2019 IEEE 39th International Conference on Distributed Computing Systems (ICDCS)

Hide and Seek: Waveform Emulation Attack and Defense in Cross-Technology Communication

Xiaonan Zhang*, Pei Huang*, Linke Guo*, and Yuguang Fang†
*Department of Electrical and Computer Engineering, Binghamton University, Binghamton, NY 13902, USA
†Department of Electrical and Computer Engineering, University of Florida, Gainesville, FL 32611, USA
Email: {xzhan167, phuang13, lguo}@binghamton.edu, fang@ece.ufl.edu

***Abstract*—The exponentially increasing number of heterogeneous Internet of Things (IoT) devices result in severe spectrum shortage and interference in the already crowded ISM band. Cross-Technology Communication (CTC) is dedicated to achieving direct communication among wireless devices with different radios and modulation schemes, which serves as an effective approach to address the above challenges. Nevertheless, CTC also provides opportunities for adversaries to manipulate IoT devices. In this paper, we identify a new attack. Built on CTC, WiFi devices are able to hide the pre-intercepted ZigBee message into their transmitted waveforms, achieving the objective of directly controlling ZigBee devices. To defend against the attack, we analyze possible strategies and consider constellation higher-order statistic analysis as the countermeasure. Extensive simulations and experiments with commodity devices (CC26x2R1) and USRP-based prototypes show the existence of the newly identified attack, and further, validate the effectiveness of the proposed defensive approach.**

***Index Terms*—Cross-Technology Communication, IoT, Emulation Attack, Physical-Layer Defense**

I. Introduction

The proliferation of the Internet of Things (IoT) applications enables ubiquitous connections among various wireless devices for bettering our daily life. According to a recent report [1], the number of IoT devices is expected to reach 55 billion by 2025, posing significant challenges on spectrum resources. Current IoT devices are using different wireless technologies, some of which share the same spectrum resources when they coexist in the common space. For example, IoT devices using the WiFi, ZigBee, and Bluetooth protocols occupying the Industrial, Scientific, and Medical (ISM) 2.4 GHz band, will lead to intense coexistence of wireless technologies. Due to their incompatibility, multiple costly and device-independent gateways are always needed to fully connect IoT devices deploying different protocols. However, the deployment of gateways not only incurs extra hardware costs but also introduces traffic overhead and communication delay. As one of the most promising paradigms, Cross-Technology-Communication (CTC) allows the direct communication among devices across different wireless technologies while bypassing the gateways, e.g., directly from WiFi to ZigBee devices or from WiFi to Bluetooth devices [2]–[4].

The work of L. Guo is partially supported by National Science Foundation (NSF) under grants ECCS-1710996, CNS-1744261, and IIS-1722731. The work of Y. Fang is partially supported by NSF under grant CNS-1717736 and IIS-1722791.

Unfortunately, the usage of CTC potentially brings severe security concerns. When the WiFi transmitter is an attacker or has been compromised by an attacker, it could control ZigBee devices directly by sending them the "well-prepared" control message. Even worse, the wide deployment, higher transmission power, and longer transmission range render larger rooms for WiFi devices to attack the short-ranged ZigBee IoT devices, such as enabling the cooling on smart thermometer during winter, unlocking the smart garage door, and turning off the security camera for break-in, etc. Since most CTC happens in the physical layer or MAC layer, existing higher-layer cryptographic approaches do not work, in the sense that most receivers get compromised soon after receiving the packets. Given that the deployment of IoT devices increases dramatically, it is critical to detect this type of attack and provide countermeasures to mitigate potential threats.

In this paper, we identify a new attack named as **CTC Waveform Emulation Attack**, where a WiFi attacker pre-intercepts the control message from the communication between a pair of ZigBee devices and further *hide*s the control message into its transmission waveform to manipulate the functionality of ZigBee devices. The WiFi emulation waveform is able to pass the decoding and demodulation processes at the ZigBee receiver, and thus it is infeasible to be detected. As a countermeasure, we propose a new defensive strategy to *seek* the malicious WiFi attacker based on constellation higher-order statistical analysis. Specifically, our contributions are listed as follows,

- We demonstrate the practicality of the waveform emulation attack from WiFi devices to ZigBee devices, where WiFi emulation waveforms are able to bypass higher-layer protocols and further control ZigBee devices.
- An effective and efficient defensive strategy is proposed to identify the WiFi emulation waveform from the authentic ZigBee waveform. To be more specific, we deploy higher order statistics to analyze the constellation diagram of the received waveform.
- Extensive simulations and experiments are conducted in both the AWGN and real environments. The results demonstrate the existence of the CTC waveform emulation attack together with the effectiveness of the proposed defensive strategy.

The rest of this paper is organized as follows. Sec. II

2575-8411/19/$31.00 ©2019 IEEE
DOI 10.1109/ICDCS.2019.00114

presents the related work. In Sec. III, we demonstrate the motivation behind the proposed CTC waveform emulation attack and the corresponding preliminary knowledge. Sec. IV introduces the adversarial model. The details of the waveform emulation attack are introduced in Sec. V while the dependence strategy is detailed in Sec. VI. We evaluate the performance of both the emulation attack and its defensive strategy in Sec. VII, followed by the conclusion in Sec. VIII.

II. Related Work

A. Cross-Technology Communication

Existing works on Cross-Technology Communication (CTC) mostly focus on how to improve the communication throughput and alleviate the cross-technology interference. B^2W^2 [3] enables the high throughput and long distance concurrent N-way cross-technology communication between Bluetooth low energy and WiFi by leveraging channel state information. Zheng *et. al* in [5] discuss interference-resilient CTC in coexisting environment. In FreeBee [6], Esense [7] and GSense [8], the communication between WiFi and ZigBee devices is enabled using RSS to measure the WiFi signal. Different from existing CTCs deploying packet-level modulation using the packet length [7], timing [6], and sequence patterns [9], [10], Li *et. al* in [2] propose a physical-level emulation technique, which motivates our newly identified attack.

B. Constellation Recognition

Automatic modulation classification (AMC) of digital modulations mounts to identify the constellation used by a digital communication system [11]. Generally, AMC algorithms can be categorized into two classes, relying on likelihood function or features of the received signal [12]. As for the QPSK constellation recognition, a hybrid likelihood ratio test (HLRT) structure is utilized to classify QPSK and BPSK modulation with unknown parameter signal level and the angle of arrival in [13], [14] respectively. Second and fourth-order moments of the received signal were applied to distinguish between QPSK and 16-QAM in [15]. Similar but different, second and fourth-order cyclic cumulants are deployed to differentiate the QPSK, 16QAM and 64QAM constellations in [16], [17]. Since the feature-based cumulant analysis has lower complexity than the likelihood function in classifying the modulation [12], we consider the cumulant analysis in our work.

III. Motivations and Preliminaries

A. Motivations

CTC enables direct communication from WiFi to ZigBee devices. Existing works mainly discuss how to improve their direct transmission efficiency [2], [6]–[8]. However, they never focus on the disadvantages brought by CTC. Therefore, we ask **what will happen if WiFi devices are compromised or controlled by attackers**. Motivated by the above work, it is highly possible for WiFi attackers to control the ZigBee devices without being detected. Specifically, WiFi devices are able to leverage CTC to emulate the observed ZigBee waveform, achieving the goal of controlling ZigBee devices while bypassing original ZigBee gateways. Due to the lack of detection methods, ZigBee devices are unable to distinguish whether the control message is coming from authentic gateways or malicious WiFi devices, and thus severe consequences will occur along with the controlled devices.

The newly identified attack is very critical due to the following reasons: 1) the waveform emulation attack fools passive ZigBee devices from the physical-layer, so the existing higher-layer cryptographic methods cannot detect it; 2) WiFi devices have longer transmission distance ($\max.\ 100m$) than ZigBee devices ($1-10m$). Hence, WiFi attackers can launch the attack without being noticed in the line-of-sight (LoS); 3) the wide deployment of WiFi-enabled mobile devices extends opportunities for launching the attack. Hence, we need to carefully design detection methods to mitigate the attack.

B. Preliminaries

We first impart preliminary knowledge about the functionalities of ZigBee and WiFi devices.

1) ZigBee Transmitter and Receiver: ZigBee devices are deployed for low-cost and low-power personal area wireless networks. Adopting the IEEE 802.15.4 protocol, they operate in the unlicensed 2.4 to 2.4835 GHz ISM bands including sixteen channels. Each channel occupies 2 MHz bandwidth with 5 MHz spaced apart. The main functions at the ZigBee transmitter and receiver are illustrated in Fig. 1. At the transmitter, Direct Sequence Spread Spectrum (DSSS) is deployed to improve interference and noise resilience. In DSSS, each hexadecimal ZigBee symbol from the MAC layer is multiplied by a pseudo-random noise spreading code, forming a 32-chip sequence. Offset Quadrature Phase-Shift Keying (OQPSK) modulation offsets the timing of odd and even chips by one chip-period. Each new pair of chips is mapped into a QPSK symbol. The duration of each ZigBee symbol lasts $16\mu s$. At the receiver side, after OQPSK demodulation and clock recovery, every 32 chips are collected and remapped into a ZigBee symbol according to the predefined symbol-to-chip spreading relationship in DSSS process. Specifically, a correlation threshold is defined to control the maximum Hamming distance between the received 32-chip sequence and the predefined chip sequence that the receiver can tolerate [18]. If the Hamming distance is less than the threshold, the received chip sequence is decoded to the corresponding ZigBee symbol. Otherwise, the chip sequence is dropped.

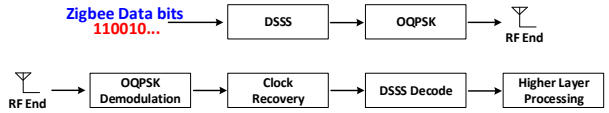


Fig. 1: ZigBee Transmitter and Receiver

2) WiFi Transmitter: WiFi transmitter deploys the IEEE 802.11g protocol to process the data from the MAC layer as shown in Fig. 2. After channel coding and interleaving, every 6 bits are mapped into one of the 64 Quadrature Amplitude Modulation (QAM) constellation points. Every 48 constellation points, together with 4 pilot symbols and 12 null

symbols are modulated onto 64 subcarriers, forming a frequency Orthogonal Frequency Division Multiplexing (OFDM) symbol. With 0.3125 MHz subcarrier space, each OFDM symbol occupies 20 MHz bandwidth. 64-point Inverse Fast Fourier Transform (64IFFT) is employed to turn each OFDM symbol into a time-domain waveform lasting $3.2\mu s$. In cyclic prefix, a guard $0.8\mu s$ interval, which is the repetition of the time-domain waveform end, is added to the beginning, forming a complete WiFi symbol lasting $4\mu s$.

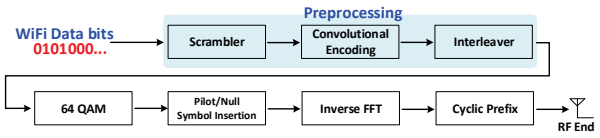


Fig. 2: WiFi Transmitter

IV. Adversarial Model

We give an example to demonstrate our adversarial model in Fig.3. Two ZigBee devices work at the central frequency 2435MHz with 2MHz bandwidth whereas WiFi devices occupy 20MHz bandwidth centered at the frequency 2440MHz. The attacking process consists of two steps as follows.

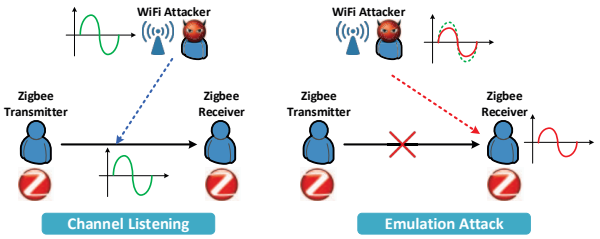


Fig. 3: CTC Waveform Emulation Attack Process

A. Channel Listening

In time slot t_1, two ZigBee devices communicate with each other (e.g., a ZigBee gateway and a smart light bulb). A WiFi attacker located close to the ZigBee receiver eavesdrops the ZigBee communication channel. Since the spectrum of ZigBee and WiFi devices are overlapped from 2434MHz to 2436MHz, the WiFi attacker observes and records the ZigBee waveform. In particular, we assume that no other devices occupy the overlapped spectrum and the WiFi attacker knows the beginning of the received ZigBee time-domain waveform.

B. Waveform Emulation

Then, the WiFi attacker moves to the ZigBee transmitter. It checks the channel availability using CSMA/CA protocol [19]. Based on [20], the WiFi attacker could sense the existence of nearby ZigBee devices. If the WiFi attacker confirms that ZigBee devices are not communicating, it emulates the received ZigBee waveform and then transmits it to the ZigBee receiver. After receiving the "legal and authentic" waveform, the ZigBee receiver continues the higher layer processing. The WiFi attacker achieves the goal of controlling ZigBee devices.

V. ZigBee Waveform Emulation Attack

A. Waveform Emulation Attack Design

1) Overview and Technical Challenges: The objective of the attack is to generate an emulated ZigBee waveform (it is actually the WiFi waveform) that is similar to the observed ZigBee waveform. Motivated by [2], a reverse process is carried out on ZigBee waveform as shown in Fig. 4. Compared with the normal WiFi transmission process in Fig.2, the WiFi attacker operates the observed ZigBee waveform reversely to get the corresponding WiFi data bits, which are sent to the ZigBee receiver after the normal WiFi transmission process. Note that each ZigBee symbol lasts $16\mu s$ whereas each WiFi symbol lasts $4\mu s$. Hence, the WiFi attacker needs to create 4 WiFi symbols to emulate one complete ZigBee symbol. Here, we focus on using one WiFi symbol to emulate $1/4$ time-domain waveform of one ZigBee symbol.

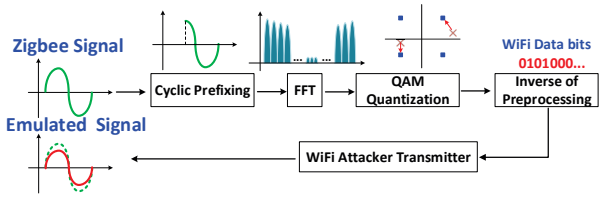


Fig. 4: ZigBee Waveform Emulation

Creating a WiFi waveform that is similar to the ZigBee waveform is a non-trivial task. In Fig.4, all three operations lead to distortions to the observed ZigBee waveform (also called original ZigBee waveform hereinafter) as follows.

- Cyclic Prefixing: In each $4\mu s$ emulated ZigBee waveform, the first $0.8\mu s$ waveform is the repetition of the last $0.8\mu s$. However, the observed $4\mu s$ ZigBee waveform does not have such a characteristic. Following the WiFi transmission protocol, the WiFi attacker has to leave out the first $0.8\mu s$ ZigBee waveform and emulate the following $3.2\mu s$, which is the input of the FFT process.
- FFT: As the reverse to IFFT in Fig.2, the WiFi attacker transforms the $3.2\mu s$ ZigBee waveform from time domain to frequency domain. After 64-point FFT operation, the frequency information of ZigBee waveform is penetrated into 64 subcarriers occupying 20MHz bandwidth. Because the operation bandwidth of the ZigBee receiver is 2MHz, it only receives the information on at most 7 subcarriers (2MHz $\approx 7 \times 0.3125$MHz) of the emulated ZigBee waveform. The information lost on other subcarriers results in irreversible distortions. How to reduce such distortions up to the hilt becomes the main challenge.
- QAM Quantization: To get WiFi data bits, the WiFi attacker has to find related 64QAM constellation points for the frequency information on subcarriers. Due to the different modulation schemes between the ZigBee and WiFi devices, the frequency information of the original ZigBee waveform cannot exactly match QAM constellation points. Their mapping process intrinsically introduces distortions. Hence, another challenge is how to uttermost diminish such difference at the WiFi attacker.

In the following, we solve the proposed challenges.

2) Choosing Frequency Points after FFT: At the WiFi transmitter, the time-domain waveform $x(n)$ after 64-point IFFT is expressed as,

$$x(n) = \frac{1}{K}\sum_{k=1}^{K} X(k)e^{-j2\pi kn/N}, \quad n = 1, 2, \cdots, N \qquad (1)$$

where $X(k)$ is the frequency component on subcarrier $e^{-j2\pi kn/N}$. $N = K = 64$. The waveform in the time domain is actually composed by the K frequency components on K subcarriers in the frequency domain. $X(k)$ also represents the importance of the subcarrier $e^{-j2\pi kn/N}$ to the waveform. Since the ZigBee receiver only receives 7 subcarriers carrying the ZigBee information, the WiFi attacker chooses the subcarriers with the largest frequency components in Fig.4.

In practice, the WiFi attacker cannot choose the frequency components for each observed ZigBee waveform due to the complexity. Since the central frequency and the bandwidth is fixed, the distribution of $X(k), k = 1, 2, \cdots, K$ is similar for each waveform. Thus, the WiFi attacker only determines the subcarrier indexes k in which the frequency components are kept. A two-step algorithm is proposed to decide the index, the *coarse estimation* and *detailed estimation*. We describe it based on the example in Table. I, where we list the frequency components of each observed ZigBee waveform in each column. Note that we ignore the frequency components with the subcarrier indexes $8-54$. In the coarse estimation, the WiFi attacker highlights all the frequency components above the threshold (3 in the example), marked as red in Table. I. In the detailed estimation, the WiFi attacker determines 7 subcarrier indexes, at which the most highlighted frequency components locate. Finally, the subcarriers with $1-4$ and $62-64$ indexes are chosen. The frequency components on these subcarrier indexes are sent into the QAM quantization.

TABLE I: Frequency Points of ZigBee Waveform

Index	1	2	3	4	5	6
1	19.8135	14.4096	14.9512	40.0943	19.8135	14.4096
2	14.2990	50.3424	44.0796	27.5399	14.2990	50.3424
3	11.1025	28.8303	23.1920	14.1483	11.1025	28.8303
4	8.3671	12.1972	14.9302	17.9765	8.3671	12.1972
5	5.6639	1.4931	5.5869	2.2252	5.6639	1.4931
6	3.0938	1.6792	3.5464	2.5908	3.0938	1.6792
7	1.0538	2.1977	1.4703	2.8351	1.0538	2.1977
...	...	...	...	...	...	...
55	1.1616	0.1748	2.5695	1.4498	1.1616	0.1748
56	0.8171	1.0029	3.2787	0.9751	0.8171	1.0029
57	0.6807	0.6807	3.0777	0.6807	0.6807	0.6807
58	1.6783	0.7128	4.6410	0.8608	1.6783	0.7128
59	2.6743	2.0764	5.2603	4.1972	2.6743	2.0764
60	2.9140	3.0542	5.9928	2.7222	2.9140	3.0542
61	1.5631	4.4502	14.0955	3.4206	1.5631	4.4502
62	4.3057	7.1549	11.4675	13.7336	4.3057	7.1549
63	39.2439	7.8455	8.4652	22.6196	39.2439	7.8455
64	40.7812	14.1395	22.7630	20.6058	40.7812	14.1395

3) Quantizing Chosen Points: According to the Parseval's theorem related to the FFT/IFFT, the energy of the waveform in the time domain is equaled to that in the frequency domain after Fourier transform. Taking the linear property, we have the following equation for the errors introduced by the QAM quantization on the chosen frequency points,

$$\int_{t=-\frac{T}{2}}^{t=\frac{T}{2}} |\hat{x(t)} - x(t)|^2 dt = T\sum_{k} |\hat{X(k)} - X(k)|^2 \qquad (2)$$

where $x(t)$ is the original ZigBee time domain waveform. The emulated ZigBee waveform is denoted as $\hat{x(t)}$. $X(k)$, $\hat{X(k)}$ are their corresponding FFT points.

The difference-energy equation (2) shows that minimizing the waveform distortion in time-domain under energy metric is equivalent to minimizing the total deviation of frequency components after QAM quantization. Therefore, QAM quantization is to choose the closest QAM constellation point in term of Euclidean distance to each of the chosen frequency points. However, the WiFi attacker just knows the 64 QAM structure as follows,

$$X(k) = \alpha\left(X_I(k) + jX_Q(k)\right) \qquad (3)$$

where $X_I(k), X_Q(k) \in\{-7, -5, -3, -1, +1, +3, +5, +7\}$ are the real and imagine part of the complex symbol $X(k)$, and α is used to scale the constellation. The attacker has to choose a scalar for QAM constellation first before quantizing the chosen frequency points. The QAM quantization becomes an optimization problem with the variable α, where the objective is to minimize the total Euclidean distance between the chosen frequency points and the QAM constellation points. Specifically, the optimization problem is formulated as follows,

$$\begin{aligned} \min_{\alpha} \quad & \sum_{k}\left(\hat{X_I(k)} - \alpha X_I(k)\right)^2 + \left(\hat{X_Q(k)} - \alpha X_Q(k)\right)^2 \\ \text{s.t.} \quad & \alpha \geq 0 \end{aligned} \qquad (4)$$

in which $\hat{X_I(k)}$ and $\hat{X_Q(k)}$ are the known real and imaging parts of the chosen frequency point $\hat{X(k)}$. Since the $X_I(k)$ and $X_Q(k)$ depend on the discrete values, the WiFi attacker employs a numerical global research method to obtain the value of the scaler, followed by finding the QAM constellation for each frequency point.

4) Carrier Allocation: Since the preprocessing in Fig.2 is invertible, the WiFi data bits related to the emulated ZigBee waveform is easily obtained given the quantized QAM constellation points. Here, we assume the WiFi attacker has obtained WiFi data bits. It modulates those bits and transmits the emulated ZigBee waveform to the ZigBee receiver. As shown in Fig.2, the pilot/null subcarrier insertion follows with 64QAM process, which is actually the subcarrier allocation process among data, pilot, and the null points. A common subcarrier allocation scheme is to put 48 data points into subcarriers $[-26, -22]$, $[-20, -8]$, $[-6, -1]$, $[1, 6]$, $[8, 20]$, and $[22, 26]$, respectively, and allocate subcarriers -21, -7, 7 and 21 to the pilot points. In addition, because WiFi and ZigBee devices operate on different central frequencies and occupy different bandwidths, the WiFi attacker allocates the quantified constellation points with ZigBee information to the subcarriers which can be received by the ZigBee receiver. Since the WiFi attacker knows the central frequency of the

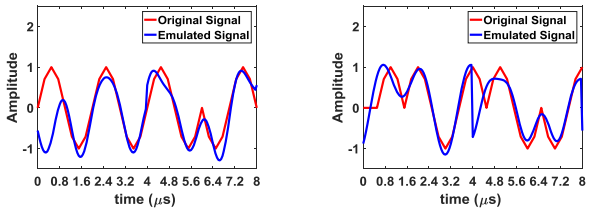
ZigBee receiver, it can set its central frequency to achieve the above goal. Taking the ZigBee 17 channel as an example, it works at the central frequency 2435MHz. The WiFi attacker sets its central frequency at 2440MHz, under which the data subcarriers $[-20, -8]$ are allocated to the constellation points carrying the ZigBee information.

B. Emulation Attack Simulation

Although distortions exist, the emulated ZigBee waveform still can pass the ZigBee receiver detection and decoding processes. As an initial validation, we simulate the CTC waveform emulation attack on the USRP N210 devices [21].

1) Simulation Process: Our simulation process follows the attacking process: channel listening and waveform emulation. We first generate an original ZigBee waveform using a ZigBee transmitter with 2MHz bands and 4MHz sampling rate. To adapt to the 20MHz sampling rate at the WiFi attacker, we interpolate the ZigBee waveform with parameter 5, creating 80 points in each WiFi symbol duration. Then, we put the last 64 points into FFT and choose the frequency points at the location $1-4$ and $62-64$, which are sent into the QAM quantization with an optimized scaler $\alpha = \sqrt{26}$. The preprocessing is ignored and the produced QAM constellation points are sent into 64-point IFFT. We add the last 16 points of the IFFT output to the beginning as the cyclic prefix. A new 80-point emulated ZigBee waveform is formed, which is actually a WiFi waveform with the sample rate 20MHz and sent to the ZigBee receiver.

2) Simulation Result: Fig. 5 plots the In-Phase and Quadrature waveform of both the original and emulated ZigBee waveforms (signals), respectively. We can see that the WiFi attacker can perfectly emulate each quarter segment of ZigBee waveform using one WiFi symbol except for the first $0.8\mu s$.



(a) In-Phase (b) Quadrature

Fig. 5: Emulated waveform Comparison

Meanwhile, we test whether the emulated ZigBee waveform can pass the detection and demodulation processes at the ZigBee receiver. In the experiment, although $0.8\mu s$ waveform in the emulated ZigBee waveform is totally different with the original one, the receiver successfully decodes the emulated waveform, which demonstrates the effectiveness.

To see whether the emulated ZigBee waveform survives in the noise environment, we also conduct the receiving test at the ZigBee receiver, where Additive White Gaussian Noise (AWGN) is added to the emulated ZigBee waveform. In each signal-to-noise ratio (SNR) value, we perform 1000 waveform transmissions from the WiFi attacker to the ZigBee receiver. The successful rate is listed in Table. II, which shows that the WiFi attacker can totally control the ZigBee devices by launching our proposed emulation attack with higher SNRs.

TABLE II: Emulation Attack Performance Under AWGN

SNR	7dB	9dB	11dB	13dB	15dB	17dB
Successful Rate	42.4%	69.2%	87.4%	93.3%	97.2%	100%

VI. Defensive Mechanism Design

In the previous section, the WiFi attacker fools the ZigBee receiver to believe that the received waveform comes from an authentic ZigBee transmitter. At the ZigBee receiver, it seems that there is no way to differentiate between the waveform from the WiFi attacker and that from the ZigBee transmitter. Existing schemes built upon higher-layer protocols are not able to thwart the proposed waveform emulation attack.

A. Defensive Strategy Analysis

Our intuition of defending the CTC waveform emulation attack is to find differences between the observed and emulated ZigBee waveforms. Although the emulated waveform is close to the observed one, different transmission schemes must leave enough "footprints", which paves the way for detection.

1) Possible Strategy Analysis: We analyze defensive strategies by scrutinizing the information flow from Fig. 1. At the first glance, cyclic prefix sheds light for us. In each emulated ZigBee waveform, the beginning and the end are the same. If the ZigBee receiver detects the repetition, it could potentially conclude that the suspicious waveform comes from the WiFi attacker. However, this methodology is not reliable. In practice, the waveform received by the ZigBee device suffers from the AWGN and even fading effects, which prevents ZigBee devices from recognizing the above repetition. We also consider using the output of OQPSK demodulation, which is the signal frequency related to the sample rate, for identifying the authentic ZigBee transmitter [22]. However, the sampling rates for both the observed and emulated ZigBee waveforms are the same at the ZigBee receiver, and thus it is infeasible to identify the attacker. Last but not least, in DSSS demodulation, the hard decision is deployed to decode the chip sequence from the chip samples. Since there are intrinsic errors between the observed and emulated ZigBee waveforms, the chip sequence from these two waveforms must be different, which may be a good candidate for detection strategies. Unfortunately, since DSSS demodulation can tolerate a certain number of errors on chip sequences for decoding, both of them will be decoded as the same ZigBee symbol even if the received chip sequences are different.

2) Constellation Analysis: The QAM quantization motivates us to differentiate the received waveform in the view of the constellation. If the waveform comes from an actual ZigBee transmitter, it has the QPSK constellation in the time domain; if not, the waveform has the 64-QAM constellation in the frequency domain. Without transforming to the frequency domain, the constellation analysis can be easily done in the

time domain. Compared to the actual ZigBee waveform, the emulated waveform has much larger offsets coming from the quantization errors and the FFT process (i.e., losing non-overlapping high-frequency components), both of which serve as the basis for detecting the waveform emulation attack.

To identify the emulated ZigBee waveform, we first get complex symbols from the received time-domain waveform. Considering the DSSS decoding in Fig. 1, every 32 float values are collected, which are then determined as binary 0 or 1 chip and mapped into one ZigBee symbol according to the predefined symbol-to-chip spreading relationship. At the ZigBee transmitter, the output of DSSS is OQPSK modulated. Hence, we consider to use the input of the DSSS demodulation to construct a new QPSK constellation diagram. Specifically, we divide those input as odd and even parts, where odd parts are put to the real axis and even parts being put to the imaginary axis. Therefore, the defensive strategy becomes a simplified constellation recognition problem. In particular, we carry out the digital modulation classification [11] to determine whether the newly constructed constellation diagram belongs to a QPSK structure or not.

In what follows, we mainly consider two scenarios for emulation attack detection. In the ideal scenario, the received waveform only suffers AWGN at the ZigBee receiver side. In the practical scenario, the frequency/phase offset happens at the received waveform due to the complex channel condition.

B. Emulation Attack Detection under Ideal Scenario

Higher-order statistic is a common and easy method used in digital modulation classification problem, which efficiently characterizes the shape of the distribution of the noisy baseband samples. Given the newly constructed constellation diagram, we focus on the fourth-order cumulant characteristics.

1) Preliminaries: For a complex-value random variable x, its second-order moments are defined in the following,

$$C_{20} = E[x^2], \quad C_{21} = E[|x|^2] \tag{5}$$

As for the fourth-order moments and cumulants, they can be defined in three different ways,

$$\begin{aligned} C_{40} &= \mathtt{cum}(x,x,x,x) \\ C_{41} &= \mathtt{cum}(x,x,x,x^*) \\ C_{42} &= \mathtt{cum}(x,x,x^*,x^*) \end{aligned} \tag{6}$$

where x^* represents the conjugate the random variable x, and for zero-mean random variables w, x, y, and z,

$$\begin{aligned} \mathtt{cum}(w,x,y,z) = E(wxyz) - E(wx)E(yz) - \\ E(wy)E(xz) - E(wz)E(xy) \end{aligned} \tag{7}$$

2) Sample Estimation: According to [23], we use the collected complex sample $d_i, i = 1, 2, \cdots, D$ output from the Clock Recovery to estimate (5) and (6) as follows,

$$\widetilde{C}_{20} = \frac{1}{D}\sum_{i=1}^{D} d_i^2, \quad \widetilde{C}_{21} = \frac{1}{D}\sum_{i=1}^{D} |d_i|^2 \tag{8}$$

where $\widetilde{\cdot}$ denotes the sample average. Considering the fourth-order cumulant estimation using complex samples, we have,

$$\begin{aligned} \widetilde{C}_{40} &= \frac{1}{D}\sum_{i=1}^{D} d_i^4 - 3\widetilde{C}_{20}^2 \\ \widetilde{C}_{41} &= \frac{1}{D}\sum_{i=1}^{D} d_i^3 d_i^* - 3\widetilde{C}_{20}\widetilde{C}_{21} \\ \widetilde{C}_{42} &= \frac{1}{D}\sum_{i=1}^{D} |d_i^4| - |\widetilde{C}_{20}|^2 - 2\widetilde{C}_{21}^2 \end{aligned} \tag{9}$$

In (8), the sample estimates of the second-order cumulants include the effect of the noise random variable. Thus, a local estimate of its variance has to be obtained and subtracted from $\widetilde{C}_{20}$ and $\widetilde{C}_{21}$. In addition, such a noise effect affects the estimate of the fourth-order cumulants according to (9). However, the constellations are not necessarily normalized after decoding at the ZigBee receiver in practice. To deal with the problem, the fourth-order cumulant estimates are usually normalized as $\widehat{C}_{4q} = \widetilde{C}_{4q}/\widetilde{C}_{21}^2$, where $q = 0, 1, 2$. The final normalized fourth-order cumulant estimates are then compared with the corresponding theoretical cumulants in order to decide the constellation type, which are shown in Table. III.

TABLE III: Theoretical Cumulants for $C_{21} = 1$

Modulation	C_{20}	C_{40}	C_{42}
BPSK	1	−2.0000	−2.0000
QPSK	0	1.0000	−1.0000
PSK(> 4)	0	0.0000	−1.0000
4-PAM	1	−1.3600	−1.3600
8-PAM	1	−1.2381	−1.2381
16-PAM	1	−1.2094	−1.2094
16-QAM	0	−0.6800	−0.6800
64-QAM	0	−0.6190	−0.6190
256-QAM	0	−0.6047	−0.6047

3) Defensive Strategy: As shown in Table. III, both C_{40} and C_{42} are used to decide constellation types among PSK, PAM and QAM. Specific to our defensive strategy, since the reconstructed constellation is known to be QPSK modulation, we mainly compare how far the estimated fourth-order cumulants are to the theoretical values.

We first define a Voronoi tessellation [24] of the feature space as $\mathbf{v} \triangleq [C_{40}, C_{42}]^T$, where C_{40} and C_{42} are the theoretical values as listed in Table. III. Similarly, our estimated fourth-order cumulants $\widehat{C}_{40}$ and $\widehat{C}_{42}$ compose a new vector $\phi = \left[\widehat{C}_{40}, \widehat{C}_{42}\right]^T$. The Euclidean distance D_E is deployed to be the distance measure metric between the Voronoi tessellation $\mathbf{v}$ and our estimated vector ϕ, where $D_E = ||\phi - \mathbf{v}||_{\mathbf{2}}$. We decide whether the received waveform is transmitted by the ZigBee transmitter or the WiFi attacker by deploying the hypothesis testing. Specifically, we have,

$$\begin{cases} H_0 : \text{From the ZigBee Transmitter} \\ H_1 : \text{From the WiFi attacker} \end{cases} \tag{10}$$

If the waveform comes from the WiFi attacker, the error brought the FFT and QAM quantization puts a negative effect

to the decision of the constructed constellation type. Here, we introduce a threshold Q to help us make the decision,

$$D_E^2 \underset{H_0}{\overset{H_1}{\gtrless}} Q \tag{11}$$

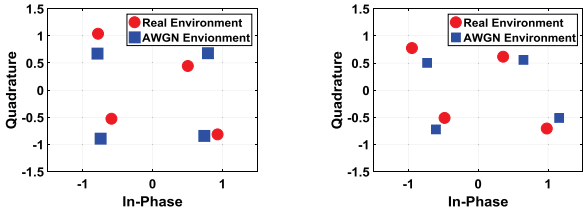
We will give the value of Q according to our experiments.

C. Emulation Attack Detection under Real Scenario

To show the constellation performance difference in AWGN and real environments, we first give an example of the newly constructed constellation as shown in Fig. 6. Given the chip samples, we deploy k-means clustering algorithm [25] to help find the constructed constellation points. Denote the chip samples as a set $S_c = \{s_{c1}, s_{c2}, \cdots, s_{cC}\}$, where C is the number of chip samples, k-means clustering algorithms aim at partitioning all the chip samples into 4 sets $\mathbf{S} = \{S_1, S_2, S_3, S_4\}$ so as to minimize the within-cluster sum of squares. Mathematically, its objective is to find:

$$\underset{\mathbf{S}}{\operatorname{argmin}} \sum_{i=1}^{4} \sum_{S_c \in S_i} ||S_c - \mu_i||^2 \tag{12}$$

where μ_i is the mean of points in S_i.



(a) ZigBee waveform (b) Emulated waveform

Fig. 6: Constellation Diagram Comparison

From Fig. 6, the constellation in the real environment has an obvious phase offset compared to that in AWGN environment. Facing the phase offset effect, we reconsider the higher-order statistics deployed in the constellation recognition for the AWGN environment. Denote the frequency offset and the phase offset as Δf and θ, respectively. According to [26], C_{40} is scaled by $e^{j(\Delta f + \theta)}$. To avoid the frequency and phase offset, we consider the estimate of the absolute value $|C_{40}|$ instead of C_{40} in the Voronoi tessellation in the real environment.

VII. Performance Analysis and Evaluation

We start from a thorough complexity analysis on both attack and defensive approaches. Following it, we implement experiments to further demonstrate the effectiveness of ZigBee waveform emulation attack and the proposed defensive strategy in both the simulation and real environment, respectively.

A. Complexity Analysis

1) Waveform Emulation Attack: The attacking process mainly consists of FFT and QAM quantization. The N-point FFT is done with $\mathcal{O}(N \log N)$. The coarse estimation after FFT is a binary hard-decision process with $\mathcal{O}(M)$, where M denotes the number of samples. Following it, we sum up the binary elements in each row and get a final vector, where each element denotes the number of the highlighted waveform samples related to the subcarrier index. The detailed estimation is to sort the vector and find the first 7 maximized elements, which has the complexity $\mathcal{O}(n)$, where n is the number of total subcarriers. The QAM quantization includes finding the optimal scalar and mapping the frequency components of the ZigBee waveform to the QAM constellation. Meanwhile, our global search method is based on the mapping process. According to [2], choosing the closest N QAM points in term of total Euclidean distance to each of K FFT points of desired waveforms is easily done in $\mathcal{O}(K)$.

In general, FFT has a complexity $\mathcal{O}(N \log N)$. However, N fixed at 64 while others depend on the number of the samples from coming ZigBee waveform. Therefore, the waveform emulation attack can be done easily in $\mathcal{O}(M)$, where M is the number of the coming ZigBee samples.

2) Defensive Approach: The main part of our defensive strategy is to calculate the fourth-order cumulants. According to [23], the fourth-order cumulants estimation can be done in $\mathcal{O}(N)$, where N denotes the complex sample number. Therefore, our proposed defense strategy is efficient to be implemented with the order of the sample number.

B. Simulation Settings

We construct two complete communication links including APP layer, MAC layer, and PHY layer from the ZigBee transmitter to the ZigBee receiver and from the WiFi attacker to the ZigBee receiver, respectively. We assume that the WiFi attacker has knowledge about the waveform sent by the ZigBee transmitter. The WiFi attacker follows the waveform processing detailed in Section V. Besides, we add another function in the ZigBee receiver to achieve the defensive strategy as described in Section VI.

An additive white Gaussian noise (AWGN) with the noise variance σ^2 is transmitted along with the observed and emulated ZigBee waveforms respectively. The power of the transmitted waveform is normalized and we define the signal-to-noise ratio SNR as $SNR = \frac{1}{\sigma^2}$. For each communication link, the transmission and reception process repeat 100 times. We collect the physical-layer data in the first 50 times to calculate the threshold in (11) at the ZigBee receiver. The rest of the physical-layer data is used in the hypothesis testing showing the effectiveness of the proposed defensive strategy.

C. Simulation Results

1) Performance of Waveform Emulation Attack:: We denote the text from **00000** to **00099** as the input of the APP layer. The ZigBee transmitter sends its waveform directly to the ZigBee receiver. The WiFi attacker emulates its waveform and then transmits the emulated one to the ZigBee receiver. We demonstrate the chip-level performance in Fig.7, which shows the Hamming distance distribution of the received chips. When the waveform comes from the ZigBee transmitter, the received chip sequences are exactly the same with the predefined chip sequences. There are 4 to 8 error chips between each chip

sequence and the predefined one when the emulated ZigBee waveform is received. Since DSSS has the error resilience, the sequences with error chips could be decoded as the correct ZigBee symbols given a feasible threshold. In our simulation, all of the emulated waveforms are decoded correctly with a feasible threshold of 10. Such an observation further testifies that the WiFi attacker could control the ZigBee device by considering the principle of the DSSS explained in Sec. III-B1.

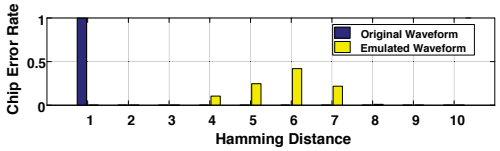
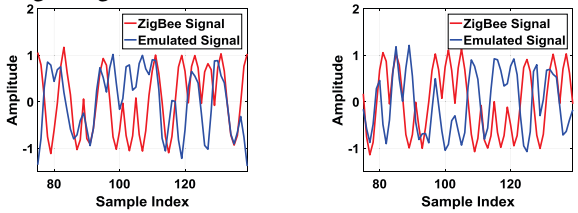


Fig. 7: Hamming Distance Distribution Comparison

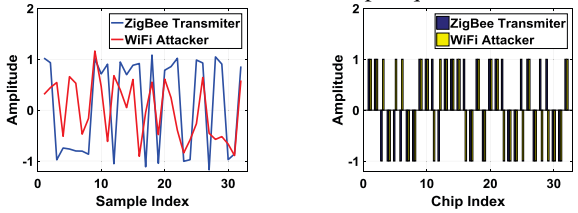
2) Performance of Possible Strategy: We first show the defensive approach performance of the possible strategies given in Sec.VI-A1. For the experiment, we choose high SNR to avoid the noise effect. Fig.8 shows the received In-Phase and Quadrature waveform at SNR = 17dB, respectively. It is hard to find the repeated segment from the waveform. Thus, we can hardly identify the emulation attacker by comparing the beginning and the end of the received waveform.



(a) In-Phase (b) Quadrature

Fig. 8: Waveform Comparison

We also demonstrate the output of the OQPSK demodulation process in Fig.9a, which shows the signal frequency in relation to the sample rate. It is obvious that the trends of these two waveforms are the same, and thus we cannot use the output from the OQPSK demodulation to distinguish the transmitters. In addition, we show the chip sequence performance after hard decision in DSSS demodulation in Fig. 9b. Although the chip sequence performance under the ZigBee and emulated waveform cases are totally different, the ZigBee receiver can obtain the same ZigBee symbol. Thus, we cannot detect the WiFi attackers from the chip sequences.

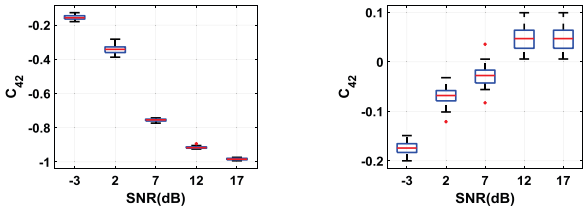


(a) Phase (b) Chip Amplitude

Fig. 9: Received waveform Comparison

3) Performance of Constellation-based Approach: To demonstrate the effectiveness of our proposed constellation-based defensive strategy, we conduct experiments at different SNRs to evaluate the fourth-order cumulants C_{40} and C_{42} performances of waveforms from the ZigBee transmitter and the WiFi attacker, respectively.

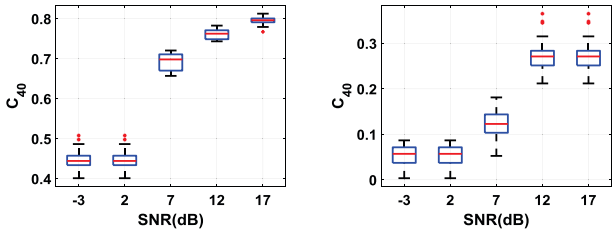
As shown in Fig. 10, we mainly compare the value of C_{42}, where more approaching to the theoretical value -1 will be categorized as authentic ZigBee transmitters. In Fig.10a, it shows the C_{42} performance of the original ZigBee waveforms. With the increase of SNR, the value of C_{42} will be much closer to -1. However, the C_{42} value of emulated waveforms are far from the theoretical value and keeps on changing to an opposite way. Due to the errors in the QAM quantization and the information lost on the non-overlapped frequencies, the newly constructed constellation under the emulated waveform intrinsically has an offset to the QPSK constellation. As the SNR becomes lower, the noise with larger variance decreases such offset on the contrary. Therefore, the trends of the C_{42} under the emulated waveform and ZigBee waveform cases are opposite. Such an observation validates the effectiveness of our proposed defensive strategy.



(a) ZigBee waveform (b) Emulated waveform

Fig. 10: C_{42} Performance

The fourth-order cumulant C_{40} performance is also demonstrated in Fig. 11. The calculation methods are the same as the C_{42}. Comparing the value of C_{40} under the ZigBee waveform in Fig.11a and the emulated waveform case in Fig.11b, the C_{40} value under the ZigBee waveform case is more close to the theoretical value 1 than that under the emulated waveform. However, the ZigBee receiver cannot distinguish the transmitter using the above trends because it cannot get the C_{42} and/or C_{40} performance of the received waveform at different SNRs at once. Therefore, the predetermined threshold decision is needed for WiFi attacker detection.



(a) ZigBee waveform (b) Emulated waveform

Fig. 11: C_{40} Performance

4) Effectiveness of Threshold Decision: When receiving a waveform, the ZigBee receiver cannot know the transmitter except for calculating the value of $\widehat{C}_{40}$ and $\widehat{C}_{42}$. For detection purpose, it needs a threshold to decide whether the waveform

is from the ZigBee transmitter and the WiFi attacker. Note that we have demonstrated that the packet reception rate is low at the SNR below 7dB when the waveform is coming from the WiFi attacker as listed in Table. II. Thus, we reconsider the fourth-order estimation performance at the SNR above 7dB. Instead of the Euclidean distance, we first calculate average Euclidean distance square using the first 50 waveform samples under both the ZigBee waveform and emulated waveform at each SNR, which are listed in the Table. IV. We observe that there is a large gap between the ZigBee waveform and emulated waveform, which makes our decision on the threshold easier. To find out the specific threshold value, we calculate D_E as,

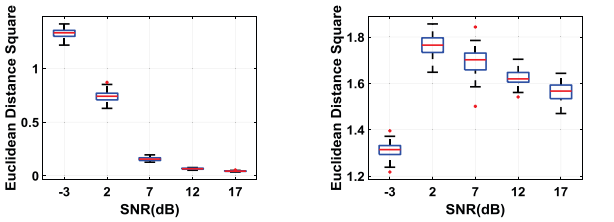
$$\begin{aligned} {D_E}^2 = ||\phi - \mathbf{v}||_2^2 &= (\widehat{C}_{40} - C_{40})^2 + (\widehat{C}_{42} - C_{42})^2 \\ &= (\widehat{C}_{40} - 1)^2 + (\widehat{C}_{42} + 1)^2. \end{aligned}$$

We decide the threshold of $\widehat{C}_{42}$ as -0.5 and $\widehat{C}_{40}$ as 0.5. Therefore, the final threshold Q in (11) becomes 0.5.

TABLE IV: Averaged Euclidean Distance Square (${D_E}^2$)

SNR	7dB	12dB	17dB
ZigBee waveform	0.1546	0.0642	0.0421
Emulated waveform	1.7140	1.6238	1.5536

The average of the Euclidean distance square over 100 ZigBee waveform samples and 100 emulated waveform samples is shown in Fig. 12. We observe that the maximum ${D_E}^2$ is below 0.5 at the SNR above 7dB for the ZigBee waveform while the minimum ${D_E}^2$ is above 0.5 for the emulated waveform at the corresponding SNR. Since the WiFi attacker can fool the ZigBee devices at the SNR above 7dB, the ZigBee receiver can distinguish the ZigBee waveform and the emulated waveform effectively by using our proposed defensive strategy.



(a) Tested ZigBee waveform (b) Tested Emulated waveform

Fig. 12: Defense Strategy Performance

D. Experimental Settings and Results

We conduct the experiment using the USRP N210 and the commodity device TI LaunchPad CC26x2R1 [27]. The USRP N210 is equipped with AD and DA converters before the RF front ends and UBX-40 daughter boards operating in the 2.4GHz range as transceivers. Its corresponding software toolkit is GNURadio [28]. The LaunchPad CC26x2R1 is part of the micro-controller unit (MCU) platform supporting the IEEE 802.15.4g protocol. In the experiment shown in Fig.13, we deploy one USRP N210 as the ZigBee transmitter and WiFi attacker alternately.

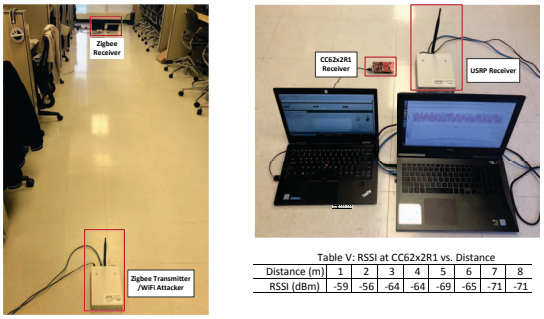


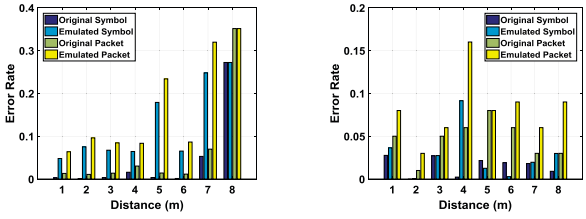
Table V: RSSI at CC62x2R1 vs. Distance

Distance (m)	1	2	3	4	5	6	7	8
RSSI (dBm)	-59	-56	-64	-64	-69	-65	-71	-71

Fig. 13: Experimental Setting

The ZigBee transmitter works on the spectrum centered at 2435MHz with the sample rate 4MHz. Whereas the WiFi attacker operates at the center of 2440MHz with the sample rate 20MHz. Their power gains are set at 0.75. Because the ZigBee receiver decodes the sequence only after getting a zero sequence, we add 10 zero points at the beginning of each emulated packet. The other USPR N210 and the launchpad CC26x2R1 play the role of the ZigBee receiver. They are centered at the 2435MHz. The received power gain of the USRP receiver is set to 0.75. The distance between the transmitter and receivers ranges from 1m to 8m. During the experiment, there are human activities such as walking. We illustrate the value of received waveform strength indication (RSSI) at the launchpad CC26x2R1 in Table. V in Fig.13. RSSI is an indication of the power level being received by the receiving radio after the antenna loss [29].

We mainly focus on the performance of the waveform emulation attack in the practical environment. As the same as that in the simulation, the ZigBee transmitter and the WiFi attacker send the text from **00000** to **00099**, respectively, we evaluate the error rates of packets and symbols at the USRP receiver and CC26x2R1 respectively. As shown in Fig.14, the error rates of both packets and symbols are lower than those of the emulated packets and symbols. This is because the noise and interference in the real scenario enlarge the difference between the original and emulated ZigBee waveform at the ZigBee receiver. Meanwhile, the packet error rate is larger than the symbol error rate because the packet is received correctly only if all the symbols in the packet are exactly received.



(a) Receiver: USRP (b) Receiver: CC26x2R1

Fig. 14: Waveform Emulation Attack Performance

As demonstrated in Fig.14a, the error rates of both packets and symbols are less than 0.1 for both original and emulated ZigBee waveforms when the distance between the transmitter and the USRP receiver is below 5m. When the distance increases, e.g., 7m, the WiFi attacker could not fool the ZigBee

device due to the large error rate. At the distance 8m, the USRP receiver cannot decode the original ZigBee waveform either. Thus, it is obvious that a WiFi attacker performs worse than the ZigBee transmitter at the USRP receiver. However, shown in Fig.14b where CC26x2R1 is deployed as the receiver, the error rates of both the packets and symbols are less than 0.1 even if the distance between the WiFi attacker and the receiver is long, e.g., 8m. Since the commodity ZigBee device has stronger demodulation functions than the USRP, we conclude that the proposed waveform emulation attack could effectively fool the ZigBee device even from a long distance.

TABLE V: Emulation Attack Performance

Distance	1m	2m	3m	4m	5m	6m
ZigBee Waveform	0.0004	0.0007	0.0011	0.0103	0.0003	0.0007
Emulated Waveform	1.1426	1.8706	1.4818	1.3215	2.0024	1.2152

Since the effective attack distance is below $6m$ using USRP as the ZigBee receiver, we set the distance between the ZigBee transmitter/the WiFi attacker and the ZigBee receiver from $1m$ to $6m$. We calculate the averaged ${D_E}^2$ based on the 5000 waveform samples under both the original and emulated waveform cases. Their ${D_E}^2$ values are listed in Table. V. As can be seen, ${D_E}^2$ values for the original ZigBee waveforms are under 0.1 whereas those for the emulated waveforms are above 1. We could choose the decision threshold from the interval $[0.1, 1]$ such that the WiFi attacker is detected when it tries to control the ZigBee devices. The observation shows the feasibility of our detection methods in the real environment.

VIII. Conclusion

In this paper, we discovered a new emulation attack built on CTC, where the WiFi device fully controls the ZigBee device directly while bypassing the ZigBee gateway. To defend against this attack, we proposed a countermeasure to identify the WiFi attacker by analyzing constellation higher-order statistics. We perform a thorough evaluation on the USRP platform in both AWGN and the real environment. The experimental results demonstrated the effectiveness of the CTC emulation attack and its defensive strategy.

References

[1] "Iot report how internet of things technology is now reaching mainstream companies and consumers," https://www.businessinsider.com/internet-of-things-report.

[2] Z. Li and T. He, "Webee: Physical-layer cross-technology communication via emulation," in *Proceedings of the 23rd Annual International Conference on Mobile Computing and Networking*. ACM, 2017, pp. 2–14.

[3] Z. Chi, Y. Li, H. Sun, Y. Yao, Z. Lu, and T. Zhu, "B2w2: N-way concurrent communication for iot devices," in *Proceedings of the 14th ACM Conference on Embedded Network Sensor Systems CD-ROM*. ACM, 2016, pp. 245–258.

[4] X. Guo, Y. He, X. Zheng, L. Yu, and O. Gnawali, "Zigfi: Harnessing channel state information for cross-technology communication," in *Proceedings of ACM INFOCOM*, 2018.

[5] X. Zheng, Y. He, and X. Guo, "Stripcomm: Interference-resilient cross-technology communication in coexisting environments," in *IEEE Int. Conf. Comput. Commun.(INFOCOM)*, 2018, pp. 15–19.

[6] S. M. Kim and T. He, "Freebee: Cross-technology communication via free side-channel," in *Proceedings of the 21st Annual International Conference on Mobile Computing and Networking*. ACM, 2015, pp. 317–330.

[7] K. Chebrolu and A. Dhekne, "Esense: communication through energy sensing," in *Proceedings of the 15th annual international conference on Mobile computing and networking*. ACM, 2009, pp. 85–96.

[8] X. Zhang and K. G. Shin, "Gap sense: Lightweight coordination of heterogeneous wireless devices," in *INFOCOM, 2013 Proceedings IEEE*. IEEE, 2013, pp. 3094–3101.

[9] W. Jiang, Z. Yin, S. M. Kim, and T. He, "Transparent cross-technology communication over data traffic," in *INFOCOM 2017-IEEE Conference on Computer Communications, IEEE*. IEEE, 2017, pp. 1–9.

[10] Z. Yin, W. Jiang, S. M. Kim, and T. He, "C-morse: Cross-technology communication with transparent morse coding," in *INFOCOM 2017-IEEE Conference on Computer Communications, IEEE*. IEEE, 2017, pp. 1–9.

[11] P. A. Forero, A. Cano, and G. B. Giannakis, "Distributed feature-based modulation classification using wireless sensor networks," in *Military Communications Conference, 2008. MILCOM 2008. IEEE*. IEEE, 2008, pp. 1–7.

[12] O. A. Dobre, A. Abdi, Y. Bar-Ness, and W. Su, "Survey of automatic modulation classification techniques: classical approaches and new trends," *IET communications*, vol. 1, no. 2, pp. 137–156, 2007.

[13] L. Hong and K. Ho, "Bpsk and qpsk modulation classification with unknown signal level," in *MILCOM 2000. 21st Century Military Communications Conference Proceedings*, vol. 2. IEEE, 2000, pp. 976–980.

[14] ——, "Modulation classification of bpsk and qpsk signals using a two element antenna array receiver," in *Military Communications Conference, 2001. MILCOM 2001. Communications for Network-Centric Operations: Creating the Information Force. IEEE*, vol. 1. IEEE, 2001, pp. 118–122.

[15] C. J. Le Martret and D. Boiteau, "Modulation classification by means of different orders statistical moments," in *MILCOM 97 Proceedings*, vol. 3. IEEE, 1997, pp. 1387–1391.

[16] P. Marchand, C. Le Martret, and J.-L. Lacoume, "Classification of linear modulations by a combination of different orders cyclic cumulants," in *spwhos*. IEEE, 1997, p. 0047.

[17] P. Marchand, J.-L. Lacoume, and C. Le Martret, "Multiple hypothesis modulation classification based on cyclic cumulants of different orders," in *Acoustics, Speech and Signal Processing, 1998. Proceedings of the 1998 IEEE International Conference on*, vol. 4. IEEE, 1998, pp. 2157–2160.

[18] A. Goldsmith, *Wireless communications*. Cambridge university press, 2005.

[19] G. Bianchi, L. Fratta, and M. Oliveri, "Performance evaluation and enhancement of the csma/ca mac protocol for 802.11 wireless lans," in *Proceedings of PIMRC'96-7th International Symposium on Personal, Indoor, and Mobile Communications*, vol. 2. IEEE, 1996, pp. 392–396.

[20] C.-J. M. Liang, N. B. Priyantha, J. Liu, and A. Terzis, "Surviving wi-fi interference in low power zigbee networks," in *Proceedings of the 8th ACM Conference on Embedded Networked Sensor Systems*. ACM, 2010, pp. 309–322.

[21] "Usrp n210," https://www.ettus.com/product/details/UN210-KIT.

[22] B. Bloessl, C. Leitner, F. Dressler, and C. Sommer, "A gnu radio-based ieee 802.15. 4 testbed," *12. GI/ITG KuVS Fachgespräch Drahtlose Sensornetze (FGSN 2013)*, pp. 37–40, 2013.

[23] A. Swami and B. M. Sadler, "Hierarchical digital modulation classification using cumulants," *IEEE Transactions on communications*, vol. 48, no. 3, pp. 416–429, 2000.

[24] S. Fortune, "Voronoi diagrams and delaunay triangulations," in *Computing in Euclidean geometry*. World Scientific, 1995, pp. 225–265.

[25] P. S. Bradley and U. M. Fayyad, "Refining initial points for k-means clustering." in *ICML*, vol. 98. Citeseer, 1998, pp. 91–99.

[26] A. Swami and B. Sadler, "Modulation classification via hierarchical agglomerative cluster analysis," in *Signal Processing Advances in Wireless Communications, First IEEE Signal Processing Workshop on*. IEEE, 1997, pp. 141–144.

[27] "Simplelink cc26x2r1 sdk overview," http://dev.ti.com/tirex/content/simplelink_zigbee_sdk_plugin_1_60_00_14/docs/zigbee_user_guide/html/zigbee/simplelink_cc2652_sdk_overview/simplelink_cc2652_sdk_overview.html.

[28] "Gnu radio," https://www.gnuradio.org/.org.

[29] "Cc2652r simplelink multiprotocol 2.4-ghz wireless mcu," http://www.ti.com/lit/ds/symlink/cc2652r.pdf.