IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received June 15, 2020, accepted July 4, 2020, date of publication July 10, 2020, date of current version July 22, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3008431

Anti-Malicious Attack Algorithm for Low-Power Wake-Up Radio Protocol

HYUNHEE PARK, (Member, IEEE)
Department of Information and Communication Engineering, Myongji University, Gyeounggi-do 17058, South Korea
e-mail: hhpark@mju.ac.kr

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government Ministry of Science and ICT (MSIT) 2019R1F1A1060742.

ABSTRACT The in-depth discussion on the development of green technology at the 25th Wireless World Research Forum (WWRF) Conference in November 2010 indicated the growing interest in energy conservation technology that needs to be applied in next generation mobile communication. Green Internet of Things (IoT) technology is expected to be included as part of green communication. In this study, to meet the low power-consumption requirements of IoT devices, IEEE 802.11ba—a standard that minimizes power consumption significantly—is introduced. IEEE 802.11ba minimizes the power consumption of wireless devices by using power-saving technology that activates wireless local area network (WLAN) chips only when necessary. Thus far, no discussions have been provided for wake-up radio (WUR) technology regarding non-sleep attacks, where the WUR receiver ends up waking up the main radios accidentally or remains in the non-sleep mode because of malicious attacks such as spoofing. Therefore, in this paper, a general operating procedure of WUR for ensuring low-power consumption and an algorithm for detecting malicious attacks are proposed. Further, an operating process for responding to malicious attacks is defined. The extensive simulation results show that the proposed anti-malicious attack WUR (AMA-WUR) protocol reduces the average packet delay by 62.84% compared with original WUR protocol, and reduces the average power consumption by 93.71% with original WUR protocol while the flooding attack vulnerabilities.

INDEX TERMS Wake up radio, IoT, low power consumption, DoSL attack, malicious attack.

I. INTRODUCTION

The telecommunications field has been expanding considerably along with a proportional increase in the electrical energy consumption of the devices. Often, performance criterion such as spectral efficiency or data rate are optimized at the cost of greater energy consumption [1]. However, with recent advances in wireless telecommunication technologies, the development of low-cost and low-energy wireless Internet of Things (IoT) devices can be achieved. Nevertheless, energy consumption remains an important factor as it can determine overall network lifetime [2].

The IoT technology is expected to provide users with smart solutions; unfortunately, power sources are not sufficiently smart yet [3]. The IoT devices typically have a shorter lifetime than expected because they often have a limited source of energy [4], [5]. When the battery on an IoT device is drained, communication is suspended until the battery is replaced; this suspension extends to other devices as the dead IoT device cannot be relied upon when relaying data [6]. Therefore, it is critical to ensure power for IoT devices is available, especially sensors, in order to maintain a stable communicating environment; this requirement makes energy efficiency and power management an important criteria for future IoT applications.

Communication radios such as radio frequency (RF) chip account for a significant percentage of energy consumption in an IoT device compared to other device components such as sensor, controller, etc. [7]. This is because of a phenomenon known as idle listening, which occurs when devices have to stay active to the communication medium and listen for incoming signals from neighboring devices even when no data are transmitted or received. In addition, the idle listening works for when the device has to stay active mode to the communication medium. The idle listening function is required to prevent data latency and packet retransmission when the destination device is in sleep mode. The idle listening process commonly used by IoT devices contributes to

The associate editor coordinating the review of this manuscript and approving it for publication was Ding Xu.

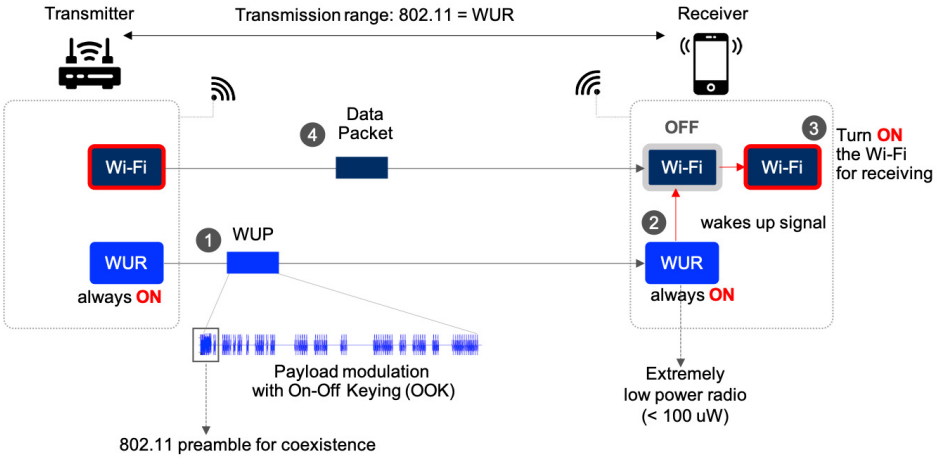


FIGURE 1. Basic WUR operation: The transmitter can be the AP and the receiver can be the IoT sensor device using the WUR chipset.

overall energy consumption. Another issue that contributes to energy consumption in device transceivers is overhearing, which occurs while continuously listening to incoming signals, where the IoT devices may end up receiving unrelated signals [8]. A Wireless Sensor Network (WSN) is defined as a small-scale gathering device of a sensor used for monitoring, sensing, capturing, and processing the data around an application [9], [10]. As a result, these devices are resource-needy and at the same time extremely dependent on battery control, storage, computation, data size and, accessible bandwidth [11]. The WSN can be defined as a network of tiny devices, called sensor nodes, which are spatially distributed and work cooperatively to communicate information gathered from the monitored field through wireless links. Specifically, sensor nodes can be IoT devices when the data gathered by the different nodes is sent to a sink that either uses the data locally or is connected to other networks. For this, a duty cycle is considered an important design component to extend the life of WSNs. The duty cycle is a technique where a device is periodically placed into the sleep mode which is an effective method of reducing energy dissipation in WSNs. As the duty cycle increases, the devices can sleep longer and more energy will be saved, whereas few of the devices are available to participate in data routing, which will decrease the overall throughput and increase transmission latency. Therefore, the duty cycle can alleviate energy consumption occurring from idle listening and overhearing at the IoT devices. The radio remains in the sleep mode by default and switches to the active mode via internal clock synchronization in order to send and receive data. The main task for the duty cycle is to wake up devices at the exact time when the packets are sent (i.e., reception). If a sleep period longer than the active period is reserved, more energy is conserved; however, the increased latency increases the possibility of packets being lost during the network latency and sleep period. In addition, synchronized packets used within the devices to set the wake-up time in the duty cycle may increase overheads, which may increase energy wastage. Several techniques [12] have been proposed to address this challenge, such as the use of spatial scheduling [13] and cognitive radio [14].

Another solution to address idle listening and overhearing in network systems is using a wake-up radio (WUR) [15]. A WUR wakes the system from low-power radio (e.g., can be WUR) by connecting to the main radio (e.g., can be primary connectivity radio (PCR)) when an incoming signal is detected. Using this device, the sleep mode can be maintained without the waking up the main radio or when receiving an incoming signal. This completely solves the idle listening problem in IoT devices without clock synchronization overheads. Wake-up messages and data can be communicated on different channels, allowing both message types to be transmitted simultaneously, in turn reducing collision possibility. To improve the performance and efficiency of a WUR, various hardware designs and protocols have been developed [16], [17].

There are two types of WURs based on power-usage rate: active WUR and passive WUR. The active WUR is the second low-power radio that receives constant power from an external power supply such as a battery. In contrast to the active WUR, a passive WUR does not require power from batteries or other physically connected power supplies. Rather, it secures the energy from the transmitted wake up signals. Passive WUR requires minimal energy through this energy securing process; however, the receiver sensitivity of the passive WUR is relatively low, which results in a short wake-up distance range.

As shown in Figure 1, when the data to be sent to the devices is generated in the access point (AP; i.e., transmitter), the AP sends a wake up packet (WUP) to the device (i.e., receiver), and the IEEE 802.11ba chipset of the target device, which receives the WUP, immediately wakes up the

IEEE 802.11 series (e.g., 802.11n/g/ac/ax) chipset through the wake-up signal. Subsequently, actual data transmission is performed using the IEEE 802.11 series chipset, which is a traditional wireless local area network (WLAN) technique. Thus, IEEE 802.11ba can drastically minimize energy consumption by operating IEEE 802.11 series only when data reception is required [18]. However, if the WUR device cannot manage to enter the sleep mode, the standardized WLAN technology exhibits no difference in power consumption compared to typical WLAN technology [19]. In addition, an IoT device will ultimately become a wireless device with a low data rate as it uses on-off keying (OOK) for data modulation [20]. Furthermore, in the case of IoT devices using small capacity batteries, the problem becomes more serious [21]. For instance, an IoT device will operate only when there is power; however, if the WUR technology is unable to enter the sleep mode (i.e., an always awake state such as idle listening) because of an unknown attack—such as spoofing attack, denial-of service (DoS) attack, and denial-of sleep (DoSL) attack—it is no different from having the power consumption of a typical wireless LAN. A DoS attack aims to render devices unavailable by interrupting the device's normal functioning. DoS attacks typically function by overwhelming, or flooding, a targeted machine with requests until normal traffic is unable to be processed, resulting in a denial of service to users. DoSL attacks exhaust the batteries of target devices by increasing their duty cycle. By forcing nodes to awake at unnecessary times or by inducing additional duty (e.g., listening, retransmissions), these attacks aim at reducing the expected lifetime of the constituted IoT network [22], [23]. Popular sleep-denial attacks either transmit unauthenticated packets or replay a recorded traffic [24]. Even though unauthenticated packets would be discarded due to failed authentication, decoding causes receivers to waste energy. Such spoofing attacks that deny services by preventing IoT devices from entering the sleep mode on the network are called DoSL attacks [24], [25].

As a countermeasure for DoSL attack, Rainer and Hans-Joachim defined wake-up token (WUT) for secure WUR operation and strengthened the authentication process of sending and receiving data [26]. The receiver would have a list of WUT and, if the WUT list matches the token reference values received for authentication, the device wakes up from sleep mode to maintain active mode. As the attacker would not know the WUT, sleep attacks become impracticable. However, in this case, an overhead issue may arise due to requirement for a WUR device to maintain WUT information for all devices. In general, WUR is not suitable for receiving and storing all the WUT information as it is one of the communication techniques that maintain a very low data rate due to OOK modulation. In addition, sleep attacks could take place if an attacker can acquire the WUT information. In addition, WUR key exchange protocol has been proposed for DoSL attacks [12]. The proposed key exchange protocol is executed for an establishment of a common secret key between legitimate peers. The secret key generated from the setup phase is used in the process of WUR device generating and updating pseudo-random WUR sequence during normal network operation [12]. This study also proposes an algorithm that prevents unauthorized WUR devices from participating in WUR communication by generating a secret key for DoSL-like attacks. Although it is possible to counter these attacks by conducting additional authentication process for a participating device in WUR communication, the overhead issue could occur in the process of generating and sending encryption keys.

The previous research proposed for secure WUR communications had limitations where only the WUR devices that have been authorized through authentication process could participate in communications. To the best of our knowledge, we are the first to propose and validate a fully architected protocol for non-sleep DoS attacks in WUR networks with capabilities of defined wake-up packet.

In this study, a case wherein the WUR receiver accidentally wakes up the main radio because of a malicious attack on an IoT WUR device operating with a limited energy source such as small capacity battery is discussed. Various attacks that can occur in WUR networks are classified by type, and new detection methods for managing these attacks are proposed. In addition, using these detection techniques, a WUR system structure that can effectively detect and prevent malicious attacks is presented. Therefore, in this study, the features of attacks against WUR devices are analyzed and a method to detect the attacks before the WUR device internally wakes up the PCR is discussed. Further, if an actual attack is detected, a method of notifying the AP through the WUR network without waking up the PCR is proposed.

The remainder of this paper is structured as follows. In Section II, the types of attacks on the WUR networks are analyzed and the structure of WUR packets is explained. In Section III and Section IV, the detection and prevention methods named the anti-malicious attack WUR (AMA-WUR) algorithm for the mentioned attacks are detailed. In Section V, a system is implemented to describe the simulation results of detecting attacks on WUPs. Finally, in Section VI, the significance of this study and future research are discussed.

II. PRELIMINARIES

A. RELATED MALICIOUS ATTACK TYPES

Malicious attacks can be divided into flooding attacks, which generate a large amount of packet traffic; connection attacks, which require an excessive number of sessions; and other attacks that utilize other application features [27].

- A flooding attack is a type of attack that interrupts normal service provision by depleting the resources of the target system and network by randomly transmitting normal packets. In Request flooding attack, the attacker send sessions that contains more number of requests than the normal users, which leads to flooding. When a flooding attack towards a WUR device occurs on an IoT-based WUR network configured to operate at low

power, power consumption increases drastically and the target WUR device cannot receive normal services.

- A connection attack is a kind of distributed denial-of-service (DDoS) attack with excessive number of half-open sessions. Traffic patterns can be established for a transmission control protocol synchronize sequence numbers (TCP SYN)-flood type attack. Incomplete (e.g., half-open) connections mean that the session has not completed the TCP three-way handshake; hence, the session in not established. When a connection attack toward a WUR device occurs on an IoT-based WUR network, the WUR session is not established. As the WUR session is not established, the threshold value for network connections increases significantly, while the target WUR device cannot receive normal service.
- An application-based attack is a kind of DDoS attack that sends out requests following the communication protocol; thus, these requests are indistinguishable from legitimate requests in the network layer. For instance, the attacker sends sessions that contain larger amount of high workload requests. The ultimate aim of the attacker is to devour resources like central processing unit (CPU) and memory of the server and degrade them. Furthermore, the attacker sends a hypertext transfer protocol (HTTP) request slowly in a piece-meal manner (one at a time) and the request is not complete initially. As a result, the server keeps the indulged resources in waiting stage until it receives the entire data. This attack is categorized into Slowloris attack, HTTP fragmentation attack, slow post attack, and slow reading attack.
- In amplification-based flooding attacks, the attacker initiates small domain name system (DNS) queries with forged source IP addresses that provoke a large extent of network traffic. And the DNS response messages are significantly larger than DNS query messages. As a result, this large extent of network traffic is directed towards the targeted system in order to incapacitate it. When an amplification-based attack toward a WUR device occurs, WUR resources are exhausted. These attacks can threaten WUR devices easily using general applications, such as HTTP and DNS of the WUR device.

As WUR devices that can operate in a green IoT environment have security vulnerabilities, prevention methods are defined for WUR frame transmission in the IEEE 802.11ba standards [18]. An AP that supports WUR operation (i.e., WUR AP) can send protected WUR frames to a device that supports WUR operation (i.e., WUR DEV). In the IEEE 802.11ba standards, the WUR integrity group temporal key (WIGTK) and WUR temporal key (WTK) are defined for use in individually protecting the WUR wake-up frames [18]. Based on a secure key-based protection, a WUR device discovers the AP's security policy through passively monitoring the Beacon frames or through active probing. After discovery, the WUR device performs Simultaneous Authentication of Equals (SAE) authentication using authentication frames with the AP. However, these definitions are only present in the standards. As the WUR device is designed to have a very low modulation rate, such as OOK modulation, it could not include the abovementioned secure keys due to the vendor-specific implementation. Therefore, it is directly exposed to the previously mentioned malicious attacks. Consequently, an anti-malicious attack algorithm for a WUR device designed to have a low modulation rate is essential.

B. OPERATING PROCEDURES OF WUR PACKET

Each WUR frame consists of the following basic components:

- A *MAC header*, which comprises frame control, identifier (ID), and type dependent (TD) control fields;
- A variable-length *frame body*, which, if present, contains information specific to the frame *type*;
- A frame check sequence (*FCS*) field, which contains either a 16-bit cyclic redundancy check (CRC) or a 16-bit message integrity code (MIC).

Figure 2 depicts the general MAC frame format for WUR frames. The MAC header of WUR frame consists of Frame Control, ID, and TD Control fields. The Frame Body field is optional in certain WUR frame types. The MAC header and the last FCS field comprise the minimal WUR frame format and are present in all WUR frames, including reserved types. The format of the frame control field is also illustrated in Figure 2. The Type field indicates the type of the WUR frame. The ID field contains an identifier for the WUR frame, which is selected from Figure 3. In Figure 3, basic service set identifier (BSSID), organizational unique identifier (OUI), and least significant bit (LSB) are stand for basic service set identification, organization unique identifier, and least significant bit, respectively. The identifier depends on the

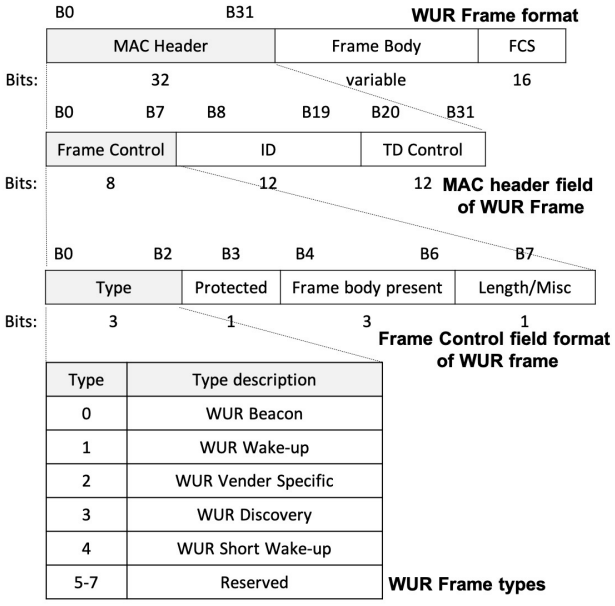
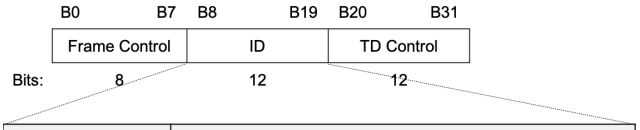


FIGURE 2. WUR frame format.



ID field	Identifier description
Transmitter ID	Identifier of the transmitting AP
Nontransmitter ID	Identifier of the nontransmitted BSSID
Group ID	Identifier of a group of receiving WUR non-AP STAs
WUR ID	Identifier of an individual receiving WUR non-AP STA
OUI ID	The 12 LSBs of the OUI

FIGURE 3. Identifiers of WUR frames.

type of WUR frame. With the WUR frames defined above, two devices perform WUR tasks through the Transmit ID for the AP and WUR ID for the WUR device. The WUR devices that have confirmed IDs through the association process may perform one-to-one communication.

III. DETECTION METHOD FOR MALICIOUS ATTACKS

Figure 4 shows the operation process for waking up a WUR device using an AP with a normal ID.

- Step 1: The AP sends a WUP to the device (DEV) carrying the DEV's wake-up ID (WUID).
- Step 2: The DEV's WUR receiver (WURx) receives the WUP and wakes up its PCR (i.e., Wi-Fi radio).
- Step 3: The DEV sends a wake-up response frame defined WUR acknowledgment (ACK) to the AP using its PCR, indicating that the DEV's PCR is woken up because it received a WUP (as opposed to sending an RTS, data, or a wake-up response frame indicating that the DEV wakes up on its own, when the DEV indeed wakes up on its own).
- Step 4: After the AP receives the wake-up response frame,
  1) because the AP did send the WUP to the DEV's WURx, the AP considers the DEV to be safe.
  2) then, the AP proceeds with the data exchange with the DEV's PCR.

As shown in Figure 4, upon receiving the WUP, the WUR receiver wakes up the PCR internally to ensure it remains in the receiving mode. However, problems may occur if the received WUP is not from a normal WUR transmitter but from an attacker with malicious intent. Repeated attacks such as this can drain the battery quickly and eventually deactivate the WUR device as well.

Malicious attacks on a WUR device running on a small capacity battery can cause the WUR to wake up its PCR falsely and disable the device [28]. Potential solutions can be encrypting the low-power Wakeup Request (WUReq), or having the WUReq frame carry secure information that is known to both the AP and the destined device. However, these solutions increase the complexity and power consumption of the WURx.

In this study, two typical attack methods that can occur in WUR networks are proposed as follows: an attacker can send continuous WUPs in the form of DDoS attack [29], and an attacker can send spoofed WUP for malicious attacks.

A. AMA-WUR ALGORITHM FOR DoSL ATTACKS

When a malicious attack occurs, packets transmitted in the network have certain characteristics as described below.

- Source addresses are distributed widely. Although ports can vary depending on the attack tools, a large number of packets may be directed to a specific target device, which results in the concentrated distribution for a destination address.
- To detect such malicious attacks, it is essential to determine the load capacity of the network in its normal state. Specific parameters for network attack analysis should be defined, and thresholds for the parameter values of the normal state should be set. Some of the popular parameters used are CPU usage and load, packet size and packet header information distribution, distribution of protocols that show the distribution of network services by types, maximum and average values of the overall traffic volume, presence of concentration on a certain host, monitoring of flows using spoofing addresses, and network flow information used to identify a flow surge on a network. By combining these parameters, the network traffic can be analyzed to increase the reliability of traffic characteristic information.

Because there are many possible combinations of parameters and there is a need for the data analysis of incoming network packets, actual implementation becomes complex and leads to a wastage of system resources. Thus, an algorithm that configures thresholds automatically according to the characteristics of a given system is proposed to minimize implementation complexity and a false-positive rate while discarding the need for analyzing WUR network packets.

As described in Section II, WUR frames only carry WUR ID information and do not include other information such as that related to specific ports. Therefore, a normal state is determined by considering the number of WUR frames that can be sent per enhanced distributed channel access (EDCA) transmission opportunity (TXOP). Since the WUR protocol uses an 802.11-based communication protocol with EDCA protocol and TXOP duration, it requires contention-based channel access [30]. This information is defined in the IEEE 802.11ba standards as follows: as described in channel access, a WUR AP can transmit multiple WUR wake-up frames in a TXOP. That is, in this study, an attack is determined as the DoSL attack if an abnormally large number of WUPs is received based on the calculation of average utilization rate and instantaneous utilization rate, and if the threshold of WUPs occurring in one typical EDCA TXOP is performed.

A unit time of the TXOP is expressed as τ, and it is calculated as follows. To determine the TXOP for a given service interval (SI), the average arriving transmission rate

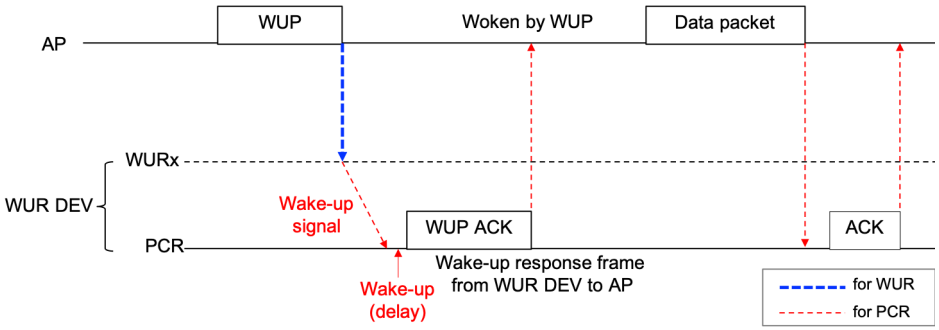


FIGURE 4. WUR normal wake-up procedure.

of WUP is calculated using

$$\alpha_j = \left\lceil \frac{S * \rho_j}{l_j} \right\rceil, \tag{1}$$

where α_j, ρ_j, l_j, and S represent the arriving transmission rate of WUP, jth average transmission rate, size of nominal WUP scale, and SI, respectively. The jth TXOP interval length is calculated using

$$\tau_j = \max_j \left(\frac{\alpha_j * l_j}{R_j} + t_o, \frac{F_{max}}{R_j} + t_o \right), \tag{2}$$

where R_j represents the transmission rate of the physical layer supporting WUR, t_o represents the time overhead by inter-frame space (IFS) and ACK duration, and F_{max} represents the maximum WUR frame size allowed.

For the operating process of the trained threshold algorithm proposed in this study, the average utilization rate and instantaneous utilization rate of each TXOP are required as parameters for comparing the normal and attack states of the WUR network. These rates can be calculated as follows. If the random TXOP unit time calculated using Equation (2) is expressed by τ, then the number of WUPs received during a certain TXOP time can be calculated. Using α_j from Equation (1), the average utilization rate for each TXOP can be obtained as

$$\Omega = \frac{\sum_{\tau_j=0}^{T} \alpha_j}{T_B}, \tag{3}$$

where T_B denotes the WUR beacon interval. In addition, the instantaneous utilization rate for each TXOP interval can be calculated using

$$\Omega_j = \frac{\alpha_{j-1} + \alpha_j}{2}, \quad j \geq 1. \tag{4}$$

This algorithm undergoes a certain training period after the association of WUR AP and WUR DEV. During this training period, the system calculates each TXOP interval and WUP utilization rate. Moreover, the system gradually increases the threshold limit for each TXOP; after the training period, each port has its own optimal threshold limit number specialized for the system. Based on this learned threshold limit, the DoSL attack can be detected with a reduction in the false-positive rate on the user's normal traffic.

B. AMA-WUR ALGORITHM FOR SPOOFED ATTACKS

An attack by spoofed packets cannot be solved using the previously described trained threshold. To solve the case of such an attack, the WURx undergoes a process of determining whether the WUP is an attack instead of immediately waking up the PCR upon receiving a spoofed WUP. Thus, for attack detection using the previously mentioned trained threshold, if the WUP occurs such that the designated threshold in the TXOP intervals is exceeded, an attack can quickly be detected. However, if a single spoofed WUP is received and the designated threshold is not exceeded, the attack may not be detected at all.

IV. PREVENTION METHOD FOR MALICIOUS ATTACKS

Once the trained threshold is obtained after the training period of the algorithm described earlier, common DoSL WUP attacks can be detected. As shown in Figure 5, if WURx determines that the received packet through a training period is an attack, it will send a WUP ACK to an AP to notify it of an attack without waking up the PCR. The AP, which receives the WUP ACK, grants a new WUID for the corresponding WUR device. Once the WUR device receives this new WUID, it will send an ACK using the new WUID. Ultimately, herein, the PCR of the corresponding WUR did not wake up. Furthermore, the DoSL attack was immediately detected, and a new WUID was received from the AP in response.

As the second malicious attack case, to solve this problem of spoofed attacks, WURx is designed to immediately send a WUP ACK to the AP instead of waking up the PCR even when the threshold limit is not exceeded upon continuous incoming WUPs. The WUP ACK sent from this process is defined such that it includes a wake-up reason code. If the AP that receives the WUP ACK with a wake-up reason code determines that the WUP was not sent by the AP, it sends a new WUID to notify WURx of the attack. After acquiring a new WUID, WUR device transmits an ACK using the new WUID. Ultimately, in this study, the PCR of the corresponding WUR device did not wake up, the notification of an attack from the AP was received, and a new WUID was acquired from the AP in the response.

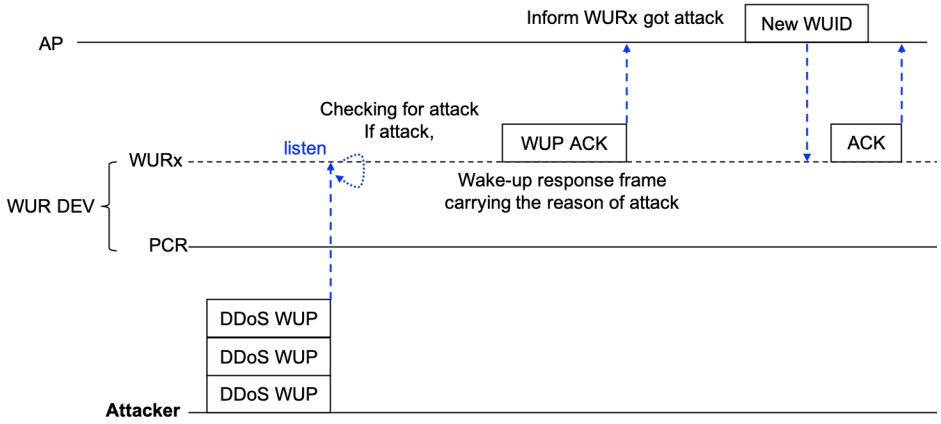


FIGURE 5. Example of DoSL attack and AMA-WUR prevention method.

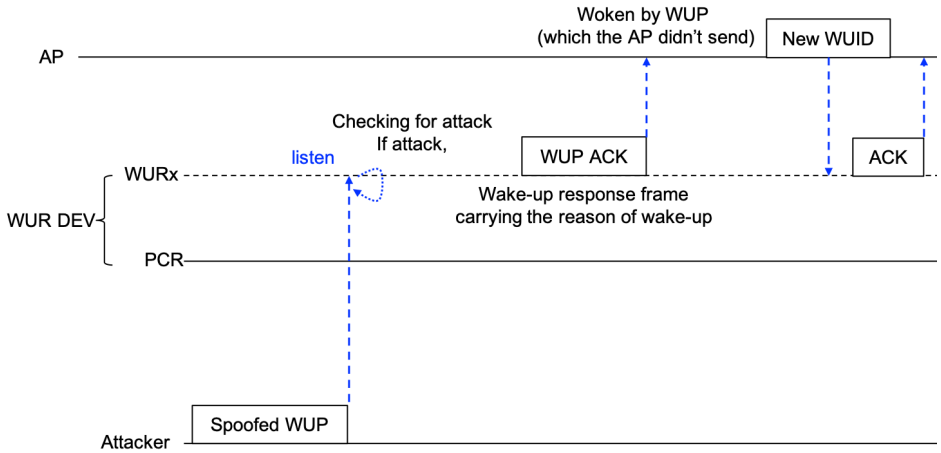
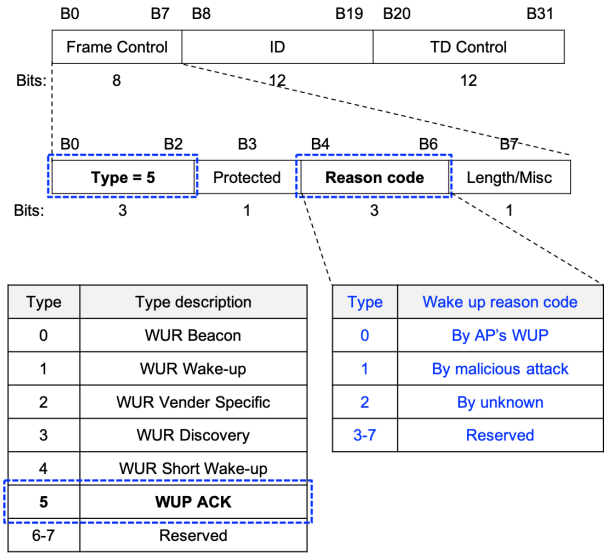


FIGURE 6. Example of spoofed attack and AMA-WUR prevention method.

New WUP ACKs are shown in Figures 5 and 6. To define a new WUP ACK, the MAC header field format of the WUR frame is used, as shown in Figure 2, and it as defined in IEEE 802.11ba. The newly formatted frame control defined in Figure 2 is shown in Figure 7 as follows. As shown in Figure 7, the WUP ACK is defined using one reserved bit (i.e., Type = 5) in the WUR frame types. When the type is 5, the 3-bit of *Frame body present* that follows Type are set as the wake-up reason code. Wake-up reason codes are in the form of 3 bits and they can be defined. Wake-up reason code types are set as follows: Type 0 of wake up reason code for general responses to the call of AP; Type 1 of wake up reason code for the cases wherein the WUR device recognizes a specific malicious attack through the training phase; Type 2 of wake up reason code for the cases of unknown reasons; and Types 3–7 of wake up reason code reserved for any additional definitions. The AP can quickly determine that the WUR device is exposed to malicious attacks if the received WUP ACK by Type 0 is not the WUP it transmitted.



Type	Type description
0	WUR Beacon
1	WUR Wake-up
2	WUR Vender Specific
3	WUR Discovery
4	WUR Short Wake-up
5	**WUP ACK**
6-7	Reserved

Type	Wake up reason code
0	By AP's WUP
1	By malicious attack
2	By unknown
3-7	Reserved

FIGURE 7. WUP ACK type and wake up reason code.

V. EXTENSIVE SIMULATION

A. SIMULATION SETUP

In this study, the WUR operation process is simulated using the OMNeT++ network simulator engine, which is a discrete event simulator. OMNeT++ (www.omnetpp.org) is an extensible, modular, component-based C++ simulation library and it has domain-specific functionality such as sensor networks, wireless ad-hoc networks, Internet protocols, and performance modeling. The MiXiM (mixed simulator) framework, which is implemented for WSN, is used in modeling a hardware platform consisting of dual radio systems in OMNeT++. First, WUR DEVs comprising different spatial device densities from a WUR AP are constructed; here, 5–50 WUR DEVs are used. For this case, the WUR DEVs are randomly deployed to monitor a 10,000m^2 area (100 × 100 m). The WUR AP is located at the center of this monitoring area. Then, a 24-h scenario is proposed to simulate a building monitoring the case. The reference WUR design considered in this study is based on the subcarrier modulation WUR introduced in [31] and [32]. Consider a WUR topology as shown in Fig. 8, with an average packet arrival rate of $\lambda = 10$ packets/s and variable traffic load represented by different numbers of devices, $N \in 5, 10, 15, \ldots, 50$, in the network. In a typical Wi-Fi network, up to approximately 30 devices are assumed to be deployed in the simulation environment of 100 x 100m area due to relative performance. For example, when there are present 30 devices in deployed WLAN, the best effort traffic reaches a peak, and then the performance falls quickly [33]. However, in order to reflect the system scalability, the experiment is expanded to feature up to 50 WUR devices in a 100 × 100m area. Additionally, the percentage of having an attack on each topology is set as random. In other words, an attack may or may not take place on a topology composed of 5 devices. Likewise, an attack may or may not take place on a topology composed of 50 devices. Further, one device becomes an attacker in a single topology in any given case. For example, if an attack takes place in a topology composed of 30 devices, 29 devices become normal WUR devices while one device becomes the attacker. $\alpha = 0.1$ is defined as the percentage of attack occurrence. The remaining parameters are configured based on the specifications listed in Table 1.

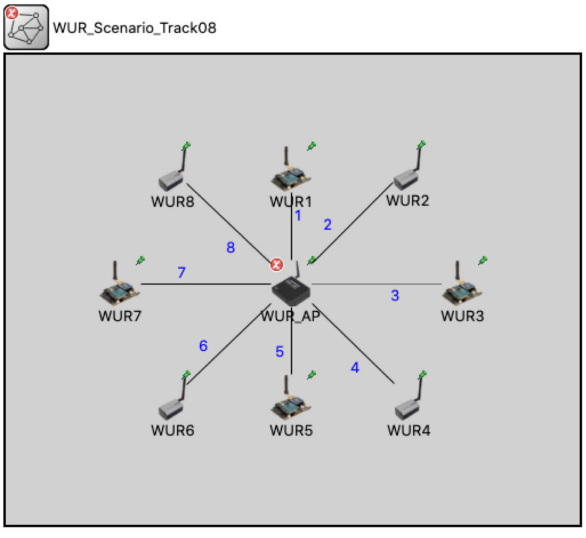


FIGURE 8. Example of simulation topology on OMNeT++.

TABLE 1. Simulation parameter configuration [19], [34], [35].

Radio Type	Parameter	Value
Main radio	Supply voltage	3 V
	Data rate	10, 100, 250 kbps
	Transmission current	17.4 mA
	Reception current	18.8 mA
	Idle current	20 μA
	Time slot	9 μs
	Short Inter-frame Space(SIFS) duration	16 μs
	Payload size	35 bytes
	ACK frame size	11 bytes
Wake-up radio	WUP duration	6, 12ms
	WUP transmission current	15.2 mA
	WURx current	8 μA
	Sleep current	3.5 μA
	Backoff current	5.16 μA
	CCA current	20.2 μA
	MCU switching time	1.8ms
	CCA duration	4 μs
	W for CW	16, 32, 64 slots
	WUP size	2, 4 bytes
	Maximum WUP attempts	7 times

The IEEE 802.11ba standards define the carrier sense multiple access/collision avoidance (CSMA/CA) protocol for the operation of the WUR medium access control (MAC) protocol. Therefore, in this study, the general WUR MAC protocol is defined as CSMA Enabled-WUR (CE-WUR). The operation procedure of the CE-WUR protocol is presented in Fig. 9. The CE-WUR works in a manner similar to the unslotted CSMA/CA MAC protocol of IEEE 802.15.4; however, it is tailored to WUPs. In CE-WUR, upon the detection of an event by a device, it first performs a backoff (BO) procedure without checking whether the channel is idle. As soon as the BO waiting time ends, it checks the channel status by performing a clear channel assessment (CCA). If it finds the channel idle for a duration of CCA, it sends a WUP; otherwise, it repeats the BO and CCA procedure. Further, CCA duration is defined as the duration in which the device can check the preamble, and CCA in the WLAN environment is defined as the smallest time duration. For example, if the time slot is defined as 9μs in the WLAN, CCA duration is set as 4μs, and the CCA detection probability from this 4us duration is defined as 90% or more. CCA typically has two mechanisms: preamble detection and energy detection. In general, the preamble detection threshold is statistically defined as 4dB SNR in detecting the 802.11 preamble. On the other hand, the energy detection (ED) threshold is set as 20dB higher than that of the preamble detection (PD) (i.e., ED = PD + 20dB). As such, these two CCA thresholds may fluctuate due to the differences in radio reception sensitivity. In this paper, the WUR device is assumed to conduct energy detection for the CCA. In such a case, RF can be adequately detected before an attack takes place as the CCA threshold is greater than that of the preamble detection.

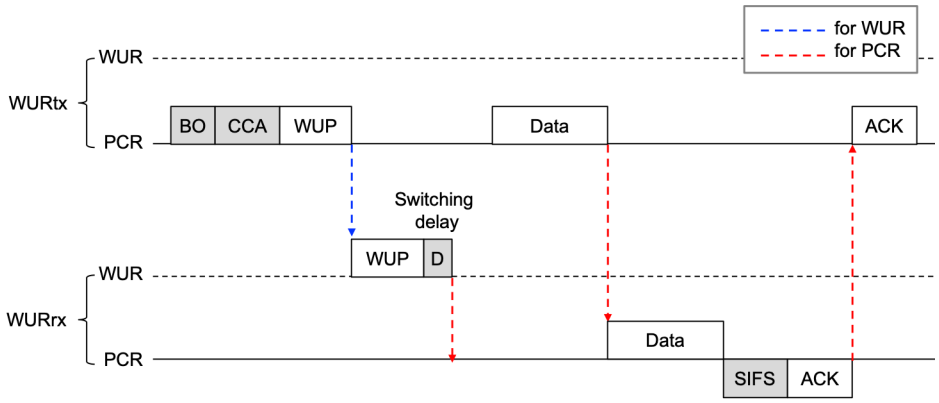


FIGURE 9. Operation procedure of the CSMA-CA enabled WUR protocol.

A WUR detects a received signal and compares it with the carrier sense threshold per slot. For this, CCA duration of WUR is defined as the duration in which the device can check the RF energy.

A WUR decreases its backoff counter by 1 if the channel is idle in a slot, and leaves it unchanged otherwise. When its backoff counter is decreased to 0, a WUR activates. A WUR module manages the contention window (CW) based on the number of slots, besides initializing the backoff counter. Although the interaction between a WUR and a WLAN module causes a delay, this delay can be made small by circuit design [36]. Therefore, for the simplicity, in this work we make two assumptions: (i) Each WLAN module has the same wake-up latency, and (ii) A WUR has the same sensitivity as carrier sense threshold of a WLAN module where the receiver achieves a total power consumption of $12\mu W$ at -50dBm sensitivity and data rate of 250kbps [7].

The IoT devices are WUR-enabled and are operated in the transmitter-initiated mode. Collisions occur if the transmissions of more than one WUR device overlap with other WURs. Under such a scenario, consider a network cluster consisting of $N+1$ WUR devices including one cluster head (e.g., WUR AP) and N WUR member devices, as shown in Fig. 8. The N WUR devices compete with each other in an asynchronous mode for data reporting towards the WUR AP over a single hop. We assume there are M backoff stages with $CW_{min} = W$, $CW_{max} = 2^M W$, and $CW_i = 2^i W$ for the ith backoff stage [36]. In addition, according to [37], the wake-up latency is assumed to be $T_W = 200\mu s$ for time taken for a WUR module to wake up completely with $T_s = 9\mu s$.

Each WUR device has a finite queue capacity and is equipped with a WUR transceiver in addition to its PCR. Assume that, at each WUR device (except the WUR AP), packets are generated based on a Poisson process with an arrival rate of λ. In order to show WUP flooding that could take place by an attacker during one EDCA TXOP duration, entropy theory has been implemented in this paper. In other words, entropy H is defined as:

$$H = -\sum_{f=1}^{n} P_f \log_2 P_f, \tag{5}$$

P_f is the number of observed flooding attack events during the enhanced distributed channel access transmission opportunity (EDCA TXOP) period. After considering number of inflow WUPs to WUR device, it is decided whether the next step proceeds or not. If number of inflow WUPs is very small, it does not have a substantial effect on the WUR network. During this time period, collected time is called 'time window'. In time window, the number of flooded WUPs flowing in EDCA TXOP is measured. If number of collected WUPs during this time window is over volume threshold (T1), it is considered as a first attack warning and they are sent to the next detection step. If it is not over volume threshold (T1), the traffic is not considered as an attack. Entropy about destination IP address flowing in the EDCA TXOP is calculated during time window and then, it is inspected to over entropy threshold (T2) of destination IP address. If WUPs flowing in the EDCA TXOP are heading to a certain destination IP address, entropy decreases. If flooded WUPs in the EDCA TXOP are heading to different destination IP addresses, entropy increases. Therefore, if entropy of destination IP address is smaller than entropy threshold (T2) of destination IP address, it is decided that they are heading to a certain destination IP address and classified as an attack. The channel is considered to be error free, and no hidden device exists in this cluster.

B. SIMULATION RESULTS

1) AVERAGE PACKET DELAY

In Figure 10, a comparison of the average packet delay for successful packet transmission is shown. In the corresponding case, the data rate used was 100kbps and the number of WUR devices used was increased from 5 to up to 50. Further, the average packet arrival rate of lambda was defined as 10 packets/s and the WUP duration was set as $12ms$. Additionally, the P_f value of entropy was set as 0.05 to reflect the random attack environment. As for the CE-WUR, packets are transmitted once the channel is determined to be idle after the BO and CCA processes. Thus, the average packet delay significantly increases with a simple increase in the network size (i.e., increase in the number of WUR devices). In addition, the case was configured to generate

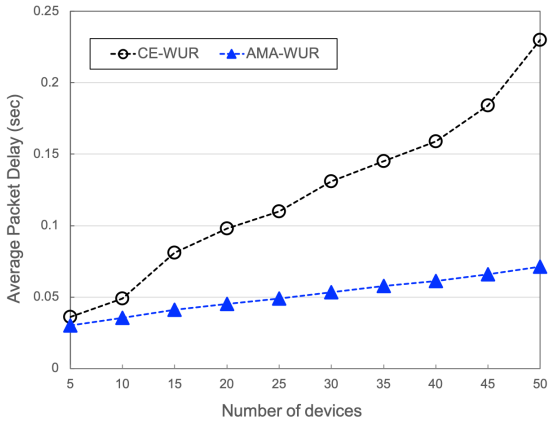


FIGURE 10. Average packet delay.

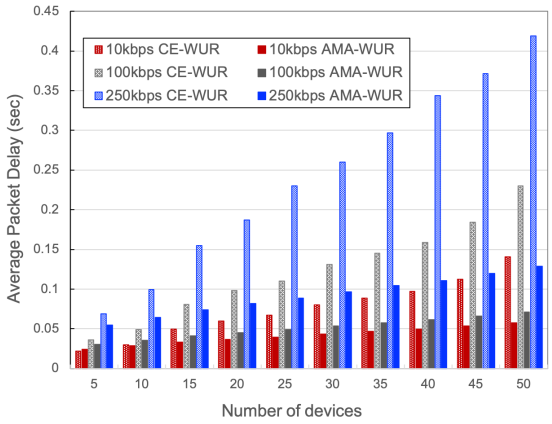


FIGURE 11. Average packet delay with different data rates.

one random attacker in each topology (based on the number of devices) environment (e.g., if the number of DEVs is 10, there are 10 WUR devices) based on the entropy probability. In other words, if an attack takes place in a topology having j number of DEVs, the number of normal devices is $j - 1$. When an attack occurs, the spoofed WUP is assumed to be transmitted based on the random average packet arrival traffic load of 100–1000 packets/s.

As shown in Figure 10, the arrival packet delay rapidly increases for CE-WUR when an attack occurs. This is due to the failed packet transmission from the spoofed WUR. On the other hand, although the proposed AMA-WUR tended to show an increase in the average packet delay with increase in the number of WUR devices, it showed resilience to the attack. This was achieved by having the WUR device, which received the spoofed WUR, immediately inform the AP through the WUP ACK and receive a new WUID so that the communication can be performed using the newly received WUID without having to wake up from the attack. Considering a topology with 30 devices as an example, the proposed AMA-WUR showed a reduction of 59.24% in average packet delay compared to CE-WUR.

Figure 11 shows a comparison of average packet delay on successful packet transmission based on three different data rates. The data rates used in this simulation were 10kbps, 100kbps, and 250kbps, and the number of WUR devices was increased from 5 to 50 devices. In addition, the average packet arrival rate of λ was set as 10 packets/s, and the WUP duration was set as $6ms$. As with the previous simulation (Figure 10), the simulation environment featured random attacks, and the P_f value was set as 0.05. As a result, both CE-WUR and the proposed AMA-WUR showed an increase in the average packet delay with increase in the number of devices. In particular, CE-WUR showed a rapid increase in packet delay as the topology became more complex. On the contrary, although the proposed AMA-WUR showed an increase in the average packet delay with increasing number of WUR devices, it showed resilience to the attack. Unlike earlier (Figure 10), the WUP duration was reduced by half and set as $6ms$ for this simulation, and both techniques did not appear

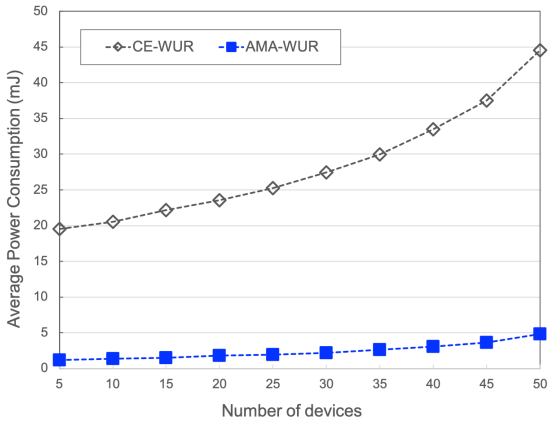


FIGURE 12. Average power consumption.

to have a significant impact on packet delays when compared to the WUP duration of $12ms$. This is because the WUP duration is included in the data rate that can be processed. Considering a topology composed of 30 devices as an example, when the data rate is 10kbps, the AMA-WUR shows a reduction of 46.25% in average packet delay compared to CE-WUR. Further, considering the same topology composed of 30 devices, when the data rate is 250kbps, AMA-WUR shows a reduction of 62.84% in average packet delay compared to CE-WUR. The results show a performance increase in AMA-WUR with increase in data rate when compared to CE-WUR. This is because data throughput also increased in AMA-WUR despite the attacks, whereas packet delay rapidly increased in the case of CE-WUR due to attacks.

2) AVERAGE POWER CONSUMPTION

Figure 12 shows a comparison of average power consumption under the same conditions as for the simulation corresponding to Figure 10. Considering a topology composed of 5 devices as an example, the proposed AMA-WUR shows a reduction of 94.08% average power consumption compared to CE-WUR. This is because AMA-WUR does not require additional power when the proposed DoSL attack packet

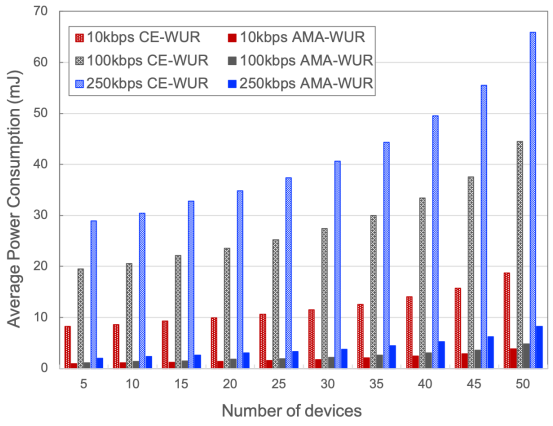


FIGURE 13. Average power consumption with different data rates.

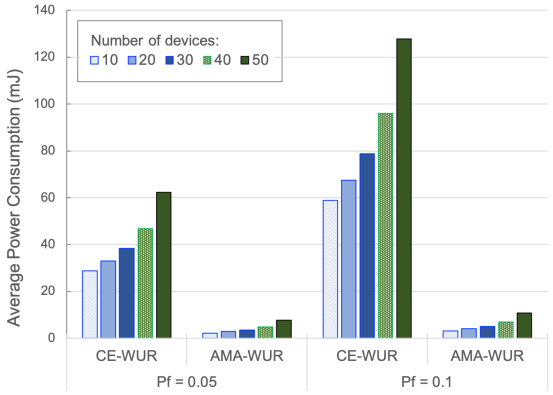


FIGURE 14. Average power consumption with different attack rates.

occurs except for the power consumed for the thresholds. In other words, if a DoSL attack takes place and is observed as an attack, power consumption does not increase as the main radio is not awaken and the attack is ignored internally. On the other hand, in the case of CE-WUR, if a DoSL attack occurs within the topology, WUR does not enter sleep mode; rather, it constantly receives packets, resulting in a drastic increase in power consumption.

As the same applies to the case where the data rate increases, as observed in Figure 13, the average power consumption shows a similar tendency to that of Figure 12 when the data rate increases. Considering a topology composed of 30 devices with a data rate of 250kbps as an example, the AMA-WUR achieves a reduction of 90.69% in average power consumption compared to CE-WUR.

In order to consider an environment having random attacks, the impact on the average power consumption according to the P_f value of entropy is examined. Here, two cases of P_f value as 0.05 and 0.1 are examined. In addition, the data rate is set as 250kbps and the average packet arrival rate of λ in a properly operating topology is set as 10 packets/s. When an attack occurs, it is assumed that the spoofed WUP is transmitted at a random average packet arrival traffic load of 100 to 1000 packets/s. Figure 14 shows the average power consumption for different number of WUR devices. The results shown on the left side of the graph are with P_f value of 0.05 and the results on the right side are with P_f value of 0.1. As shown in Figure 14, power consumption of CE-WUR drastically increased with increase in the number of random attacks. For example, in the case of 30 WUR devices, CE-WUR showed power consumption increase of 105.05% when P_f is 0.1 versus when it is 0.05. On the other hand, in the same case of 30 WUR devices, AMA-WUR shows a power consumption increase of only 41.83% when P_f is 0.1 compared to when P_f is 0.05. Additionally, when comparing power consumption between CE-WUR and AMA-WUR in the case of 30 WUR devices, the proposed AMA-WUR shows 90.90% lower power consumption than CE-WUR when P_f is 0.05 and 93.71% lower power consumption when P_f is 0.1. This is because AMA-WUR, unlike CE-WUR, does not switch to awake state (i.e., not waking up the main radio) when a spoofed attack occurs and discards the corresponding packet. In addition, AMA-WUR directly requests the AP for a new WUID; thus, it is not exposed to any additional attacks.

VI. CONCLUSION

Small-sized IoT sensors that operate using limited energy are key technologies for green communication. In this paper, the vulnerabilities of WUR devices to malicious attacks such as the DoSL attack were discussed. Then, an AMA-WUR protocol to solve such an issue was proposed. Further, in addition to the case of DoSL attacks, a method of waking up the PCR after checking the WUR AP rather than immediately waking it for the attack cases of spoofing attacks was proposed. As the PCR is not woken up immediately, the average packet delay and average energy consumption both showed higher values when compared with traditional CE-WUR; however, when the average packet delay and average energy consumption results were compared for the case of actual attack occurrence, the AMA-WUR showed superior results. The extensive simulation results show that the proposed anti-malicious attack WUR (AMA-WUR) protocol reduces the average packet delay by 62.84%, and reduces the average power consumption by 93.71% while the flooding attack vulnerabilities. In the case of an extreme attack, the packet delay increased drastically in the traditional CE-WUR to cause failure of packet transmission, and the energy consumption also increased drastically to cause the corresponding WUR device to become a dead device. Thus, security vulnerabilities should be considered when designing protocols of the WUR devices constructed for green communication.

REFERENCES

[1] X. Zhang, M. Zhang, F. Meng, Y. Qiao, S. Xu, and S. Hour, "A low-power wide-area network information monitoring system by combining NB-IoT and LoRa," *IEEE Internet Things J.*, vol. 6, no. 1, pp. 590–598, Feb. 2019.

[2] B. Jolly, "The last thing IoT device engineers think about: End of battery life behavior for IoT devices," in *Proc. IEEE 62nd Int. Midwest Symp. Circuits Syst. (MWSCAS)*, Dallas, TX, USA, Aug. 2019, pp. 837–840.

[3] S. Misra, S. K. Roy, A. Roy, M. S. Obaidat, and A. Jha, "MEGAN: Multipurpose energy-efficient, adaptable, and low-cost wireless sensor node for the Internet of Things," *IEEE Syst. J.*, vol. 14, no. 1, pp. 144–151, Mar. 2020.

[4] N. K. Giang, J. Im, D. Kim, M. Jung, and W. Kastner, "Integrating the EPCIS and building automation system into the Internet of Things: A lightweight and interoperable approach," *J. Wireless Mobile Netw., Ubiquitous Comput., Dependable Appl.*, vol. 6, no. 1, pp. 56–73, Mar. 2015.

[5] C.-S. Shih, J.-J. Chou, and K.-J. Lin, "WuKong: Secure run-time environment and data-driven IoT applications for smart cities and smart buildings," *J. Internet Services Inf. Secur.*, vol. 8, no. 2, pp. 1–17, May 2018.

[6] G. Kakamanshadi, S. Gupta, and S. Singh, "A new optimal relay nodes selection method for wireless sensor networks," in *Proc. 5th Conf. Knowl. Based Eng. Innov. (KBEI)*, Tehran, Iran, Feb. 2019, pp. 787–793.

[7] H. Bello, Z. Xiaoping, R. Nordin, and J. Xin, "Advances and opportunities in passive wake-up radios with wireless energy harvesting for the Internet of Things applications," *Sensors*, vol. 19, no. 14, pp. 1–33, 2019.

[8] J. Kim, N.-O. Song, B.-J. Kwak, K. Kim, and J.-K.-K. Rhee, "Highly adaptive and scalable random access based on idle-time," in *Proc. IEEE Int. Conf. Commun. Workshops (ICC Workshops)*, Paris, France, May 2017, pp. 435–440.

[9] J. Tan, W. Liu, T. Wang, N. N. Xiong, H. Song, A. Liu, and Z. Zeng, "An adaptive collection scheme-based matrix completion for data gathering in energy-harvesting wireless sensor networks," *IEEE Access*, vol. 7, pp. 6703–6723, 2019.

[10] A. Alhalafi, L. Sboui, R. Naous, and B. Shihada, "GTBS: A green task-based sensing for energy efficient wireless sensor networks," in *Proc. IEEE Conf. Comput. Commun. Workshops (INFOCOM WKSHPS)*, San Francisco, CA, USA, Apr. 2016, pp. 136–143.

[11] P. Rawat, K. D. Singh, H. Chaouchi, and J. M. Bonnin, "Wireless sensor networks: A survey on recent developments and potential synergies," *J. Supercomput.*, vol. 68, no. 1, pp. 1–48, Apr. 2014.

[12] R. C. Carrano, D. Passos, L. C. S. Magalhaes, and C. V. N. Albuquerque, "Survey and taxonomy of duty cycling mechanisms in wireless sensor networks," *IEEE Commun. Surveys Tuts.*, vol. 16, no. 1, pp. 181–194, 1st Quart., 2014.

[13] B. Shrestha, E. Hossain, and S. Camorlinga, "Hidden node collision mitigated CSMA/CA-based multihop wireless sensor networks," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Budapest, Hungary, Jun. 2013, pp. 1570–1575.

[14] O. B. Akan, O. Karli, and O. Ergul, "Cognitive radio sensor networks," *IEEE Netw.*, vol. 23, no. 4, pp. 34–40, Jul./Aug. 2009.

[15] H. Park and E.-J. Kim, "Wake-up radio-resilient scanning mechanism for mobile device in IEEE 802.11ba," *Sensors Mater.*, vol. 30, no. 12, pp. 2961–2968, May 2018.

[16] F. A. Aoudia, M. Gautier, M. Magno, O. Berder, and L. Benini, "Leveraging energy harvesting and wake-up receivers for long-term wireless sensor networks," *Sensors*, vol. 18, no. 5, pp. 1–27, 2018.

[17] F. Hutu, A. Khoumeri, G. Villemaud, and J.-M. Gorce, "A new wake-up radio architecture for wireless sensor networks," *EURASIP J. Wireless Commun. Netw.*, vol. 2014, no. 1, pp. 1–10, Dec. 2014.

[18] *IEEE Draft Standard for Information Technology—Telecommunications and Information Exchange Between Systems Local and Metropolitan Area Networks—Specific Requirements Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications Amendment: Wake-Up Radio Operation*, Standard IEEE P802.11ba/D3.0, May 2019.

[19] F. Z. Djiroun and D. Djenouri, "MAC protocols with wake-up radio for wireless sensor networks: A review," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 1, pp. 587–618, 1st Quart., 2017.

[20] W. Zhang, W. Liu, T. Wang, A. Liu, Z. Zeng, H. Song, and S. Zhang, "Adaption resizing communication buffer to maximize lifetime and reduce delay for WVSNs," *IEEE Access*, vol. 7, pp. 48266–48287, 2019.

[21] L. Guntupalli, D. Ghose, F. Y. Li, and M. Gidlund, "Energy efficient consecutive packet transmissions in receiver-initiated wake-up radio enabled WSNs," *IEEE Sensors J.*, vol. 18, no. 11, pp. 4733–4745, Jun. 2018.

[22] S. Naik and N. Shekokar, "Conservation of energy in wireless sensor network by preventing denial of sleep attack," *Procedia Comput. Sci.*, vol. 45, pp. 370–379, Jan. 2015.

[23] D. Spenza, M. Magno, S. Basagni, L. Benini, M. Paoli, and C. Petrioli, "Beyond duty cycling: Wake-up radio with selective awakenings for long-lived wireless sensing systems," in *Proc. IEEE Conf. Comput. Commun. (INFOCOM)*, Apr. 2015, pp. 522–530.

[24] D. R. Raymond, R. C. Marchany, M. I. Brownfield, and S. F. Midkiff, "Effects of denial-of-sleep attacks on wireless sensor network MAC protocols," *IEEE Trans. Veh. Technol.*, vol. 58, no. 1, pp. 367–380, Jan. 2009.

[25] C. Gritti, M. Önen, R. Molva, W. Susilo, and T. Plantard, "Device identification and personal data attestation in networks," *J. Wireless Mobile Netw., Ubiquitous Comput., Dependable Appl.*, vol. 9, no. 4, pp. 1–25, Dec. 2018.

[26] R. Falk and H.-J. Hof, "Fighting insomnia: A secure wake-up scheme for wireless sensor networks," in *Proc. 3rd Int. Conf. Emerg. Secur. Inf., Syst. Technol.*, Athens, Greece, 2009, pp. 191–196.

[27] S. Hu, D. Yue, X. Xie, X. Chen, and X. Yin, "Resilient event-triggered controller synthesis of networked control systems under periodic DoS jamming attacks," *IEEE Trans. Cybern.*, vol. 49, no. 12, pp. 4271–4281, Dec. 2019.

[28] A. T. Capossele, V. Cervo, C. Petrioli, and D. Spenza, "Counteracting denial-of-sleep attacks in wake-up-radio-based sensing systems," in *Proc. 13th Annu. IEEE Int. Conf. Sens., Commun., Netw. (SECON)*, London, U.K., Jun. 2016, pp. 1–9.

[29] C. Atapour, I. Agrafiotis, and S. Creese, "Modeling advanced persistent threats to enhance anomaly detection techniques," *J. Wireless Mobile Netw., Ubiquitous Comput., Dependable Appl.*, vol. 9, no. 4, pp. 71–102, Dec. 2018.

[30] H. Park, "Adaptive backoff enabled WUR on non-cellular local IoT for extreme low power operation," *Future Gener. Comput. Syst.*, vol. 108, pp. 62–67, Jul. 2020.

[31] G. U. Gamm, M. Sippel, M. Kostic, and L. M. Reindl, "Low power wake-up receiver for wireless sensor nodes," in *Proc. 6th Int. Conf. Intell. Sensors, Sensor Netw. Inf. Process.*, Brisbane, QLD, Australia, Dec. 2010, pp. 121–126.

[32] G. U. Gamm and L. M. Reindl, "Range extension for wireless wake-up receivers," in *Proc. Int. Multi-Conf. Syst., Sygnals Devices*, Chemnitz, Germany, Mar. 2012, pp. 1–4.

[33] E. Rozner, V. Navda, R. Ramjee, and S. Rayanchu, "NAPman: Network-assisted power management for WiFi devices," in *Proc. Int. Conf. Mobile Syst., Appl., Services (MobiSys)*. New York, NY, USA: Association for Computing Machinery, 2010, pp. 91–106.

[34] J. Oller, I. Demirkol, J. Casademont, J. Paradells, G. U. Gamm, and L. Reindl, "Has time come to switch from duty-cycled MAC protocols to wake-up radio for wireless sensor networks?" *IEEE/ACM Trans. Netw.*, vol. 24, no. 2, pp. 674–687, Apr. 2016.

[35] J. Oller, I. Demirkol, J. Casademont, J. Paradells, G. Gamm, and L. Reindl, "Performance evaluation and comparative analysis of SubCarrier modulation wake-up radio systems for energy-efficient wireless sensor networks," *Sensors*, vol. 14, no. 1, pp. 22–51, Dec. 2013.

[36] S. Tang and S. Obana, "Reducing false wake-up in contention-based wake-up control of wireless LANs," *Wireless Netw.*, vol. 25, no. 5, pp. 2333–2349, Jul. 2019.

[37] X. Zhang and K. G. Shin, "E-MiLi: Energy-minimizing idle listening in wireless networks," *IEEE Trans. Mobile Comput.*, vol. 11, no. 9, pp. 1441–1454, Sep. 2012.

HYUNHEE PARK (Member, IEEE) received the Ph.D. degree from the School of Electronics and Computer Engineering, Korea University, South Korea, in August 2011. From September 2011 to February 2013, she was a Research Professor with the Information Technology Center, Korea University. She was also with the Mobile Networks and Communications Laboratory, Korea University. From January 2013 to November 2014, she was a Postdoctoral Researcher with the INRIA Research Center, where she works in DIONYSOS Research Group. She was a Postdoctoral Researcher with Telecom Bretagne, where she undertakes the system implementation for multi-path TCP on wireless networks. From November 2014 to February 2017, she was a Senior Researcher with LG Electronics for Wi-Fi standardization (IEEE 802.11ax, Wake Up Radio, Wi-Fi Alliance, and so on). From March 2017 to February 2020, she was an Assistant Professor with the Department of Computer Software, Korean Bible University. Since 2020, she has been an Assistant Professor with the Department of Information and Communication Engineering, Myongji University, South Korea. She is currently a Supervisor of Data Analysis and Networking (DAN) Laboratory. Her research interests include wireless communications and networks, QoE, and resource management.