2 of Chapter 4. However, we do not present those figures here as the effect of those attacks are not as much significant as the effect we already showed here for the other attacks.

5.4 Bandwidth Utilization

Until now, we have seen the effect of possible attacks on the regular nodes in the network in terms of the probabilities of getting a free channel while the attack is going on and by these we realize that how much resources the attacker devices are consuming and hence, making the regular devices suffer from the DoS attacks. We have also seen the total success probability for transmitting a packet and the end to end delays.

Another important measure of the network performance that might be degraded due to the ongoing attacks is the bandwidth utilization of the network. We would

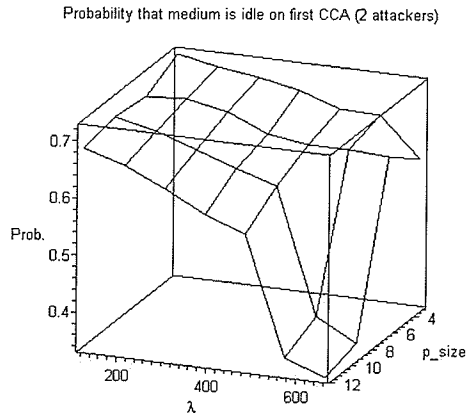
quantify that by measuring the throughput of the network that is the average throughput for all the regular devices in the network. Throughput is the ratio of the time spent towards transmitting actual data and the time used for channel accessing and transmitting a packet successfully. If the network faces a high number of collisions then a significant decrease in throughput may occur while good throughput is crucial for a secured as well as a purposeful network.

Therefore, we have also measured the throughput while we induced different attacks that we have already mentioned in Section 5.3. Here, we present two diagrams that are in Figure 5.7(a) and Figure 5.7(b), which show the throughput when two different attacks are induced by the attackers. We only present the Figures that show significant degradation while the attacks are going on. We could notice from both of the Figures that the average throughput for the regular devices is always under 1% while the attacker devices capture the channel for a longer time by inducing various attacks.

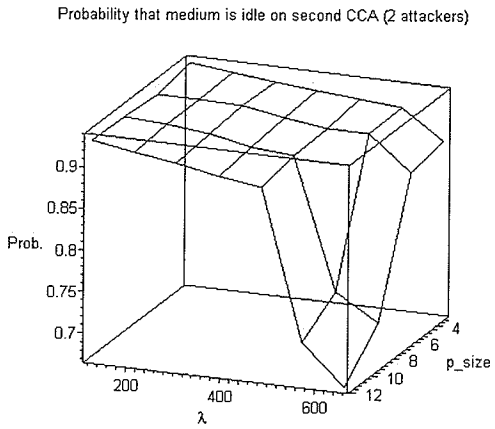
5.5 Summary

We have seen the effects of different attacks that we have identified in Chapter 4 through simulation results in this Chapter. While the effects we have seen may indeed pose formidable risks to normal network operations, it should be noted that they are not probably very cost-effective to launch. These attacks could not be stopped even if we use the existing security techniques such as encryption, authentication. They would only work once the network is aware of the intruder and hence, use authentication or encryption for those devices. Hence, we would need an intrusion detection

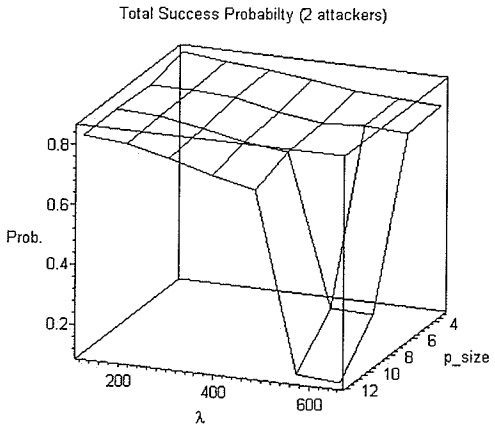
technique that may help identify the malicious devices. In the next Chapter, we describe our proposed simple intrusion detection method and the simulation results.



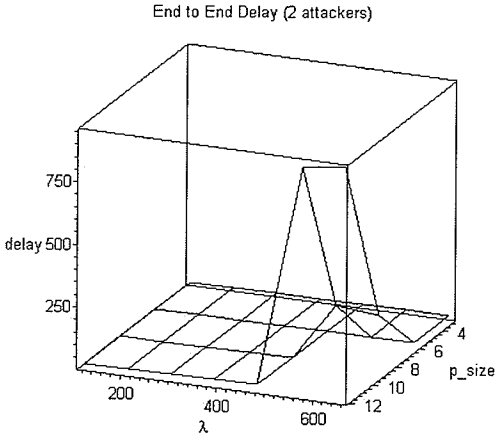
(a) α



(b) β

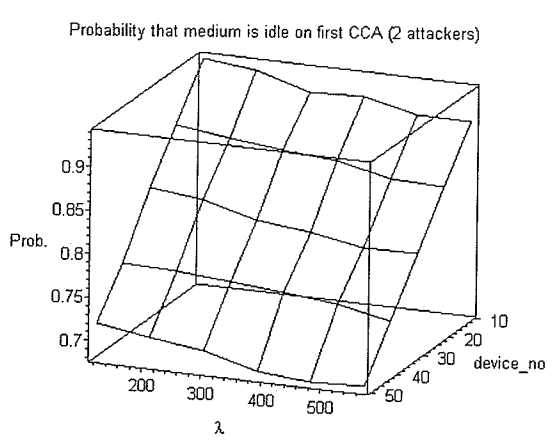


(c) γ

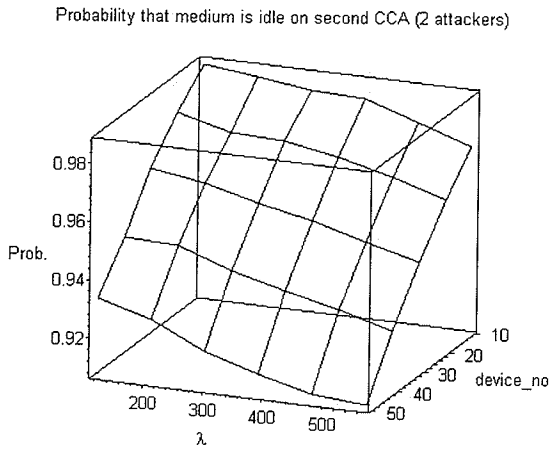


(d) End-to-end delay

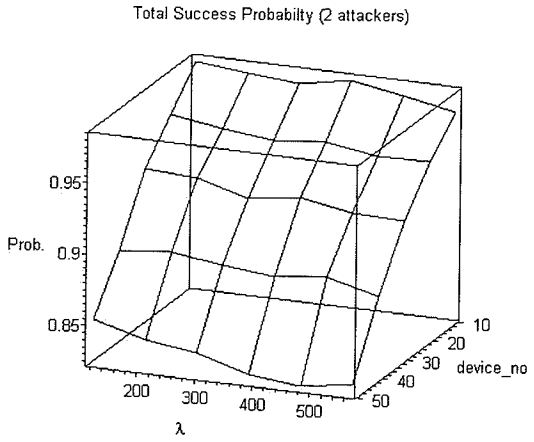
Figure 5.4: α, β, γ and incurred delay of the regular nodes when 2 attackers send variable larger sized packets (larger than the regular devices) with variable packet arrival rate



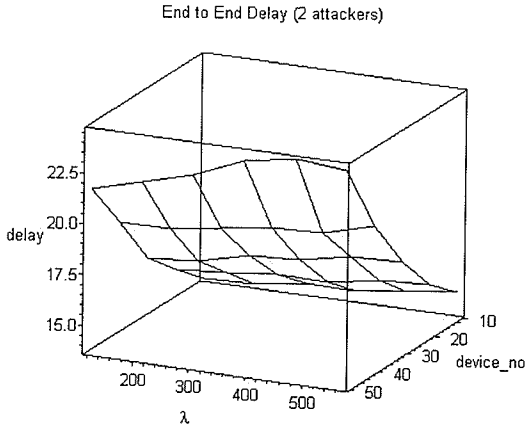
(a) α



(b) β

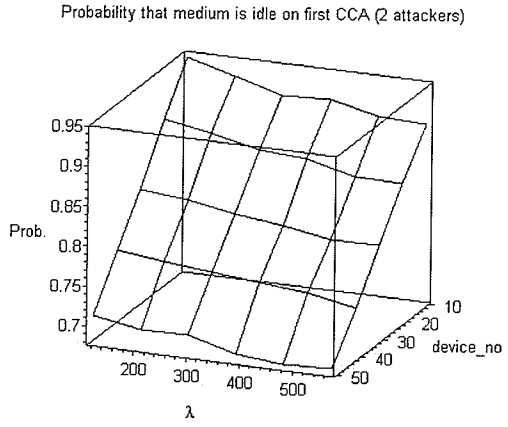


(c) γ

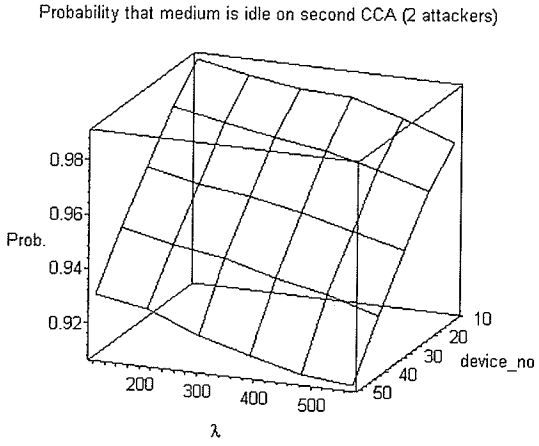


(d) End-to-end delay

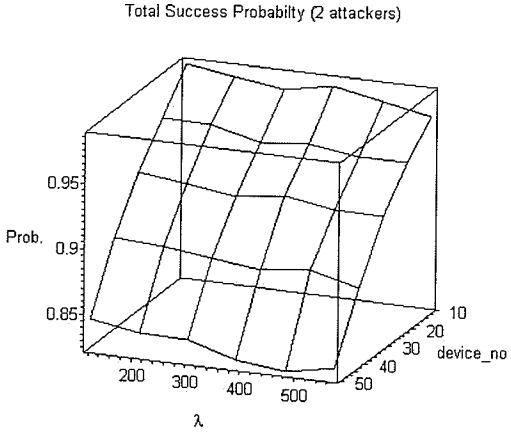
Figure 5.5: α, β, γ and incurred delay of the regular nodes when 2 attackers omit one CCA and send variable number of packets in the presence of variable number of regular devices



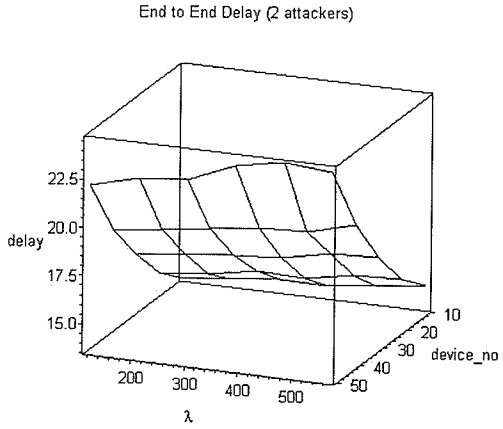
(a) α



(b) β

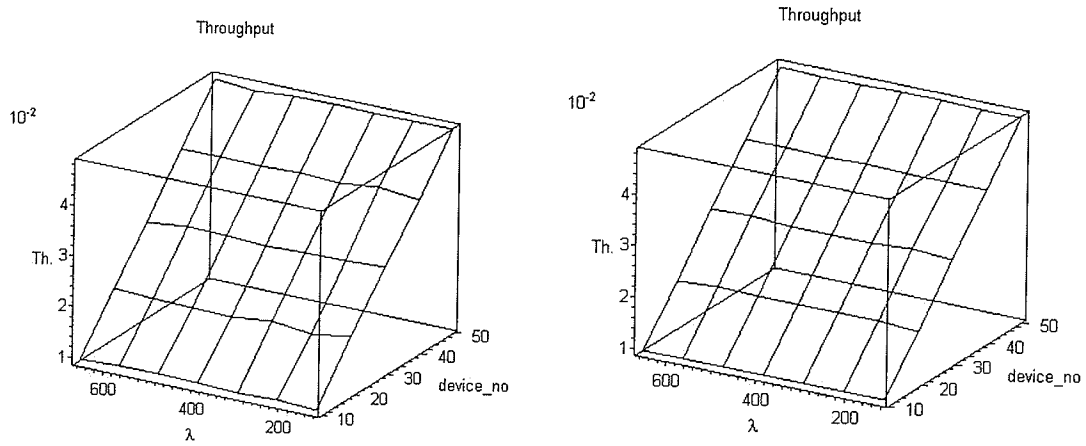


(c) γ



(d) End-to-end delay

Figure 5.6: α, β, γ and incurred delay of the regular nodes when 2 attackers omit both CCAs and send variable number of packets in the presence of variable number of regular devices



(a) Throughput when 2 attackers induce attacks by sending huge number of packets

(b) Throughput when 2 attackers omit both CCAs and send variable number of packets in the presence of variable number of regular devices

Figure 5.7: Throughput Measurement

Chapter 6

Intrusion Detection Method And Experimental Results

In order to design a simple yet effective intrusion detection method for WSN, it is important to determine what should be classified as an intrusion or an anomaly as it depends strongly on the application. Also, while designing an intrusion detection method, we have to define the scope of the method so as to comply with the underlying protocol that is being used for a particular WSN application. Moreover, to be practical to implement on WSNs, intrusion detection techniques should be lightweight and scalable.

Keeping these constraints in mind, we have adopted a simple averaging method similar to the one proposed in [24]. Averaging is applied to packet interarrival time, which can easily be measured by the PAN coordinator, rather than its inverse – the packet arrival rate, which is somewhat more difficult to measure. However, the averaging method proposed in [24] relies on two time windows that require substantial

memory resources (e.g., the long term window requires more than a thousand values) and thus may not be suitable to apply in a resource constrained sensor network environment. Fortunately, a simpler averaging technique that uses resources in a very efficient manner is available: the Exponential Weighted Moving Average, or EWMA, the details of which are presented in the discussion that follows.

6.1 EWMA

Exponential Weighted moving average (EWMA) is related to simple moving average. It is a weighted simple moving average that puts progressively increasing weight on more recent values of the averaged variable. The formula for EWMA is:

$$\overline{f}_i = \alpha f_i + (1 - \alpha)\overline{f}_{i-1} \tag{6.1}$$

where f_i is the recent value of any factor (packet inter arrival time in our case), α is the weighting factor, and $\overline{f_i}$ and $\overline{f}_{i-1}$ is the new EWMA that would be calculated and the previous EWMA value of the same factor respectively.

α is the weighting factor here to control the value of average. If we want to put more weight on the recent value then the α need to be increased. On the other hand, if we want to give more weight to the older values, α need to be decreased. In the network research area, EWMA is sometimes used to minimize the fluctuation of the values. For example, EWMA can be used when working with radio signal strength that fluctuates and is not stable enough to be used as it is. This property is beneficial in our context since the nodes generate random traffic, and the use of EWMA can help smooth the random fluctuations of packet interarrival time. In Figure 6.1, we

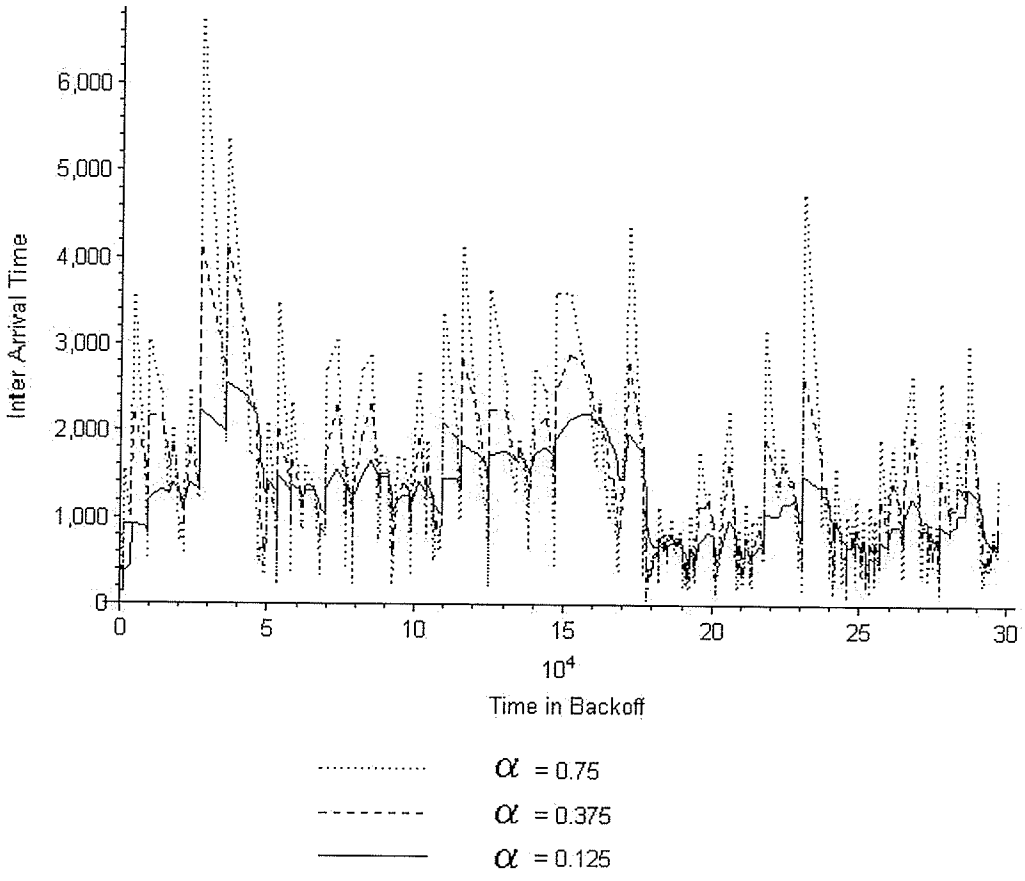


Figure 6.1: EWMA of the packet interarrival time of an arbitrary device

plot the EWMA values of an arbitrary device with three α values that is calculated as functions of time. As mentioned earlier, the different α values give us different short or long term averages. We choose three moderate α values where two ($\alpha = 0.375$ and $\alpha = 0.125$) of them give us long term averages since the more weight is given to the past EWMA values and one ($\alpha = 0.75$) gives us the short term averages as we give more weight to the recent interarrival times.

In our simulation, we create an arrival process with fixed bursts of ON and OFF intervals that helps to present an actual intrusion detection experiment. Here, the

attacker devices behave anomalously during the ON intervals and behave normally during OFF interval. For each ON burst, we create a Poisson subprocess. That means in all the periods the regular devices generated Poisson traffic with a constant arrival rate of 120 packets per minute, while two attacker devices periodically switched between two different packet arrival rates in the ON and OFF periods: the lower rate is kept fixed and equal to the packet arrival rate of the regular devices; the other is kept higher and variable. We keep the duration of the ON and OFF intervals fixed and equal to make our further calculation simple. We assume that an intruder will be changing interarrival times by either constantly transmitting or disregarding previous long waiting times between the bursts. Moreover, a warmup time is used in order to allow the cluster to reach a steady state before the actual attacks are introduced. From this arrangement we would know how frequently we get false alarms and how much time our system takes to recover from an attack (we discuss these issues at the end of Section 6.2.3). Figure 6.2 illustrates this compound process.

Using the above attacking arrival pattern we have calculated the EWMA of the attacker devices and shown the differences between two cases when there is attack and when there is no attack. We could notice from Figure 6.3 and Figure 6.4 how the EWMA immune to the fluctuation of the channel when we compare it to the regular interarrival time. In Figure 6.4, we have shown the pattern of the attacking arrival rate too.

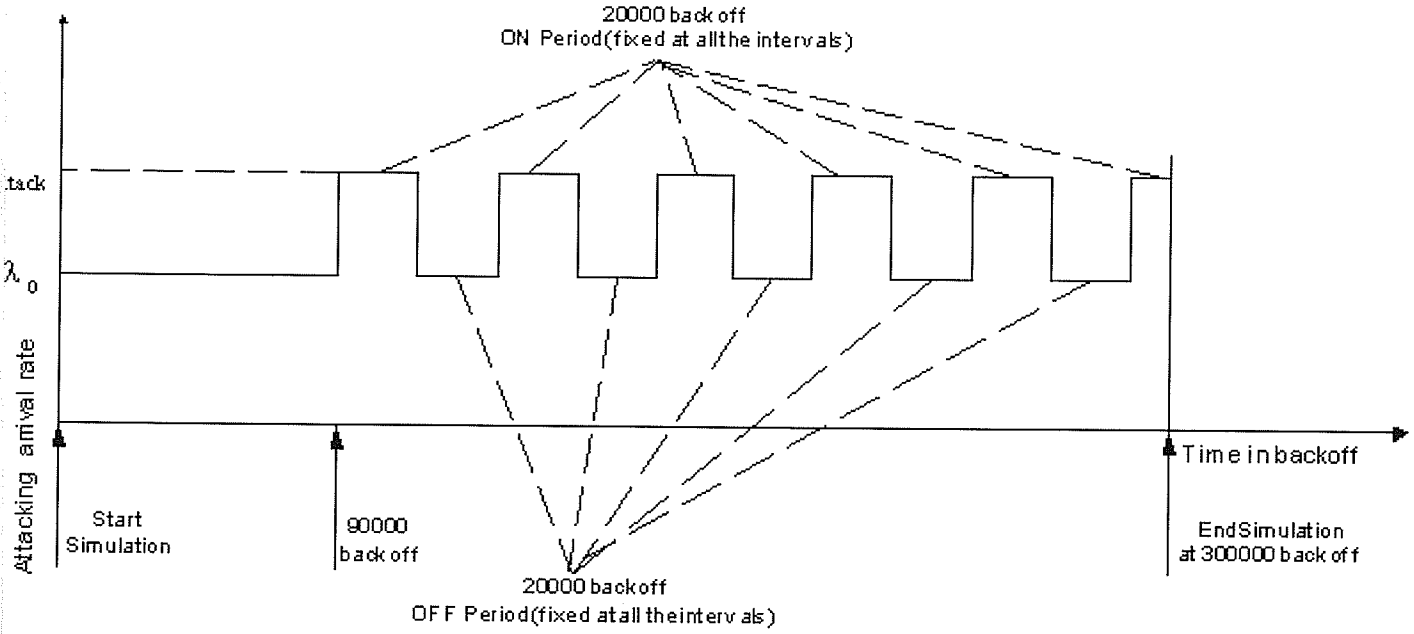


Figure 6.2: The ON and OFF Arrival Process

6.1.1 Standard Deviation of different EWMAs

For our EWMA-based intrusion detection method we first need to decide on which α value we will work on. To decide on the α value, we need to see which EWMA with a particular α value leads to a smaller standard deviation. This is because smaller standard deviation means smaller fluctuation and hence, the probability of getting less false alarms will be higher.

We have conducted an experiment to observe the changes of the standard deviation of different EWMAs with different α values. We have calculated 10 (this number is arbitrarily chosen) EWMAs with 10 different values of α (0.10 to 1.0). Again, the factor we consider to calculate the EWMA is the packet inter-arrival time of every device. For each EWMA, the PAN coordinator calculates the EWMA at every packet

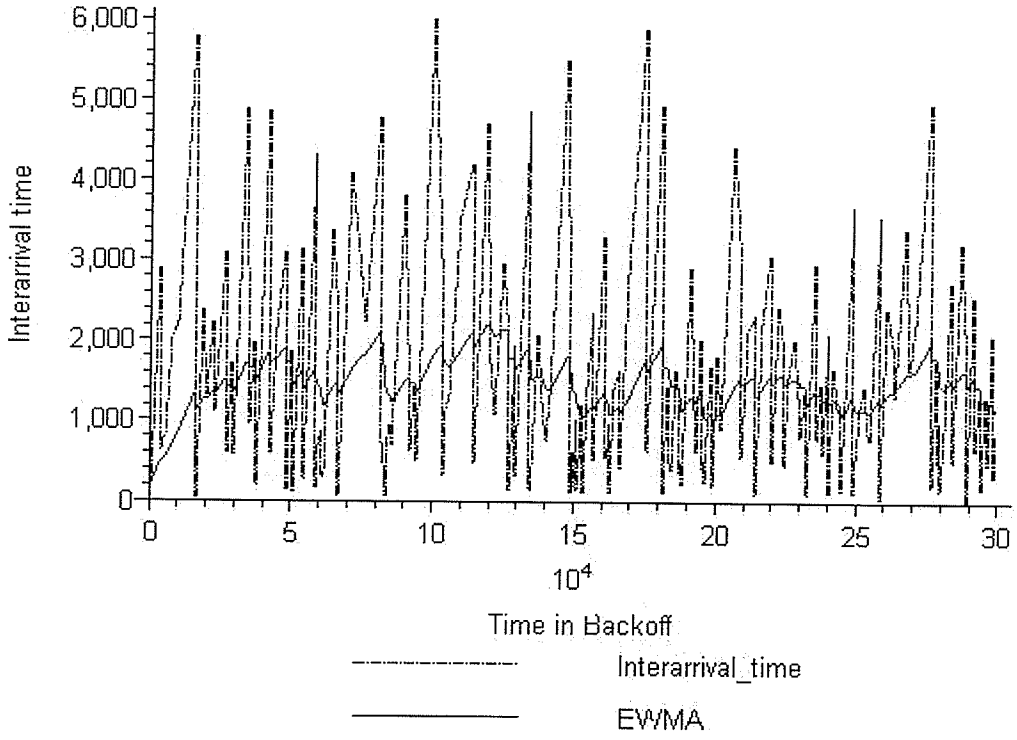


Figure 6.3: EWMA (long term) and interarrival time of a device that induces no attack.

it receives and keeps a sum of the all the instantaneous value of this particular EWMA and squared sum of all the instantaneous value of this EWMA. Then, we calculate the standard deviation of this particular EWMA with particular weighting coefficient α. The formulas are shown here:

$$\mu(\tau_i) = E(\tau_i) = \frac{1}{N}\sum_{j=0}^{N}\tau_{i,j} \tag{6.2}$$

$$\sigma(\tau_i) = \sqrt{(E(\tau_i^2) - (E(\tau_i))^2)} = \sqrt{\frac{1}{N-1}\sum_{j=0}^{N}(\tau_{i,j} - \mu(\tau_i))^2} \tag{6.3}$$

Where $\mu(\tau_i)$ and $\sigma(\tau_i)$ are the mean and the standard deviation of the particular EWMA denoted as τ_i. In this experiment, this is repeated for each of the 10 distinct

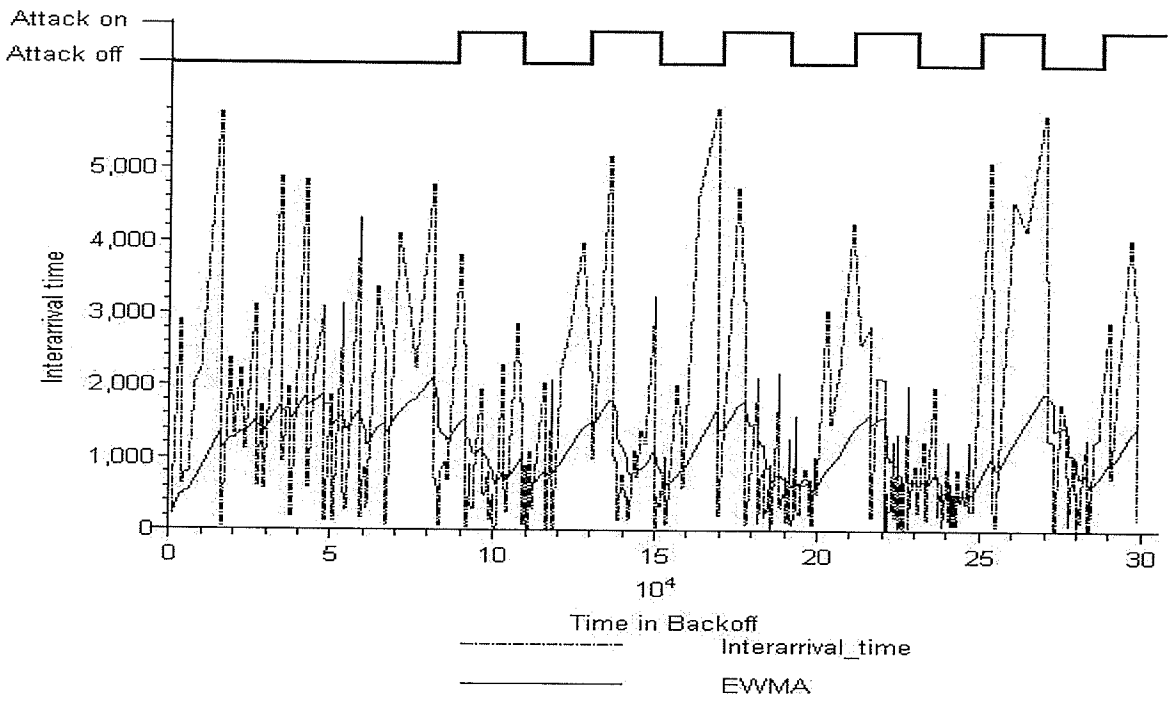


Figure 6.4: EWMA (long term) and interarrival time of an attacker device that induces attack

EWMAs having 10 distinct α values.

We have calculated this for each device in our network. Moreover, in this experiment different arrival rates of the attackers have been considered. Figure 6.5 and Figure 6.6 show 4 diagrams where the standard deviations of 10 distinct EWMA is shown for some arbitrarily chosen devices. For the attacker devices (device no 51 and 52 in our network) we also present the original EWMA values that we calculated for 10 different α values as well as present the standard deviation of these mean values. We need to mention here that the attacker devices begins its attack by switching to the ON-OFF pattern described above only after 90000 backoff periods. (The entire simulation time is 300000 backoff periods, as can be seen from Figure 6.2).

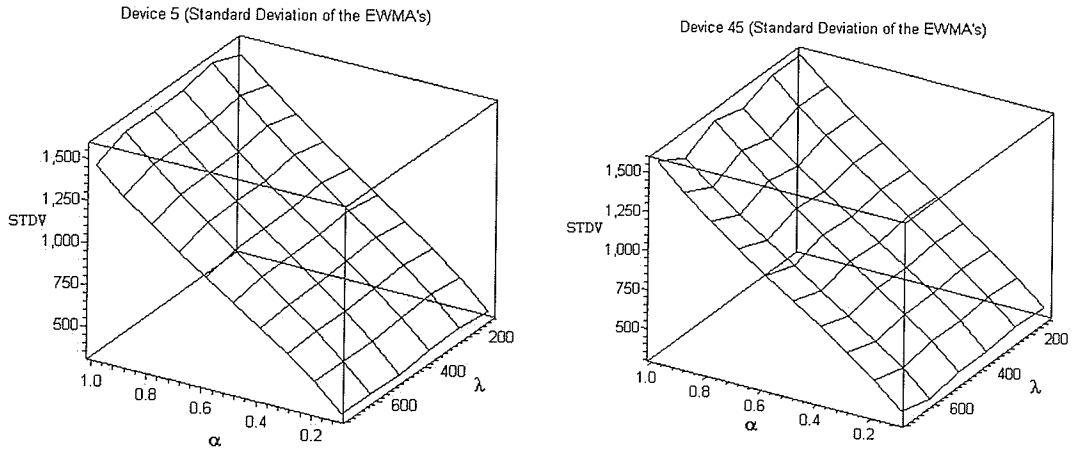


Figure 6.5: Standard deviation of EWMA'S of device 5 and 45 with different arrival rates of the attacker devices

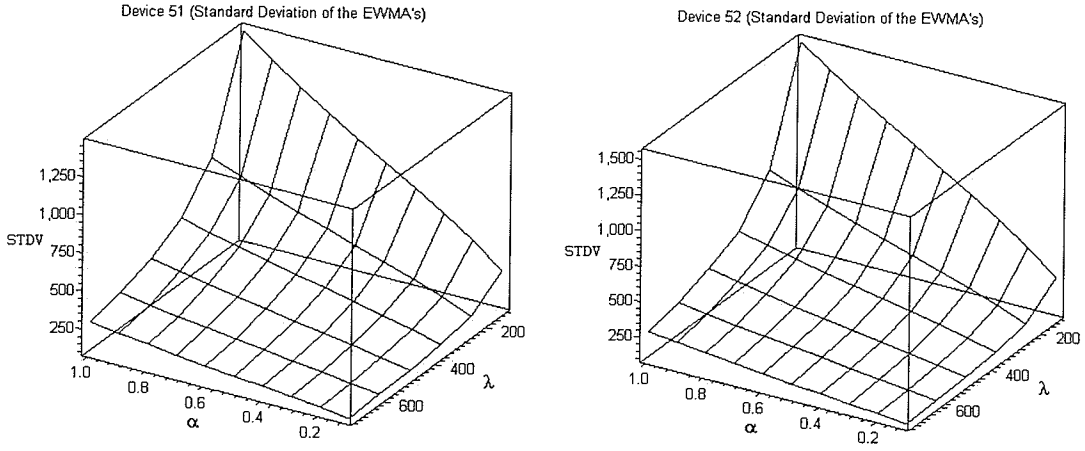


Figure 6.6: Standard deviation of EWMA'S of device 51 and 52 (the Attacker devices)

We can observe from the Figures that the standard deviation of different EWMA's tend to be higher when the α value is higher. That means, a smaller α gives us smaller standard deviation and hence smoothen the obvious fluctuation in our network.

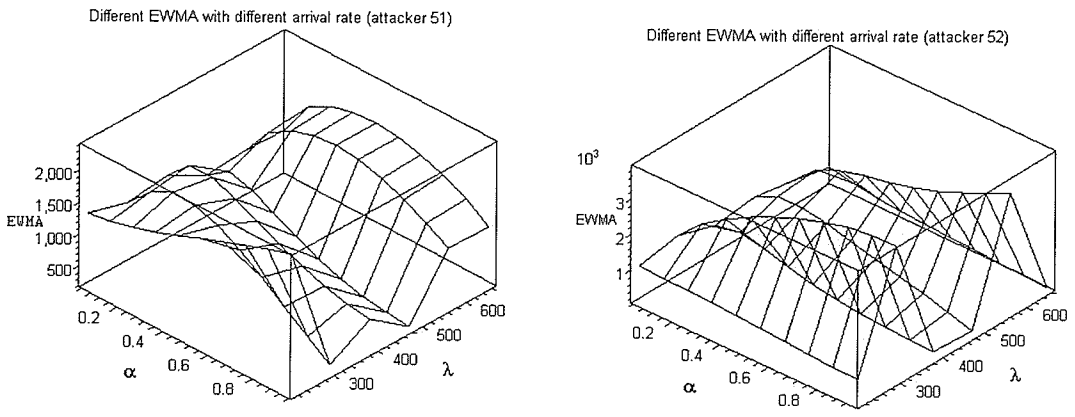


Figure 6.7: EWMA Values for Device 51 and 52

6.2 Detection Mechanism

An important feature in star topology network is the communication pattern; almost all communications will be in the form of many to one transmission as nodes reporting sensor reading to a more capable nodes (i.e., the PAN coordinator). Therefore, the PAN coordinator should take all the necessary steps to detect anomalous activities in our network. What this means is, simply, that our intrusion detection mechanism should devise whether any particular node sends packets more frequently than its peers. To model the mechanism, we first need to define the normality criteria for each of the device in the network. Then, the algorithm will compare the recent trend with the normal (or historically normal) trend of device traffic. The purpose is clear as an attacker in our network might not start attack at the beginning of the network formation as it has to associate with the network and establish itself as a regular device and then induces attacks. Furthermore, we can also compare the behavior of a single device with the behavior pattern of the rest of the network. This is meant to show how differently this particular device is behaving compared to all

the other devices in the network. This increases the chance of an anomalous device to be detected, even if the comparison of recent traffic to historical averages for that particular device does not succeed in detecting a possible intrusion.

Similar to the technique proposed in [24], our detection technique requires the calculation of two EWMAs ($EWMA_1$ and $EWMA_2$) of packet inter arrival times with different α values (α_1 and α_2) for each device in the network. $EWMA_1$ represents the average packet inter arrival time of the entire network and hence, $EWMA_1$ also represents the long term average (i.e., it keeps a smooth trend of the previously taken average of the packet inter arrival times) of the entire network. On the other hand, $EWMA_2$ keeps a short term average of each device in the network.

In Section 6.1.1, we have shown that smaller α values lead to smoothing of the fluctuations in packet arrival rates. That means, the EWMA with smaller α gives more steady moving average of the packet inter arrival times. On the other hand, the EWMA with larger α value puts more weight on the recent value and, hence, gives an average that is biased toward most recent activity.

Therefore, two separate EWMAs of packet arrival rates with different α values are updated with each incoming packet by the PAN coordinator and the EWMA ($EWMA_2$)using larger α (α_2) is compared against the EWMA ($EWMA_1$) using smaller α (α_1) which is accepted as normal. Moreover, by $EWMA_1$, the PAN coordinator also stores the exponential weighted moving average of the packet interarrival times for all the devices that keeps the cumulative patterns of the entire system.

Now, a deviation from the normality criteria is deemed anomalous as nodes follow a predictable traffic generation behavior (depending on the application) in our

network. A device i is anomalous if the following EWMA comparison holds:

$EWMA_{2i} < W \times EWMA_1$

Here, the threshold W is used to tune the strictness of the detection process. By strictness we mean the system should not let any sort of malicious activities undetected at any situation. That means, using a small value of W we are increasing the probability that the left hand side term of the comparator in the above expression would be smaller and hence a device would be detected as malicious.

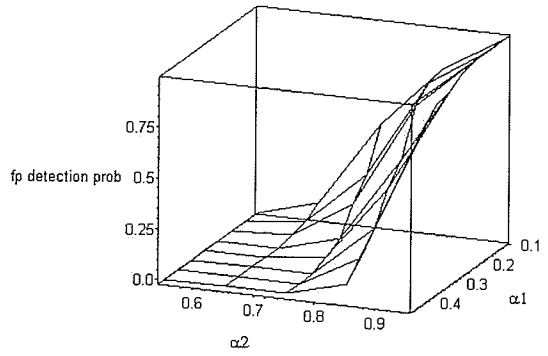
The selection of proper α_1, α_2, and W values depending on security vulnerabilities and available resources has crucial influence as these values strongly affect both the probability of detection and detection time as shown with simulation results. All the experimental results shown are the average of multiple simulation result using different random seeds.

6.2.1 Effect of α_1 and α_2

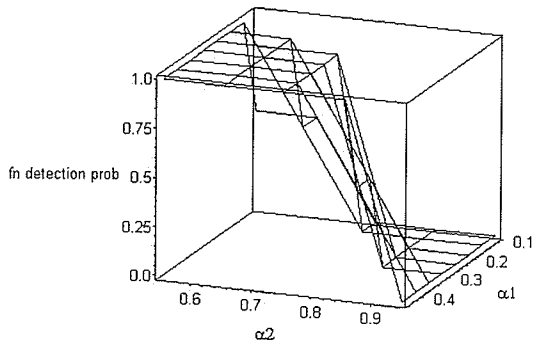
Though it is clear while comparing $EWMA_2$ with $EWMA_1$ the corresponding α value needs to be larger for $EWMA_2$ and smaller for $EWMA_1$, we need to observe the effect using different combinations of α_1 and α_2.

The range of α value is 0 to 1. In the first experiment, we take $W = 0.10$, vary α_1 from 0.10 to 0.45 and α_2 from 0.55 to 0.90. According to the simulation results, we plot three graphs: for the probability of detecting false positives and false negatives and for the average delays of detection, which are depicted in Figure 6.8.

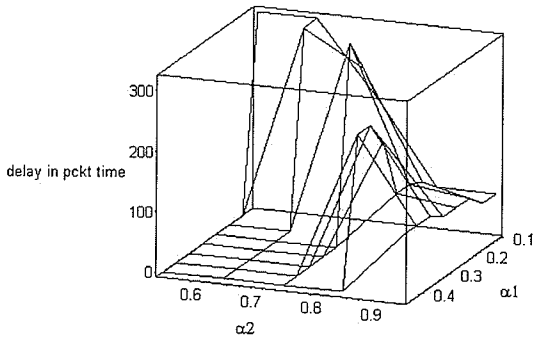
In Figure 6.8(a) and 6.8(b), we see that the increase of α_2 (moving from left to right on the diagram, with a fixed value for α_1) makes the detection mechanism



(a) Probability of detecting false positives



(b) Probability of detecting false negatives



(c) Detection delay (in packet times)

Figure 6.8: Probability of detecting false positives and false negatives with different α_1 and α_2 values

stricter. Although a higher percentage of actual attacks will be detected as such, the frequency of false alarms will also increase. Also, the strictness of the system results in shorter delays in detecting the anomalous activities (Figure 6.8(c)).

The experimental results show that the larger the difference between α_1 and α_2, the stricter the detection system – i.e., the probability of detection increases. If we

analyze the reason then we find that the little difference between α_1 and α_2 does not help to detect any significant difference between current and previous activities of devices. If the two α values tend to be equal then whatever the recent value is the two EWMAs become almost same. Therefore, the comparison between $EWMA_1$ and $EWMA_2$ has no effect on detecting significant changes. Therefore, this is a crucial decision to be made about the α values. If the strictness is more important then we need to amplify the difference between two α values and we need to decrease the difference if we want to avoid getting false positives.

We needed to decide upon values for α_1 and α_2 in order to carry on our next experiments. By analyzing Figure 6.8(a) and 6.8(b), we see that for some combination of α_1 and α_2 such as (0.10, 0.85) (0.2, 0.85), (0.25, 0.85), (0.3, 0.85) we can minimize the probabilities of both getting false positives and getting false negatives. Depending on the application, we may want to minimize one or the other; in sensing applications it might be preferable to minimize the probability of false negatives. That means we need a strict detection system and at the same time we want to reduce the false positives as much as possible. If we look at the delays carefully we can pickup the same values. However, to decide on the values we also have to see among these combination of α_1 and α_2, which one give us better result with different values for the threshold that is W.

6.2.2 Effect of weight W

The threshold W plays the vital role in our detection mechanism and we analyze the effect of different values for W through the next set of experiments. In this set

of experiments, we have used different combinations of W, α_1 and α_2. We have used the following combinations:

1. Keep α_2 fixed and vary W and α_1 and measure the probability of false positives and false negatives and the delay of detection in packet times.

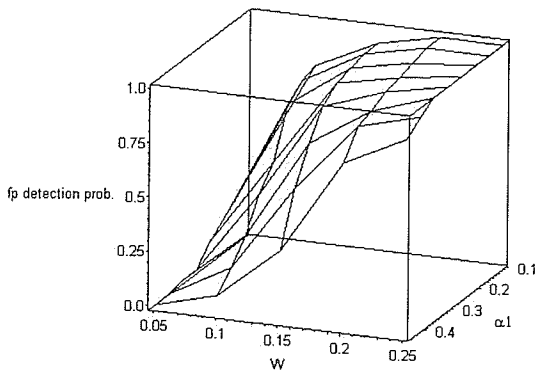
2. Keep α_1 fixed and vary W and α_2 and measure the probability of false positives and false negatives and the delay of detection in packet times.

According to the experiments we plot six graphs where we have shown the probability of getting false positives and false negatives and the delay with W varies from 0.05 to 0.20, α_1 varies from 0.10 to 0.45 and α_2 varies from 0.55 to 0.90. The figures 6.9(a), 6.9(b) and 6.10(a), 6.10(b) show that the increase of W also increases the probability of detection. By analyzing the figures we observe that $W = 0.1$ is one of the best choice to minimize the false alarm without sacrificing the strictness of our detection system. We need to keep in mind that the delay of detection should also be minimized. Figures 6.9(c) and 6.10(c) show that when probability of detection increases the delay of detection decreases. Therefore, through these measurements (that include Section 6.2.1 and Section 6.2.2) we have decided to use the following values:

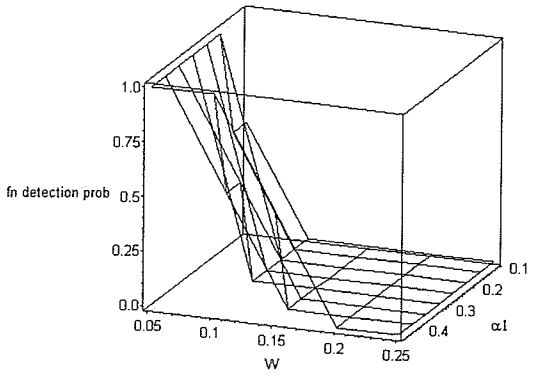
- 0.10 for the smoothing coefficient of the long term EWMA that is α_1;

- 0.85 for the smoothing coefficient of the short term EWMA that is α_2; and

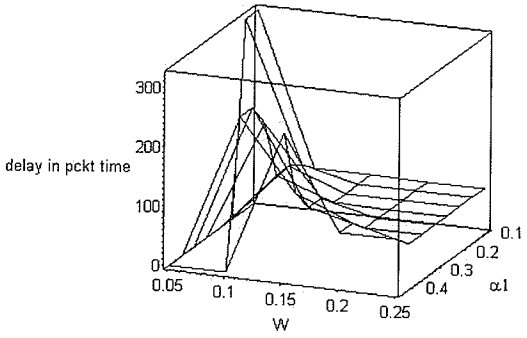
- 0.10 for the threshold W.



(a) Probability of detecting false positives



(b) Probability of detecting false negatives

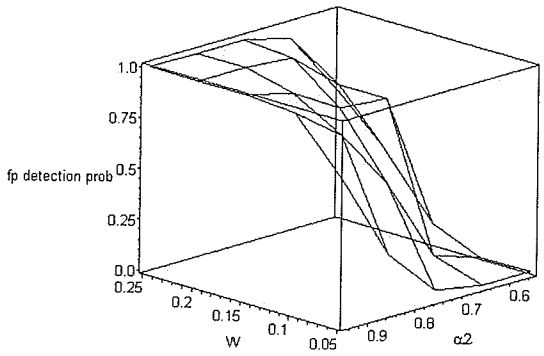


(c) Detection delay (in packet times)

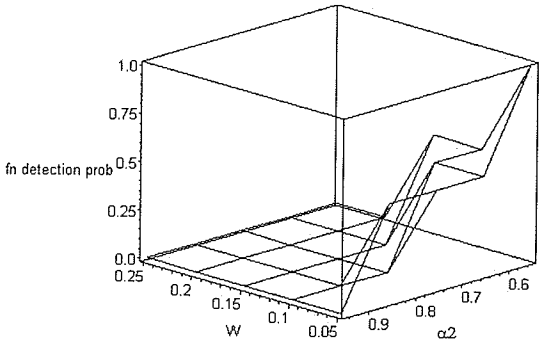
Figure 6.9: Probability of detecting false positives and false negatives with different α_1 and W values

6.2.3 Handling False Positives

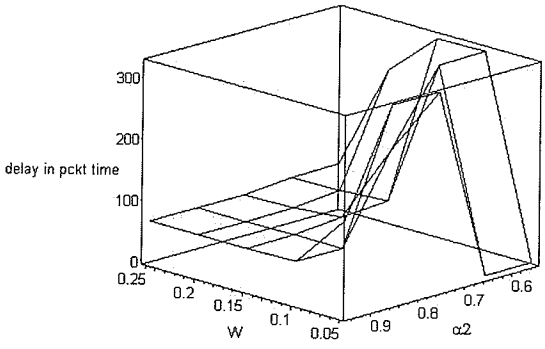
We already mentioned that the packet arrival rate fluctuates a lot in our network. Therefore, the detection system that relies on the behavior of the packet arrival rates of individual device may not avoid generating large number of false positives. From



(a) Probability of detecting false positives



(b) Probability of detecting false negatives



(c) Detection delay (in packet times)

Figure 6.10: Probability of detecting false positives and false negatives with different α_2 and W values

all the previous Figures we may see that the results are less stable than desired due to random fluctuations caused by the random traffic. Namely, the ratio of the long- and short-term EWMAs tend to dither, i.e., oscillate within a small range around the threshold, which results in a large number of threshold crossings and leads to instability of results. For example, for the values of $W = 0.10$, $\alpha_1 = 0.10$, and

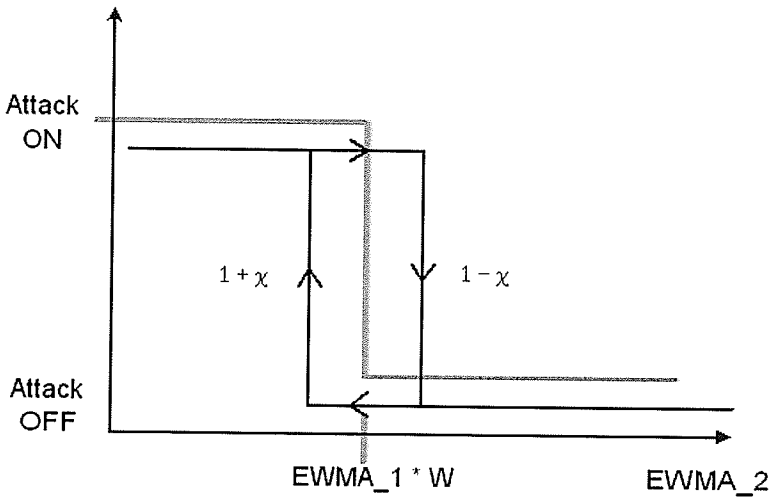


Figure 6.11: χ inclusion to reduce the oscillation of the detection

$\alpha_2 = 0.85$ we found that we get about 78% false positives even though we get 0% false negatives and less delay, which is only 65 in packet times. Therefore, even with our chosen W, α_1, and α_2 values that lead to almost no false negatives, we have significant number of false positives.

In order to prevent such oscillations, we modified our detection method that can filter some of the false positives generated by our detection system where at the same time we have low false negatives. We have introduced a small hysteresis in the decision process, as depicted in Figure 6.11. Hysteresis is phenomenon in which the response of a system to an external influence (attack in our case) depends not only on the present outcome of that influence but also on the previous history of the system. Basically, two different threshold values are used:

- when detecting the onset of an attack:

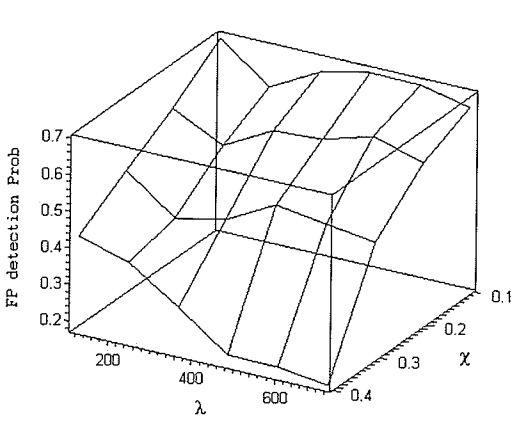
$$\overline{EWMA}_2 < \overline{EWMA}_1 \times W(1-\chi); \tag{6.4}$$

- when detecting the end of an attack:

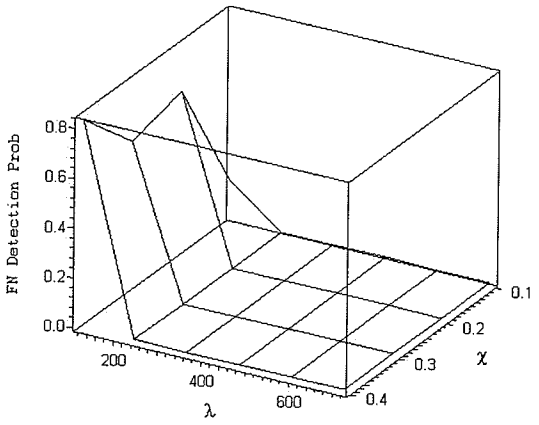
$$\overline{EWMA}_2 < \overline{EWMA}_1 \times W(1+\chi). \tag{6.5}$$

The introduction of hysteresis can help to reduce the impact of the dithering in the decision variable, and thus lead to substantial improvements in the stability of the algorithm. We intend to keep the value of χ small, which slightly changes (increase or decrease) the threshold W. From the previous experiments we find that $W = 0.10$ gives us better result in terms of number of false positives and false negatives during the detection period. Instead of keeping a fixed threshold for all the period we want to change the value of the threshold W dynamically depending on the outcome of the mean value comparison (i.e. we know if there is an ongoing attack or not) so that we tune the threshold itself to minimize the effect of oscillation.

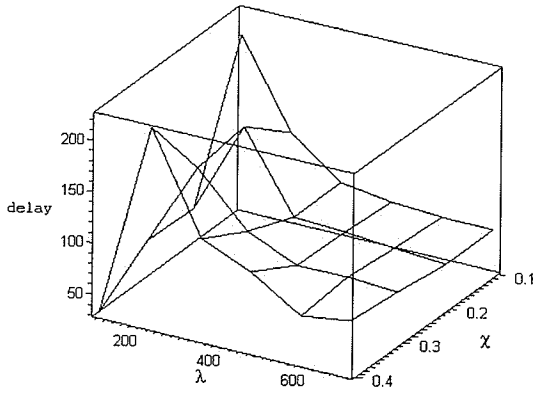
The optimum value of hysteresis χ can be determined as follows. We calculate the ratio of the standard deviation and the EWMA of a particular attacker node that has significantly different arrival pattern. This is reasonable as the oscillation of the detection pattern is the cause of fluctuations of the arrival pattern and, by extension, of fluctuations of the EWMA of the interarrival time. Earlier, we have conducted an experiment where we have calculated the EWMA (mean) and the standard deviation of the mean to determine which α value we will use for our detection method in Section 6.1 and 6.1.1.



(a) Probability of detecting false positives



(b) Probability of detecting false negatives



(c) Detection delay (in packet times)

Figure 6.12: The probability of detecting false positives, false negatives and the incurred delay to detect the attacker with different χ values and different attacker arrival rates (by our modified method)

A further question arises when we want to choose a particular α value to get the ratio of a particular standard deviation and a particular EWMA from Figure 6.6. To this end, we have chosen an arbitrary attacker device (device 51). In Section 6.2.1

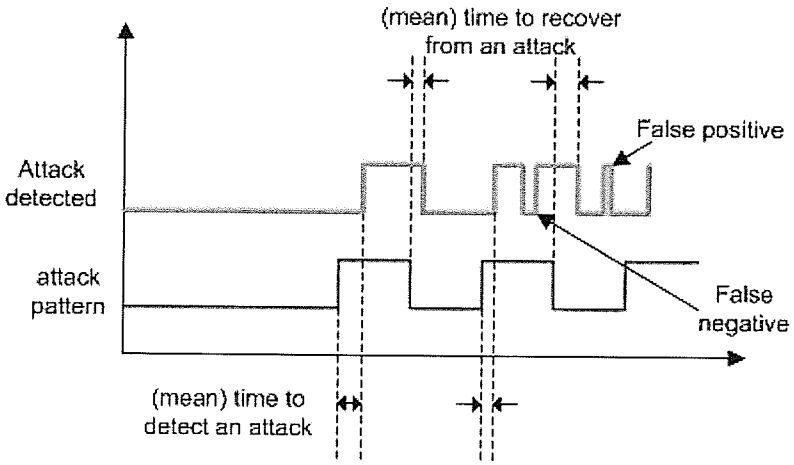


Figure 6.13: Attack pattern and detection pattern

we have shown that the two α values we have found to work well are $\alpha_1 = 0.10$ and $\alpha_2 = 0.85$. From Figure 6.6 and Figure 6.7, the workable χ values (the ratio of the standard deviation and the EWMA for these two α values of attacker device 51) for this purpose is below 0.40. To find the optimum χ value, we have conducted further experiments using $\chi = 0.10, 0.20, 0.30$ and 0.40. We measured the probability of detecting false positives and false negatives as well as the delay with these four χ values and different attacker arrival rate. The outcome of these experiments are shown in Figure 6.12(a), 6.12(b) and 6.12(c). We would like to mention here that all the results here were obtained as averages of ten experiment runs with different random seeds.

As can be seen from Figure 6.12(a) and 6.12(b), the probability of false alarms is significantly lower compared to the data in figures in Section 6.2.1 and 6.2.2. Also, we get lower delays, which is shown in Figure 6.12(c).

6.2.4 Additional performance measures

Until now, we have measured the performance of our detection method in terms of lower false positive rate, low false negative rate and less delay in packet times. To check the performance of the modified algorithm we have used other performance measurements such as the mean time between false alarms (MTBFA), mean time to detect a true attack (MTD), and mean time to recovery from a true attack (MTTR). These measures provide the same information but in a more practical format, as they can directly be interpreted in a practical setting. Figure 6.13 shows how we can calculate the MTD and MTTR. Here, we have also shown where we find the false positives and the false negatives and it refers to all the other data that we took in Section 6.2.1 and 6.2.2.

These factors show how robust (resilient) a developed system is. Among these, the MTBFA is quite similar to the false positive rate except it tells how frequently (an average estimation) we a get a false alarm by our detection system. The other attribute namely the MTD shows how quickly (which is also an average value) our method could detect an attack. Lastly, the MTTR gives us an average time of how fast our method can understand that the attack has stopped and hence, allow the network to resume normal operations again.

Hence, we conducted another set of experiments to measure these performance and for this experiment we have used the same setup as the previous experiments with different χ values and different arrival rates of the attackers. The results of this experiment are shown in Figure 6.14.

From Figure 6.14(a), 6.14(b) and 6.14(c), we notice that with the increasing arrival

rates of the attackers and increasing χ values we get small MTD, large MTBFA and small MTTR.

As can be seen, the mean time to detect an attack rapidly decreases when the attacker increases its packet generation rate above, say, 150 to 180 packets per minute. At higher packet generation rates, MTD decreases more slowly, while from Figure 6.12(b) we see that the probability of missing a real attack becomes very low . Typically, the algorithm will detect an attack in around 3 to 5 thousand backoff periods, which corresponds to the range of 1 to 1.5 seconds.

Mean time between false alarms and mean time to recover after an attack behave in a similar manner: namely, both of them start at higher values around a few thousand backoffs for attacker arrival rates close to that of regular devices, and then slowly decrease when the attacker arrival rate increases.

From all these Figures (i.e. Figure 6.12 and 6.14) we may notice that in terms of an optimum MTD, MTBFA and MTTR values $\chi = 0.40$ also hold here. In the next experiment, we considered different packet arrival rate of the attackers and different packet sizes of the attackers. Although in our algorithm we do not keep any statistics of the packet sizes of the packets coming from any devices in the network, we want to see how our method works if we introduce the attacks where both the packer arrival rate and the packet sizes of the attacker devices are being changed simultaneously.

As can be seen from Figure 6.15 and Figure 6.16, none of the performance measures is significantly sensitive to attacker packet size and hence, the performance measures of the method is quite similar to the ones we have shown in Figure 6.12 and 6.14. However, longer packets do take up more of the previous bandwidth [20].

6.3 Performance of the intrusion detection algorithm

Finally, we have considered the case where the 'randomness' of regular device traffic varies. Namely, in many sensing applications the arrival process of regular packets may be closer to true periodic arrivals than to random Poisson arrivals. To model this effect, the interarrival rate of packets for regular devices was modified to be

$$\frac{1}{t_i} = (1-\rho)\,\lambda + \rho\tau \tag{6.6}$$

where τ denotes a random variable with negative exponential distribution and mean value of $1/\lambda$ (i.e,. the inverse of the mean of a Poisson process), while ρ is a 'randomness' coefficient. When $\rho = 1$, packet arrivals are pure Poisson, whereas for $\rho = 0$, packet arrivals are uniformly distributed with the frequency of λ; values in between those two extremes correspond to higher or lower 'randomness' of packet arrivals.

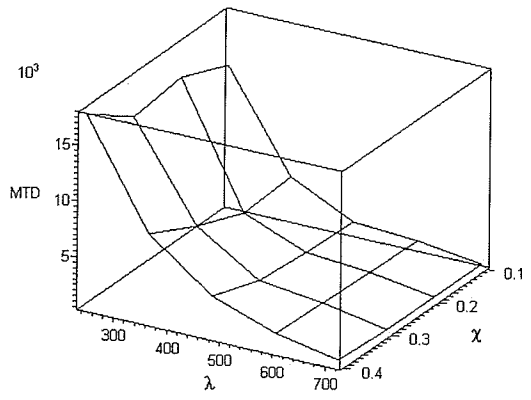
The same performance indicators as in the previous Section, i.e., the mean time to detect an attack, probability of missing a real attack or probability of detecting wrong device(i.e., the probability of false negatives and false positives), mean time between false alarms, and mean time to recover after an attack, are shown in Figure 6.17 and Figure 6.18. As before, regular devices generate Poisson traffic at the rate of 120 packets per minute

As can be seen, the mean time to detect an attack (Figure 6.18) does not depend very much on the value of ρ, whereas we see that with the different arrival of the

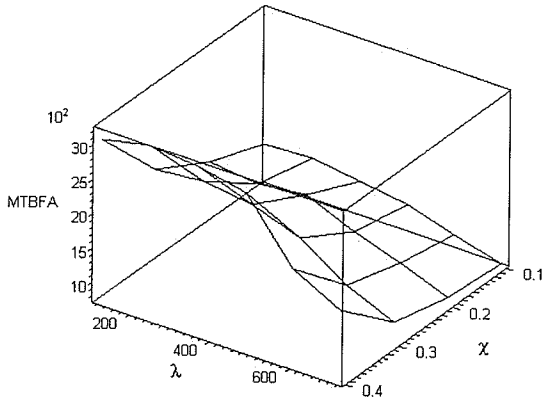
regular devices the probability of detecting false positives and false negatives becomes lower (from Figure 6.17).

6.4 Summary

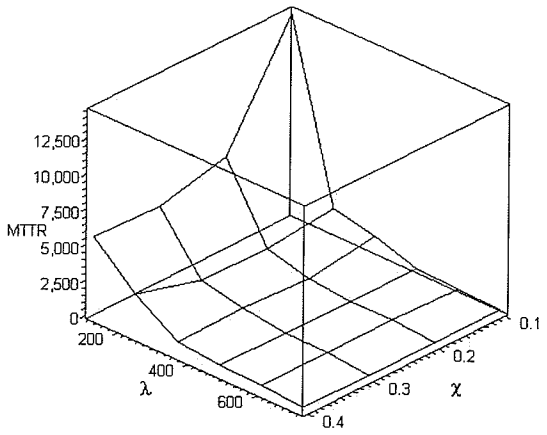
In this Section, we have introduced a simple, traffic based intrusion detection method that requires modest resource requirements. We have experimentally found the required parameter values that give a reasonable tradeoff between the probabilities of getting false positives and false negatives. We want to stress that the computational and memory requirements of the intrusion detection mechanism are kept at an absolute minimum, through the use of the exponentially weighted moving averages, which means that the proposed mechanism is feasible for use in a wireless sensor network environment where individual devices have severe resource limitations.



(a) Mean time to detect a true attack (MTD)

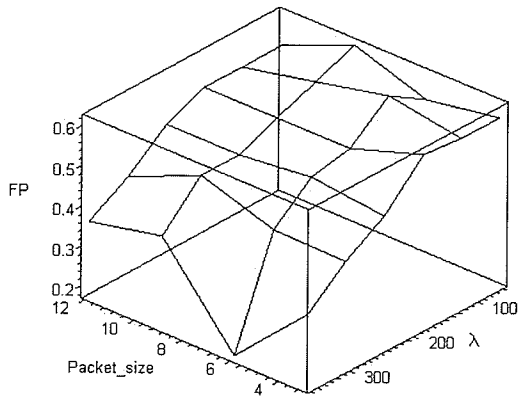


(b) Mean time between false alarms (MTBFA)

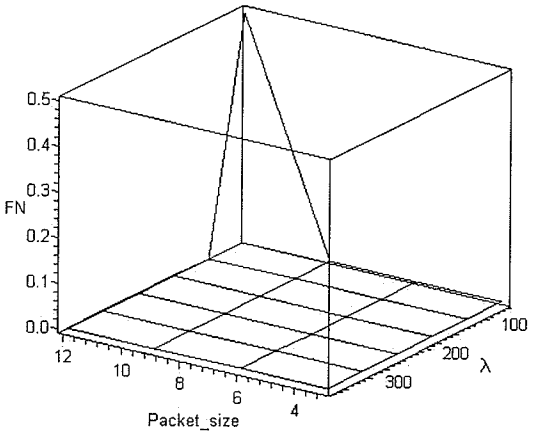


(c) Mean time to recovery from a true attack (MTTR)

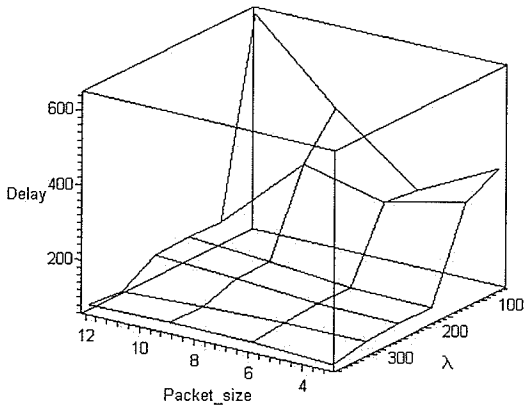
Figure 6.14: Additional performance indicators with different χ values and different attacker arrival rates.



(a) Probability of detecting false positives

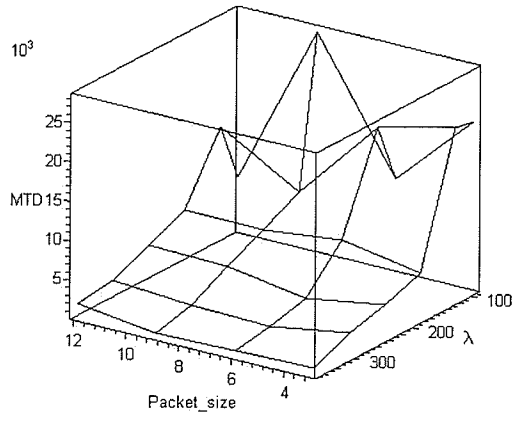


(b) Probability of detecting false negatives

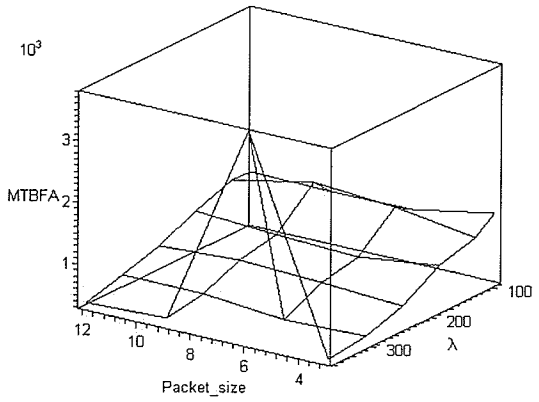


(c) Detection delay (in packet times)

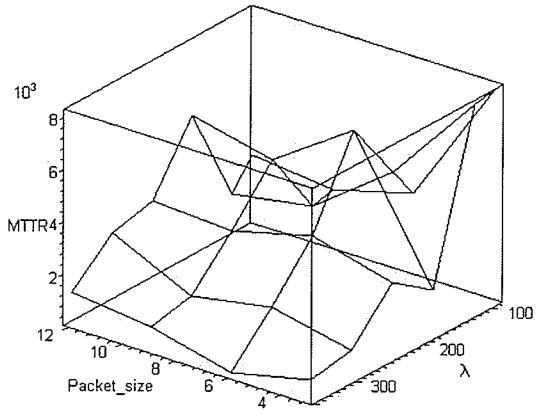
Figure 6.15: The probability of detecting false positives, false negatives and the incurred delay to detect the attacker with different *packetsizes* and different attackers arrival rate(by our modified method)



(a) MTD

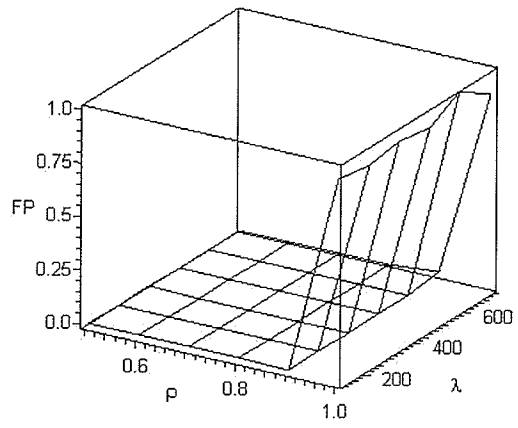


(b) MTBFA

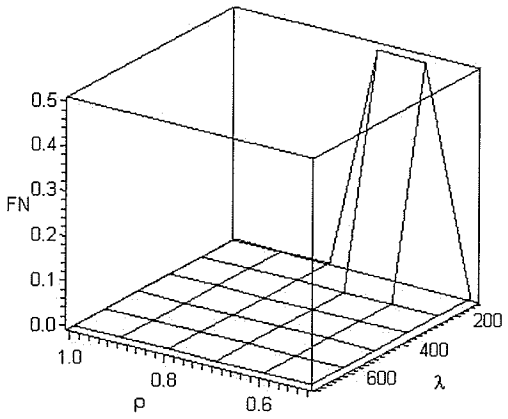


(c) MTTR

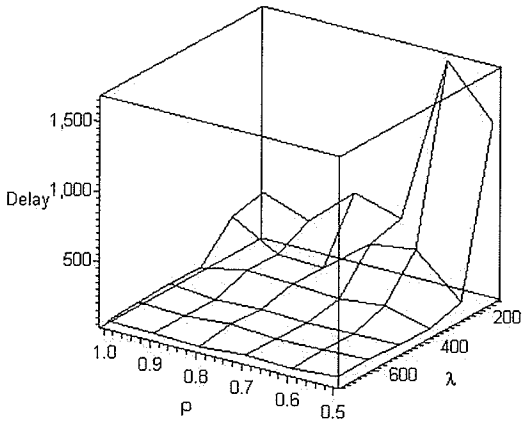
Figure 6.16: The MTD, MTBFA and MTTR with different *packetsizes* and different attackers arrival rate(by our modified method)



(a) Probability of detecting false positives

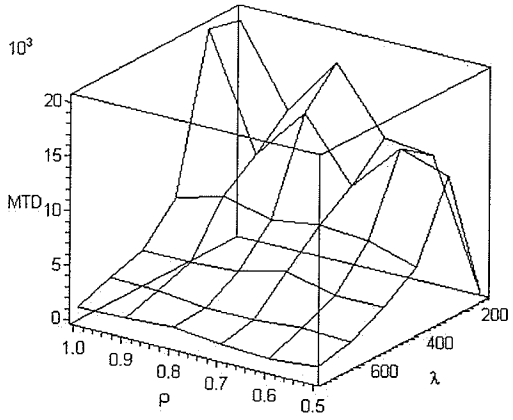


(b) Probability of detecting false negatives

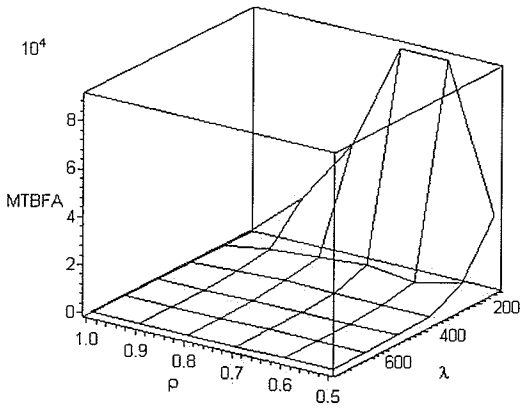


(c) Detection delay (in packet times)

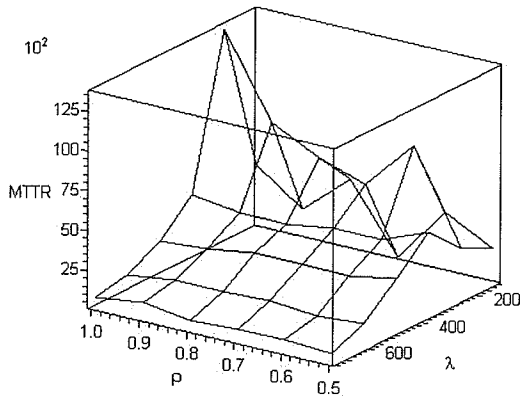
Figure 6.17: The probability of detecting false positives, false negatives and the incurred delay to detect the attacker with different ρ and different attacker's arrival rate.



(a) MTD



(b) MTBFA



(c) MTTR

Figure 6.18: The MTD, MTBFA and MTTR with different ρ and different attacker's arrival rate.

Chapter 7

Conclusion and Future Work

The security in the MAC layer of a wireless sensor network application is very important as the MAC layer has special vulnerabilities compared to the other layers of the network. This is because, the MAC layer manages and maintains communications among the sensor nodes. Also, in WSN, the shared medium is open for all and hence it makes easy to induce different sorts of attacks by exploiting the medium access control protocol employed in the network. Moreover, MAC layer anomalies may be caused by malicious operations of the upper layers and not just the MAC layer operations. Although MAC layer provides several major security functions, such as authentication, authorization, data encryption, we cannot always guarantee the availability of WSN service and resources with the help of these security solutions. Moreover, in WSN each application is distinct and hence, the underlying MAC protocol would be distinct to cope with the application requirements. Therefore, we need to study the potential MAC protocol for future WSN application with the perspective of ensuring security.

In this research, we have identified a number of possible MAC layer vulnerabilities in WSN by analyzing the IEEE 802.15.4 sensor network, which is recently recommended for low bit-rate WPAN and hence, a prospective WSN. All the security challenges of WSNs are also well applicable for the 802.15.4 compliant WSNs and indeed are very vulnerable to attacks in hostile environments. Therefore, any sensor network application that will deploy the IEEE 802.15.4 MAC protocol must account for these security vulnerabilities to ensure a security in a targeted application.

We also developed a simple traffic volume-based intrusion detection technique for IEEE 802.15.4 sensor clusters operating in beacon enabled, slotted CSMA-CA mode and propose defense strategies to guard against the attacks. For our intrusion detection method, we use the MAC layer features such as the packet arrival rate of the sensor nodes, packet size, packet inter-arrival times to build the statistical model using the Exponential Weighted Moving Average (EWMA) method in the PAN coordinator in the network. We have also introduced a small hysteresis in the decision process in order to avoid false alarms due to dithering. We have shown that the powerful PAN coordinator is able to detect the MAC layer intruders that induces simulated attacks with a fewer number of false positives and less delay. The EWMA based intrusion detection technique helps us to minimize the memory and power use of the PAN coordinator, since we do not need any buffer maintenance to record the packet arrival time and the packet inter-arrival time. Since the network uses CSMA-CA, the results can easily be generalized to other CSMA-based protocols of WSN. The results show that such an approach is indeed possible, although the attacks which inject spurious traffic at rates close to that of regular devices are rather difficult to detect.

Now, the ultimate question that may arise after we found that there is an ongoing attack going on in the cluster is, what is the next action? There could be several options for further actions. First, the coordinator may decide to switch to a different channel in order to alleviate the attack and inform all the compromised devices accordingly. This procedure would be very appropriate when there is a secure communication channel, possibly with separate encryption to each of those devices. Second, the coordinator may simply inform all the network devices that this particular cluster has been compromised; therefore, the corresponding sensing application may be shut down by a human operator as being alerted by the application. Also, the nodes could be forced to switch to sleep mode and do not participate in any communication for a certain period, which is suggested in [26], after an attack is detected. However, we have not considered these possible actions in this work. Our future work on this thesis might include these further actions after the intrusion is detected. Also, we plan to analyze the impact of the possible attacks that we found on a IEEE 802.15.4 compliant network where both the uplink and downlink channel are implemented and enhance our intrusion detection technique to include all the features of such network. Furthermore, we will focus on the behavior of the algorithm in case of regular traffic, attacker traffic, or both of them follow a non-Poisson distribution and we will also investigate possible ways in which a sensor cluster can alleviate an ongoing attack. We believe that our work will contribute to obtain detail findings of the security threats while executing a MAC protocol in future WSN applications as well as presently unknown attacks in IEEE 802.15.4 compliant networks. These new findings may help show why security measures should be considered immediately and profoundly.

Bibliography

[1] N. Abramson. Development of the alohanet. *IEEE Transactions on Information Theory*, 31(2):119–123, March 1985.

[2] I. F. Akyildiz and I. H. kasimoglu. Wireless sensor and actor networks: research challenges. *Ad Hoc Networks Journal(Elsevier)*, 2:351–367, October 2004.

[3] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci. Wireless sensor networks: a survey. *IEEE Communications Magazine*, 40(8):102–114, August 2002.

[4] J. P. Anderson. Computer security threat monitoring and surveillance. Technical report, James. P. Anderson Co., Fort Washington, Pennsylvania, 1980.

[5] J. Bellardo and S. Savage. 802.11 denial-of-service attacks: Real vulnerabilities and practical solutions. In *Proc. 12th USENIX Security Symposium*, August 2003.

[6] E. Callaway, P. Gorday, L. Hester, José A. Gutiérrez, M. Naeve, B. Heile, and V. Bahl. Home networking with IEEE 802.15.4: A developing standard for low-

rate wireless personal area networks. *IEEE Communications Magazine*, 40(28): 70–77, August 2002.

[7] E. H. Callaway, Jr. *Wireless Sensor Networks, Architecture and Protocols.* Auerbach Publications, Boca Raton, FL, 2004.

[8] H. Chan and A. Perrig. Security and privacy in sensor networks. *IEEE Computer*, pages 103–105, October 2003.

[9] V. Gupta, S. Krishnamurthy, and M. Faloutsos. Denial of service attacks at the MAC layer in wireless ad hoc networks. In *MILCOM 2002*, July 2002.

[10] J. A. Gutiérrez, E. H. Callaway, Jr., and R. L. Barrett, Jr. *Low-Rate Wireless Personal Area Networks.* IEEE Press, New York, NY, 2004.

[11] IEEE 802.11. LAN MAN standards committee of the ieee computer society, wireless lan medium access control (mac) and physical layer (phy) specification. IEEE standard 802.11, IEEE, New York, NY, 1999.

[12] IEEE 802.15.4. Standard for part 15.4: Wireless MAC and PHY specifications for low rate WPAN. IEEE standard 802.15.4, IEEE, New York, NY, October 2003.

[13] O. Kachirski and R. Guha. Effective intrusion detection using multiple sensors in wireless ad hoc networks. In *Proc. 36th Annual Hawaii International Conference on System Sciences (HICSS)*, January 2003.

[14] C. Karlof, N. Sastry, and D. Wagner. TinySec: A link layer security architecture

for wireless sensor networks. In *Proc. ACM SenSys*, pages 162–175, November 2004.

[15] C. Karlof and D. Wagner. Secure routing in sensor networks: Attacks and countermeasures. *Special Issue on Sensor Network Applications and Protocols*, 1: 293–315, September 2003.

[16] M. Khan, Fereshteh Amini, Jelena Mišić, and Vojislav B. Mišić. The cost of security: performance of ZigBee key exchange mechanism in an 802.15.4 beacon enabled cluster. In *Proc. WSNS'06*, Vancouver, CA, October 2006.

[17] L. Kleinrock and F. Tobagi. Packet switching in radio channels:part i - carrier sense multiple access modes and their throughput delay characteristics. *IEEE Transactions on Communications*, 23(12):1400–1416, December 1975.

[18] C. Lu, B. Blum, T. Abdelzaher, J. Stankovic, and T. He. RAP: A real-time communication architecture for large-scale wireless sensor networks. In *Proc. 8th Real-Time and Embedded Technology and Application Symp.(RTAS)*, 2002.

[19] J. Mišić, S. Shafi, and V. B. Mišić. Analysis of 802.15.4 beacon enabled PAN in saturation mode. In *Proc. SPECTS 2004*, San Jose, CA, July 2004.

[20] J. Mišić, S. Shafi, and V. B. Mišić. Avoiding the bottlenecks in the mac layer in 802.15.4 low rate wpan. In *Proc. HWISE2005*, volume 2, pages 363–367, Fukuoka, Japan, 2005.

[21] J. Mišić, S. Shafi, and V. B. Mišić. Performance of beacon enabled ieee 802.15.4 pan with downlink and uplink traffic. In *Proc. 4th International IFIP TC6 Net-*

working Conference NETWORKING 2005, pages 228–239, Waterloo, Ontario, May 2005.

[22] V. B. Mišić, J. Begum, and J. Mišić. MAC layer security issues in 802.15.4 sensor networks. In *Proc. Workshop on Wireless Networks and Communications Systems(WiNCS)*, Philadelphia, PA, July 2005.

[23] J. Newsome, R. Shi, D. Song, and A. Perrig. The Sybil attack in sensor networks: Analysis and defenses. In *Proc. IEEE International Conference on Information Processing in Sensor Networks (IPSN)*, April 2004.

[24] I. Onat and A. Miri. A real-time node-based anomaly detection algorithm for wireless sensor networks. In *Proc. International Conference on Sensor Networks*, pages 422–427, 2005.

[25] A. Perrig, D. Wagner, and J. Stankovic. Security in wireless sensor networks. *Communications of the ACM*, 47(6):53–57, June 2004.

[26] Q. Ren and Q. Liang. Secure media access control (MAC) in wireless sensor networks: Intrusion Detections and Countermeasures. In *Proc. 15th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*, volume 4, pages 3025–3029, September 2004.

[27] RSoft Design, Inc. *Artifex v.4.4.2.* San Jose, CA, 2003.

[28] N. Sastry and D. Wagner. Security considerations for IEEE 802.15.4 networks. In *Proc. ACM Workshop on Wireless Security (WiSe)*, October 2004.

[29] E. Shi and A. Perrig. Designing secure sensor network. *IEEE Wireless Communications*, pages 38–43, December 2004.

[30] W. Stallings. *Network Security Essentials: Applications and Standards*. Prentice-Hall, New Jersey, 2000.

[31] A. D. Wood and J. A. Stankovic. Denial of service in sensor networks. *IEEE Computer*, 35(10):54–62, October 2002.

[32] W.Ye and J. Heidemann. Medium access control in wireless sensor networks. Technical Report ISI-TR-580, USC/Information Sciences Institute, October 2003. URL `http://www.isi.edu/~johnh/PAPERS/Ye03c.html`.

[33] Y. Zhang and W. Lee. Intrusion detection in wireless adhoc networks. In *Proc. Mobile Computing and Networking*, pages 275–283, 2000.

[34] Y. Zhou, D. Wu, and S. Nettles. Analyzing and preventing MAC-layer denial of service attacks for stock 802.11 systems. In *Proc. First Annual International Conference on Broadband Networks*, pages 22–88, San José, CA, October 2004.