Attacking Beacon-Enabled 802.15.4 Networks

Sang Shin Jung, Marco Valero, Anu Bourgeois, and Raheem Beyah

CAP Research Group
Department of Computer Science, Georgia State University
Atlanta, GA 30303, USA
{sangsin,mvalero,abourgeois,rbeyah}@cs.gsu.edu
home page: http://www.cs.gsu.edu/cap

Abstract. The IEEE 802.15.4 standard has attracted time-critical applications in wireless sensor networks (WSNs) because of its beacon-enabled mode and guaranteed time slots (GTSs). However, the GTS management scheme's built-in security mechanisms still leave the 802.15.4 MAC vulnerable to attacks. Further, the existing techniques in the literature for securing 802.15.4 either focus on non beacon-enabled mode 802.15.4 or cannot defend against insider attacks for beacon-enabled mode 802.15.4. In this paper we illustrate this by demonstrating attacks on the availability and integrity of a beacon-enabled 802.15.4 sensor network. To confirm the validity of the attacks, we implement the attacks using Tmote Sky Motes for sensor nodes, where the malicious node is deployed as an inside attacker. We show that the malicious node can easily exploit information retrieved from the beacon frames to compromise the integrity and availability of the network. We also discuss possible defense mechanisms against these attacks.

Key words: Insider attacks, Beacon-enabled 802.15.4, wireless sensor networks, MAC misbehavior

1 Introduction

Wireless sensor networks (WSNs) have emerged quickly and attracted a number of diverse applications. The use of these applications ranges from residential to government. For example, AlertMe home monitoring [1] is a residential system that enables secure indoor and outdoor home environment monitoring with simple contact and passive infrared (PIR) sensors. If AlertMe detects intruders, it immediately reports the intrusion to the homeowner. The military is also using WSNs to detect an adversary's behavior and location. For example, seismic sensors can be used to detect the movement of heavy artillery (e.g., tanks) in the battlefield. In either case, not receiving information about the environment in a time-sensitive manner can have significant consequences. To provide support for time-sensitive communication, the IEEE 802.15.4 standard provides a beacon-enabled mode. Unlike non beacon-enabled mode, the beacon-enabled mode in 802.15.4 networks facilitates real-time delivery of data using the GTS management scheme during the contention free period (CFP) [2, 3, 4, 5, 6, 7]. In the beacon-enabled mode, a centralized node (i.e., personal area network (PAN) coordinator) broadcasts beacon frames to synchronize the nodes in the network, manages GTS allocation/de-allocation requests from the end devices, and assigns dedicated slots for transmission from these nodes. Many researchers have focused on improving the performance or energy efficiency of beacon-enabled 802.15.4. For example, the IPP-HURRY research group has analyzed the delay bound of GTS allocation to maximize the throughput of each GTS allocation for real-time sensor networks [3, 4]. In addition, in [5] the authors present a case study of Siemens Industry Automation Division that requires real-time delivery of short alarms/messages. The case study evaluates GTS allocation to maximize low latency of its scheme. Although there has been a significant emphasis on improving the performance of the beacon-enabled 802.15.4 protocol, there has been little work on securing this mode of the 802.15.4 protocol. This is significant, given that the GTS management scheme in beacon-enabled 802.15.4 networks does not verify the ID of each sensor node that requests GTSs. Therefore, an inside attacker can easily compromise the guaranteed data transmissions from the time-sensitive applications in the beacon-enabled network by either impersonating existing legitimate nodes' IDs or creating IDs for nodes that do not exist (i.e., implement a Sybil attack [8] at the MAC layer).

In this paper, we demonstrate four attacks that are possible by an inside attacker who impersonates legitimate nodes or generates multiple fake IDs. This is accomplished by the inside attacker targeting the vulnerabilities of the GTS management scheme in a beacon-enabled 802.15.4 network. The contributions of this paper include the discovery of vulnerable properties of the beacon-enabled mode in the IEEE 802.15.4 standard, the implementation and analysis of four potential insider attacks associated with those vulnerabilities, and the presentation of defense mechanisms against the attacks.

The rest of this paper is organized as follows. We review some related works including several security protocols for WSNs and attacks on beacon-enabled IEEE 802.15.4 in Section 2. In Section 3, we explain the GTS management scheme and its vulnerabilities. In Section 4, we define the network and attack model used to implement four potential attacks. In Section 5, we introduce our four attacks against the GTS management scheme. In Section 6, we describe the implementation of the attacks. In Section 7, we show the result of each attack based on the collected data. We briefly mention possible defenses against these attacks in Section 8 and conclude our work in Section 9.

2 Related Work

In this section we categorize current 802.15.4 defense mechanisms into beaconless mode and beacon-enabled mode according to the literature and highlight their limitations. We also discuss the difference between our attacks on beacon-enabled 802.15.4 networks and others previously demonstrated.

Defense Mechanisms in Beacon-Less Mode

In [9, 10, 11], the received signal strength indication (RSSI) was proposed to identify nodes conducting a Sybil attack. The basic idea of RSSI-based methods is that sensor nodes at different locations can be differentiated by the different RSSIs. In [10], M. Demirbas et al. calculate the ratio of RSSIs to improve traditional RSSI-based solutions. In [9], J. Yang et al. propose K-means cluster analysis that can be applied to RSSI readings. However, RSSI-based solutions can be evaded by malicious nodes with mobility. Another defense method is a cryptographic approach. Most of these approaches presents either light-weight methods such as light-weight identity certificates [12] or key distribution and management algorithms [13, 14, 15, 16] to distinguish between legitimate nodes and malicious nodes using multiple stolen or forged IDs. However, it is not practical for resource constrained sensor devices to utilize highly expensive key distribution methods. Some link layer secure protocols such as SPINS, TinySec, and MiniSec [17, 18, 19] respectively are designed specifically for energy constrained sensor nodes and provide data authentication and secrecy at the link layer. However, these protocols are susceptible to failures when a compromised node in the network acquires a shared pair-wise or network-wide secret key. Although the aforementioned protocols have merit, they do not apply to beacon-enabled 802.15.4 networks. Further, they cannot be directly applied to beacon-enabled mode because it utilizes many different features such as time-sensitive GTSs.

Defense Mechanisms in Beacon-Enabled Mode

Few defense methods have been proposed for beacon-enabled mode. One RSSI-based solution for beacon-enabled mode was proposed by F. Amini et al. in [11]. The authors proposed an RSSI solution where they introduced the use of a disc number and a device ID. However, if a malicious node is close enough to a legitimate node in the same personal area network (PAN), its RSSI may be confused with the RSSI of the legitimate node. The IEEE 802.15.4 standard [20] also has built-in security features to provide data secrecy and data authenticity. However, in [21], N. Sastry et al. point out that these security features have vulnerabilities related to the initial vector (IV) management, key management, and integrity protection. Another link layer secure protocol implementation for beacon-enabled mode was presented in [22]. Alim et al. introduce EAP-Sens which provides entity authentication and key management to validate each device ID with an extensible authentication protocol (EAP) [23] and EAP-generalized pre-shared key (EAP-GPSK) [24]. Even though Alim et al. mention that EAP-Sens is not vulnerable to a man-in-the-middle attack due to its shared key method, EAP-Sens is still vulnerable to attacks when there is an inside attacker. Overall, neither the aforementioned detection mechanisms nor secure link layer protocols in beacon-enabled mode are effective in the case of compromised nodes acting as inside attackers.

Attacks on Beacon-Enabled 802.15.4 Networks
In [25], R. Sokullu et al. use ns-2 simulations to demonstrate GTS attacks on the 802.15.4 MAC, particularly in beacon-enabled mode. The GTS attacks were divided into four different scenarios: One Intelligent Attacker (OIA), One Random Attacker (ORA), Two Intelligent Attackers (TIAs), and Two Random Attackers (TRAs). Both the OIA and TIAs scenarios target the maximum number of GTSs assigned to one legitimate node. In contrast, the ORA and TRAs scenarios attack just one randomly chosen GTS. The main goal of the GTS attacks in [25] is to create collisions during the CFP to deny the use of GTSs. In contrast, our four attacks seek to exploit the beacon-enabled 802.15.4 MAC by providing scenarios of unfairness and exhaustion [26, 27].

In addition to presenting different types of attacks compared to those discussed in [25], our attacks were implemented on real devices (i.e., Tmote Sky Motes) rather than in simulation. This latter point is extremely important for 802.15.4 MAC layer attacks, because in addition to the challenge of accurately modeling physical layer interference, simulations do not take into account constraints imposed by the hardware, operating system, and applications, which can lead to simplified attack scenarios. This is especially pronounced in resource-constrained devices (e.g., Tmote Sky Motes). For example, to implement the Sybil attack (at the MAC layer) in TinyOS, we modified the timer function of TinyOS (in TimerC.nc) to make it multithreaded so each fake node could use an instance. Each instance now has to compete internally (within TinyOS) to gain access to the node's resources (e.g., processor, transceiver), making this attack much more difficult to conduct. This small, but noticeable nuance is not present in simulation tools.

3 Problem Statement

In this section, we briefly explain the GTS management scheme of the IEEE 802.15.4 standard and we state three vulnerabilities of the scheme.

3.1 GTS Management Scheme

The IEEE 802.15.4 standard [20] operating in beacon-enabled mode defines the superframe (SF) that consists of contention access period (CAP), contention free period (CFP), and inactive period as shown in Figure 1. According to the standard, the personal area network (PAN) coordinator periodically transmits beacon frames at intervals defined by the *aBeaconOrder* variable. The beacon frames contain the number of GTSs and these directions used by nodes to transmit data during the CFP. The structure of the beacon frame and the GTS field are shown in Figure 2 (a) and (b) respectively. As shown in Figure 1, the PAN coordinator defines that each superframe can have maximum of seven GTSs for the CFP other than $aMinCAPLength$ in [20]. The slots of GTSs must be

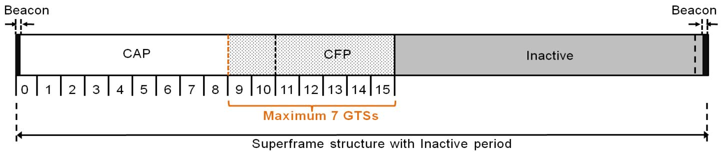


Fig. 1. GTSs in Superframe structure.

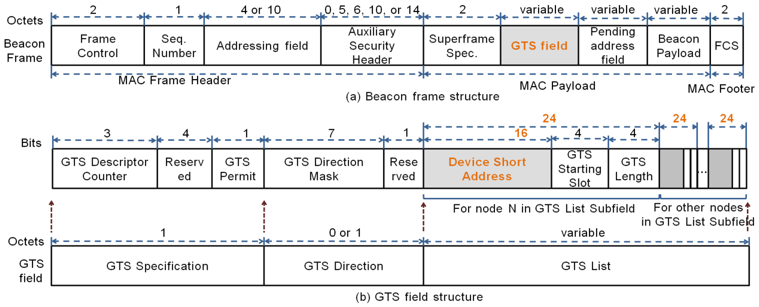


Fig. 2. Details of MAC frame structure: (a) beacon frame structure and (b) GTS field structure in beacon frame.

assigned to legitimate nodes issuing GTS allocation requests to the PAN coordinator. Then, the assigned slots should be released by the PAN coordinator after receiving a GTS deallocation request from the same legitimate node.

Below we briefly explain the normal GTS allocation and deallocation processes.

GTS Allocation: If a legitimate node has data to transmit, it generates a GTS allocation request. The PAN coordinator will allocate an available GTS to the legitimate node, and all subsequent beacon frames will contain the GTS descriptor defining the device address, GTS slot and direction. Upon receiving the beacon with the GTS descriptor, the legitimate node will schedule the pending packet to be transmitted at the allocated GTS. The GTS allocation process is shown in Figure 3.

GTS Deallocation: The GTS deallocation occurs after the GTS descriptor has been transmitted for $aGTSDescPersistenceTime$ beacons by the PAN coordinator or when the legitimate node using the GTS sends an explicit GTS deallocation request. The GTS deallocation process is shown in Figure 3.

3.2 Vulnerabilities of GTS Management Scheme

The PAN coordinator manages a list of GTSs to control the network access during the CFP. However, the GTS management scheme has the following vulnerabilities.

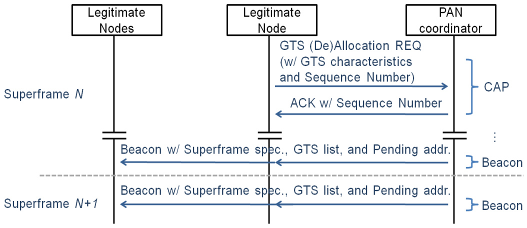

Legitimate Nodes
Legitimate Node
PAN coordinator
GTS (De)Allocation REQ (w/ GTS characteristics and Sequence Number)
CAP
ACK w/ Sequence Number
Superframe N
Beacon w/ Superframe spec., GTS list, and Pending addr.
Beacon
Superframe N+1
Beacon w/ Superframe spec., GTS list, and Pending addr.
Beacon

Fig. 3. GTS allocation and deallocation procedure.

CAP Maintenance: According to the IEEE 802.15.4 standard, the PAN coordinator can perform several preventative actions to keep $aMinCAPLength$. One of these actions is to deallocate unused GTSs within every $2 * n$ SFs, where n is defined as either $2^{(8-macBeaconOrder)}$ $(0 \leq macBeaconOrder \leq 8)$ or $(9 \leq macBeaconOrder \leq 14)$. However, if a malicious node keeps constantly sending either GTS requests or data at the assigned GTSs during the CFP, the preventative action is ineffective.

Verification of Sensor Nodes' IDs: In the 802.15.4 GTS management scheme, the PAN coordinator manages the Identities (IDs) of legitimate nodes requesting one or more GTSs. The PAN coordinator assigns GTSs to the nodes, deallocates the assigned slots, and avoids duplicated GTS requests from the same legitimate node. However, as shown in Figure 4 the PAN coordinator only checks the sensor nodes' IDs (a short 2-octet address) and the sequence number of the packets. Thus, a malicious node can easily evade the verification process for sensor nodes' IDs by using new forged IDs or impersonating legitimate nodes in the network.

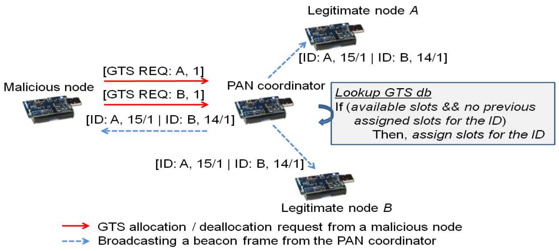

Legitimate node A
[ID: A, 15/1 | ID: B, 14/1]
[GTS REQ: A, 1]
[GTS REQ: B, 1]
Malicious node
PAN coordinator
[ID: A, 15/1 | ID: B, 14/1]
Lookup GTS db
If (available slots && no previous assigned slots for the ID)
Then, assign slots for the ID
[ID: A, 15/1 | ID: B, 14/1]
Legitimate node B
GTS allocation / deallocation request from a malicious node
Broadcasting a beacon frame from the PAN coordinator

Fig. 4. A malicious node impersonating the IDs of legitimate node A and B.

4 Experiment Design

In this section, we present the network design, the attack model, and the hardware and software components used in this work.

4.1 Network Design

In this paper, we use sensor nodes supporting the IEEE 802.15.4 standard in beacon-enabled mode. The nodes performing legal activities in the network are called legitimate nodes, while the bad nodes are called malicious nodes. The nodes are organized in a cluster which has a base station (i.e., PAN coordinator) collecting messages from each sensor. We use Tmote Sky Motes [28] as sensor nodes and PAN coordinator. Tmote Sky Mote has a CC2420 radio chip [29] and supports the 802.15.4 standard [20] in both beacon-less and beacon-enabled mode.

4.2 Attack Model

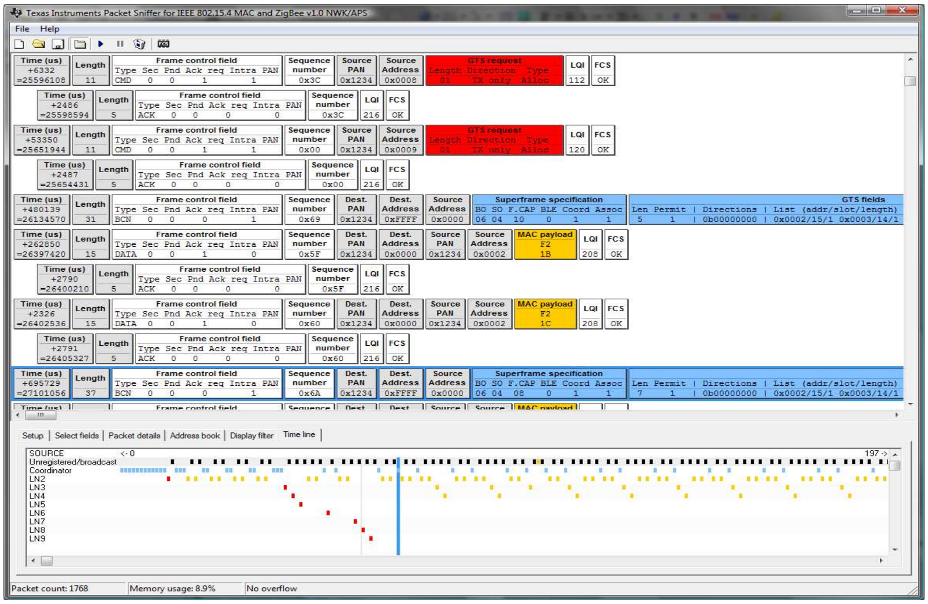
Similar to the threat models defined in [26] and [30], we assume that a malicious node behaves badly as a mote-class, inside, and active attacker. As a mote-class adversary, a malicious node has the same capabilities as that of any legitimate node. Therefore, we use Tmote Sky Motes for the malicious node. As an inside and active attacker, a malicious node listens to broadcasting beacons and interferes with the communication between legitimate nodes and the PAN coordinator.

4.3 Hardware and Software Components

We used four Tmote Sky Motes [28]: one PAN coordinator, two legitimate nodes, and one malicious node. Our attack experiments use the IEEE 802.15.4 open-ZB open source implementation [31]. In particular, we used version 1.2 of the source code in conjunction with TinyOS v1.15 [32]. In addition, we used the Texas Instruments (TI) CC2420 Evaluation Board/Evaluation Module (EB/EM) [33] in conjunction with the TI Chipcon packet sniffer [34] to capture and analyze packet traffic in the network. Only four nodes were used because the open source implementation used became unstable above four nodes in the network. However, it is important to note that these attacks are *independent* of the number of nodes deployed in the network. Figure 5 shows examples of captured packets from the TI Chipcon packet sniffer. Figure 6 shows Tmote Sky Motes and CC2420 EB/EM.

5 Overview of Attacks

We divided the four attacks into two categories depending on the types of IDs that the malicious node uses to perform the illicit activities. The first category is *existing IDs* in the PAN where a malicious node uses the ID of a legitimate node in the PAN. The second category is *non-existing IDs* in the PAN where malicious nodes use any non-existing ID in the PAN and pretend to be newly deployed nodes in the network. In the former category, the malicious node can affect exhaustion of legitimate nodes. In the latter, it causes exhaustion and unfairness against legitimate nodes.



```
Time (us)  | Length | Frame control field                | Sequence | Source | Source  | GTS request                 | LQI | FCS
+6332      |        | Type Sec Pnd Ack req Intra PAN     | number   | PAN    | Address | Length Direction  Type      |     |
=25596108  | 11     | CMD   0   0     1        1         | 0x3C     | 0x1234 | 0x0008  | 01     TX only    Alloc     | 112 | OK

   Time (us) | Length | Frame control field                | Sequence | LQI | FCS
   +2486     |        | Type Sec Pnd Ack req Intra PAN     | number   |     |
   =25598594 | 5      | ACK   0   0     0        0         | 0x3C     | 216 | OK

Time (us)  | Length | Frame control field                | Sequence | Source | Source  | GTS request                 | LQI | FCS
+53350     |        | Type Sec Pnd Ack req Intra PAN     | number   | PAN    | Address | Length Direction  Type      |     |
=25651944  | 11     | CMD   0   0     1        1         | 0x00     | 0x1234 | 0x0009  | 01     TX only    Alloc     | 120 | OK

   Time (us) | Length | Frame control field                | Sequence | LQI | FCS
   +2487     |        | Type Sec Pnd Ack req Intra PAN     | number   |     |
   =25654431 | 5      | ACK   0   0     0        0         | 0x00     | 216 | OK

Time (us)  | Length | Frame control field                | Sequence | Dest.  | Dest.   | Source  | Superframe specification        | GTS fields
+480139    |        | Type Sec Pnd Ack req Intra PAN     | number   | PAN    | Address | Address | BO SO F.CAP BLE Coord Assoc     | Len Permit | Directions | List (addr/slot/length)
=26134570  | 31     | BCN   0   0     0        1         | 0x69     | 0x1234 | 0xFFFF  | 0x0000  | 06 04 10     0    1     1       | 5    1     | 0b00000000 | 0x0002/15/1 0x0003/14/1

Time (us)  | Length | Frame control field                | Sequence | Dest.  | Dest.   | Source | Source  | MAC payload | LQI | FCS
+262850    |        | Type Sec Pnd Ack req Intra PAN     | number   | PAN    | Address | PAN    | Address | F2          |     |
=26397420  | 15     | DATA  0   0     1        0         | 0x5F     | 0x1234 | 0x0000  | 0x1234 | 0x0002  | 1B          | 208 | OK

   Time (us) | Length | Frame control field                | Sequence | LQI | FCS
   +2790     |        | Type Sec Pnd Ack req Intra PAN     | number   |     |
   =26400210 | 5      | ACK   0   0     0        0         | 0x5F     | 216 | OK

Time (us)  | Length | Frame control field                | Sequence | Dest.  | Dest.   | Source | Source  | MAC payload | LQI | FCS
+2326      |        | Type Sec Pnd Ack req Intra PAN     | number   | PAN    | Address | PAN    | Address | F2          |     |
=26402536  | 15     | DATA  0   0     1        0         | 0x60     | 0x1234 | 0x0000  | 0x1234 | 0x0002  | 1C          | 208 | OK

   Time (us) | Length | Frame control field                | Sequence | LQI | FCS
   +2791     |        | Type Sec Pnd Ack req Intra PAN     | number   |     |
   =26405327 | 5      | ACK   0   0     0        0         | 0x60     | 216 | OK

Time (us)  | Length | Frame control field                | Sequence | Dest.  | Dest.   | Source  | Superframe specification        |
+695729    |        | Type Sec Pnd Ack req Intra PAN     | number   | PAN    | Address | Address | BO SO F.CAP BLE Coord Assoc     | Len Permit | Directions | List (addr/slot/length)
=27101056  | 37     | BCN   0   0     0        1         | 0x6A     | 0x1234 | 0xFFFF  | 0x0000  | 06 04 08     0    1     1       | 7    1     | 0b00000000 | 0x0002/15/1 0x0003/14/1
```

Fig. 5. Captured packets from TI Chipcon packet sniffer.

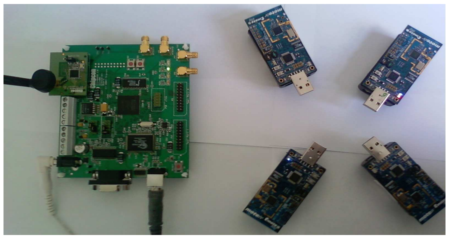


Fig. 6. Tmote Sky Motes and CC2420 EB/EM.

5.1 Existing Identities in the PAN

In this category, a malicious node impersonates the existing legitimate nodes in the PAN. The attack is of the form of DoS against data transmissions during the CFP. The idea is to block data transmission of legitimate nodes, which denies legitimate nodes requiring GTSs access to the link.

DoS against Data Transmissions During CFP

If a malicious node is in the transmission range of the PAN coordinator, it can eavesdrop on the messages sent by legitimate nodes and also intercept the beacons sent by the PAN coordinator. Since the beacons include the GTS list (Figure 2 (b)), the malicious node can recognize not only how many legitimate nodes are in the PAN, but also what legitimate nodes request and use GTSs to send data during the CFP. In this attack, a malicious node sends GTS deallocation requests using legitimate nodes' IDs to the PAN coordinator. Figure 7 shows an example of this attack where two legitimate nodes send GTS allocation requests before sending data during the CFP of the next SF. However, a malicious node knowing that the two nodes are in the GTS list can terminate the data transmissions of the legitimate nodes by sending a GTS deallocation request with the legitimate nodes' IDs.

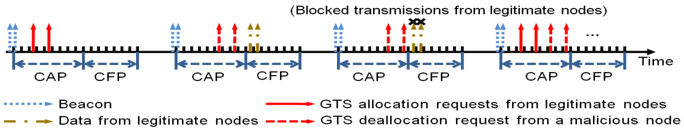

(Blocked transmissions from legitimate nodes)
...
Time
CAP
CFP
CAP
CFP
CAP
CFP
CAP
CFP
Beacon
Data from legitimate nodes
GTS allocation requests from legitimate nodes
GTS deallocation request from a malicious node

Fig. 7. A malicious node blocking a legitimate node sending data during CFP.

False Data Injection

While a legitimate node is not in the GTS list, a malicious node can send a GTS allocation request and try to send data using the legitimate node's ID. Having checked the node's IDs and sequence number, the PAN coordinator accepts the data sent by the malicious node that contain false information. Figure 8 shows how this attack works; if a legitimate node is transmitting current temperature data during the CAP, the malicious node sends a GTS allocation request with the spoofed ID, and pretends to be the legitimate node to inject false data during CFP.

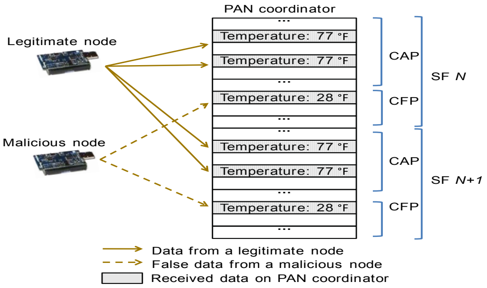


Fig. 8. A malicious node sending false temperature to the PAN coordinator.

5.2 Non-existing Identities in the PAN

In this category of attacks, a malicious node forges 7 different IDs depending on the maximum number of available GTSs. Two attacks herein perform exhaustion and unfairness attacks by occupying all 7 GTSs and not allowing legitimate nodes to reserve GTSs.

DoS against GTS Requests

To perform this attack, a malicious node keeps monitoring the available GTS slots with the intent of completely occupying them. Then, the attacker sends several GTS allocation requests to fill up all the available GTSs in the SF. The advantage of this attack is that the malicious node can reduce its energy consumption because once it occupies all 7 GTSs, it does not need to send out any data or commands. The malicious node simply dissects beacon frames to see if the PAN coordinator performs the preventative action for the CAP maintenance. Figure 9 shows that after legitimate node A and B send GTS deallocation requests, the malicious node completely fills all 7 GTSs with two additional GTS allocation requests. The goal of this attack is *not* for the attacker to use the bandwidth requested, rather it is to prevent the legitimate nodes from transmitting.

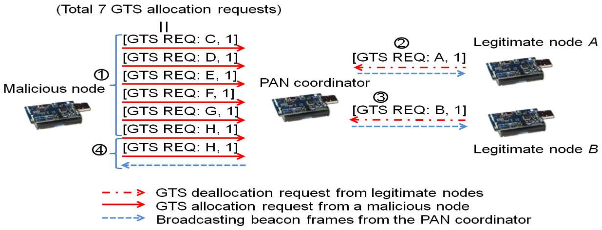


Fig. 9. A malicious node filling up all 7 GTSs. 1: the malicious node sends five GTS allocation requests. 2 and 3: legitimate node A and B send GTS deallocation requests. 4: the malicious node sends the rest of GTS allocation requests.

Stealing Network Bandwidth

Similar to the DoS against GTS requests, in this attack, an attacker observes the GTS list in order to eventually occupy the available GTS slots. However, in this attack, the malicious node sends data at the assigned time slots. The purpose of data transmission is to prevent the PAN coordinator from dropping the assigned GTSs. As shown in Figure 10, the second CFP has data transmitted from both legitimate nodes and a malicious node. However, since legitimate nodes send GTS deallocation requests during the second CAP, the malicious node sends a GTS allocation requests to occupy the new free GTS. Eventually, only the malicious node sends data during the fourth CFP. The time slots will never be vacant during the CFP of every SF, which can cause both exhaustion and unfairness against legitimate nodes. This also affects the PAN coordinator who cannot go into sleep mode (denial of sleep attack [35]).

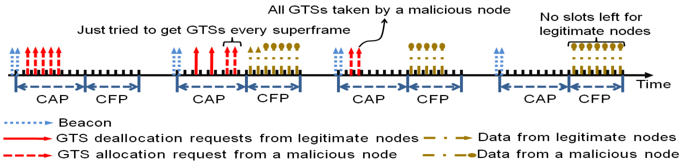

All GTSs taken by a malicious node
Just tried to get GTSs every superframe
No slots left for legitimate nodes
CAP
CFP
CAP
CFP
CAP
CFP
CAP
CFP
Time
Beacon
GTS deallocation requests from legitimate nodes
GTS allocation request from a malicious node
Data from legitimate nodes
Data from a malicious node

Fig. 10. A malicious node stealing all 7 GTSs during CFP.

6 Implementation of Attacks

6.1 Existing Identities in the PAN

We assume that there is one PAN coordinator, two legitimate nodes: LN2 and LN6, and one malicious node: MN4 as shown in Figures 11 and 12. MN4 impersonates the IDs of LN2 and LN6 after eavesdropping on beacon frames.

DoS against Data Transmissions During CFP

As shown in Figure 11, this attack works through two SFs. In the first SF, LN2 and LN6 send GTS allocation requests to the PAN coordinator to reserve one GTS. Then, the PAN coordinator broadcasts the beacon with the GTS list to inform LN2 and LN6 of their assigned slots. Along with LN2 and LN6, MN4 also receives the beacon. Therefore, MN4 knows how many legitimate nodes are in the GTS list and what their IDs are. In the second SF, MN4 sends GTS deallocation requests with the impersonated LN2 and LN6's IDs. The PAN coordinator removes LN2 and LN6 from the GTS list and will not receive data during the CFP of the next SF. Since LN2 and LN6 have no allocated GTSs anymore, they will not able to send their messages during the CFP of the third SF.

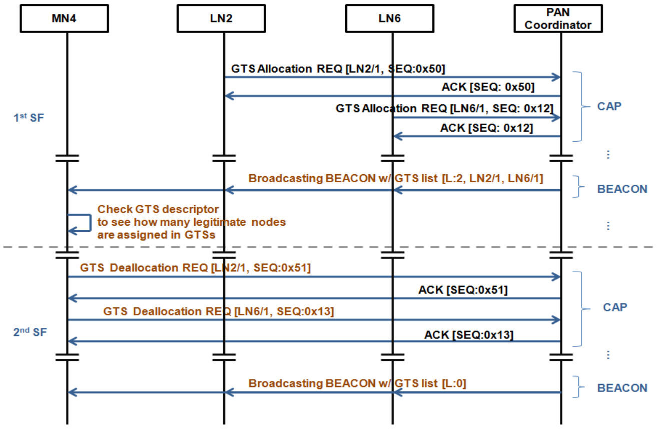

MN4
LN2
LN6
PAN Coordinator
GTS Allocation REQ [LN2/1, SEQ:0x50]
ACK [SEQ: 0x50]
GTS Allocation REQ [LN6/1, SEQ: 0x12]
ACK [SEQ: 0x12]
CAP
1st SF
Broadcasting BEACON w/ GTS list [L:2, LN2/1, LN6/1]
BEACON
Check GTS descriptor to see how many legitimate nodes are assigned in GTSs
GTS Deallocation REQ [LN2/1, SEQ:0x51]
ACK [SEQ:0x51]
GTS Deallocation REQ [LN6/1, SEQ:0x13]
ACK [SEQ:0x13]
CAP
2nd SF
Broadcasting BEACON w/ GTS list [L:0]
BEACON

Fig. 11. The sequences of DoS against Data Transmissions During CFP.

False Data Injection

Unlike DoS against Data Transmissions During CFP, this attack exploits GTS allocation requests to transmit false data. Figure 12 shows such a case that LN2 has already been assigned to one GTS. In this case, MN4 starts after LN2 sends a GTS deallocation request in the first SF. Then, the PAN coordinator removes LN2's ID on the GTS list of the next beacon. Since MN4 is aware that LN2 is not in the GTS list, it immediately tries to get one GTS by sending a GTS allocation request with LN2's ID. Once MN4 successfully takes the GTS, it starts sending false data with LN2's ID in the third SF.

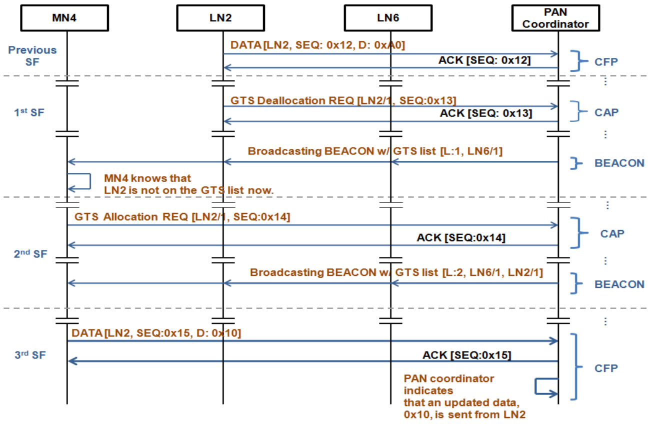

MN4
LN2
LN6
PAN Coordinator
Previous SF
DATA [LN2, SEQ: 0x12, D: 0xA0]
ACK [SEQ: 0x12]
CFP
1st SF
GTS Deallocation REQ [LN2/1, SEQ:0x13]
ACK [SEQ: 0x13]
CAP
Broadcasting BEACON w/ GTS list [L:1, LN6/1]
BEACON
MN4 knows that
LN2 is not on the GTS list now.
GTS Allocation REQ [LN2/1, SEQ:0x14]
ACK [SEQ:0x14]
CAP
2nd SF
Broadcasting BEACON w/ GTS list [L:2, LN6/1, LN2/1]
BEACON
DATA [LN2, SEQ:0x15, D: 0x10]
3rd SF
ACK [SEQ:0x15]
CFP
PAN coordinator
indicates
that an updated data,
0x10, is sent from LN2

Fig. 12. The sequence of False Data Injection.

6.2 Non-existing Identities in the PAN

For forging non-existing IDs, we also have one PAN coordinator, two legitimate nodes: LN2 and LN6, and one malicious node: MN4 that pretends to be a different ID from ones of LN2 and LN6. In this case, MN4 eavesdrops on the beacons to learn what IDs do not belong in the PAN.

DoS against GTS Requests

As shown in Figure 13, this attack needs several superframes to allow MN4 to fill all 7 GTSs. In each SF, MN4 knows how many GTSs are available and sends GTS allocation requests in order to reserve the remaining slots of GTSs. Once MN4 takes all 7 GTSs, it stops sending GTS allocation requests to reduce its energy consumption and monitors the beacons to start sending GTS allocation requests again if the PAN coordinator drops the unused GTSs by a preventative action for the CAP maintenance.

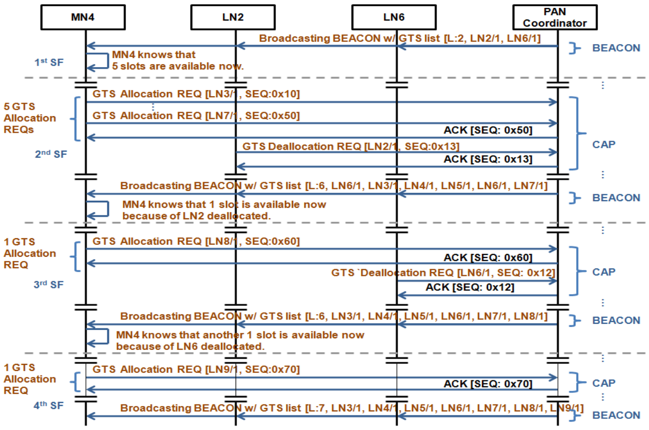

MN4
LN2
LN6
PAN Coordinator
Broadcasting BEACON w/ GTS list [L:2, LN2/1, LN6/1]
BEACON
1st SF
MN4 knows that
5 slots are available now.
GTS Allocation REQ [LN3/1, SEQ:0x10]
5 GTS Allocation REQs
GTS Allocation REQ [LN7/1, SEQ:0x50]
ACK [SEQ: 0x50]
GTS Deallocation REQ [LN2/1, SEQ:0x13]
2nd SF
ACK [SEQ: 0x13]
CAP
Broadcasting BEACON w/ GTS list [L:6, LN6/1, LN3/1, LN4/1, LN5/1, LN6/1, LN7/1]
BEACON
MN4 knows that 1 slot is available now
because of LN2 deallocated.
1 GTS Allocation REQ
GTS Allocation REQ [LN8/1, SEQ:0x60]
ACK [SEQ: 0x60]
GTS `Deallocation REQ [LN6/1, SEQ: 0x12]
3rd SF
ACK [SEQ: 0x12]
CAP
Broadcasting BEACON w/ GTS list [L:6, LN3/1, LN4/1, LN5/1, LN6/1, LN7/1, LN8/1]
BEACON
MN4 knows that another 1 slot is available now
because of LN6 deallocated.
1 GTS Allocation REQ
GTS Allocation REQ [LN9/1, SEQ:0x70]
ACK [SEQ: 0x70]
CAP
4th SF
Broadcasting BEACON w/ GTS list [L:7, LN3/1, LN4/1, LN5/1, LN6/1, LN7/1, LN8/1, LN9/1]
BEACON

Fig. 13. The sequence of DoS against GTS request.

Stealing Network Bandwidth

Figure 14 shows that a malicious node takes the last slot out of GTSs, 6 slots of which were already assigned to the malicious node. Then, it can utilize all 7 GTSs during the CFP to transmit data. The difference from the previous DoS against GTS Requests is that since this attack continues to transmit data at each time slot of the CFP, the PAN coordinator will not take a preventative action for the CAP maintenance.

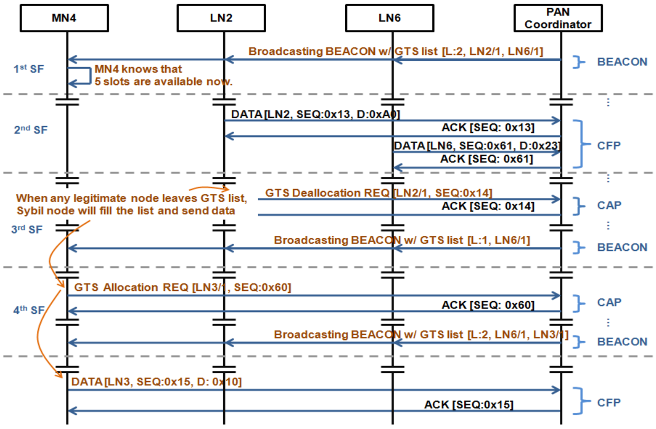

MN4
LN2
LN6
PAN Coordinator
Broadcasting BEACON w/ GTS list [L:2, LN2/1, LN6/1]
BEACON
1st SF
MN4 knows that
5 slots are available now.
DATA [LN2, SEQ:0x13, D:0xA0]
2nd SF
ACK [SEQ: 0x13]
DATA [LN6, SEQ:0x61, D:0x23]
ACK [SEQ: 0x61]
CFP
When any legitimate node leaves GTS list,
Sybil node will fill the list and send data
GTS Deallocation REQ [LN2/1, SEQ:0x14]
ACK [SEQ: 0x14]
CAP
3rd SF
Broadcasting BEACON w/ GTS list [L:1, LN6/1]
BEACON
GTS Allocation REQ [LN3/1, SEQ:0x60]
ACK [SEQ: 0x60]
CAP
4th SF
Broadcasting BEACON w/ GTS list [L:2, LN6/1, LN3/1]
BEACON
DATA [LN3, SEQ:0x15, D: 0x10]
ACK [SEQ:0x15]
CFP

Fig. 14. The sequence of Stealing Network Bandwidth.

7 Attack Analysis

We have verified our implementation with the packet sniffer [34] to monitor the packet transmission while each attack is executing. We utilize the PAN coordinator to log humidity and temperature sent by a legitimate node during both the CAP and the CFP. In addition, the throughputs in Figures 15, 16, and 17 are based on the total number of data in bytes divided by the elapsed time. The total data is counted only during the CFP. For each test of the four attacks, we measured the packet transmission for 100 to 400 seconds depending on the complexity of each attack.

DoS against Data Transmission During CFP

Figure 15 shows the decline of data throughputs on LN2 and LN6 while MN4 is sending GTS deallocation requests with LN2 and LN6's IDs. Around the 50-second mark of the experiment, a malicious node sends two GTS deallocation requests back to back. It also sends the same two GTS deallocation requests whenever it receives a beacon-notification. Therefore, the data throughputs from LN2 and LN6 during the CFP are dropped to 0bps. During the moment after 50-second mark, even though LN2 and LN6 try to send GTS allocation requests, the requests cannot be accomplished because of continuously sending GTS deallocation requests from MN4.

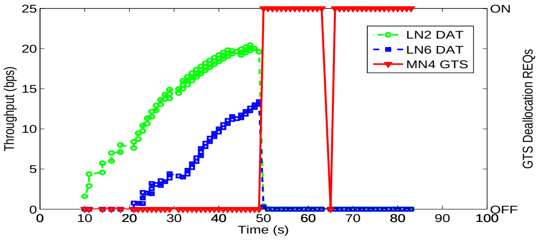


Fig. 15. Legitimate nodes (LN2 and LN6) data throughput during CFP by a malicious node (MN4). LN2 DAT and LN6 DAT: Data from LN2 and LN6 and MN4 GTS: GTS deallocation requests from MN4.

False Data Injection

Figure 16 shows the change of humidity and temperature from LN2. We tested this attack inside of a building, the humidity and temperature conditions were approximately 41% and 72°F respectively. However, since MN4 sends false data readings of 90% of humidity and 28°F temperature during the CFP, this results in many fluctuations of data for 20 seconds around the 73 to 93-second mark. Since 28°F is below the freezing point, the false data of temperature might lead to a warning sign in a practical situation.

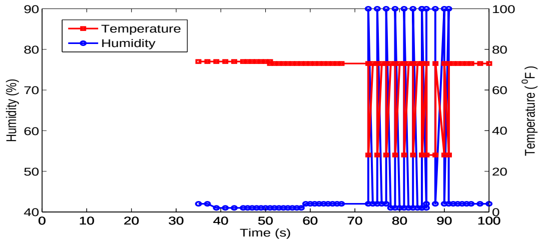


Fig. 16. Fluctuation of humidity and temperature.

DoS against GTS Requests

Figure 17 shows two instances of this attack. LN2 and MN4 are started at the same time (around the 20-second mark). By sending a GTS request, LN2 quickly occupies one GTS and transmits data during the CFP. Similarly, MN4 quickly occupies the remaining 6 of the 7 GTSs. While LN2 is transmitting data, MN4 continuously sends GTS allocation requests in an attempt to occupy the last GTS. Once LN2 releases its GTS at the 50-second mark, the coordinator allows MN4 to occupy the last GTS. MN4 now stops sending GTS allocation requests to conserve energy. LN2 sends a GTS allocation request around the 60-second mark and the 90-second mark, but the coordinator does not assign LN2 a GTS (because MN4 has them all). To see another iteration of this, we turn off the PAN coordinator around the 130-second mark to force it to perform the preventative CAP maintenance action manually (this is because the IEEE 802.15.4 source code from the open-ZB does not handle this situation as it should). Accordingly, the PAN coordinator does not have any requested GTSs. Around the 140-second mark, we turn on the PAN coordinator and LN2 successfully is allocated one GTS and it transmits data during the corresponding CFP for about 70 seconds. MN4 now begins sending GTS allocation requests between the 150-second mark and 200-second mark and is able to occupy 6 GTSs. Also, when LN2 releases its GTS around the 200-second mark, MN4 immediately occupies all 7 GTSs again.

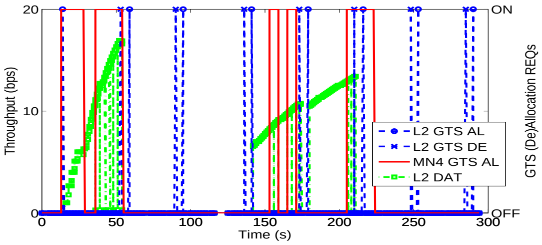


Fig. 17. A malicious node (MN4) filling up all 7 GTSs. LN2 DAT: LN2 Data, LN2 GTS AL: LN2 GTS allocation request, LN2 GTS DE: LN2 GTS deallocation request, and MN4 GTS AL: GTS allocation requests from MN4.

Stealing Network Bandwidth

Figure 18 shows the data throughputs of LN2 and MN4 and the GTS allocation requests of MN4. While LN2 has one GTS and transmits data during the CFP, MN4 starts sending GTS allocation requests with 7 forged IDs around 20-second mark and transmits data at the assigned GTSs. One of 7 GTSs allocation requests of MN4 is discarded at the first attempt because one GTS is already assigned to LN2. However, as soon as LN2 releases its GTS around the 50-second mark, MN4 occupies the last GTS immediately and has all 7 GTSs. MN4 probably consumes its energy by itself. However, LN2 and the PAN coordinator can use a lot of energy because LN2 attempts a GTS allocation request to get one GTS, and the PAN coordinator needs to receive data from the nodes.

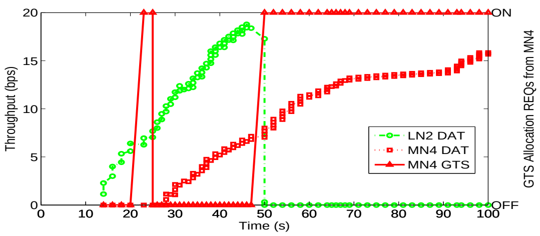


Fig. 18. A malicious node (MN4) stealing GTSs during CFP. LN2 DAT and MN4 DAT: Data from LN2 and MN4 respectively and MN4 GTS: GTS allocation requests from MN4.

8 Possible Countermeasures

We can consider several countermeasures against an inside attacker launching attacks in a beacon-enabled 802.15.4 network. [1] Even though light-weight authentication for each node might be a viable solution, authentication with a reliable key distribution and management is a expensive method for resource limited sensor nodes. In addition, the 802.15.4 standard states that key management and entity (e.g., sensor node) authentication can be implemented on top of the MAC layer [20]. Therefore, we present less expensive methods that can defend against our implemented attacks.

Reliable GTS Management Scheme: According to the 802.15.4 standard [20], its security features already have an access control list (ACL) mode. However, the functionality of the ACL mode does not cover the GTS management scheme. The access control should be extended to restrict the available numbers

[1] Due to space constraints, the countermeasures will be addressed in detail in our future work.

of GTSs to each node and keep track of thine reserved GTSs. The access control mechanism should exam the frequency of sending GTS (de)allocation requests from each node. If the frequency of GTS requests is too high from one node, it may become suspicious that a malicious node is trying to hold the CAP by sending a number of GTS requests (one of the commands in the 802.15.4) because all the commands can be sent during the CAP if there is no contention. In addition, the access control mechanism should keep track of the interval between GTS requests from the same node. If the interval of the same GTS request is too short, this could be an indication that a malicious node is interfering with a legitimate node sending GTS requests.

Multiple Channels: Another possible prevention against an inside attacker either impersonating legitimate nodes or forging new nodes' IDs is that the PAN coordinator might use different pre-defined channels for each legitimate node that may be changed after a short period of time (i.e., frequency hopping). Then, a malicious node would need to take a while to scan the communication channel with each change. Even though the malicious node discovers one proper channel, it can pretend to be a legitimate node for a very short time since the legitimate node can change the communication channel with the PAN coordinator. Moreover, the malicious node will have to spend a large amount of time to scan other channels for other legitimate nodes.

9 Conclusion and Future work

In this paper, we first described some existing vulnerabilities of the GTS management scheme in the IEEE 802.15.4 standard. We also investigated security protocols proposed in the recent years and security features adopted in the standard. However, to date, no method considers insider attacks against beacon-enabled 802.15.4 networks. Therefore, we have targeted the GTS management scheme in a beacon-enabled IEEE 802.15.4 network and implemented four possible attacks on integrity and availability: (1) DoS against sending data during CFP, (2) False data injection, (3) DoS against GTS requests, and (4) Stealing network bandwidth. We also analyzed the results for each attack. For our future work, we will consider ways for malicious nodes to save energy while attacking, develop other types of attacks in the MAC layer, and implement the defense mechanisms discussed in Section 8.

Acknowledgements

This work was partly supported by NSF Grant No. CAREER-CNS-844144.

References

1. Alert Me homepage. http://www.alertme.com/products/home-monitoring
2. Mishra, Amitabh and Na, Chewoo and Rosenburgh, Dwayne: On Scheduling Guaranteed Time Slots for Time Sensitive Transactions in IEEE 802.15.4 Networks. In: Military Communications Conference, 2007. MILCOM 2007. IEEE, pp. 1-7, (2007)
3. Koubaa, A. and Alves, M. and Tovar, E.: i-GAME: an implicit GTS allocation mechanism in IEEE 802.15.4 for time-sensitive wireless sensor networks. In: Real-Time Systems, 2006. 18th Euromicro Conference on, 10 pp.-192, (2006)
4. Koubaa, A. and Alves, M. and Tovar, E.: GTS allocation analysis in IEEE 802.15.4 for real-time wireless sensor networks. In: Parallel and Distributed Processing Symposium, 2006. IPDPS 2006. 20th International, 8 pp. (2006)
5. Feng Chen and Talanis, T. and German, R. and Dressler, F.: Real-time enabled IEEE 802.15.4 sensor networks in industrial automation. In: Industrial Embedded Systems, 2009. SIES '09. IEEE International Symposium on, pp. 136-139, (2009)
6. Pangun Park and Fischione, C. and Johansson, K.H.: Performance Analysis of GTS Allocation in Beacon Enabled IEEE 802.15.4. In: Sensor, Mesh and Ad Hoc Communications and Networks, 2009. SECON '09. 6th Annual IEEE Communications Society Conference on, pp. 1-9, (2009)
7. Mehta, A. and Bhatti, G. and Sahinoglu, Z. and Viswanathan, R. and Zhang, J.: Performance analysis of beacon-enabled IEEE 802.15.4 MAC for emergency response applications. In: Advanced Networks and Telecommunication Systems (ANTS), 2009 IEEE 3rd International Symposium on, pp. 1-3, (2009)
8. Douceur, J. R.: The Sybil Attack. In: IPTPS02: 1st International Workshop on Peer-to-Peer Systems, (New York, NY, USA), IPTPS, (2002)
9. Yang, J. Chen, Y. and Trappe, W.: Detecting sybil attacks in wireless and sensor networks using cluster analysis. In: Mobile Ad Hoc and Sensor Systems, 2008. MASS 2008. 5th IEEE International Conference on, pp. 834-839, 29 2008-Oct. 2 (2008)
10. Demirbas, M. and Song, Y.: An rssi-based scheme for sybil attack detection in wireless sensor networks. In: World of Wireless, Mobile and Multimedia Networks, 2006. WoWMoM 2006. International Symposium on a, pp. 5 pp. 570, (2006)
11. Amini, F. Misic, J. and Pourreza, H.: Detection of sybil attack in beacon enabled IEEE 802.15.4 networks. In: Wireless Communications and Mobile Computing Conference, 2008. IWCMC08. International, pp. 1058-1063, Aug. (2008)
12. Zhang, Q. Wang, P. Reeves, D. and Ning, P.: Defending against Sybil attacks in sensor networks. In: Distributed Computing Systems Workshops, 2005. 25th IEEE International Conference on, pp. 185-191, June (2005)
13. Du, W. Deng, J. Han, Y. S. and Varshney, P. K.: A pairwise key predistribution scheme for wireless sensor networks. In: CCS03: Proceedings of the 10th ACM conference on Computer and communications security, (New York, NY, USA), pp. 42-51, ACM, (2003)
14. Liu, D. and Ning, P.: Establishing pairwise keys in distributed sensor networks. In: CCS03: Proceedings of the 10th ACM conference on Computer and communications security, (New York, NY, USA), pp. 52-61, ACM, (2003)
15. Du, W. Deng, J. Han, Y. Chen, S. and Varshney, P.: A key management scheme for wireless sensor networks using deployment knowledge. In: INFOCOM 2004. Twenty-third AnnualJoint Conference of the IEEE Computer and Communications Societies, vol. 1, pp. 597, March (2004)
16. Eschenauer, L. and Gligor, V. D.: A key-management scheme for distributed sensor networks. In: CCS02: Proceedings of the 9th ACM conference on Computer and communications security, (New York, NY, USA), pp. 41-47, ACM, (2002)

17. Perrig, A. Szewczyk, R. Wen, V. Culler, D. and Tygar, J. D.: SPINS: security protocols for sensor networks. In: MobiCom' 01: Proceedings of the 7th annual international conference on Mobile computing and networking, (New York, NY, USA), pp. 189-199, ACM, (2001)
18. Karlof, C. Sastry, N. and Wagner, D.: Tinysec: a link layer security architecture for wireless sensor networks. In: SenSys04: Proceedings of the 2nd international conference on Embedded networked sensor systems, (New York, NY, USA), pp. 162-175, ACM, (2004)
19. Luk, M. Mezzour, G. Perrig, A. and Gligor, V.: MiniSec: a secure sensor network communication architecture. In: IPSN07: Proceedings of the 6th international conference on Information processing in sensor networks, (New York, NY, USA), pp. 479-488, ACM, (2007)
20. Wireless medium access control and physical layer specications for low-rate wireless personal area networks. In: IEEE Standard, 802.15.4-2003. ISBN 0-7381-3677-5. May (2003)
21. Sastry, N. and Wagner, D.: Security considerations for ieee 802.15.4 networks. In: WiSe04: Proceedings of the 3rd ACM workshop on Wireless security, (New York, NY, USA), pp. 32-42, ACM, (2004)
22. Alim, M. A. and Sarikaya, B.: EAP-Sens: a security architecture for wireless sensor networks. In: WICON08: Proceedings of the 4th Annual International Conference on Wireless Internet, (ICST, Brussels, Belgium, Belgium), pp. 1-9, ICST, (2008)
23. Aboba, L. B. B. Vollbrecht,J. C. J. and Levkowetz, H.: Extensible Authentication Protocol EAP. June (2004) http://tools.ietf.org/html/rfc3748
24. Clancy, T. and Tschofenig, H.: Extensible Authentication Protocol Generalized Pre-Shared Key EAP-GPSK method. February (2009) http://tools.ietf.org/html/rfc5433
25. Radosveta Sokullu O. D. and Korkmaz, I.: On the IEEE 802.15.4 MAC layer attacks: GTS attack. In: Sensor Technologies and Applications, 2008. SENSORCOMM08. Second International Conference on, pp. 673-678, Aug. (2008)
26. Roosta, T. Shieh, S. and Sastry, S.: Taxonomy of security attacks in sensor networks and countermeasures. In: The First IEEE International Conference on System Integration and Reliability Improvements. Hanoi, pp. 13-15, (2006)
27. Wood, A. and Stankovic, J.: Denial of service in sensor networks. Computer, vol. 35, pp. 54-62, Oct (2002)
28. Moteiv Corporation, tmote-sky-datasheet, (2006) http://www.moteiv.com
29. Chipcon product from Texas Instruments, CC2420. http://focus.ti.com/lit/ds/symlink/cc2420.pdf
30. Karlof, C. and Wagner, D.: Secure routing in wireless sensor networks: Attacks and Countermeasures. In: Sensor Network Protocols and Applications, 2003. Proceedings of the First IEEE. 2003 IEEE International Workshop on, pp. 113-127, May (2003)
31. Open-zb homepage. http://www.open-zb.net/
32. TinyOS homepage. http://www.tinyos.net/
33. Chipcon Products from Texas Instruments, User Manual Rev. 1.0 CC2420DK Development Kit. http://focus.ti.com/lit/ug/swru045/swru045.pdf.
34. Texas Instruments Incorporated, SmartRFPacket Sniffer User Manual Rev. 1.9. http://focus.ti.com/docs/toolsw/folders/print/packetsniffer.html
35. Anthony, J. A. S. Wood, D.: A Taxonomy for Denial-of-Service Attacks. In: Wireless Sensor Networks. CRC Press, (2004)