Short Paper: Reactive Jamming in Wireless Networks—How Realistic is the Threat?

Conference Paper · June 2011

DOI: 10.1145/1998412.1998422 · Source: DBLP

4 authors, including:

Matthias Wilhelm
Momentum Engineering Inc., Tokyo, Japan
27 PUBLICATIONS **676** CITATIONS
SEE PROFILE

Jens B. Schmitt
Technische Universität Kaiserslautern
275 PUBLICATIONS **3,361** CITATIONS
SEE PROFILE

Vincent Lenders
Armasuisse
154 PUBLICATIONS **3,208** CITATIONS
SEE PROFILE

Short Paper: Reactive Jamming in Wireless Networks—How Realistic is the Threat?

Matthias Wilhelm, Ivan Martinovic*, Jens B. Schmitt, and Vincent Lenders‡
Disco Labs, TU Kaiserslautern, Germany ‡Armasuisse, Switzerland
{wilhelm,martinovic,jschmitt}@cs.uni-kl.de vincent.lenders@armasuisse.ch

ABSTRACT

In this work, we take on the role of a wireless adversary and investigate one of its most powerful tools—radio frequency jamming. Although different jammer designs are discussed in the literature, reactive jamming, i.e., targeting only packets that are already *on the air*, is generally recognized as a stepping stone in implementing optimal jamming strategies. The reason is that, while destroying only selected packets, the adversary minimizes its risk of being detected. One might hope for reactive jamming to be too challenging or uneconomical for an attacker to conceive and implement due to its strict real-time requirements. Yet, in this work we disillusion from such hopes as we demonstrate that flexible and reliable *software-defined* reactive jamming is feasible by designing and implementing a reactive jammer against IEEE 802.15.4 networks. First, we identify the causes of loss at the physical layer of 802.15.4 and show how to achieve the best performance for reactive jamming. Then, we apply these insights to our USRP2-based reactive jamming prototype, enabling a classification of transmissions in real-time, and reliable and selective jamming. The prototype achieves a reaction time in the order of microseconds, a high precision (such as targeting individual symbols), and a 97.6 % jamming rate in realistic indoor scenarios for a single reactive jammer, and over 99.9 % for two concurrent jammers.

Categories and Subject Descriptors

C.2.0 [**Computer Communication Networks**]: General—*Security and protection (e.g., firewalls)*

General Terms

Security, Experimentation, Performance

Keywords

Reactive jamming, software-defined jammer, 802.15.4, WSN

*This work was partially funded by the Carl-Zeiss Foundation Fellowship.

WiSec'11, June14–17, 2011, Hamburg, Germany.

1. INTRODUCTION

The simplicity of deployment and administration as well as low-cost hardware result in an increased reliance on wireless communication systems. However, the blocking of wireless communication, i.e., jamming, is one of the major security threats and understanding the impact and complexity of such attacks and their countermeasures is of great interest to the networking community (see, e.g., [2, 11, 16, 17]), as this physical attack against the availability is unique to wireless networks and hard to mitigate on higher layers.

In the literature, several jammer categories have been identified [11, 17] according to their channel-awareness and statefulness. Constant and random jammers are the prevalent form of jammers as they are easy to implement, but lack channel-awareness. On the other end of the spectrum, reactive jammers base their jamming decisions on both the current and previous channel states. This is very desirable from the attacker's point of view since it has several benefits: *(i)* it allows for effective and efficient jamming [6], as only short jamming bursts are required to destroy complete packets; *(ii)* reactive jamming is challenging to detect [15], because only limited interference with other nodes is experienced, which minimizes the risk of exposure; and *(iii)* it enables the implementation of optimal jamming strategies, since channel-awareness is a major factor for such strategies. For example, Bayraktaroglu et al. [1] show that a smart jammer that takes the sender's state into account can be four orders of magnitude more efficient than a constant jammer. On the other hand, reactive jamming is challenging to accomplish due to the strict real-time requirements for detection and subsequent jamming. The form of jamming signals and the jamming precision become crucial for a successful destruction of packets. Hence, the question arises: Is reactive jamming a realistic threat in wireless networks in terms of technical feasibility and economic viability?

In this work, we deliver the bad news that, indeed, flexible reactive jamming is feasible in 802.15.4 networks by using low-cost software-defined radios (SDR), which are easy to configure and adapt to different application scenarios. Thus, research efforts in jamming detection and countermeasures should assume more sophisticated, yet economical, adversaries. To assist in the experimental evaluation of the main factors in reactive jammer designs, we provide a USRP2-based flexible experimental platform to the research community.[1] To achieve the best jamming performance, we analyze the causes of loss at the physical layer of 802.15.4

[1]Visit `http://disco.cs.uni-kl.de` or contact the main author for the necessary resources.

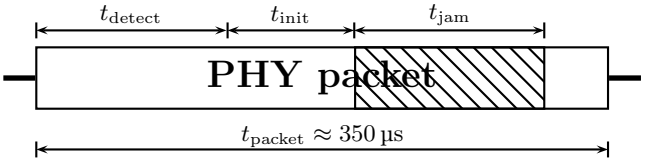


Figure 1: Time constraints: the reactive jammer must detect a transmission, initiate the jamming process and interfere with the transmission to prevent a packet reception.

and derive guidelines for successful reactive jamming against ZigBee-based networks. We assess the applicability of our approach by systematically evaluating the performance of our prototype system in several experimental settings. The results justify that reactive jamming should be considered a real threat with a low entry barrier.

2. DESIGN CHALLENGES

We set several goals for a reactive jamming platform: an accurate detection of RF transmissions as well as reliable and precise jamming, all while a packet is still on the air. Additionally, the aim is to achieve 100 % transmission cancelation even in challenging conditions often found in indoor wireless sensor networks (WSNs), e.g., assuming multipath fading and strong signals at the receiver due to short distances.

To get an impression of the timing requirements, see Fig. 1. The system must detect a transmission and decide whether it must be jammed or not (with the required time t_{detect}), schedule and initialize the sending of a jamming signal (with delay t_{init}), and send a short, yet sufficient jamming burst to destroy the packet (t_{jam}), all while it is being transmitted.[2] The concurrent jamming must exceed the shortest interference time t_{jam}^{min} to cause a packet loss. Therefore, we require

$$t_{\text{detect}} + t_{\text{init}} + t_{\text{jam}}^{min} \leq t_{\text{packet}},$$

i.e., to react quickly enough to hit the packet for the minimal required jamming duration. In the case of IEEE 802.15.4, the shortest packets are ACKs, with a duration of $t_{\text{packet}} = 352\,\mu\text{s}$. Despite these tight requirements, the system design should still be flexible and reconfigurable, so a fully programmable reactive jammer based on the software-defined radio paradigm is what we aim for.

Challenge 1: How can we ensure successful jamming? Our goal is to destroy all selected packets while keeping the jamming duration as short as possible. This poses a quite different problem compared to proactive jamming performance evaluations in the literature [1, 5]. The reactive jammer must be able to destroy transmissions at the receiver even if a sender has already started a transmission. In §3, we provide an overview for causes of loss on the physical layer and identify the jamming signal that causes the minimal packet reception ratio (PRR). In §5.1 we evaluate the minimal jam duration and show that the required jamming burst can be as short as $t_{\text{jam}}^{min} = 26\,\mu\text{s}$ to ensure a PRR of 0 %.

Challenge 2: How do we achieve real-time performance? Nychis et al. [9] show that the host-based SDR architecture (where the processing is done by a PC) introduces additional latency (e.g., 2 ms on average in case of the USRP2) into the system. We mitigate this problem by implementing our system on the USRP2's FPGA, which enables a high-speed detector design and deterministic timing (see §4 for details). Our experimental results show that we achieve a jamming initialization time of $t_{\text{init}} \approx 15\,\mu\text{s}$, while still keeping the flexibility of SDR-based systems.

Challenge 3: How do we react to 802.15.4 packets only? Our goal is a high detection accuracy, but with a minimal introduced delay t_{detect}. Along this way, different implementation choices can be made, e.g., a simple power detector is easy to implement and offers a short reaction time, but cannot classify transmissions accurately (e.g., it may not discriminate between different wireless technologies). Therefore, the detector of our prototype is designed to search for modulated 802.15.4 PHY headers, thus restricting our jamming to 802.15.4 packets only. The detector adds an additional delay of less than 4 µs. Overall, the experiments in §5 show that our prototype reacts quickly enough to detect and reliably destroy ZigBee transmissions.

3. EFFECTIVE REACTIVE JAMMING

We concentrate on physical layer attacks against 802.15.4 instead of jamming approaches against MAC mechanisms [5, 7] such as attacking the clear channel assessment (CCA). In this section, we identify the causes of packet loss on the physical layer of 802.15.4, as well as which jamming signals and timings are consequently the most effective ones against such transmissions. The results are verified through systematic experiments in a WSN testbed with MICAz motes. We identify the factors that influence the jamming performance, and select the optimal jamming signal, which we subsequently implement as part of the reactive jamming system.

3.1 802.15.4 Background

Before going into details, we briefly cover aspects of the 802.15.4 physical layer that are necessary for the later discussion of jamming against such networks. Although IEEE Std. 802.15.4-2006 [4] defines four different physical layers for the wireless interconnection of devices in wireless personal area networks (WPANs), we limit ourselves here to the 2.4 GHz PHY because of its widespread use. The standard defines 16 channels labeled Channel 11–26, with a bandwidth of 2 MHz each and a 5 MHz interspacing. Bytes in the PHY protocol data unit (PPDU) are transmitted at a rate of 250 kbps. They are divided into groups of 4 bit, which are then mapped to a set of 16 symbols. These symbols are spread with the corresponding 32 bit pseudo-noise (PN) chipping sequence, i.e., 802.15.4 uses direct sequence spread spectrum (DSSS) with a spreading factor of eight. This stream of chips is then modulated onto the carrier using O-QPSK with half-sine pulse-shaping, and transmitted over the wireless medium to the receiver.

Reception process. The reception process can be explained in terms of the PPDU headers (SHR and PHR), shown in Fig. 2. The essential components are shown in more detail, and ellipses show the required reception steps for these components. When a carrier is detected, the receiver synchronizes with the predefined preamble sequence (eight "0" symbols in the standard) to compensate the phase and frequency offset of the incoming transmission. This is necessary as the sender and receiver are not synchronized; with this step, the receiver recovers the timing of both chips and symbols, and the symbol clock adjusts to the symbol

[2] We do not consider the propagation delay in our analysis, we assume short distances between all devices.

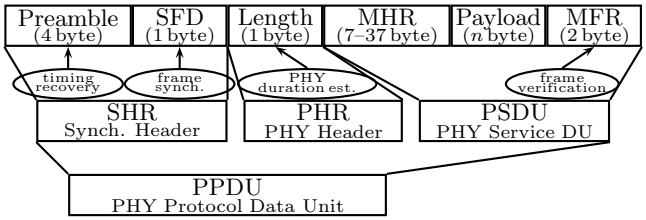


Figure 2: The structure of IEEE 802.15.4 packets.

boundaries. The receiver then expects a specific two symbol sequence, the Start-of-Frame Delimiter (SFD) that marks the beginning of the PHY header (PHR) and the following MAC layer frame. This process is called frame synchronization. The PHR consists of 7 bit containing the length of the following PHY service data unit (PSDU), which allows for duration estimation of the transmission. At each symbol clock tick, a decision is made as to which of the 16 possible symbols was the one most likely transmitted during the last period. At the end of the PSDU, the MAC footer (MFR) contains a 16 bit integrity checksum (frame check sequence, FCS) using CRC16 that verifies whether the frame is received without errors. If this is the case, the received frame is passed to the higher layers. We refer the reader to [10, 14] for a more detailed treatment of the 802.15.4 PHY and the properties of different transceiver designs.

3.2 Causes of Loss on the 802.15.4 PHY

To understand the underlying reasons for the effectiveness of different jamming approaches, we need to identify the causes of packet loss on the 802.15.4 physical layer. For a study focused on the IEEE 802.11 physical layer, refer to the work of Gummadi et al. [3].

Symbol misdetection and integrity errors. Once a frame is detected, the most likely symbol that has recently been transmitted is chosen on each symbol clock tick. Strong jamming transmissions concurrently with a symbol cause a symbol misdetection if a sufficient number of chips are flipped, consequently generating bit errors on higher layers. Integrity checks such as the CRC16 check of 802.15.4 detect these errors, resulting in a packet drop as no forward error correction (FEC) is used in 802.15.4. Thus, a single symbol error is sufficient to destroy a complete packet. Similarly, the MHR contains addressing information and the frame type, which can trigger packet drops if damaged even before the integrity of the frame is checked.

Failed timing recovery. If a jammer interferes with the preamble at the beginning of the transmission, it can cause the timing recovery to fail. A corrected phase and frequency offset are crucial for a successful packet reception, as otherwise symbol decisions are based on sub-optimal (non-peak) sampling times that decrease the SNR dramatically. This makes the symbol decisions more prone to errors, even if the jammer interfered during the preamble only. Additionally, a failure to lock onto a transmission can also cause the frame synchronization (discovering the SFD) to fail, such that a packet is overheard completely even if the incoming signal is strong.

Frame sync and damaged PHY headers. With this strategy, a jammer interferes with the SFD or PSDU length field. After the SFD is detected, the receiver knows that a frame is arriving and starts to interpret a number of incoming symbols determined by the frame length. A proactive jammer can insert SFDs on the channel to trigger frame detection events at a receiver. The receiver then fails to detect any further transmission for a period of time as it is already occupied with decoding channel noise. In addition, a reactive jammer is able to selectively block the SFD symbols such that a receiver does not detect a frame, or to introduce an error in the frame length field that also results in a misinterpretation of the frame's fields.

Limited dynamic range. Common commercial receiver designs use mechanisms that make receivers more robust in regular situations, but have a jamming amplification effect, such as Automatic Gain Control (AGC). AGC is a control loop that adjusts the amplification of incoming baseband signals to fill the complete dynamic range of the analog-to-digital converter (ADC). This enables transceiver designers to use cheap ADCs with low resolution, such as 4–6 bit [10]. However, on the downside, an adversary can exploit the AGC mechanism in two ways: either through a pre-emptive locking of the receiver to low amplification, which makes other signals too weak to receive (causing failed timing recovery or frame synchronization), or by reactively sending a strong jamming signal to the receiver that uses a high gain setting (causing clipping in the ADC and therefore symbol misdetection). Interestingly, both of these strategies affect following symbols after the jamming has ceased, as the control loop does not react instantaneously.

3.3 Effectiveness of Jamming Waveforms

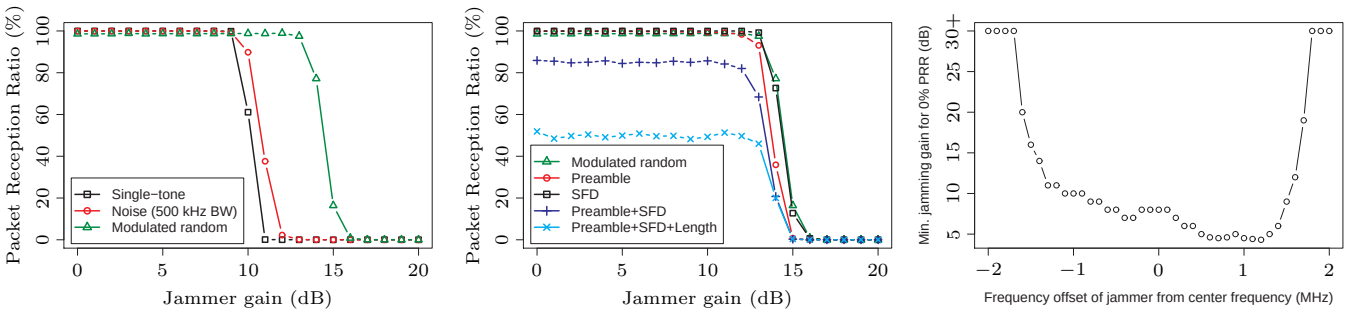
Based on the previous discussion, we want to identify jamming waveforms that are the most effective against 802.15.4. By waveform, we refer to the shape of the RF signal transmitted on the channel, specified by a sequence of I/Q samples. We check the susceptibility to three different jamming waveforms that trigger the causes presented in the previous section: symbol, timing, and frame sync errors. The signals we consider are *(i)* wideband noise, *(ii)* a narrowband continuous wave (single-tone jamming), and *(iii)* 802.15.4 modulated signals with different content, such as random symbols, preambles or SFDs to interfere with the PHY packet reception process.

3.3.1 Experimental Setting

We conduct the experiments in a room with a surface area of 4 m × 3 m, with two MICAz motes programmed as sender and receiver placed at 2 m apart, and a USRP2 as the jammer in the same room. The USRP2 is equipped with an XCRV2450 board with a maximum transmit power of 100 mW (20 dBm), and 3 dBi omnidirectional antennas. The jamming waveforms are generated on a host PC using GNU Radio. We use constant jamming and deactivate the clear channel assessment functionality of the sender such that it transmits irrespective of the channel state to ensure that we only observe physical layer effects. We do not use reactive jamming at this point because this would introduce new uncertainties into the experiment, however, the results also apply to reactive jamming. We vary the transmission power of the jammer (denoted as *jammer gain*) and measure the resulting PRR at the receiver, i.e., packets that successfully passed the CRC check despite jamming.

3.3.2 Results

A comparison of the jamming effectiveness for different waveforms is given in Fig. 3a for waveforms that mainly



(a) Impact of waveforms that interfere with the symbol decision.

(b) Waveforms interfering with timing recovery and frame synchronization.

(c) Impact of relative position of the tone jamming signals in the channel.

Figure 3: The results of the jamming experiments to identify the most effective jamming waveform.

cause symbol misdetection, and Fig. 3b for those waveforms that interfere with the 802.15.4 PHY reception process. The results for the different waveforms are analyzed below.

Noise jamming. Wideband interference is always present in wireless communications, such that the receivers are specifically designed to withstand its influence. Its main effect is chip flipping that increases the likelihood of symbol misdetection. However, for a limited power budget (e.g., 20 dBm for the USRP2) the jamming signal's power is spread over a wider spectrum, depending on the bandwidth of the signal. This is the main factor why noise jamming has a limited efficiency in our tests; we achieved the best results with a BW of 500 kHz, yet it was always a few dB less efficient in comparison to single-tone jamming to achieve a PRR of 0 %.

Single-tone jamming. We used a constant signal that is modulated on the carrier, resulting in a continuous wave. This very narrowband signal may be expected to perform badly as only a small portion of the 802.15.4 channel bandwidth is affected. However, several effects cause a superior jamming efficiency in our experiments. First, this waveform interferes with timing recovery, the receiver detects the jamming signal as a second carrier signal, and the frequency mismatch makes a phase correction impossible. The second effect is that it has the largest signal amplitude of the tested waveforms; it offers more power per Hertz with a limited power budget as the signal is more concentrated on the channel. This causes AGC to react faster, which results in chip misdetection on smaller power levels in comparison to other jamming waveforms.

Because of the first effect, the relative position of the tone in the channel is an important factor. We experimented with different offset values from the channel's center frequency, and the results are shown in Fig. 3c.[3] We observe that the channel filter of the MICAz transceiver has a width of 3 MHz, which cancels out-of-band interference. Additionally, a jamming signal directly on the center frequency is less effective in comparison to a 1 MHz frequency offset (on the corner frequency of the modulation), which complies with results in the literature [12]. Surprisingly, this effect is not symmetric. We can only speculate why this is the case, but an artifact from either the USRP2's behavior (nonlinearities in the transmitter chain) or the receiver chip is a potential explanation.

[3]Note that the measurements result from a different experimental setup and the jammer gain values are not directly comparable to the other results.

802.15.4 modulated jamming. We generated the modulated signals using the UCLA ZigBee implementation [14]. We evaluated five patterns: random symbols, preamble (`0x00`), SFD (`0xA7`), synchronization header SHR (preamble+SFD), and SHR+PHR headers (preamble+SFD+length). Each of the sequences has a different effect on the receiver. Random symbols interfere with the symbol recognition and can therefore flip symbols (Fig. 3a). We expected the preamble or SFD symbols to interfere with the timing recovery, but these two waveforms are comparable to random symbols in their jamming efficiency. The reason is that the receiver locks onto stronger preambles (the *radio capture* effect), and that SFDs without preambles are not detected by the receiver because of lacking timing recovery.

Network degradations with weaker jamming transmissions are observed for the SHR and SHR+PHR waveforms. The receiver can lock onto such jamming signals even if they are weaker than the legitimate signal. Thus, even with a smaller jammer power a severe reduction in the PRR is possible as the receiver is busy decoding noise (see the comparison in Fig. 3b). This effect can be amplified further through the use of a valid length field after the SFD, forcing the receiver to stay longer in the reception state. For a proactive jammer this attack is attractive, because even weak signals at the receiver can still cause severe reductions in the PRR.

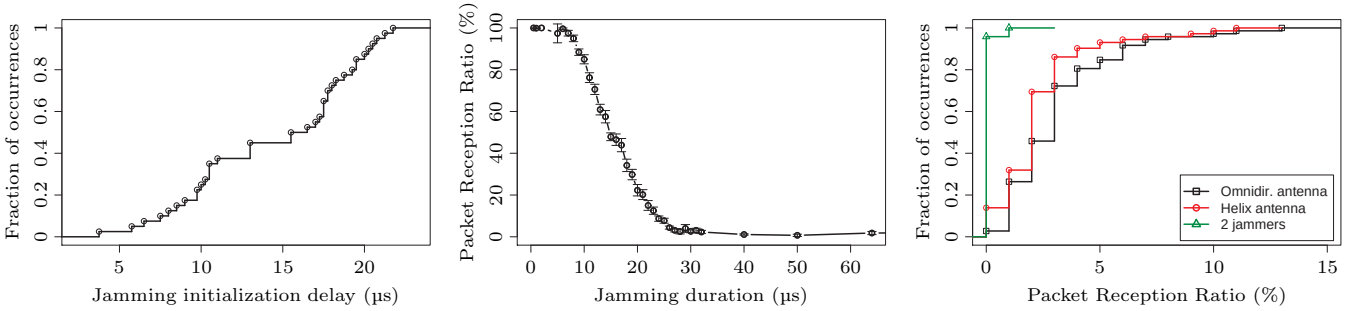
3.4 Guidelines for Effective Reactive Jamming

Considering reactive jamming, 802.15.4 modulated symbols are not as effective, since the receiver is already locked on the transmission. Due to the design choices of the transceiver in the MICAz sensor motes, single-tone jamming proves more efficient for reactive jamming than actual 802.15.4 waveforms with a limited power budget. This waveform reliably jams transmissions of the sensor motes in our experiments, and it is easily generated in software. The most efficient placement of the tone is at 1 MHz above the center frequency of the channel.

4. IMPLEMENTATION DETAILS

In this section, we explain how the widely used USRP2 can be turned into a reactive jammer.

USRP2 integration. The USRP2 platform is equipped with a Xilinx Spartan-3 FPGA running with a clock speed of 100 MHz, which provides sufficient performance and a fine-grained timing resolution of 10 ns/cycle. Additionally, the USRP2 has enough free resources (only 40 % of the FPGA is occupied) to add our prototype while reusing the function-



(a) Reaction time t_{init} of the jammer.

(b) PRR under different jamming durations (t_{jam}).

(c) Jamming performance of the system in the indoor grid scenario.

Figure 4: The results of the system performance evaluation.

ality of the original system. We modified the UHD FPGA code and firmware from Ettus Research. The operation of the USRP2 is controlled by a softcore processor in FPGA logic that executes firmware code written in C. This offers an easy integration path for our system and a maximum of reuse. However, the sequential program execution of the firmware may introduce larger time deviations into the system. The magnitude of these effects are evaluated in the next section. We added a detection module in FPGA logic that receives complex samples from the RX DSP pipeline and interrupts the firmware on a detection event. We altered the firmware to await such interrupts and to initiate the jamming process, which causes the USRP2 to start sending a ready-made jamming waveform on the channel.

Detector implementation. For every clock cycle, a new complex RF sample is available as input to the detector module. Considering the symbol duration of 16 µs in 802.15.4, we have 1600 clock cycles per symbol available, which enables complex detector designs. We implemented a PHY header (preamble+SFD) detector in our prototype. First, we perform an MSK demodulation on the signal (as explained in [14]), and feed the resulting stream of chips into a correlating receiver that detects a SHR on the channel accurately.[4] Once a SFD is detected, an interrupt is triggered at the programmable interrupt controller. Our detector adds a 4 µs delay after the SFD because of the time needed for correlation.

5. EXPERIMENTS

5.1 Micro-Benchmarks

Reaction delay. First, we determine the jamming initialization time after a packet is detected (t_{init}). More precision enables a "surgical" jamming where we can operate on a (sub-) symbol level. Further, we evaluate whether the firmware-based approach with its indeterministic timing is sufficient for our strict timing requirements. For this experiment, we place a MICAz mote close to the RX antenna of the jammer and start detecting its transmissions. The jammer schedules a jamming request as soon as an SFD is detected and initiates the transmission of the jamming signal. Using a second USRP2, we monitor and collect samples from the channel and measure the time from the end of the SFD and the beginning of the jamming signal. We use power envelope detection to identify the start of the packet and the start of the jamming signal; the resulting t_{init} is the elapsed time between these two events, minus 4 µs from the detector.

The empirical CDF (ECDF) of the experimental results is shown in Fig. 4a. We observe a delay of $t_{\text{init}} = 14.4$ µs on average, which is mainly caused by the firmware latency. For the summary of delay components, the RX/TX turnaround from the daughterboard accounts for 1 µs, a small number of FPGA cycles is spent in the TX DSP pipeline, the rest (and the deviations) is caused by the interrupt handling and the additional processing in the firmware.[5]

Necessary jamming durations. Another interesting parameter is the shortest duration t_{jam}^{min} necessary to achieve reliable jamming. Two MICAz motes are programmed as sender and receiver. To ensure that the jamming duration is the only factor in this performance measurement, the receiver is placed close to the jammer's TX antenna. For each jam duration we consider, we transmit 100 packets and measure the PRR at the receiver, with 10 repetitions each. We use a single-tone as the jamming waveform.

The experimental results are shown in Fig. 4b, 95 % confidence intervals are provided for the PRR means. The experiments show that a duration of approximately 26 µs is sufficient to reliably jam 802.15.4 transmissions. In theory, the destruction of a single symbol (16 µs) should be enough to cause a dropping probability of 93.75 % (there is still a 1 in 16 chance that the correct symbol is chosen), but due to symbol misalignments we require a slightly longer jamming duration to ensure interference with a complete symbol.

5.2 System Performance Evaluation

We evaluate our entire system in an indoor scenario to show that the system operates reliably even in challenging propagation environments.

Setting and methodology. The experiments are conducted in a 6 m × 6 m room. We measure the PRR in the presence of our reactive jammer for different positions of the receiver; here, we consider 72 positions arranged on a grid. We transmit 100 frames per position, giving an overall number of 7200 packets to jam in each experiment.

We use tone jamming, and several antenna configurations for the USRP2 jammer: for RX, an omnidirectional antenna is used, for TX we evaluate two antenna setups: *(i)* a second omnidirectional antenna, and *(ii)* a 13 dBi directional helix antenna. The helix antenna is expected to increase the

[4] In recent work, we extended this design to demodulate complete packets to get real-time access to their content.

[5] An improved implementation (that was available only after the review process) achieves faster and more deterministic results by using the softcore processor's internal interrupt handler instead of interrupt polling.

signal power at the receiver with its directional characteristics and to reduce the influence of antenna misalignments. The antennas are placed on one side of the room. We use two MICAz motes as sender and receiver, which are moved simultaneously with a constant distance of 1 m to provide them with excellent reception conditions.

Results. The ECDF of the different experiments is shown in Fig. 4c. For the omnidirectional TX antenna, 227 packets out of 7200 arrived at the receiver; the jammer has an average success rate of 96.85 %. The helix TX antenna boosts the success rate further; 168 arrived at the receiver, making an average success rate of 97.67 %. This shows that the antenna choice increases the jamming performance, yet not dramatically. Only few positions can be considered as problematic with a PRR of more than 5 % (18 % resp. 8 % of the positions in the experiments). Our analysis of missed packets showed that a jamming process can prevent a subsequent packet detection because of self-interference, because only a single board is used for both transmitting and receiving.

Two reactive jamming systems can be used concurrently to achieve better results, as this helps to minimize jamming opportunities missed by the system, and enables a better positioning of the TX antenna. No coordination is used, and both reactive jammers act independently when they detect incoming SFDs. The second TX antenna is placed on the other end of the room. This setup allowed only the reception of single packets at 3 different positions, a total of 3 packets was received in 7200 tries, resulting in a successful jamming rate of 99.96 %. This shows that redundancy is more powerful than the antenna choice in our scenario.

6. DISCUSSION

Our experimental results show that effective reactive jamming is in reach for an adversary. While our implementation presented here is specifically designed for 802.15.4, adaptations for different technologies are mainly a matter of exchanging the detector for different standards, and choosing an effective jamming waveform. Probably, the most crucial factor remains the reaction time. Nevertheless, when considering other technologies, the duration of an ACK frame for 802.11g (without legacy devices) is $t_{\text{packet}} \approx 30\,\mu\text{s}$, while our current prototype implementation reacts in 20 µs. This shows that even high-speed communication standards such as WLAN can be targeted with the system described here.

Turning bad news into good ones, we remark that our results also support recent research activities, which discuss that jamming does not only belong to an attacker but can also protect devices in the network from receiving malicious transmissions [11]. In previous work [8], we showed that injection attacks against WSNs can be mitigated in a cooperative manner by jamming packets with suspicious signal fingerprints. As we considered standard sensor motes in our experiments, a special admission frame prior to the actual data frames was necessary to relax the timing constraints in order to decouple the jamming decision from the actual jamming process. With a reactive jammer, such a protection scheme is conceivable for unmodified systems. By using different detectors, various jamming triggers can be defined, such as a full demodulation of the transmission to access the packet's content. This would also allow for a sophisticated real-time classification of packets [13], e.g., by address fields, ACKs only, or by the signal's physical properties such as the direction of arrival, device location, or RF fingerprints.

7. CONCLUSION

In this work we justified that real-time reactive jamming based on the software-defined radio paradigm is feasible and must be considered a realistic threat. Our analysis is based on a prototype implementation, which achieves a high precision of reactive jamming even if using low-cost COTS hardware such as the USRP2. Using this prototype, we provided insights to the causes for loss, and offered guidelines for successful reactive jamming against WSNs with an experimental study on physical layer effects. We evaluated the performance of our prototype system in a realistic MICAz testbed, and showed that the proposed system design offers not only a high precision but also the possibility of adapting the system to new requirements, such as reactively jamming 802.11 networks. In summary, the goal of this work was to practically demonstrate that reactive jamming should be considered as a weapon in the arsenal of the attacker. Thus research in jamming countermeasures becomes an even more important and delicate research issue in the future.

8. REFERENCES

[1] E. Bayraktaroglu, C. King, X. Liu, G. Noubir, R. Rajaraman, and B. Thapa. On the performance of IEEE 802.11 under jamming. In *Proc. of IEEE INFOCOM*, pages 1265–1273, Apr. 2008.

[2] J. T. Chiang and Y.-C. Hu. Cross-layer jamming detection and mitigation in wireless broadcast networks. *IEEE/ACM Transactions on Networking*, 19(1):286–298, Feb. 2011.

[3] R. Gummadi, D. Wetherall, B. Greenstein, and S. Seshan. Understanding and mitigating the impact of RF interference on 802.11 networks. In *Proc. of ACM SIGCOMM*, pages 385–396, Aug. 2007.

[4] IEEE Computer Society. IEEE Standard 802.15.4-2006: Wireless medium access control and physical layer (PHY) specifications for low-rate wireless personal area networks (WPANs). `http://www.ieee802.org/11/`, Sept. 2006.

[5] Y. W. Law, M. Palaniswami, L. V. Hoesel, J. Doumen, P. Hartel, and P. Havinga. Energy-efficient link-layer jamming attacks against wireless sensor network MAC protocols. *ACM Transactions on Sensor Networks*, 5(1):6:1–38, Feb. 2009.

[6] M. Li, I. Koutsopoulos, and R. Poovendran. Optimal jamming attacks and network defense policies in wireless sensor networks. In *Proc. of IEEE INFOCOM*, pages 1307–1315, May 2007.

[7] G. Lin and G. Noubir. On link layer denial of service in data wireless LANs. *Wireless Communications and Mobile Computing*, 5(3):273–284, May 2005.

[8] I. Martinovic, P. Pichota, and J. B. Schmitt. Jamming for good: a fresh approach to authentic communication in WSNs. In *Proc. of ACM WiSec*, pages 161–168, Mar. 2009.

[9] G. Nychis, T. Hottelier, Z. Yang, S. Seshan, and P. Steenkiste. Enabling MAC protocol implementations on software-defined radios. In *Proc. of USENIX NSDI*, pages 91–105, Apr. 2009.

[10] N.-J. Oh and S.-G. Lee. Building a 2.4-GHz radio transceiver using IEEE 802.15.4. *IEEE Circuits and Devices Magazine*, 21(6):43–51, Nov. 2005.

[11] K. Pelechrinis, M. Iliofotou, and S. V. Krishnamurthy. Denial of service attacks in wireless networks: The case of jammers. *IEEE Communications Surveys & Tutorials*, PP(99):1–13, second quarter 2011 (to appear).

[12] R. A. Poisel. *Modern communications jamming principles and techniques*. Artech House Publishers, Boston, MA, Nov. 2003.

[13] A. Proaño and L. Lazos. Selective jamming attacks in wireless networks. In *Proc. of IEEE ICC*, pages 1–6, May 2010.

[14] T. Schmid. GNU Radio 802.15.4 en- and decoding. Technical Report TR-UCLA-NESL-200609-06, UCLA NESL, Sept. 2006.

[15] M. Strasser, B. Danev, and S. Čapkun. Detection of reactive jamming in sensor networks. *ACM Transactions on Sensor Networks*, 7(2):16:1–29, Aug. 2010.

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

[17] W. Xu, W. Trappe, Y. Zhang, and T. Wood. The feasibility of launching and detecting jamming attacks in wireless networks. In *Proc. of ACM MobiHoc*, pages 46–57, May 2005.