Fine-grained analysis of packet losses in wireless sensor networks

Bilal Shebaro

Daniele Midi
Purdue University, dmidi@purdue.edu

Elisa Bertino
Purdue University, bertino@cs.purdue.edu

Follow this and additional works at: http://docs.lib.purdue.edu/ccpubs

Part of the Engineering Commons, Life Sciences Commons, Medicine and Health Sciences Commons, and the Physical Sciences and Mathematics Commons

Shebaro, Bilal; Midi, Daniele; and Bertino, Elisa, "Fine-grained analysis of packet losses in wireless sensor networks" (2014). Cyber Center Publications. Paper 646.
http://dx.doi.org/10.1109/SAHCN.2014.6990368

This document has been made available through Purdue e-Pubs, a service of the Purdue University Libraries. Please contact epubs@purdue.edu for additional information.
Fine-Grained Analysis of Packet Losses in Wireless Sensor Networks

Bilal Shebaro, Daniele Midi, Elisa Bertino
Computer Science, Cyber Center, and CERIAS,
Purdue University, West Lafayette, IN 47907, USA
Email: {bshebaro, dmidi, bertino}@purdue.edu

Abstract—Packet losses in a wireless sensor network represent an indicator of possible attacks to the network. Detecting and reacting to such losses is thus an important component of any comprehensive security solution. However, in order to quickly and automatically react to such a loss, it is important to determine the actual cause of the loss. In a wireless sensor network, packet losses can result from attacks affecting the nodes or the wireless links connecting the nodes. Failure to identify the actual attack can undermine the efficacy of the attack responses. We thus need approaches to correctly identify the cause of packet losses. In this paper, we address this problem by proposing and building a fine-grained analysis (FGA) tool that investigates the causes of packet losses and reports the most likely cause of these losses. Our tool uses parameters, e.g. RSSI and LQI, present within every received packet to profile the links between nodes and their corresponding neighborhood. Through real-world experiments, we have validated our approach and shown that our tool is able to differentiate between the various attacks that may affect the nodes and the links.

I. INTRODUCTION

In computer networks, situational awareness (SA) has become a main requirement for highly secure network systems [1]. For example, the Department of Homeland Security networks require cyber situation awareness for security analysts to evaluate possible cyber threats and to defend against them [2]. SA typically refers to being aware of what is happening in the network of interest, in order to understand how information, events, and defense actions will impact the security of the network, both immediately and in the near future. When dealing with wireless sensor networks (WSNs), a class of events that is relevant for SA is represented by packet losses. Such events may lead to the loss of relevant information and may undermine data quality solutions based on redundant transmission of data [3]. As WSNs have been deployed for disaster recovery, tactical missions, and patient monitoring [4]–[10], the sensitivity of these applications leave no room for any data errors. However, in order to achieve full SA, it is not sufficient to detect that packets have been lost. It is also crucial to obtain correct diagnoses about the causes of the losses, as packet losses could be due to misbehaving or compromised nodes or to attacks on the links. For sensor systems to survive, this knowledge is crucial for responding to the attacks and for recovery and debugging purposes.

Current intrusion detection systems (IDSes) are typically only able of detecting packet losses and are thus unable to determine the cause of the losses, whether it is node or link related [11]–[13]. Attacks like selective forwarding and blackhole attacks are examples of node related attacks that result in partial or total packet losses, while interference is an example of a link related attack. Current intrusion detection techniques thus need to be extended with approaches able to perform a correct diagnosis of the cause of packet losses in the WSN of interest.

In this paper we address such need by designing and implementing an approach that performs fine-grained analysis (FGA) of packet losses in order to diagnose the causes of the losses. Our approach is based on the analysis of all the links in the WSN of interest in order to determine whether the cause of a packet loss is due to a malicious node or a link problem. Our approach profiles the network links using the received signal strength indicator (RSSI), link quality indicator (LQI), and the packet reception rate (PRR). Whenever a packet loss is reported, we use these profiles to perform a thorough analysis to determine the cause of the packet loss. In addition, in cases in which a packet loss is due to an interference among wireless links, our approach is able to estimate the location of the interference source to determine the network regions affected by this interference. Such information is crucial, for example, for the network administrators in order to determine the affected nodes and take the correct response actions, such as requiring the unaffected sensors to re-route their data through links and nodes not affected by the interference.

The design of our FGA tool has many advantages that make it suitable for asynchronous systems such as WSNs. First, the analysis is event-driven and is carried out simultaneously at every investigating node. Second, multiple simultaneous investigations are possible if more than one node is malfunctioning at different parts of the network. Third, the investigation is carried out in a stealthy manner, giving no chances for the malicious node to interfere with the investigation results.

In contrast to previous approaches [14], [15], our approach does not require additional nodes or resources to operate, but rather it takes advantage of existing link properties. Because sensor nodes have limited resources and power budget, our approach uses resident parameters (RSSI/LQI) associated within every sent/received packet to perform the required analysis. Our approach is fully distributed and does not rely on the base station (BS) to perform the analysis; instead, it depends on the direct neighbors of the faulty node or link.
Our FGA tool can be used in many different applications such as forensics and real-time response systems. Forensic analysts may choose to record and save the link parameters of every packet to investigate and determine the nodes involved in suspicious data transfers. Moreover, our FGA tool can be used in real-time incident response systems (IRS) for better and more accurate responses.

The rest of this paper is organized as follows. Section II introduces some background information on the RSSI and LQI parameters. Section III elaborates on how we build the link and neighborhood profiles. We list the diagnosis steps used in analyzing packet losses in Section IV. In Section V we show, in the case of interference, how we locate the affected network region. In Section VI we describe the experimental setup and implementation details of our FGA tool, followed by the experimental results. Finally we discuss related work in Section VIII followed by some conclusions and future work in Section IX.

II. BACKGROUND

In this section, we introduce the basic parameters that are used in profiling the links among nodes in a WSN. As hardware platform, we use the CC2420 radio chips that are installed on the Telos nodes (see Section VI for more details about the CC2420 radio chip). The CC2420 chip provides two useful measurements: the RSSI and the LQI. The RSSI represents the signal power of the received packet and is measured in dBm. Its value is calculated over 8 symbol periods and stored in the RSSI_VAL register of the CC2420 radio chip. Chipcon, the manufacturer of these radios, has specified in the CC2420 datasheet that the signal power value is computed in dBm as $RSSI_{VAL} + RSSI_{OFFSET}$, where $RSSI_{OFFSET}$ is about $-45$. The RSSI value ranges between $-50$ and $-100$, with the higher value (less negative) representing a stronger signal.

The LQI is a measure of the current quality of the received signal and can be viewed as the chip error rate of the received symbol. According to the CC2420 specification, the measured LQI is actually the average correlation of each symbol obtained by comparing the symbol that is supposed to be received and the symbol actually received (signal + noise). The LQI value ranges between 50 and 100, with the higher values representing better quality frames.

Finally, a third parameter used in our approach is the PRR, which is defined as the ratio of the number of successfully received packets over the number of packets sent between two neighbor nodes. A high PRR means a better link quality and a healthy communication link.

III. NETWORK PROFILING MANAGEMENT

In this section, we show how we use the RSSI, LQI, and PRR parameters to profile each link between two neighbor nodes, and how we profile each node’s neighborhood.

---

1LQI values are generally calculated by software to convert their values to a range of 0 – 255. The values are computed using the RSSI and the average correlation values. In this paper, we refer to the LQI as the average correlation, that is, the value that CC2420 chip denotes as LQI.
B. Neighborhood Profiling

We also define the profile of the neighborhood of each node by averaging all the link profiles of the direct neighbors of the node. The neighborhood profile is critical in cases of strong interference for localizing the source of the interference and giving a better understanding on the areas of the network that are possibly affected, as discussed in Section V.

C. Adding, Removing, or Relocating Nodes

Our FGA tool is designed to perform the least possible changes in case of network modifications. As sensor nodes have limited processing power and storage, we have to ensure that our design can tolerate network changes, such as adding new nodes to or removing nodes from the network, as well as relocating existing nodes.

When a new node is added, the occurring changes only partially affect the direct neighbors of the introduced node. While the BS updates the network map, it requests the new node to build its link profiles with its neighbor nodes. The new node will also discover its direct neighbors and request them to profile their new common links. At this point, each neighbor node will add one more link profile to its existing list of profiles, and locally update its neighborhood profile as well.

In case a node has been removed, there will be no need for changes, as the node’s neighbors will have an extra link profile that has no negative impact on the analysis that might have to be carried in the future, unless storage becomes an issue. In situations when re-profiling is requested, all the profiles of removed nodes will be eliminated.

Relocating nodes is very similar to the situation of adding new nodes. Since our profiling parameters (RSSI, LQI, PRR) are affected by location, re-positioning a node requires re-building its link and neighborhood profiles. As the BS updates its network map, the relocated node may be introduced to new neighbors where neighborhood profiling will be requested.

IV. Diagnosis

In this section, we show our analysis steps to differentiate between different types of attacks that may cause packet losses. We first assume the network topology to be designed such that every node has at least two possible paths to reach the BS. We also assume that an IDS is deployed at each node so that each sensor is capable of detecting packet losses as well as data modifications [16]–[19]. In our work, the IDS will trigger the FGA only when the IDS detects that there is a packet drop attack that needs to be investigated upon.

Purpose of the analysis: The purpose of the FGA is to differentiate between the attacks that target the nodes from those that target the links between them. Through the detected attack, we can conclude whether the cause of packet drops is node or link related. For instance, selective forwarding attacks and blackhole attacks are node related attacks that cause partial and total packet losses, respectively. However, radio interference, or jamming, attacks may also be responsible for packet losses as they have a negative effect on the network links and regions depending on the interference source location and strength \(^2\). Basically, the existence of interference can affect the RSSI and LQI values of received packets that passed through a noisy environment, and can sometimes impair the signal quality of other packets to the point that they become unreadable.

Profile Comparison: A basic building block in our analysis technique is represented by the profile comparison algorithm shown in Figure 2. The comparison, that has as input a node under investigation, first sets \(PRR_{curr}\) as the current PRR of this node to its investigating neighbor, and PRR threshold \((PRR_{thres})\) as the value that differentiates between the cases of partial and total packet losses. \(\Delta_{RSSI}\) and \(\Delta_{LQI}\) denote the corresponding difference between the initial and current RSSI and LQI values respectively, and \(Interf_{thres}\) as the interference threshold representing the minimal difference in link profiles (RSSI, LQI) that can determine the existence of interference. The value of the \(Interf_{thres}\) is determined by measuring the maximum fluctuation occurred during the initial profiling of the network links.

When \(PRR_{curr} \geq PRR_{thres}\), the profile comparison algorithm of each neighbor node would only need to compare the link profiles that connect it with the node under investigation. However, when \(PRR_{curr} < PRR_{thres}\), the profile comparison algorithm of each neighbor node would need to re-profile all its links and compare the new profiles with the originally captured link profiles.

Analysis Steps of the FGA: Our analysis is driven to understand the components of each link profile together with the neighborhood of each node to better evaluate the cause of packet losses. The basic idea is for the nodes, whose IDS reported packet drops, to re-profile their links with the suspicious node and compare them to the existing profiles generated during initial setup. Link re-profiling algorithm is carried simultaneously by several investigating nodes and without the knowledge of the suspicious node to avoid any misdirection with the investigation results.

\(^2\)In this paper, we refer to the term interference for both intentional (jamming) and unintentional disrupt of signal communication between sensor nodes.
We design the FGA algorithm to be event-driven, suitable for asynchronous systems such as WSNs. The triggering events and their respective actions are shown in Figure 3. Basically, when the IDS of a node, say Node n, observes packet drops at one of its neighbor nodes, say n\textsubscript{bad}, the FGA of Node n is triggered to investigate the cause of the packet drops observed at n\textsubscript{bad}. The FGA of Node n starts snooping packets from n\textsubscript{bad} and records their corresponding RSSI and LQI values in a re-profiling array R\textsubscript{bad}[N] of size N. For every subsequent notification by the IDS of Node n that another packet has been dropped by n\textsubscript{bad}, a value of 0 is pushed into R\textsubscript{bad}[]. This process is carried out in parallel at every node that observes packet drops from n\textsubscript{bad} and is investigating accordingly. At the time, say TC, when one of the investigating nodes fills up its re-profiling array R\textsubscript{bad}[], it computes its PRR and applies the profile comparison algorithm (see Figure 2) to make a vote on what is the most likely cause of packet drops at n\textsubscript{bad}. The resulting vote is broadcasted in the format of: \langle n\textsubscript{bad}, NodeID[], Vote, TC\rangle, where NodeID[] is an array that initially contains the ID of the first voting node but eventually will contain the IDs of all voting nodes.

Every other node that is still investigating on n\textsubscript{bad}, say Node n', will stop recording packets from n\textsubscript{bad} at the time of receiving the first broadcasted vote, and will only consider those packets with timestamp less than or equal to TC. This is necessary because up until the first vote is broadcasted, the voting nodes.

If enough packets have been recorded in R\textsubscript{bad}[] at each Node n' according to a preset threshold, each Node n' will compute its own vote according to the profile comparison algorithm (see Figure 2) and will aggregate it with the originally received vote from another investigating node. The vote aggregation result is based on the following:

**High Radio Interference:** If at least one node votes for High Radio Interference, the aggregated vote is High Radio Interference and therefore packet drops at n\textsubscript{bad} are link related. A node will vote for High Radio Interference when the RSSI and LQI values of its neighborhood profile are significantly affected due to such strong interference.

**Low Radio Interference:** If at least one node votes for Low Radio Interference and none of the other nodes votes for High Radio Interference, the aggregated vote is Low Radio Interference and therefore packet drops at n\textsubscript{bad} are link related. A node will vote for Low Radio Interference when the RSSI and LQI values of the link profile with n\textsubscript{bad} are significantly changed in their RSSI and LQI values when compared with its original link profile connecting both nodes.

**Selective Forwarding Attack:** If at least one node votes for Selective Forwarding and none of the other nodes votes for any type of interference in the network medium, the aggregated vote is Selective Forwarding and therefore packet drops at n\textsubscript{bad} are node related. A node will vote for Selective Forwarding when none of its received packets from n\textsubscript{bad} has any significant changes in their RSSI and LQI values when compared with its original link profile connecting both nodes.

**Blackhole Attack:** If at least one node votes for Blackhole when none of other investigating nodes votes for selective forwarding or any type of interference in the network medium, the aggregated vote is Blackhole and therefore packet drops at n\textsubscript{bad} are node related. A node will vote for a blackhole attack when none of its neighborhood link profiles is affected by interference nor any packets have been forwarded from n\textsubscript{bad}.

Every node that computes its vote (its own or aggregated) will broadcast the vote with a preset timeout T\textsubscript{v}. Any node that receives a vote with the NodeID[] array including new node ID(s) will consider the received vote as more updated and therefore will be aggregated with its own vote (if not already included), then broadcasted. T\textsubscript{v} is reset every time a vote is broadcasted. All votes received from n\textsubscript{bad} are ignored and do not reset T\textsubscript{v}. When T\textsubscript{v} expires, the vote aggregation is considered complete, meaning that no more votes are circulating. In addition, every neighbor node will have the most updated vote result, with NodeID[] listing the IDs of all the nodes that broadcasted their vote.

The strength of our approach relies on the fact that multiple investigations can be carried out if more than one n\textsubscript{bad} exists; however, we assume that the investigating nodes are trustworthy and are not reporting false votes. On the other hand, since the RSSI and LQI values may be subject to sampling bias that could diverse the voting result of a single node, we achieve better voting accuracy in network topologies that allow each node to have at least two direct neighbors and more than one path to reach the BS. This redundancy is typical in engineering a real-world sensor network and therefore makes our voting results less diverse.

V. **Locating Interference Sources**

In critical (real-time) sensor applications, it is necessary to be able to anticipate possible future attacks on the network. This is achieved by enriching the system with learning techniques that can warn the network of possible attacks, based...
on previous faulty scenarios. In cases when radio interference is causing packet losses, it is necessary to locate the source of the interference in order to identify the network region (set of nodes/links) that may be also affected by the noise source. This is also relevant for the response systems to avoid re-routing packets through the affected region, which may affect the efficiency of the response action.

Locating the source of interference is necessary as it may assist in evaluating the trust level of sensor readings. When interference is the cause of packet losses at a certain node or link, its effect may also reach other regions where the IDS might require further diagnosis. This could leave the whole network in an inconsistent state for a longer period of time, spent between detection and analysis of the same cause. Therefore it is necessary for the FGA to locate the source of interference and inform the BS about the affected region as part of its analysis. Even though the FGA is totally distributed and do not rely on the BS in detecting the cause of packet drops, it will however require the BS’s knowledge of the network map in locating the source of interference.

![Diagram of network with interference source](image)

**Fig. 4:** $\vec{V}_i$ and $\Delta_i$ representing Node i’s location and neighborhood profile.

When the FGA tool determines that the cause of packet losses is due to interference, it attempts to locate the interference source in order to estimate the nodes and links that may be possibly affected. The approach used for detecting the interference source works as follows. Let $n_{bad}$ be the first node reported to the BS as affected by interference. The BS then requires all the direct neighbors $N_{bad}$ of this node to provide their neighborhood profiles, as discussed in Section III. Once the new profiles are computed and compared to the original neighborhood profiles, each node sends to the BS the profile difference, denoted as $\Delta_i$, for evaluation.

The technique used to locate the source of interference leverages the formula for the calculation of the weighted centroid of finite points [20] defined as:

$$\sum_{i=1}^{n} W_i \vec{V}_i$$

where $W_i$ is the weight at Node $i$ computed as the exponential of $\Delta_{i}$, and $\vec{V}_i$ is the vector with the spatial coordinates $(\vec{x}, \vec{y}, \vec{z})$ of Node $i$ according to the actual topology of the network known by the BS. Figure 4 also shows the vectors that represent each node’s location according to a given origin $O$ and their corresponding $\Delta$. The BS will use these $\Delta_i$s and $\vec{V}_i$s, together with knowledge of the topology, to locate the interference source.

VI. EXPERIMENTAL ANALYSIS

In this section, we report experimental results to assess the efficiency and accuracy of our FGA tool. We first introduce our experimental setup, then we perform real-world experiments to test our FGA tool on different attack scenarios.

A. Experimental Setup

Our setup consisted of 25 TelosB [21] wireless sensor motes using TinyOS 2.1, which were placed at different locations. These motes operate at 2.4 GHz ISM band, with an effective data rate of 256 kbps, which is a much higher rate than that of older radios. For experimental purposes, we set up one sensor to act as a BS and created a server Java program to interact with the BS through the USB port. All other nodes were programmed with the same code and waited for commands from the BS. Each node was programmed to perform its analysis locally and independently.

We also built a simplistic routing system on top of the standard messaging layer that offers point-to-point multi-hop direct communication. We applied the concepts of the Internet routing protocols so that every node is capable of automatically building its own routing table and self-discovering/learning the best routes towards any other node in the network. Thus, after a few initial packet exchanges, most of the routes are automatically discovered and so is the whole network. Our improved routing system enhances the node’s communication overhead by reducing useless packet transmissions. Through the BS, we used the multi-hop protocol to allow the administrator to send commands to all the nodes, as well as to single specific nodes through our Java program that was connected to the BS.

We also set up a special mote, the jammer, to act as the source of interference in order to test the FGA accuracy in detecting and locating the noise [22]. This mote was programmed to emit dummy packets every 5 msec. with an increasing counter value. We avoided keeping the other mote radios busy with useless interrupts, while maximizing the interference on the radio medium itself. This helped us to get a more realistic representation of a real-world interference attack.

The accuracy of the FGA is related to the considered topology of the sensor network. The minimum topology requirements would be for each node to have at least two possible paths to reach the BS. This requires that every link has at least one additional node watching it, forming a triangular structure for every hop towards the BS. This requirement is basic for most real-world sensor networks, considering that redundancy is always required by the engineering of every topology. Figure 5 shows a snapshot of the topology of a portion of our network that is the closest to the BS. We performed real-world experiments using this topology to compute the corresponding profiles of each link. Then we used these profiles to detect causes of packet drops as detailed in Section IV.

B. FGA Testing on Different Attack Scenarios

We carried out our FGA testing at different parts of the network to better evaluate its efficiency, and we experimented...
all possible cases by placing the interference source at different locations. Since only the direct neighbor nodes participate in the FGA of a suspicious node, we will refer to Figure 5 in our experiments, as attacks causing packet losses will occur among these nodes.

Our experiments were divided into 6 scenarios:

**Experiment 1: Building Profiles at Initial Network Setup:** At initial network setup, the BS requests every node to start building its initial profiles with its direct neighbors, and saves these profiles locally at every node. The time needed to profile our 25 sensor nodes, which each sends 100 dummy messages to each other node every 10 milliseconds, was a total of 25 sec., thus 1 sec. to profile each node links. Figure 6 shows sample profile values we collected from Nodes 2, 3, 4, 5, 6 and 7 (see Figure 5).

**Experiment 2: Interference Effect on Link Profiles:** In order to test the impact of interference on link profiles, we activated our interference mote and placed it close to Node 3. We manually requested re-profiling to see the changes in the RSSI and LQI values of the affected links, mainly around Node 3. Figure 6 shows the corresponding profiles of these links. Notice the changes in the RSSI values with no significant changes in the LQI values due to low radio interference we purposely used. Also, since the LQI values that CC2420 radios report are independent from the RSSI parameter, the link quality is stable once the signal strength is good enough.

**Experiment 3: Selective Forwarding Attack:** To test the FGA tool against this attack, we configured our network so that Node 5 would send messages to the BS, whereas Node 2 would be the intermediate node as chosen by the routing protocol. Nodes 3, 4, 6 and 7 would be the direct neighbors of Node 2 and thus monitor its behavior.

The intermediate Node 2 was programmed to simulate a Selective Forwarding attack by dropping packets with 10% probability, and we made sure that there was no interference within range. Our results show that when 20 packets were dropped by Node 2, the IDSes of Nodes 3, 4, 6 and 7 successfully detected a packet drop attack and triggered the FGA. In our experiment, Node 4 was the first to completely fill its re-profiling array of 10 slots at time TC, compute, and broadcast its vote. Each of the Nodes 3, 6 and 7 had already recorded the RSSI and LQI values of 9 received packets from Node 2 at the time stamped in Node 4’s vote. The FGA of each node in turn aggregates and broadcast its vote accordingly until all votes of the investigating nodes are aggregated. The final vote aggregation reported a “Selective Forwarding” attack.

**Experiment 4: Low Interference Attack:** We placed the interference mote to carry out low interference near Node 2. The interference however was not strong enough to isolate Node 2 completely. However, when 20 packets were dropped by Node 2, the IDSes of Nodes 3, 4, 5, 6 and 7 detected a packet drop attack and activated their FGA. Node 3 was the first to fill its re-profiling array slots and sent out its vote. Nodes 4 and 6 received the vote and stopped recording with 9/10 slots filled, while Node 5 and 7 stopped recording with 8/10 slots filled. Since our threshold for accepting a node’s vote is to have at least 70% of its re-profiling array slots filled, we considered the votes of all 5 investigating nodes. Even though Nodes 4 and 7 voted for “Selective Forwarding” as their recorded packets did not show any significant influence of existing interference, Nodes 3, 5, and 6 voted for “Low Interference” which is the aggregated vote result according to the methodology we presented in Section IV.

**Experiment 5: Strong Interference Attack:** This attack scenario was the same as the previous attack scenario with the difference that in this scenario we strengthened the interference to such a point that Node 2 was completely isolated from the network. Nodes 3, 4, 6 and 7 carried out the analysis with Node 4 filling out its re-profiling array first and broadcasted its vote. Nodes 6 and 7 received the vote and stopped recording with 7/10 slots filled, while Node 3 with only 5/10 slots filled. Since our threshold for accepting a node’s vote is to have at least 70% of its re-profiling array slots filled, we do not consider the vote of Node 3. Because of the applied strong interference, the re-profiling arrays of Nodes 4, 6 and 7 contained 0’s and therefore had to re-profile their links with their direct neighbors to perform the profile comparison algorithm. Even though Node 6 was the only node to report "High Radio Interference", its vote was dominant according to the methodology we presented in Section IV.

**Experiment 6: Locating the source of interference:** In this experiment, we performed the technique described in Section V to locate the source of interference that was causing packet losses. We placed the interference source at different locations, specifically near Nodes 2, 3 and 4, to compute the accuracy of our location method. Figure 7 shows the actual and estimated location of the interference source when placed near Node 3. Once the BS receives the neighborhood
profile difference, it can locate the interference source using its precise/approximate coordinates of each node. Figure 8 shows the accuracy of our method by comparing the actual location of the interference source to the computed ones resulted from various interference locations.

**Summary of Results:** Figure 9 shows the accuracy of the FGA tool with respect to the actual number of packet loss test cases we performed. The FGA was able to perform correct diagnoses in ~ 90% of the cases when selective forwarding was the cause of packet losses (Experiment 3), in ~ 95% of the cases when low interference was the cause (Experiment 4), and in ~ 100% of the time when strong interference was the cause of packet losses (Experiment 5). These results show the accuracy of our FGA technique for sensor network applications.

**VII. SECURITY ANALYSIS**

In this section, we present the security analysis of the FGA tool to analyze possible ways that attackers might use to conceal the cause of packet drops. The aim of our security analysis is to mitigate these ways and discuss how likely it is to compromise the results of the FGA.

**Transmission Power Manipulation.** When a node is suspected to be maliciously dropping packets, the FGA tool executes the re-profiling algorithm as described in Section IV. However, an attacker may try to misdirect the FGA by using a different power than the usual power to transmit the dummy messages required for re-profiling, which will make the neighbor nodes think the packet loss is caused by link interference. However, the re-profiling procedure we implement in our FGA is hidden from the attacker and thus the node under attack will not be able to differentiate between normal and re-profiling traffic. Only after the FGA results (votes) are broadcasted is when the attacker might know that the re-profiling process has already been executed.

**Correctness of Initial Link Profiles.** One possible interference source could be a result of closely placed sensors...
that could disturb the network communication signals. After a successful network initial setup, the existence of strong interference among the sensors is unlikely as it is easily detectable. However, very low interference could exist that may affect the RSSI values but without causing any packet drops. The FGA link initial profiles are collected right after a successful network initial setup, thus the RSSI values collected may be affected by the existing interference. However, we claim that those link profiles are still valid and correct as long as no packets drops have been detected by the IDS during the initial collection process. Moreover, the existence of a malicious node among the newly installed nodes is unlikely, as we believe that each node was tested at initial setup, thus also validating the correctness of the FGA initial profiles.

Reactive Jamming Attacks. In a reactive jamming attack, the jammer may modify the Start-of-Frame Delimiter (SFD bytes) of some packets before sending them, resulting in dropping packets at the destination node. As such packet modification does not increase $\delta_{\text{RSSI}}$ or $\delta_{\text{LQI}}$, the FGA algorithm might think that the destination node is malicious, but actually it is not. As packet modification is out of the scope of this paper, we rely on the IDS to check if there is any packet modification before the FGA starts carrying its re-profiling algorithm.

VIII. RELATED WORK

Our work is related to previous research on two different topics: (1) Use of forensic analysis techniques for investigating packet losses in WSNs, (2) use of RSSI/LQI for WSN performance and sensor localization.

Several approaches have been proposed for detecting packet dropping attacks, but few approaches identify the cause of packet drops through their impact on network parameters, as we investigate with our FGA tool. Ramach et al. introduced DAMON, a generic architecture that supports the monitoring of a wide range of protocol, device, and network parameters [23]. Also, Qiu et al. introduced a diagnostic system that employs trace-driven simulations to diagnose performance problems caused by packet dropping, link congestion, external noise, and MAC misbehavior [24]. However, the evidentiary data that both approaches collect is not used to determine the likely cause of packet losses, but rather it is used to diagnose performance problems. De Couto et al. designed an expected transmission count metric to estimate the packet delivery ratio on links [25], similar to the ETT metric that assigns weights to individual links based on the expected transmission time of a packet over the link [26]. Although both of these metrics examine the packet loss rate, neither of them gives information about the cause of packet losses.

Only very few forensic analysis techniques have been proposed to investigate packet losses occurring in networks. Yang et al. introduced a detection scheme based on observations made by neighboring nodes (witnesses) in order to identify misbehaving nodes that are incorrectly forwarding packets [27]. Furthermore, Ning et al. built a forensic analyzer that uses network parameters (packet size, bit rate in use, node density, interference level) and monitors logs to determine the cause of discarded packets and forwarding misbehaviors [28]. In contrast, our approach uses less network parameters and fewer computations to determine the likely causes of packet losses. Moreover, our approach differentiates between node and link related causes of packet drops, while their approach differentiates between natural induced packet losses from malicious discarding.

On the other hand, several researchers have used the RSSI and LQI parameters of the $CC2420$ for purposes different from ours. Zaruba et al. have used the RSSI readings for indoor localization, trying to locate wireless nodes in a home environment requiring a single access point [29]. However, Parameswaran et al. [30] have determined that the usage of RSSI only is sufficient for localization algorithms, but they were not always successful in getting very accurate measurements of node distances due to the presence of factors like interference. Zanca et al. [31] compared many RSSI-based localization algorithms and showed that by just using RSSI, localization may not be accurate with errors of few centimeters due to the presence of moving people or obstacles. However, Srinivasan and Levis [32] have argued that when the RSSI value and the LQI value are combined, they represent an effective indicator of localization, even with the existence of obstacles or interference. Their argument is based on measuring the correlation between RSSI, LQI, and PRR based on different RSSI power levels and observing low RSSI variations for a link over time, even without performing the hardware calibration performed by Chen and Terzis [33], due to the RSSI symmetry in links of $CC2420$ radios. Others uses of RSSI parameters were proposed by Khan et al. for troubleshooting unresponsive sensor nodes. However, their approach requires the use of external power-metering subsystem located next to the nodes to collect their power consumption traces in case of future possible failures [14].

IX. CONCLUSION AND FUTURE WORK

In this paper, we introduced an FGA tool for wireless sensor networks that uses existing link parameters to investigate the cause of packet losses, whether it is related to node attacks or to link interference attacks. The tool has been implemented and deployed on actual sensor nodes. Experiments on these sensor nodes have shown that our FGA tool is able to successfully differentiate the various attacks and determine the most likely cause of the packet losses. Also, in the case when interference is the cause of packet losses, the FGA tool is able to locate the interference source and to estimate its effect on other nodes and links.

Although our FGA tool is designed to be fully distributed at every node, it can also be centralized at the BS. The major design changes would be that all the link and neighborhood profiles as well as the comparisons performed at the analysis stage would be located at the BS. In regards to performance, the distributed approach performs faster analysis at the node level, while the centralized approach that may be slower but has no storage constraints on the sensor nodes.

In our future work, we plan to extend the functionality of the FGA tool to cover additional events and anomalies. We also plan to implement learning techniques so that our tool can learn from previous attacks and anticipate future ones.
ACKNOWLEDGEMENT

The work reported in this paper has been partially supported by the US National Science Foundation under grants CNS-1111512 and CNS-1016722.

REFERENCES


[33] Y. Chen and A. Terzis, “Calibrating rssi measurements for 802.15.4 radios.”