IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received January 18, 2020, accepted February 2, 2020, date of publication February 19, 2020, date of current version March 2, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2975016

Fusion-On-Field Security and Privacy Preservation for IoT Edge Devices: Concurrent Defense Against Multiple Types of Hardware Trojan Attacks

HAWZHIN MOHAMMED[1], SYED RAFAY HASAN[1], AND FALAH AWWAD[2]
[1]Department of Electrical and Computer Engineering, Tennessee Tech University, Cookeville, TN 38505, USA
[2]Department of Electrical Engineering, College of Engineering, United Arab Emirates University, Al Ain 15551, United Arab Emirates

Corresponding authors: Hawzhin Mohammed (hmohammed42@students.tntech.edu); Syed Rafay Hasan (shasan@tntech.edu); Falah Awwad (f_awwad@uaeu.ac.ae)

This work was supported by Information and Communication Technologies (ICT) Fund UAE, fund No. 21N206 at UAE University, Al Ain, United Arab Emirates (UAE).

ABSTRACT Internet of Things (IoT) devices have connected millions of houses around the globe via the internet. In the recent past, threats due to hardware Trojan (HT) in the integrated circuits (IC) have become a serious concern, which affects IoT edge devices (IoT-ED). In this paper, the possibility of the IoT-ED with embedded HT that can cause serious security, privacy, and availability problems to the IoT based Home Area Network (HAN) has been discussed. Conventional network attack detection techniques work at the network protocol layers, whereas IoT-ED with HT can lead to the peculiar manifestation of attack at the physical and/or firmware level. On the other hand, in the IC design, most of the HT-based attack detection techniques require design time intervention, which is expensive for many of the IoT-ED and cannot guarantee 100% immunity. The argument in this paper is that the health of modern IoT-ED requires a final line of defense against possible HT-based attacks that goes undetected during IC design and test. The approach is to utilize power profiling (PP) and network traffic (NT) data without intervening into the IC design to detect malicious activity in HAN. The proposed technique is to effectively identify multiple attacks concurrently and to differentiate between different types of attacks. The IoT-ED behaviors for five different types of random attacks have been studied, including covert channel, DoS, ARQ, power depletion, and impersonation attacks. Data fusion has been leveraged by combining the PP and NT data and is able to detect, without design time intervention, each of the five attacks individually with up to 99% accuracy. Moreover, the proposed technique can also detect all the attacks concurrently with 92% accuracy. To the best of authors' knowledge, this is the first work where multiple HT based attacks are concurrently detected in IoT-ED without requiring any design time intervention.

INDEX TERMS Internet of Things, hardware security, home area network, hardware Trojan, machine learning, power profile, ARQ attack, DoS attack.

I. INTRODUCTION

Internet of Things (IoT) is fundamentally a collection of smart devices inserted with remote correspondence capacity of wireless connection [9], [17]. They are utilized as a part of different applications in everyday life. IoT devices are prevalent in the smart city, smart grid, home area network (HAN), advanced manufacturing, health monitoring, and many other modern applications. The mobility report from Ericsson Inc. stated that the number of IoT devices are expected to grow by 27 percent annually and will reach 4.1 billion devices by 2024 [1]. On the other hand, total spending on IoT devices and administrations has come to nearly $800 billion in 2017 and is expected to reach $1.4 trillion by 2021 [2].

The United State Federal Trade Commission's report, in [3] and [4], states that an IoT device is exposed to a variety of possible safety risks. They can be used by adversaries to infiltrate into the IoT devices at the hardware, software or communication interface level. Although these risks exist with traditional computer networks, in [4] it has alluded

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

that with poor security in IoT devices these threats can be more catastrophic in modern IoT based systems. This fact is further corroborated with the reports published by Gartner Inc., which states that 1) the market for information security reached $114 billion in 2018, and will grow by 8.7% in 2019 to reach $124 billion [5], 2) worldwide IoT security spending expected to reach $3.1 billion in 2021 [6].

With the advent of the possibility of hardware intrinsic attacks, such as the Hardware Trojan (HT) attack [7], the absence of concrete measures of security in IoT devices has turned into a major concern to the academic and industrial research communities [7]–[11]. As all the IoT devices contain an Integrated Circuit (IC) chip, the HT-based attacks' vulnerabilities are universal to all IoT devices [12], [13]. This can harm the IoT-based network's performance, security, privacy, accessibility, and may even provide backdoor access leading to information leakage [14]–[17], [57].

One of the major consumers of IoT devices is the Home Area Network (HAN). In modern smart grids [18], IoT-based HAN are connected via smart meters. Hence, the security of IoT-based HAN is essential for overall well-being of the smart grid. Several researchers have studied the security challenges in IoT-based HAN. Jokar *et al.* in [19] proposed a model that uses intrusion detection and prevention systems for ZigBee-based HAN. This model employs a dynamic machine learning-based prevention technique. In [20], Faisal *et al.* proposed a technique for the comprehensive Intrusion Detection System (IDS) in Advanced Metering Infrastructure (AMI), which is designed to be reliable, dynamic and accommodates the realtime nature of traffic for each component in AMI. The authors in [20] have observed that some algorithms that use a very minimal amount of computing resources and offer a moderate level of accuracy can potentially be used for the smart meter IDS. However, these models can detect only attacks made at the network layer level. The network layer detection mechanism needs access to the content of the messages, to detect attacks, which raises privacy preservation issue. The HT-based attack can lead to a different characterization of communication traffic, which may not be identified by such traditional IDS used in [19] and [20].

The above observations made researchers believe that, since HT-based attacks cause unique characterization to communication traffic, HT-based attacks need to be further investigated. For example, a power depletion attack for IoT-based HAN networks, which is caused by HT-based attacks, mostly affects the physical parameters. More specifically, the attacker in a power depletion attack intends to shut down the IoT device and disconnect it from the HAN network or sabotage the HAN's topology in case of a multi-hop or tree network. Similarly, HT-based attacks can lead to impersonation attacks under a legitimate IoT device's identity (ID), hence these types of attack cannot be detected by traditional IDS and firewall monitoring systems. Furthermore, in HT-based covert channel attacks, the attacker IoT device may send legitimate data, simultaneously or staggered, through a legitimate and a covert channel interface. The covert channel interface may or may not be visible to traditional IDS or firewall monitoring systems [14]–[16]. Along the same lines, the HT-based attacks may also lead to a Denial-of-Service (DoS) attack, and a new type of attack, which has been named Automatic Repeat reQuest (ARQ) attack. In the ARQ attack the infected device with this HT-based attack ignores the acknowledgment signal, thus causing multiple requests from the sender. Although such attacks are detectable using traditional IDS and firewall monitoring systems [21], they cannot distinguish between ARQ and DoS attacks. This distinction between ARQ and DoS attacks is important to reduce false positives.

To fill the gap of HT-based attacks detection and overcome the unique characterization of HT-based attacks on IoT-based networks, researchers secure the IoT edge device (IoT-ED) through the utilization of design time intervention techniques. In these techniques, they protect the IC chips from HT-based attacks by modifying the layout at design time, which protects the IC chip from HT-based attacks but not the whole IoT device [8], [9], [14]. In some other works, researchers utilized a network protocol level detection mechanism to secure the network without taking hardware problem into perspective [19], [20]. On the other hand, most of the works in literature address one attack or a couple of attacks at a time without taking into consideration the variety of attacks on the IoT-based network. To overcome the above limitations, this paper, ''FusIon: On-Field Security and Privacy Preservation for IoT Edge Devices; Concurrent Defense Against Multiple types of Hardware Trojan Attacks'', proposes a novel attack detection system to concurrently detect the above mentioned attacks due to HTs. The effect of HT-based attacks inside the IoT device on the security of the HAN network have been studied. The study includes how the behavior changes, due to HT-based attacks, of some of the well-known attacks on the HAN network, namely, a DoS attack, a covert channel attack, an impersonation attack, a power depletion attack, and an ARQ attack. The network traffic (NT) data and power profile (PP) data have been utilized without intervening into the IoT devices.

The data fusion technique has been exploited to increase detection accuracy and decrease false positives. The new set of fused data has been provided to the machine learning model for classification purposes. The proposed data fusion based techniques against different attack scenarios have been verified experimentally. The results show that the proposed data fusion techniques can successfully detect HT-based attacks on IoT edge devices if they fall within the above attack categories. In order to emulate the real-world attack scenarios, implementation of random attacks following Gaussian distribution has been conducted. The frequency, duration, and the instance of attacks all are randomly distributed. The main contribution of this paper is as follows:

- For the first time randomness created due to data fusion has been exploited to detect multiple HT-based attacks on HAN. The proposed methodology can detect an

impersonation attack even if the attacker uses a legitimate ID of legitimate IoT edge devices and it can differentiate between an ARQ attack and a DoS attack.
- HT-based attacks are investigated on IoT edge devices in the HAN network using NT and PP data-sets. An on-filed, run-time approach is proposed without requiring any design time intervention to detect hardware attacks. The proposed methodology does not require accessing or destroying the IC chips in the IoT edge device as well.
- The proposed approach follows the privacy-preservation based methodology that detects attacks on HAN without accessing the contents of the messages.

The remainder of this paper is organized as follows: The related work is provided in Section II. The background is provided in Section III. Section IV describes the details of the studied network, threat model and attack scenarios. Section V describes the randomization of hardware attacks. On-field IoT-ED security and privacy preservation defense against hardware Trojan is in Section VI. The experimental setup and the results are given in Section VII. Conclusions are drawn in Section VIII.

II. RELATED WORK

IoT security is generally addressed in the literature from either the network layer level (the assumption is cloud-based security falls within the network level) or the physical layer level perspective [47]. Researchers have utilized various different techniques to detect attacks in IoT-EDs at each level. With the enormous success of machine learning, many studies have been able to successfully apply machine learning at the network level to detect attacks in IoT-EDs. Yang *et al.* in [48], investigate the active learning method for intrusion detection of wireless IoT device networks. It is seen that the active learning method can efficiently improve the performance over the traditional supervised learning methods for intrusion detection. But this work does not address hardware-level attacks or breach of firmware in IoT-EDs. Firmware breach in IoT-EDs is studied by Ling *et al.* in [17]. The authors show that attacks can be launched successfully by exploiting the insecure communication protocol of a known brand of a smart plug system. Though Ling *et al.* provide possible attack scenarios, their work does not provide any defense to such attack mechanism. The advancement in the threat of physical layer level security in the recent past warranted many researchers to look into possible solutions for these types of attacks as discussed below.

Many attacks stem out from the hardware level but manifest at the network layer level. Xiao *et al.* in [49] have recognized such IoT-ED attack models and learning-based security methods. This involves IoT-ED authentication, access control, malware detection, and secure offloading, which are proved to be assuring protection for the IoT-EDs against attacks that are sowed at the hardware level and manifest in the network layer. In the same direction, Thangavelu *et al.* in [50] introduce a distributed device fingerprinting technique (DEFT), where fingerprinting is generated by the unique pattern of network packets corresponding to each IoT-ED. The network controller and gateways in DEFT coordinate to recognize whether the incoming hardware device is a legitimate one or not.

Another type of attack, power depletion attack, has been addressed in [46] and [51]. Power depletion attacks in IoT have been surveyed in [46] by Lopez *et al.* They present their three-layered approaches to hardware attacks, namely: 1) physical layer, 2) battery management system (BMS) layer, and 3) application layer. Their unique contribution is related to analyzing vulnerabilities in BMS that can lead to a power depletion attack on IoT-ED. They presume that a power depletion attack results in a complete dysfunctional device, but they did not study its possible implication at the network layer level. Wei *et al.* in [51] report a machine learning-based method to detect malicious mobile malware in Android applications. They implemented a malicious application detection tool, named Androidetect, which implicitly saves the IoT-ED from depleting power.

A covert channel attack or data leakage attack has been addressed in [14], [15], and [22]. Cañedo *et al.* in [22] study utilizing Artificial Neural Networks (ANN) in a gateway to detect anomalies in the data transmitted from the IoT-EDs. They use the ANN to determine the healthy state of a system and connected devices. They use IoT-ED ID, sensor value, and a timestamp of transmitted data as input features to the ANN. By using these features, they prevent some attacks like the man-in-the-middle attack. In [14], Subramani *et al.* introduce a hardware Trojan attack which utilizes the error-correcting capabilities of the Forward Error Correction (FEC) encoder in the baseband circuits of an 802.11a/g transmitter. They argue that FEC provides protection against a noisy channel more than it requires. In a Trojan free communication channel, the noise distribution is centered around zero. Systematic inconsistencies in the channel noise distribution can be caused by HT. The detection scheme monitors the channel noise distribution and variations in the noise distribution that can identify HT at the sender end. In [15], Caviglione *et al.* introduce two machine learning methods using a neural network and decision tree algorithm to detect the appearance of malware exploiting power consumption. They use the power consumption value from a high-level application and file system as an input feature to the machine learning algorithms. The first method utilizes regression to predict the expected behavior of power consumption. The second method utilizes classification to provide information on covert channel communication using IoT-ED power consumption. They claim to detect seven types of covert channel attacks in Android devices. Their solution requires access to the firmware.

A combination of attacks, denial-of-service attack and covert channel attack, have been addressed in [8] and [9]. In [8], Dofe *et al.* introduce a dynamic permutation method to address both hardware Trojan and side-channel analysis attacks. The utilized permutation method decreases the probability of launching hardware attacks successfully in IoT-ED.

The dynamic permutation method prevents the hardwares attack and changes the power consumption (profile) over time. They do not need network-level protection or run-time detection. Liu *et al.* in [9] introduced a framework to monitor data communications in the IoT-EDs. They utilize different vendors for IoT-EDs to build a distributed framework where each IoT-ED monitors the trustworthiness of their neighbors. Any effort to leak confidential information or to collude with each other that lead to catastrophic failures can be detected. This distributed monitoring scheme creates a trustworthy communication channel between untrustworthy IoT-EDs. In both cases, the solution requires design time intervention.

From the literature review, so far researchers have focused on one attack at a time. In a few cases when the attacks are similar, a research work addresses two types of attacks (e.g. covert channel attack and DoS attack in [8] and [9]). Hence, there is a growing need for a solution or framework that can cater to multiple types of attacks simultaneously. So far whenever HT based attack are discussed in IoT-EDs such as [8], [9], and [14], they require design time intervention to be able to get detected. In some other cases, like [15] and [17], they require access to the firmware. Also, the existing literature related to hardware Trojan attacks in an IoT system does not discuss the possibility of them affecting network traffic. This is the second problem that has been addressed in this work by proposing a novel, on-field, run-time HT based attack detection in IoT-ED.

III. BACKGROUND

In this section, an overview of some of the concepts that help in understanding the rest of the paper has been provided.

A. IOT DEVICE CATEGORIES

IoT devices in the IoT system can be divided into two categories: 1) IoT Edge Device (IoT-ED) and (2) IoT Gateway Device (IoT-GD) [22]. An IoT-ED is typically equipped with low-computation capability, low-communication bandwidth, and low-power budget. IoT-ED usually has a unique purpose, for instance, collecting sensed data and reporting it to the IoT-GD. IoT-GD is equipped with more resources than IoT-EDs. It is responsible for collecting the sensed data from the IoT-EDs and connects them to the outside world [22].

Many of these smart devices are becoming part of modern domestic daily life as part of the HAN. These smart devices in HAN are IoT-EDs, which play an important role in the HAN network. Fig. 1 illustrates a conceptual HAN network, equipped with a smart meter (SM) or IoT-GD and multiple IoT-EDs inside SHAs. Through interfacing with the IoT-GD, IoT-ED sends its power consumption reading to the IoT-GD periodically after every short period of time. The power consumption readings of all IoT-EDs are reported to the utility company after being collected by IoT-GD [23]–[25]. A household can interface with IoT-ED through the internet for device status checking, task scheduling, task execution, and task termination.

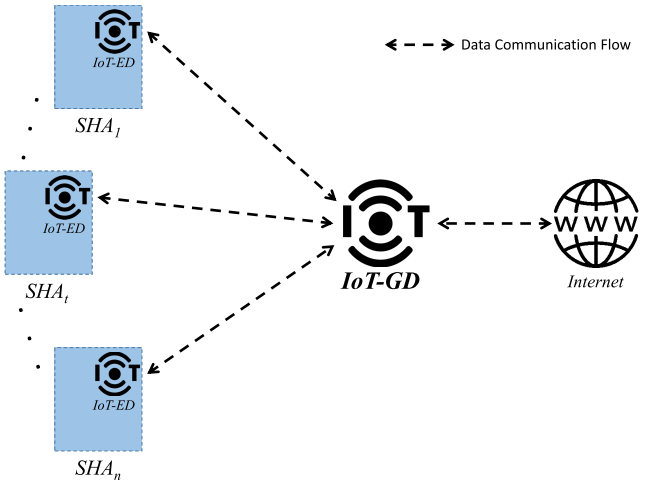


FIGURE 1. Home Area Network (HAN), where, IoT-GD (Smart Meter) is connected through a wireless connection to IoT-ED inside the Smart Home Appliances. IoT-GD also connected to the Utility company through the Internet.

B. HT DETECTION, DESIGN TIME INSERTIONS IN THE CIRCUIT

The IC design and fabrication process has become more vulnerable to hardware attacks or malicious insertions of hardware units, called hardware Trojans (HT) [26]–[29]. An HT is a malicious piece of hardware circuitry that is embedded into the IC of IoT-ED and is characterized by its physical representation and behavior [27], [29]. In IC design, due to globalization and usage of third party intellectual properties (3PIP), IC can be tampered with at any stage of the design and fabrication process. The modern embedded system contains several ICs as its major components. HTs embedded in ICs are divided into two types based on their activation techniques: 1) always-on HT; it can affect the IC operation at any time; 2) triggered HT; this Trojan can affect the IC operation on the event it is triggered [27], [29]. The trigger for the HT can be internal or external. An internally triggered HT is activated by an event that occurs within the targeted IC. The internal event may be a timer or a combination of binary inputs. An externally triggered HT requires external input to the target IC to activate the HT. The external event may be introduced by keyboard input, a keyword, a push-button, a switch or maybe a combination of events. An HT is composed of two parts: 1) payload; and 2) trigger. The payload is responsible for the malicious activity and trigger is responsible for activating the payload [27], [29].

Detection can be performed during test time or run-time. In [8] and [14], the authors protected the IC chips of IoT-ED from a HT-based attack by inserting a detection mechanism during design time. This prevents the IoT-ED from a hardware attack or helps with HT detection in run-time. But, this extra hardware comes with extra effort, cost, design time, and time to market, which results in the added cost of IoT-ED and delay to the market.

C. HT DETECTION DURING RUN-TIME WITHOUT INSERTING ANY DESIGN TIME DETECTION MECHANISM

Using IoT-ED in a network without any design-time protection increases the chances of a successful HT-based attack without detection. In this paper, the run-time detection has been presumed that it does not need an extra circuit to be inserted into the IoT-ED IC chips. This decreases the effort, cost, design time, and time to market. Hence, this claim is more in line with industrial practices.

D. DATA FUSION

Data fusion is a technique utilized by the data scientist to fuse multiple data-sets in order to obtain more reliable information [52], [53]. This helps in the classification problem of machine learning, where some data-set leads to weak classification. Merging different weak classifying data-sets leads to a data-set with strong classification. The fused data-set is more informative than the original input data-sets [30], [31].

There are two types of data fusion, namely: 1) centralized data fusion, and 2) decentralized data fusion. In centralized data fusion, the IoT-ED sends the data to a coordinator. The coordinator is responsible for fusing the data. In decentralized data fusion, the IoT-ED takes complete responsibility for fusing the data.

In Central Limit theorem, when independent random variables are fused, the merged data-set leads to a normal distribution even if the original variables are not normally distributed. Let D_1 and D_2 denote two independent random data-sets with noise variances σ_1^2 and σ_2^2, respectively. To obtain data-set D_3 which is a combination (fusion) of D_1 and D_2, the Central Limit theorem has been applied;

$$D_3 = \sigma_3^2\left(\frac{D_1}{\sigma_1^2} + \frac{D_2}{\sigma_2^2}\right) \tag{1}$$

where:

$$\sigma_3^2 = \frac{1}{1/\sigma_1^2 + 1/\sigma_2^2} \tag{2}$$

is the variance of the fused data-set. The fused data-set is a linear combination of the two data-sets (D_1 and D_2) weighted by their respective noise variances (σ_1^2 and σ_2^2).

E. NETWORK TRAFFIC (NT)

The network traffic or message counting is the number of messages in a period of time. Each application requires a certain amount of messages. The network operator gets security benefit from proper network traffic analysis. A potential indication of an attack can be obtained from an unusual volume of network traffic. Message counting is required when the data communication needs to stay within a specific number of messages per period of time. This is the case in HAN when the utility company requires one power consumption reading per unit time [32].

TABLE 1. Possibilities of attacker and defender existence in HAN network.

Case	IoT-ED	Network Deployment	Network Operator
I	*Attacker*	*Honest*	*Defender*
II	*Attacker*	*Defender*	*Honest*
III	*Honest*	*Attacker*	*Defender*
IV	*Honest*	*Defender*	*Attacker*
V	*Defender*	*Attacker*	*Honest*
VI	*Defender*	*Honest*	*Attacker*

F. POWER PROFILING (PP)

IoT-ED consumes electric power for operation. Each type of operation has a different power consumption requirement. Differences in power consumption occur as the IoT-ED performs various operations. It is a metric used for side-channel analysis in which the power consumption of the IoT-EDs has been studied [33]–[35].

G. MACHINE LEARNING TOOL (WEKA)

WEKA framework/tool is a combination of machine learning algorithms for data mining jobs [36]. WEKA is used because it is an open-source data mining package that contains tools for data pre-processing, classification, regression, clustering, and visualization. It is also well-suited for developing new machine learning schemes.

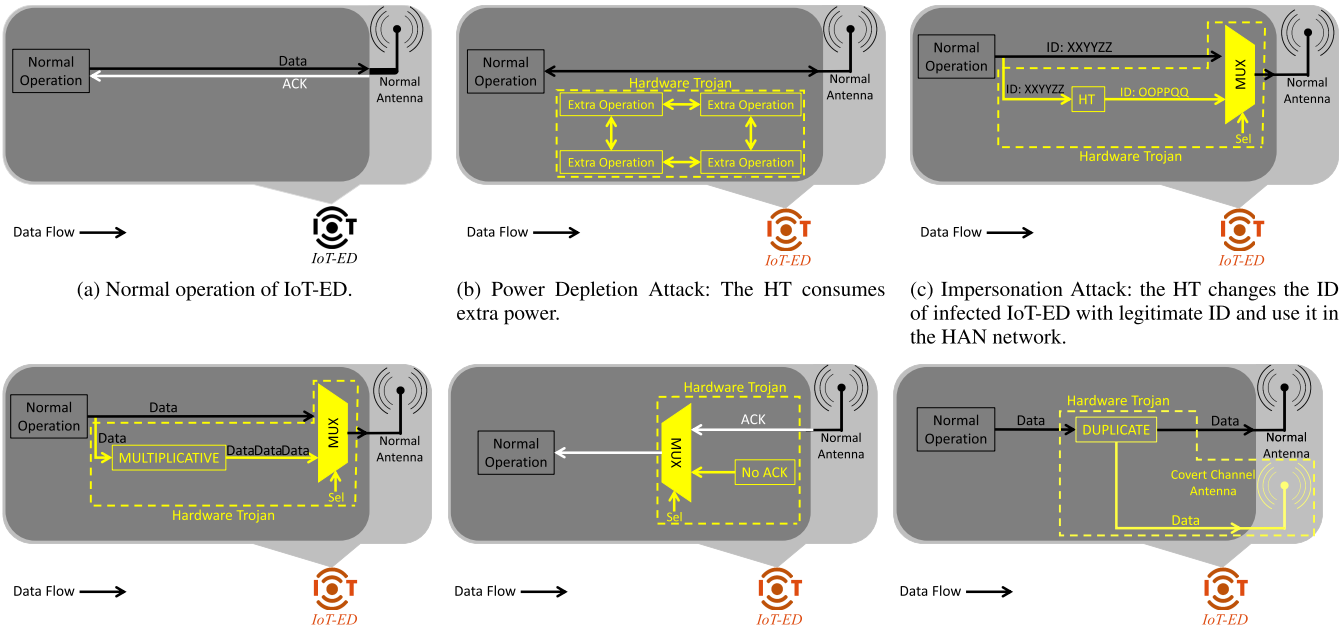
IV. THREAT MODEL AND ATTACK SCENARIOS

In this section, the network model, threat model and attack scenarios has been explained.

A. DETAILS OF THE STUDIED NETWORK AND THREAT MODEL

A threat model for any IoT-based HAN network can have three main players: 1) IoT-ED manufacturer, which includes the semiconductor supply chain, 2) the network deployment team, and 3) the network operation (monitoring) team. All of them can be assumed as one of three roles, they can be: 1) an attacker, 2) a defender or 3) an honest user (neither attacker nor defender). These cases have been illustrated in Table 1. The Threat Model I from Table 1 has been chosen with the following rationale. As the IoT-ED manufacturer and most of its components (i.e., IC manufacturer) are designed and fabricated offshore, the probability of them to be attacked or compromised is usually high. Therefore, the IoT-ED is assumed to be manufactured in unsecured environments and the attacker has access to the design cycle of ICs used in IoT-ED. Also, the network implementation is presumed to be performed under a controlled environment, hence they are honest. The network operator is the one who has to closely monitor the network as a last line of defense. They use the network for transferring data from one node to another. Here the network operator is the defender.

Home Area Network (HAN): consists of Smart Home Appliances that contain an IoT-EDs and a Smart Meter (SM) or IoT-GD (coordinator). The IoT-EDs are connected to the IoT-GD through wireless communication technology. The



(a) Normal operation of IoT-ED.

(b) Power Depletion Attack: The HT consumes extra power.

(c) Impersonation Attack: the HT changes the ID of infected IoT-ED with legitimate ID and use it in the HAN network.

(d) DoS Attack: the HT generates more data from the regular traffic data to be sent over the HAN network.

(e) HT-based ARQ Attack: the HT drops the ACK messages received from the sender.

(f) Covert Channel Attack: the HT generates extra copy of the data and send it through covert channel.

FIGURE 2. Normal case and Attacks type in IoT-ED on HAN network.

IoT-GD acts as the organizer among all the IoT-EDs in the HAN network. Also, it acts as an organizer between IoT-EDs and the utility company. IoT-GD is connected to the utility company through a high bandwidth connection that can be a wired or wireless connection. The HAN network topology is illustrated in Fig. 1. The IoT-GD has a higher computation, higher bandwidth, and higher power budget compared to IoT-EDs. This concept of the edge device and gateway device is not specific to the HAN network alone, so the proposed approach is scalable, flexible and can be applied to other IoT-based networks as well.

Threat Model: IoT-EDs consist of several IC chips. In this threat model the attacker can intrude the IC chips internally. The attacker can change the internal IoT-ED chips' structure. The attacker is assumed to be capable of stealthily adding, removing or manipulating the IC chips' internal design. If this manipulation is not detected by the IC supply chain testing then it can lead to attacks during run time. The attacker can design the HT in such a way that it can be triggered when the attacker needs or under a very rare condition. This behavior makes the HT hard to be detected in the testing phase. In this threat model, the defender (network operator) is assumed to have access to the NT and PP data of the IoT-EDs. Threat model I from Table 1 is in line with the scenarios that are mentioned in most of the recent IoT systems.

B. ATTACK SCENARIOS

In this section, the attack scenarios have been explained on the HAN that is studied in this paper.

1) NORMAL CASE

Under the normal case, the IoT-EDs report their power consumption (PP) data (black line in Fig. 2) every five minutes to the IoT-GD through a wireless connection. Then the IoT-ED receives an ACK message from the IoT-GD (white line in Fig. 2) to confirm the delivery of the message.

2) POWER DEPLETION ATTACK

A power depletion attack is an attack where HT consumes extra power of the IoT-ED with some stealthy circuit. Fig. 2b illustrates the power depletion attack on IoT-ED with HT inside the device. The HT that is surrounded by the dotted line results in producing extra computation. The IoT-ED has a limited power budget, therefore, this extra computation requirement increases the power consumption and may lead to the shutdown of the IoT-ED. This attack poses very serious concerns for IoT system security and availability. The malicious HT that resides inside the IoT-ED tries to deplete the IoT-ED's power and makes it unavailable for power consumption data reporting to the IoT-GD or replay messages in case of a multi-hub network topology. This attack is difficult to detect by the traditional IDS and firewall as the HTs deplete the IoT-ED's power within the IC and hence such attacks are not detectable through IDS or firewall.

3) IMPERSONATION ATTACK

An impersonation attack is when an HT infected device is disguised as a legitimate IoT-ED. This can be implemented by replacing the ID of the infected IoT-ED with a legitimate device ID. By learning the legitimate ID by the attacker,

HT can use the legitimate ID to inject false data into the HAN network which can harm the receiver device by causing a wrong decision, and consequently adversely affect the HAN network. This attack is difficult to be detected by the traditional IDS and firewall as the IoT-ED sends its data under the impersonate of a legitimate device ID which fools the IDS and firewall. The attacker uses the legitimate ID to report false power consumption data to the IoT-GD which wrongly affects the power decision at the coordinator or utility company side. Fig. 2c illustrates the impersonation attack on an IoT-ED, where the original IoT-ED ID is "XXYYZZ" but the HT inside the IoT-ED (surrounded with the dotted lines in Fig. 2c) changes it with a legitimate ID "OOPPQQ". In this case, the network has two IoT-EDs with the same ID that reports two different power consumption data to the IoT-GD. The number of device IDs stays the same in the network as there is no increment in the ID of the IoT-EDs but the amount of the network traffic reporting to the IoT-GD increases.

4) DOS ATTACK

A DoS attack is an attack where the attacker tries to make a machine or network resource unavailable to its expected consumer(s) by accidentally or inconclusively disturbing services. A DoS attack floods the targeted device with repetitive unnecessary requests trying to overburden structures and block some or all legitimate requests from being fulfilled. Fig. 2d illustrates the DoS attack on IoT-ED in the HAN network. The HT that is surrounded by the dotted line inside the IoT-ED multiplies the original data and sends repetitive data to the IoT-GD. The HT in the DoS attack starts to flood the HAN network with a large number of repeated messages. The compromised IoT-ED, instead of sending its power consumption data at a reasonable rate, sends the power consumption data more frequently.

5) ARQ ATTACK

The HT can maliciously affect the data transmission traffic protocol, such as an Automatic Repeat reQuest (ARQ) protocol. ARQ is an error-control system for data transmission. ARQ utilizes acknowledgment messages (ACK) and timeouts to achieve dependable data transmission over a problematic communication connection. If the sender IoT-ED does not get an acknowledgment before the timeout, then it continues to re-transmit the data until the sender gets an acknowledgment or exceeds a predefined number of re-transmissions which is called a timeout. Fig. 2e illustrates a possible HT-based ARQ attack on an IoT-ED. The HT in the IoT-ED (surrounded with the dotted lines in Fig. 2e) blocks the ACK message that has been sent back from the IoT-GD to the IoT-ED in response to the reported data. This forces the IoT-ED to re-transmit the data. Due to HT insertion, the IoT-ED may pass No ACK signal instead of an ACK signal, hence the malicious device sends multiple messages of the same type. Due to the random nature of HT trigger, this results in the IoT-ED to send a random number of the repeated power consumption data to the IoT-GD in every reporting cycle. Finally, the attacker IoT-ED accepts the ACK message at random then stop sending any more data and waits for the next reporting cycle. An ARQ attack is uniquely attributed to the HT-based ICs. To the best of the authors' knowledge, such attacks have not been addressed in the literature other than a few handful of researchers [35], [37]. Although the traditional IDS and firewall can detect the ARQ and DoS attacks, they cannot differentiate between them.

6) COVERT CHANNEL ATTACK

A covert channel attack or data leakage attack is a kind of attack that generates an ability to transfer data between devices through a hidden channel. This attack is hidden from the security system as it does not utilize the legitimate data transfer channel, and consequently cannot be identified by the traditional network security systems. Fig. 2f illustrates the covert channel attack in IoT-ED. The HT inside the IoT-ED, that is surrounded with the dotted line, sends the same data through more than one interface (where one of the interfaces is monitored and the other one is not). In the covert channel attack, the leaked data can be sent simultaneously with the legitimate data or with a time lag. This data leakage can be the duplication of the legitimate data of the IoT-ED or data that may contain secret information such as an encryption or decryption algorithm. It is not possible for traditional IDS nor firewall to detect this type of attack as the covert channel may not be monitored by the IDS system or firewall.

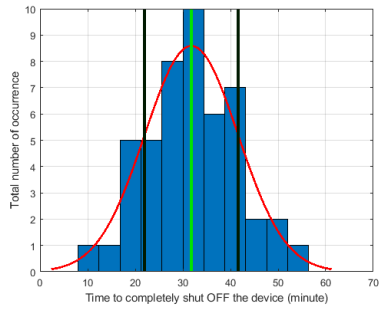
V. RANDOMIZATION OF HARDWARE ATTACKS

To emulate a realistic security scenario, randomness has been introduced into the attack. For any attack scenario, the attack on the HAN starts randomly, may last for a random duration or may repeat itself in a completely random fashion.

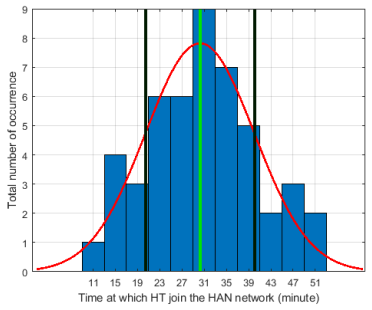
Power Depletion Attack: For power depletion attack, the IoT-ED gets disconnected from the HAN network at random time intervals. At the beginning of the network operation, the IoT-ED works normally, but after a while, it gets disconnected from the HAN network due to power depletion. The attack has been implemented following Normal distribution as shown in Fig. 3a with mean (μ) = 31.7917 and variance (σ^2) = 9.8109. This randomness results in a different power shutdown duration in each cycle of the attack. The randomness comes from a Normal distribution that is illustrated in Fig. 3a.

Impersonation Attack: In the case of the impersonation, the attacker IoT-ED joins the HAN network in a random time interval. The randomness of the impersonation attack come from a Normal distribution that is illustrated in Fig. 3b with $\mu = 28.6875$ and $\sigma^2 = 11.6439$. In this case, the attacker IoT-ED sneaks into the HAN network using an ID of a legitimate IoT-ED. Because the ID is legitimate, the attacker IoT-ED can pass the traditional security checkpoints.

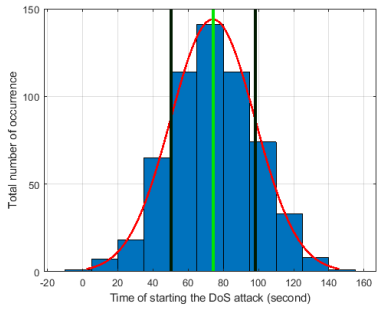
DoS Attack: In the DoS attack, the duration for which the attacker sends the message and the starting of the attack is random. After each reporting period, the attacker IoT-ED starts a DoS attack, but the beginning of the attack and the duration of the attack are random following a Normal distribution.



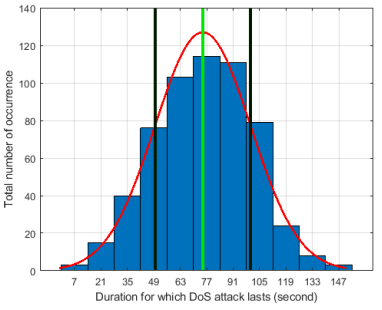
(a) Power Depletion Attack: Time to completely shut OFF the IoT-ED from power ON has Normal distribution N($\mu = 31.7917, \sigma^2 = 9.8109$).



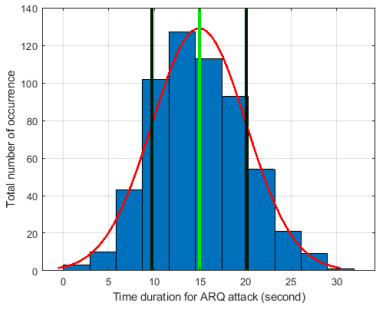
(b) Impersonation attack: Time at which HT-base malicious IoT-ED joins the network has Normal distribution N($\mu = 28.6875, \sigma^2 = 11.6439$).



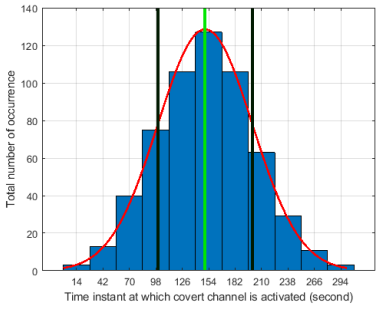
(c) DoS attack (beginning): Time from the beginning of reporting period until the first occurrence of the DoS attack N($\mu = 74.0339, \sigma^2 = 23.9390$).



(d) DoS attack (duration): Duration of DoS attack aslo follows Normal distribution N($\mu = 74.9479, \sigma^2 = 25.3365$).



(e) ARQ attack: Normal distribution of the duration of ARQ attack (ignoring the ACK message) N($\mu = 14.9097, \sigma^2 = 5.1636$).



(f) Normal distribution of covert channel attack: time instant at which covert channel is activated N($\mu = 149.0370, \sigma^2 = 50.0567$).

FIGURE 3. Normal distribution of attacks by IoT-ED on HAN network.

For instance, in the first reporting period, the IoT-ED starts the attack after 52 seconds and it lasts for 63 seconds. In the second reporting period, the DoS attack starts after a relatively longer time compared to the first attack. It starts after 127 seconds and lasts for 31 seconds and continues on in the subsequent reporting periods randomly. Figs. 3c and 3d illustrate the overall Normal distribution of the beginning time instant of the attack and the duration of this attack. The μ and σ^2 for the random beginning of the attack in each cycle are 74.0399 and 23.9390, respectively. Similarly, μ and σ^2 for the duration of the DoS attack in each reporting period are 74.9479 and 25.3365, respectively.

ARQ Attack: After each reporting period, the HT inside the IoT-ED blocks the ACK message that is received by the IoT-ED from the IoT-GD. The HT maybe designed to randomly block the number of ACK messages to increase its stealthiness. The randomness comes from a Normal distribution. For example, from the testbed, in the first reporting period, the HT inside the IoT-ED drops 5 ACK messages and in the second it drops 8. In the third reporting period, the HT drops 13 ACK messages which leads to a resend of 13 messages by the IoT-ED. Overall, the implementation of the ARQ attack is shown in Fig. 3e, which illustrates the Normal distribution of the number of messages to simulate the ARQ attack with $\mu = 14.9097$ and $\sigma^2 = 5.1636$.

Covert Channel Attack: For the covert channel attack, the attacker IoT-ED sends the leaked message through covert channel(s) at random time intervals. The randomness comes from a Normal distribution. Fig. 3 illustrates the Normal distribution of a covert channel attack of IoT-ED with $\mu = 149.0370$ and $\sigma^2 = 50.0567$. For example, from the testbed, in the first and second reporting period, the HT sends the leaked data after around 3 minutes of the legitimate reporting period, but in the third reporting period, the HT sends the data after around 5 minutes, which means with almost the next legitimate reporting data. In the fourth reporting period, the HT sends the leaked data after around 4 minutes.

VI. ON-FIELD IOT-ED SECURITY AND PRIVACY PRESERVATION DEFENSE AGAINST HT ATTACK

Traditionally most of the HT-based attack detection techniques for IoT-ED require design time modification in the IC [8], [9], [14]. However, it is possible that HT-infected IoT-ED can go undetected during the IC-testing and deployment on the field. In order to detect the triggering of such an attack, an on-field IoT-ED defense technique has been proposed in this paper. To achieve this and based on the threat model, the IoT-ED has been dealt with as a black box. In other words, the defender (which is the network operator as described in the threat model in Section IV-A) does not have access to the internal elements and on-board IC chips of the IoT-EDs. The argument in such a case is that one can monitor the IoT-EDs through three ways: 1) monitor individual communication interfaces of each IoT-ED like

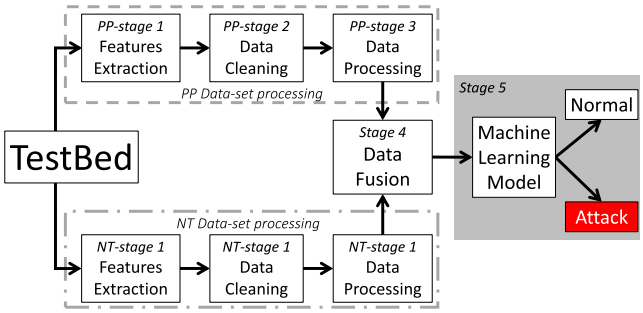


FIGURE 4. Phase I: Feature Extraction, Data cleaning, Data pre-processing, Data fusion, Training, and Testing of Machine learning model under a testbed environment for NT and PP data-sets.

wired serial interfaces, or wireless interfaces; 2) monitor the network traffic (NT) data; 3) monitor the power profile (PP) data of each IoT-ED. As monitoring each individual interface gets impractical due to the enormous number of possibilities, utilizing NT and PP data-sets are a more practical choice for detecting HT-based attacks for on-field IoT-ED defense.

To implement FusIon methodology, an algorithm has been developed, Algorithm 1. The algorithm has been divided into two phases. Phase I is the data collection phase under a testbed environment, which is from line 1 to 5 in Algorithm 1. The second phase is under an on-field run-time environment, where HT-based attacks may manifest at the physical or network layer level in an IoT-based HAN network. The second phase is executed from line 6 to 17 in Algorithm 1.

Algorithm 1 On-Field Hardware Trojan Attack Detection

Require: Testbed Environment
1: Collection and Features Extraction from NT & PP data-set
2: Data Cleaning for NT & PP
3: Data Processing for NT & PP
4: Data Fusion (NT + PP)
5: Feed data to Machine Learning Model
Ensure: Build Classifier or the Machine Learning Model On-Field
Require: Run-Time Environment
6: **while** Network Running **do**
7: Capture real time NT & PP Data
8: Feed the new data to Machine Learning Model
9: **if** Result = Normal **then**
10: No attack on the network
11: *Action* : Continue Monitoring
12: **else if** Result = Attack **then**
13: There is attack on the network
14: *Action* : Inform the Network Administrator
15: *Action* : Network Administrator remove the source of the attack or do Network optimization
16: **end if**
17: **end while**
Ensure: Running IoT-EDs in HAN without security breaches

A. PHASE I - TESTBED ENVIRONMENT

In Phase I, first the NT and PP data have been collected (line 1 of Algorithm 1). The processing of the collected data is divided into five stages, as shown in Fig. 4 (illustrated in Algorithm 1 in line 1 to 5). Three of these stages run in parallel for both data-sets (NT and PP). The last two stages are fusing the data-sets and then perform machine learning on the fused data-set. The five processing stages are explained as follows:

1) *Feature Extraction:* In Phase I - PP-stage 1 and NT-stage 1 of the proposed scheme, feature extraction has been performed on the two collected data-sets, NT and PP, which are collected and stored in ".csv" file format for each IoT-ED. It's worthy to mention here that the PP data is only related to the controlling circuit of the SHA, which is referred to as IoT-ED throughout this paper. As stated before, feature extraction has been performed for both the data-sets in parallel as shown in PP-stage 1 and NT-stage 1 of Fig. 4 (line 1 in Algorithm 1). Since this work is to detect physical-layer level attacks the focus is on extracting features related to power consumption, like voltage, current, and power. Since voltage remains almost constant, the focus is on extracting power consumption only, which is effectively a function of current consumption. On the other hand, message counting has been chosen in NT among other readings. For NT-stage 1, there were several features stored in the NT data-set, including number of messages per unit of time, source and destination MAC addresses, source and destination IP addresses, source and destination port numbers, timing, etc. Since this work is to detect HT and preserve the privacy of the IoT-ED, the focus of the work is on the number of messages per unit of time. The defender (which is the network operator) counts the messages that have been sent by the IoT-ED to report the SHA's energy usage. Also, the NT data-set is privacy-preserving, as there is no access to the contents of the messages. The NT data sending is secure as IoT-ED uses an end to end encryption algorithm.
2) *Data Cleaning:* Phase I - PP-stage 2 and NT-stage 2 of the proposed scheme, is the data cleaning process (line 2 in Algorithm 1). For both the data-sets, FusIon methodology retains the required information and removes any redundant data from the ".csv" file. The retained information in the testbed for the PP data-set is the power consumption values and for the NT data-set is the number of messages per unit of time, as stated when describing the Feature Extraction stage. Furthermore, in order to improve the robustness of the methodology, both data-sets are divided into chunks of two hour sub-data-sets; in total amount up to 48 hours of experimental results for each scenario (total of six scenarios, one normal and five attack scenarios).

3) *Data Processing:* In Phase I - PP-stage 3 and NT-stage 3, data processing occurs. The data collected by the IoT-GD is formatted to the appropriate data types and units then stored in a ".csv" file format (line 4 in Algorithm 1). This is done at the IoT-GD. Then labeling the data is performed according to the name of the IoT-ED and mode of operation.
4) *Data Fusion:* Data fusion takes place in Phase I - Stage 4. In data fusion, the two data-sets (NT and PP) have been fused (line 4 in Algorithm 1). Since the two data-sets are random and independent, combining them gives more reliable randomness to the data-set, which follows Normal distribution [38]. Next, the fused data is converted to a new format ".arff" for the machine learning algorithm. The file conversion is a requirement for the WEKA training tool [36].
5) *Machine Learning Algorithm Used:* In Phase I - Stage 5, the machine learning model is trained. The data-set results from the data fusion stage have been split into two sets. One set is for training purposes and another is for testing. For training, 66% has been used and the rest of the data has been used for cross-validation testing. This partitioning of data-sets (into training and testing data-sets) leads to a reduction in overfitting and improved generalization on unseen data. The following machine learning algorithms have been explored on the data-set 1) Support Vector Machine (SVM), 2) Artificial Neural Network (ANN), 3) Decision Tree, and 4) Random Forest. In the experiments, Random Forest has been found to provide the best results among all the above mentioned techniques, Table 2 summarize the output result of all machine learning algorithms. One reason of Random Forest providing better results for a fused data-set is that the Random Forest algorithm combines both classification and regression techniques and is suitable for the classification of a continuous data-set which is also the case in the fused data-set [39].

B. PHASE II - ON-FIELD RUN-TIME ENVIRONMENT

In Phase II, an IoT-ED attack scenario has been randomly chosen to mimic the on-field run-time environment. After collecting the two data values, corresponding to NT and PP data-sets, both data-sets have been fed into FusIon methodology in real-time while the HAN network is operational (line 7 and 8 in Algorithm 1). Fig. 5 illustrates the collection process of the NT and PP data from IoT-ED under the test. It also shows the process of how these two data values are fed to the trained machine learning model for classification in real-time to detect HT-based attacks.

In the normal case scenario (Algorithm 1), the trained machine learning model continues the monitoring process and does not take any action (line 9, 10 and 11 in Algorithm 1). If the IoT-ED is affected by any of the five attacks and the trained machine learning model classifies it as an attacker, then the FusIon methodology in Algorithm 1

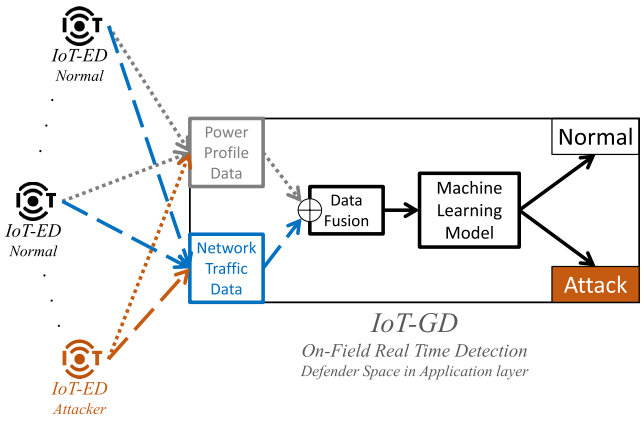


FIGURE 5. Phase II: Real-time HT detection in an on-field real-time environment, where IoT-ED is under the test.

flags an alarm and informs the defender (which is network operator) to take action (line 12 to 15 in Algorithm 1).

The messages between IoT-ED and IoT-GD are encrypted so the external attacker cannot decrypt the message and access what it contains. Also, as the encryption and network monitoring (defender) handles by two different applications at the IoT-GD, the defender can only count the number of messages and cannot access what it contains. FusIon methodology is privacy-preserving as the contents of the messages stay unraveled for the external attacker and for the defender inside the IoT-GD.

VII. EXPERIMENTAL SETUP AND RESULTS

The experimental setup and the results are explained in this section.

A. EXPERIMENTAL SETUP

In this segment, the experimental setup has been explained.

1) TESTBED SETUP

As illustrated in Figs. 1 and 5, the testbed consist of multiple IoT-EDs and an IoT-GD that every device is connected wirelessly to shape HAN network. The IoT-ED sends the reading to the IoT-GD. A publicly available power consumption data-set has been used for each home appliance to mimic a real home power consumption scenario [40]. The IoT-GD collects the power consumption data from each IoT-EDs periodically. Then the IoT-GD aggregates the collected reading and sends the aggregated data to the utility company at some predefined time instances. The network topology is a star topology, as shown in Fig. 1, where the IoT-GD is in the middle and all other IoT-EDs are at the edges of the HAN network.

Network Traffic (NT) or Message Counting: The IoT-ED sends its power consumption data to the IoT-GD then waits for the ACK signal from the IoT-GD to confirm the data delivery to the destination. IoT-ED continues working with this protocol at each predefined reporting period. A Python script has been developed that gives the IoT-GD capability

TABLE 2. Accuracy comparison among different machine learning algorithms.

	NT data-set only (%)	PP data-set only (%)	Data Fusion (NT + PP) (%)
Random Forest	57.89	67.73	92.27
Decision Tree	57.63	66.84	88.41
Support Vector Machine	50.95	33.69	62.57
Artificial neural Network	51.49	34.40	65.57

to listen to all IoT-EDs simultaneously and service all the requests. Each IoT-ED sends its power consumption data to the IoT-GD through a specific port number. These messages between the sender and the receiver are encrypted with an end-to-end encryption algorithm to prevent privacy breaching and secure the connection. The defender (network operator) counts the number of messages that have been sent by the IoT-EDs per unit time. In the experiment, the defender counts the number of messages every minute.

Power Profiling (PP): A current sensor INA219 [41] is used for measuring the PP of each IoT-EDs. The IoT-ED under test is connected to the power source through INA219. The current sensor sends the consumed power by the IoT-ED to the IoT-GD, using Python libraries once every second. Another Python script has been developed that reads the consumed power and sends the reading periodically to the IoT-GD.

The testbed experiment is carried out for 96 days. All IoT-EDs are configured using remote access through SSH terminals in order to change the mode of operations of the IoT-EDs on the fly. Two types of data that have been collected, the power profiling (PP) of the IoT-ED and the output communication messages (NT) from IoT-EDs' interface. The two data-sets (NT and PP) are fused then processed by the IoT-GD which uses this data-set to train a machine learning model (as shown in Fig. 4).

2) COMMUNICATION SETUP

TCP protocol has been used as a communication protocol because of its reliability. The IoT-ED reads the SHA power consumption and reports the reading to the IoT-GD through TCP protocol. This reporting is periodically made every five minutes. As mentioned earlier, this data is utilized to obtain an NT data-set. To maintain privacy preservation, the actual power consumption information of SHA is not accessed by the defender (network operator). The INA219 current sensor reads the consumed power by the IoT-ED and reports the reading to the IoT-GD device as well. This reporting is periodical at a rate of once per second.

3) DEVICE TYPE(S)

The testbed consists of eleven IoT-EDs of two different models namely: four Raspberry Pi 3 Model B, and seven Raspberry Pi Zero W. The Raspberry Pi 3 Model B is the third generation of Raspberry Pi. Its specification is: 1) quad-core 1.2GHz Broadcom BCM2837 64bit CPU, 2) 1GB RAM, 3) BCM43438 wireless LAN and Bluetooth Low Energy (BLE) on board, and 4) 100 base Ethernet interface. The Raspberry Pi Zero W extends the Raspberry Pi Zero family. Its specification is: 1) 1GHz, single-core CPU, 2) 512MB RAM, and 3) wireless LAN and Bluetooth Low Energy (BLE) on board. These devices have been used because they are one of the most commonly used platforms in modern IoT-based network experiments [42]–[44].

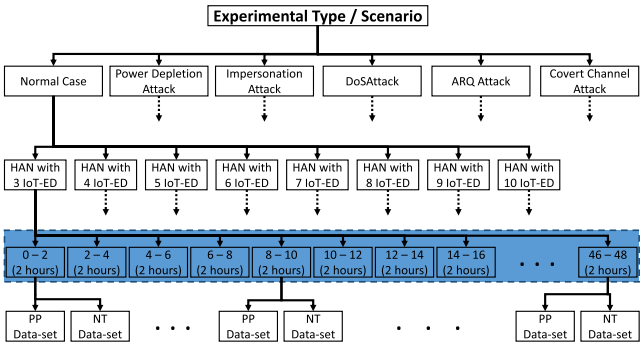


FIGURE 6. Six operation cases, each case of eight network topology (48 scenarios) for chunk of 2 hours of NT and PP data-set collection.

4) EXPERIMENTAL SCENARIOS

For HAN network topology, six scenarios have been considered; 1) Normal, when all IoT-EDs run in normal mode, 2) Power depletion attack, 3) Impersonation attack, 4) DoS attack, 5) ARQ attack, and 6) Covert channel attack. In all the cases, one device has been considered to be under attack and the rest are in normal mode. As shown in Fig. 6, eight different experiments were performed for each of the above mentioned six cases. The first scenario contains three IoT-ED and one IoT-GD. Then the number of the IoT-ED increases in each experiment by one until it reaches ten IoT-EDs in the HAN network. Fig. 6 illustrates the experimental combination that has been implemented in the testbed. In all, 48 experiments accumulating data worth for 96 days of device operation have been performed. The methodology runs each experiment for 48 hours in a chunk of 2 hours. This is illustrated as the dotted rectangle in Fig. 6. This chunk of 2 hours is to implement random duration attacks for each experiment. For each experiment, both NT and PP data values are collected, these data values are used for training a machine learning model. The NT has been counted every 1 minute and 138,240 data points have been collected. For the PP, 8,294,400 data points have been obtained as the reporting time is once every second.

As the collection of the PP data points (every one second) is more frequent than the NT data points (every one minute) to the IoT-GD, the data must be aligned for the data fusion technique to be applied. The first approach can be taking the average of the PP data and fusing it with NT data to produce new fused data to be fed to the machine learning, but this leads to the loss of some of the important information in the PP data. The second approach is to replicate the NT data and

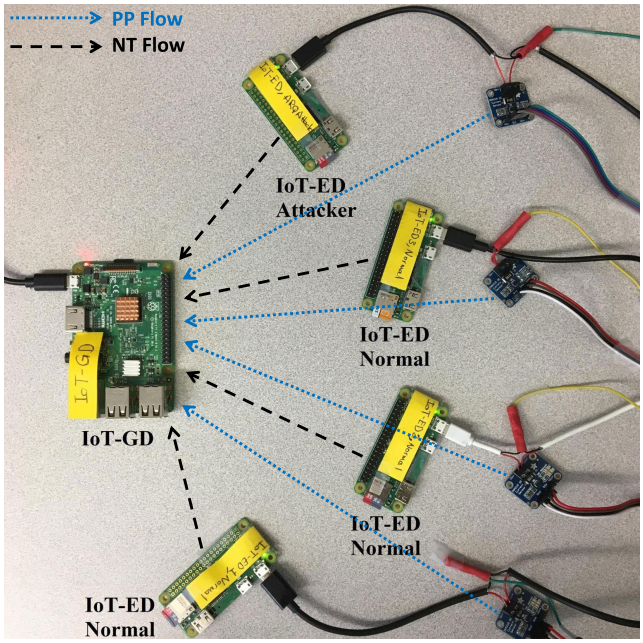


FIGURE 7. Testbed experiment example - three Normal IoT-ED, one attacker IoT-ED, and one IoT-GD.

align it with the PP data, which is the route that has been taken in this work to preserve the valuable information in the fine-grained PP data-set. This assumption can be justified by the fact that in statistically overwhelming scenarios the change in the NT data point between two consecutive samples is limited to 20%. This approach provided us with a larger data-set, which consequently helped in reducing the skewness of the distribution that leads to better accuracy.

Fig. 7 shows one of the experiments in the testbed. This experiment consists of four IoT-EDs and one IoT-GD. Three of the IoT-EDs work normally and the other IoT-ED works as an attacker. The four IoT-EDs send their data to the IoT-GD through a wireless connection (dotted black line). Their power consumption is collected by the INA219 and sent to the IoT-GD (dotted blue line).

B. RESULTS

In some of our previously reported results [45] and [35], we were able to detect the IoT-ED attacker, provided that the attacks were predetermined. In this work, the attacks have been taken to another level. The number of attacks on the HAN network has been increased. Also, all the attacks on the HAN network have been randomized i.e. the start, duration, and end of the attacks on the HAN network are randomized. This randomness decreases the detection accuracy from 95.5% in [35] to be 67.73% using the PP data-set only. We come up with the message counting (NT data-set) to detect the hardware attacks and preserve privacy. But because of the randomness in the attack, the accuracy for NT data-set only is 57.89%, which means it is not better than the PP data-set. By fusing the two data-sets (NT and PP) for the attacks with the randomness, the data fusion gives higher accuracy and reaches 92.27%. Table 3 shows the result for the NT data-set, PP data-set, and the fused data-set (NT + PP).

TABLE 3. Result of hardware Trojan detection using NT, PP, and fused data-sets in HAN network.

	Accuracy(%)	FP	Precision
NT data-set only	57.89	0.084	-
PP data-set only	67.73	0.065	0.677
Data Fusion (NT + PP)	92.27	0.015	0.925

Table 3 summarizes the result for hardware Trojan detection using data fusion including all the attacks. The reason for this increase in the classification accuracy comes from a combination of different weak classifying data-sets. Fusing two different weak classifying data-sets lead to a data-set with strong classification and the fused data-set is more informative than the original input data-sets [30], [31]. Also, the combination of the NT and PP data-sets lead to a reduction in the false positive (FP) from 0.084 for the NT data-set only and 0.065 for the PP data-set only to 0.015 for the fused data-set.

Figs. 8 shows the distribution for all data-sets, which are the NT data-set only, the PP data-set only, and the (NT + PP) data-set, i.e. the fused data-set. A perfect normal distribution data-set has a skew and kurtosis of zero. Skewness is defined as a measure of a data-set's symmetry. A perfectly symmetrical data-set has a skewness of zero. A data to be normally distributed needs to have a skew of more than -2 and less than $+2$ [54]. So, the skewness between -0.5 and $+0.5$ means the data-set is fairly symmetrical. The skewness between $+0.5$ and $+1$ or -0.5 and -1 means the data-set is moderately skewed. But when the skewness is less than -1 or greater than 1, the data-set is highly skewed. As stated earlier, lower skew is desirable, as it makes the data-set more random. Mathematically Skewness of Normal distribution is defined as follows:

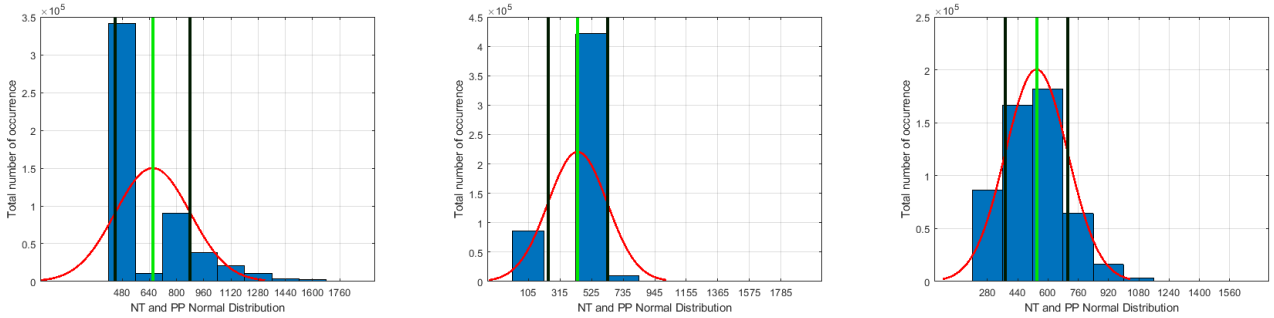
$$Skewness = \sum \frac{(X_i - \bar{X})^3}{ns^3} \tag{3}$$

where, n is the sample size, X_i is the i^{th} value, $\bar{X}$ is the mean and s is the sample standard deviation. Fig. 8a shows that the NT data-set distribution has a positive skew of $+1.74$, which means the NT data-set is highly left-skewed. Fig. 8b shows that the PP data-set distribution has a negative skew of -1.56, which means the PP data-set is highly right-skewed. But when the data-sets are fused, the distribution of the fused data-set approximates closer to Normal distribution with a skewness of -0.05 (almost zero), the fused (NT + PP) data-set is Normally distributed, as shown in Fig. 8c.

Another factor to measure the distribution of the data-sets is kurtosis. Kurtosis is a measure of the combined weight of the tails to the rest of the distribution [55].

$$Kurtosis = \sum \frac{(X_i - \bar{X})^4}{ns^4} \tag{4}$$

A positive kurtosis means there is more data at the tail than the perfect Normal distribution and a negative kurtosis means



(a) NT data-set only, D_1 distribution ($\mu_1 = 659.8979, \sigma_1^2 = 220.4516$).

(b) PP data-set only, D_2 distribution ($\mu_2 = 432.7325, \sigma_2^2 = 197.0167$).

(c) Fused data-set (D_3) distribution N($\mu_3 = 539.9391, \sigma_3^2 = 165.1474$), where $D_3 = \sigma_3^2(D_1/\sigma_1^2 + D_2/\sigma_2^2)$.

FIGURE 8. Data Distribution for raw data-sets (NT and PP) and fused data-set in IoT-ED on HAN network.

TABLE 4. State-of-the-art comparison, possibilities of attacker existence in HAN network.

Type of attacks	[34]	[46]	[9]	[8]	[21]	[14]	[15]	[17]	FusIon
Power Depletion attack	√	√	x	x	x	x	x	x	√
Impersonation attack	x	x	x	x	x	x	x	x	√
DoS attack	x	x	√	√	√	x	x	x	√
ARQ attack	x	x	x	x	x	x	x	x	√
Cover Channel attack	x	x	√	√	x	√	√	√	√

there is less data at the tail than Normal distribution. Fig. 8a shows that the NT data-set distribution has a positive kurtosis of + 2.8 and Fig. 8b shows that the PP data-set distribution has a positive kurtosis of + 1.53, which means both data-sets have more data at the tail compared to perfect Normal distribution, especially the NT data-set. But when both data-sets are fused, the kurtosis for the fused data-set (NT + PP) is reduced to + 0.35, which means it is nearer to Normal distribution.

Hence, the above explanation shows that the fused-data becomes more random. This is beneficial for machine learning algorithms. For example, the usual way to combine the randomized models into a group for a machine learning algorithm (e.g. Random Forest) is to average their predictions:

$$\Psi_{D,\theta_1,\ldots,\theta_M}(x) = \frac{1}{M}\sum \phi_{D,\theta_m}(x) \tag{5}$$

where M is a number of models, ϕ_{D,θ_m} is the randomized model (e.g. Random Forest), D is the data-set, θ_m is the random seed, and $\Psi_{D,\theta_1,\ldots,\theta_M}$ is the new group model (e.g. Random Forest after fusing the data-sets). So the expected generalization error of the group is smaller compared to the generation error of the individual randomized models. The reason is that the average prediction is the prediction that reduces the average squared error concerning the individual predictions of the models. The average prediction is the closest prediction concerning all individual predictions. Creating a fused data-set always reduces the variance of the class probability estimate, which results in a reduction of the misclassification error in a randomized model [56].

TABLE 5. Result of feeding fused data to machine learning and individual attack detection accuracy in HAN.

	Accuracy(%)	FP	Precision
Power Depletion attack	83.9	0.042	0.799
Impersonation	100.0	0.000	1.000
DoS attack	99.0	0.002	0.991
ARQ attack	99.1	0.002	0.990
Covert Channel attack	79.8	0.044	0.784
Normal case	91.9	0.003	0.986

By fusing the two data-sets (NT and PP), the accuracy increases and the false-positive decreases. Table 5 shows the result of feeding fused data to machine learning. The hardware Trojan detection accuracy is high for most of the attack, but the machine learning model is not able to differentiate among a power depletion attack, covert channel attack and normal operation, as these three cases follow the same data pattern reporting to the IoT-GD. This lead to a reduction in the overall accuracy for all attacks to be 92.27% as shown in Table 3. To increase the accuracy for these attacks, FusIon methodology recommends comparing each individual attack against the normal case operation.

Table 6 shows the accuracy for the trained machine learning model to differentiate between the normal case and one attack case at a time. The trained machine learning model is able to classify almost all the attacks with very high accuracy. It can be clearly seen from Table 6 that FusIon technique is providing better results, except for power depletion attack compared to almost all the state-of-the-art literature. Regarding power depletion attack, in [34] the authors only discussed

TABLE 6. The accuracy for the trained machine learning model to differentiate between the normal case and one attack case at a time in HAN network.

	[34]	[21]	[15]	[17]	FusIon
Normal vs PD Attack	98.6%	–	–	–	97.27%
Normal vs IM Attack	–	–	–	–	100.0%
Normal vs DoS Attack	–	98%	–	–	100.0%
Normal vs ARQ Attack	–	–	–	–	100.0%
Normal v. CC Attack	–	–	92.27%	90%	94.95%

a software/malware-based power depletion attack. While this work is the first one that detects the hardware-based attack on the field without presuming a design level modification in the hardware.

Compared to other works in literature, the proposed technique can detect five attacks concurrently with high accuracy. Table 4 compares the FusIon approach with other approaches in the literature. In [34] & [46], the authors address the power manipulation or power depletion attack inside the IoT devices. The work [34] reports their accuracy as 98.6% for only a power depletion attack. In [8] & [9], the authors address the DoS attack and covert channel attack on IoT devices. Work [21] addresses only the DoS attack on IoT devices and their accuracy is 98% for a DoS attack. Authors of [14], [15], & [17] address the covert channel attack on an IoT device. The works [15] & [17] report their accuracy as 92% and 90% respectively for a covert channel attack only.

By fusing the NT and PP data-set, the proposed technique can concurrently detect five attacks with an accuracy of 92.27% as reported in the last column of the Table 4. FusIon methodology does the decision on the IoT-GD, i.e. the trained machine learning model runs on the IoT-GD because it is secure, better and faster compared to cloud-based services [16]. Although cloud-based services provide the required resources, data communication from IoT-EDs to cloud brings up different challenges: security problems, power consumption, time lag, and bandwidth limitation. To get over these limitations, previous studies have suggested moving data processing close to the edge of the network [16], [58], [59].

VIII. CONCLUSION

In this paper, the hardware Trojan attack and its effect on IoT devices and the IoT-based HAN has been studied. This paper investigates five types of HT-based attacks on HAN. The network traffic and power profiling data-sets have been used to train a machine learning model. With randomness introduced in the five attacks, the detection accuracy decreases for both data-sets. The proposed data fusion technique leads to an increase in the detection accuracy and reduction in the false positives. The proposed methodology results show more than 99% accuracy on the individual attack detection, which is better than most of the state-of-the-art works [15], [17], [21], and [34]. Moreover, this work is able to detect five types of HT-based attacks concurrently, which is, to the best of authors' knowledge, a unique contribution.

REFERENCES

[1] Ericsson. (Nov. 2018). IoT connections outlook. [Online]. Available: https://www.ericsson.com/en/mobility-report/reports/november-2018/iot-connections-outlook.
[2] N. Gagliordi. (2017). *IoT Spending to Surpass $800 Billion in 2017, Led by Hardware: IDC*. [Online]. Available: https://www.zdnet.com/article/iotspending-to-surpass-800-billion-in-2017-led-by-hardware-idc/
[3] FTC Staff Report, "Internet of Things: Privacy and security in a connected world," Federal Trade Commission, Washington, DC, USA, Tech. Rep. 150127, 2015. [Online]. Available: https://www.ftc.gov/system/files/documents/reports/federal-trade-commission-staff-report-november-2013-workshop-entitled-internet-things-privacy/150127iotrpt.pdf
[4] FTC Staff Report, "Privacy & data security," Federal Trade Commission, Washington, DC, USA, Tech. Rep. 508, 2018. [Online]. Available: https://www.ftc.gov/system/files/documents/reports/privacy-data-security-update-2018/2018-privacy-data-security-report-508.pdf
[5] Gartner. (Aug. 2018). *Gartner Forecasts Worldwide Information Security Spending to Exceed $124 Billion in 2019*. Newsroom. [Online]. Available: https://www.gartner.com/en/newsroom/pressreleases/2018–08-15-gartner-forecasts-worldwide-information-securityspending-to-exceed-124-billion-in-2019
[6] Gartner. (Mar. 2018). *Gartner Says Worldwide IoT Security Spending Will Reach $1.5 Billion in 2018*. Newsroom. [Online]. Available: https://www.gartner.com/en/newsroom/press-releases/2018–03-21-gartner-says-worldwide-iot-security-spending-will-reach-1-point-5-billion-in-2018
[7] H. Li, Q. Liu, and J. Zhang, "A survey of hardware trojan threat and defense," *Integration*, vol. 55, pp. 426–437, Sep. 2016.
[8] J. Dofe, J. Frey, and Q. Yu, "Hardware security assurance in emerging IoT applications," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, May 2016, pp. 2050–2053.
[9] C. Liu, P. Cronin, and C. Yang, "A mutual auditing framework to protect IoT against hardware trojans," in *Proc. 21st Asia South Pacific Design Autom. Conf. (ASP-DAC)*, Jan. 2016, pp. 69–74.
[10] Y. Shin, H. C. Kim, D. Kwon, J. H. Jeong, and J. Hur, "Unveiling hardware-based data prefetcher, a hidden source of information leakage," in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur.*, Oct. 2018, pp. 131–145.
[11] J. Classen, M. Schulz, and M. Hollick, "Practical covert channels for WiFi systems," in *Proc. IEEE Conf. Commun. Netw. Secur. (CNS)*, Sep. 2015, pp. 209–217.
[12] S. Bhunia, M. S. Hsiao, M. Banga, and S. Narasimhan, "Hardware trojan attacks: Threat analysis and countermeasures," *Proc. IEEE*, vol. 102, no. 8, pp. 1229–1247, Aug. 2014.
[13] S. Bhasin and F. Regazzoni, "A survey on hardware trojan detection techniques," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, May 2015, pp. 2021–2024.
[14] K. S. Subramani, A. Antonopoulos, A. A. Abotabl, A. Nosratinia, and Y. Makris, "INFECT: INconspicuous FEC-based trojan: A hardware attack on an 802.11a/g wireless network," in *Proc. IEEE Int. Symp. Hardw. Oriented Secur. Trust (HOST)*, May 2017, pp. 90–94.
[15] L. Caviglione, M. Gaggero, J.-F. Lalande, W. Mazurczyk, and M. Urbanski, "Seeing the unseen: Revealing mobile malware hidden communications via energy consumption and artificial intelligence," *IEEE Trans. Inf. Forensics Security*, vol. 11, no. 4, pp. 799–810, Apr. 2016.
[16] A. O. Akmandor, H. Yin, and N. K. Jha, "Simultaneously ensuring smartness, security, and energy efficiency in Internet-of-Things sensors," in *Proc. IEEE Custom Integr. Circuits Conf. (CICC)*, Apr. 2018, pp. 1–8.
[17] Z. Ling, J. Luo, Y. Xu, C. Gao, K. Wu, and X. Fu, "Security vulnerabilities of Internet of things: A case study of the smart plug system," *IEEE Internet Things J.*, vol. 4, no. 6, pp. 1899–1909, Dec. 2017.
[18] U.S. Department of Energy, Office of Electricity Delivery and Energy Reliability. *What is the Smart Grid?* Accessed: Dec. 2019. [Online]. Available: https://www.smartgrid.gov/the_smart_grid/smart_grid.html
[19] P. Jokar and V. Leung, "Intrusion detection and prevention for ZigBee-based home area networks in smart grids," *IEEE Trans. Smart Grid*, to be published.
[20] M. A. Faisal, Z. Aung, J. R. Williams, and A. Sanchez, "Data-Stream-Based intrusion detection system for advanced metering infrastructure in smart grid: A feasibility study," *IEEE Syst. J.*, vol. 9, no. 1, pp. 31–44, Mar. 2015.

[21] X. Yuan, C. Li, and X. Li, "DeepDefense: Identifying DDoS attack via deep learning," in *Proc. IEEE Int. Conf. Smart Comput. (SMARTCOMP)*, May 2017, pp. 1–8.

[22] J. Canedo and A. Skjellum, "Using machine learning to secure IoT systems," in *Proc. 14th Annu. Conf. Privacy, Secur. Trust (PST)*, Dec. 2016, pp. 219–222.

[23] H. Mohammed, S. Tonyali, K. Rabieh, M. Mahmoud, and K. Akkaya, "Efficient privacy-preserving data collection scheme for smart grid AMI networks," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Dec. 2016, pp. 1–6.

[24] A. Alsharif, M. Nabil, S. Tonyali, H. Mohammed, M. Mahmoud, and K. Akkaya, "EPIC: Efficient privacy-preserving scheme with EtoE data integrity and authenticity for AMI networks," *IEEE Internet Things J.*, vol. 6, no. 2, pp. 3309–3321, Apr. 2019.

[25] S. A. Parker, W. Hunt, K. McMordie Stoughton, B. K. Boyd, K. M. Fowler, T. M. Koehler, W. F. Sandusky, G. P. Sullivan, and R. Pugh, "Metering best practices. A guide to achieving utility resource efficiency, release 3.0," Pacific Northwest Nat. Lab. (PNNL), Richland, WA, USA, Tech. Rep. PNNL-23892, 2015.

[26] X. Wang, M. Tehranipoor, and J. Plusquellic, "Detecting malicious inclusions in secure hardware: Challenges and solutions," in *Proc. IEEE Int. Workshop Hardw.-Oriented Secur. Trust*, Jun. 2008, pp. 15–19.

[27] M. Tehranipoor and F. Koushanfar, "A survey of hardware trojan taxonomy and detection," *IEEE design test Comput.*, vol. 27, no. 1, pp. 10–25, Jan./Feb. 2010.

[28] R. M. Rad, X. Wang, M. Tehranipoor, and J. Plusquellic, "Power supply signal calibration techniques for improving detection resolution to hardware trojans," in *Proc. IEEE/ACM Int. Conf. Comput.-Aided Design*, Nov. 2008, pp. 632–639.

[29] H. Salmani, M. Tehranipoor, and J. Plusquellic, "A novel technique for improving hardware trojan detection and reducing trojan activation time," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 20, no. 1, pp. 112–125, Jan. 2012.

[30] T.-D. Vu, O. Aycard, and F. Tango, "Object perception for intelligent vehicle applications: A multi-sensor fusion approach," in *Proc. IEEE Intell. Vehicles Symp. Proc.*, Jun. 2014, pp. 774–780.

[31] M. Stubler, S. Reuter, and K. Dietmayer, "A continuously learning feature-based map using a Bernoulli filtering approach," in *Proc. Sensor Data Fusion: Trends, Solutions, Appl. (SDF)*, Oct. 2017, pp. 1–6.

[32] R. E. Castellanos and P. Millan, "Design of a wireless communications network for advanced metering infrastructure in a utility in colombia," in *Proc. IEEE Colombian Commun. Conf. (COLCOM)*, May 2012, pp. 1–6.

[33] F. K. Lodhi, S. R. Hasan, O. Hasan, and F. Awwadl, "Power profiling of microcontroller's instruction set for runtime hardware trojans detection without golden circuit models," in *Proc. Design, Autom. Test Eur. Conf. Exhib. (DATE),*, Mar. 2017, pp. 294–297.

[34] L. Liu, G. Yan, X. Zhang, and S. Chen, "Virusmeter: Preventing your cellphone from spies," in *Proc. Int. Workshop Recent Adv. Intrusion Detection*. Berlin, Germany: Springer, 2009, pp. 244–264.

[35] H. Mohammed, T. Odetola, S. R. Hasan, S. Stissi, I. Garlin, and F. Awwad, "(HIADIoT): Hardware intrinsic attack detection in Internet of Things; Leveraging power profiling," in *Proc. IEEE 62nd Int. Midwest Symp. Circuits Syst. (MWSCAS)*, Aug. 2019, pp. 852–855.

[36] WEKA. (2019). *Machine Learning group At the University of Waikato*. Release 3.8. [Online]. Available: https://www.cs.waikato.ac.nz/ml/weka/index.html

[37] H. Mohammed, J. Howell, S. R. Hasan, N. Guo, F. Khalid, and O. Elkeelany, "Hardware trojan based security issues in home area network: A testbed setup," in *Proc. IEEE 61st Int. Midwest Symp. Circuits Syst. (MWSCAS)*, Aug. 2018, pp. 972–975.

[38] M. Rosenblatt, "A central limit theorem and a strong mixing condition," *Proc. Nat. Acad. Sci. USA*, vol. 42, no. 1, p. 43, 1956.

[39] L. Breiman, "Random forests," *Mach. Learn.*, vol. 45, no. 1, pp. 5–32, Oct. 2001.

[40] M. Pipattanasomporn, M. Kuzlu, S. Rahman, and Y. Teklu, "Load profiles of selected major household appliances and their demand response opportunities," *IEEE Trans. Smart Grid*, vol. 5, no. 2, pp. 742–750, Mar. 2014.

[41] Texas Instruments. (Dec. 2015). *INA219 Zero-Drift, Bidirectional Current/Power Monitor With I^2C Interface SBOS448G*. [Online]. Available: http://www.ti.com/lit/ds/symlink/ina219.pdf

[42] O. Hamdan, H. Shanableh, I. Zaki, A. R. Al-Ali, and T. Shanableh, "IoT-based interactive dual mode smart home automation," in *Proc. IEEE Int. Conf. Consum. Electron. (ICCE)*, Jan. 2019, pp. 1–2.

[43] D. Pavithra and R. Balakrishnan, "IoT based monitoring and control system for home automation," in *Proc. Global Conf. Commun. technol. (GCCT)*, Apr. 2015, pp. 169–173.

[44] V. Vujović and M. Maksimović, "Raspberry Pi as a Sensor Web node for home automation," *Comput. Electr. Eng.*, vol. 44, pp. 153–171, May 2015.

[45] J. Shelley, H. Mohammed, L. Zink, S. R. Hasan, and O. Elkeelany, "Covert communication channel detection in low-power battery operated IoT devices: Leveraging power profiles," in *Proc. SoutheastCon*, Apr. 2018, pp. 1–6.

[46] A. B. Lopez, K. Vatanparvar, A. P. Deb Nath, S. Yang, S. Bhunia, and M. A. Al Faruque, "A security perspective on battery systems of the Internet of things," *J. Hardw. Syst. Secur.*, vol. 1, no. 2, pp. 188–199, May 2017.

[47] F. K. Lodhi, I. Abbasi, F. Khalid, O. Hasan, F. Awwad, and S. R. Hasan, "A self-learning framework to detect the intruded integrated circuits," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, May 2016, pp. 1702–1705.

[48] K. Yang, J. Ren, Y. Zhu, and W. Zhang, "Active learning for wireless IoT intrusion detection," *IEEE Wireless Commun.*, vol. 25, no. 6, pp. 19–25, Dec. 2018.

[49] L. Xiao, X. Wan, X. Lu, Y. Zhang, and D. Wu, "IoT security techniques based on machine learning: How do IoT devices use AI to enhance security?" *IEEE Signal Process. Mag.*, vol. 35, no. 5, pp. 41–49, Sep. 2018.

[50] V. Thangavelu, D. M. Divakaran, R. Sairam, S. S. Bhunia, and M. Gurusamy, "DEFT: A distributed IoT fingerprinting technique," *IEEE Internet Things J.*, vol. 6, no. 1, pp. 940–952, Feb. 2019.

[51] L. Wei, W. Luo, J. Weng, Y. Zhong, X. Zhang, and Z. Yan, "Machine learning-based malicious application detection of android," *IEEE Access*, vol. 5, pp. 25591–25601, 2017.

[52] J. Llinas and D. L. Hall, "An introduction to multi-sensor data fusion," in *Proc. ISCAS. Proc. IEEE Int. Symp. Circuits Syst. (Cat. No.98CH7)*, vol. 6, Jun. 1998, pp. 537–540.

[53] B. V. Dasarathy, "Sensor fusion potential exploitation-innovative architectures and illustrative applications," *Proc. IEEE*, vol. 85, no. 1, pp. 24–38, Jan. 1997.

[54] D. George and M. Mallery, *SPSS for Windows Step by Step: A Simple Guide and Reference 17.0 Update*. Boston, MA, USA: Pearson, 2010.

[55] J. D. Wheeler. (2011). *Problems with Skewness and Kurtosis, Part One*. [Online]. Available: htpps://www.qualitydigest.com

[56] G. Louppe, "Understanding random forests: From theory to practice," 2014, *arXiv:1407.7502*. [Online]. Available: http://arxiv.org/abs/1407.7502

[57] T. A. Odetola, H. Raoof Mohammed, and S. Rafay Hasan, "A stealthy hardware trojan exploiting the architectural vulnerability of deep learning architectures: Input interception attack (IIA)," 2019, *arXiv:1911.00783*. [Online]. Available: http://arxiv.org/abs/1911.00783

[58] M. Baza, M. Nabil, N. Lasla, K. Fidan, M. Mahmoud, and M. Abdallah, "Blockchain-based firmware update scheme tailored for autonomous vehicles," in *Proc. IEEE Wireless Commun. Netw. Conf. (WCNC)*, Apr. 2019.

[59] W. AlAmiri, M. Baza, M. Mahmoud, W. Alasmary, and K. Akkaya, "Towards secure smart parking system using blockchain technology," in *Proc. 17th IEEE Annu. Consum. Commun. Netw. Conf. (CCNC)*, Las Vegas, NV, USA, Nov. 2020, pp. 1–6.

HAWZHIN MOHAMMED received the B.Sc. degree in electrical engineering from Salahaddin University, Erbil, Iraq, in 2000, and the M.Sc. degree from Tennessee Tech University, Cookeville, TN, USA, in 2017, where he is currently pursuing the Ph.D. degree with the Department of Electrical and Computer Engineering. His current research interest includes wireless network security, hardware security, the IoT security, machine learning, and deep learning.

SYED RAFAY HASAN received the B.Eng. degree in electrical engineering from the NED University of Engineering and Technology, Pakistan, and the M.Eng. and Ph.D. degrees in electrical engineering from Concordia University, Montreal, QC, Canada. From 2006 to 2009, he was an Adjunct Faculty Member with Concordia University. From 2009 to 2011, he was a Research Associate with the Ecole Polytechnique de Montreal. Since 2011, he has been with the Electrical and Computer Engineering Department, Tennessee Tech University, Cookeville, TN, USA, where he is currently an Associate Professor. He has published more than 69 peer-reviewed journals and conference papers. His current research interests include hardware design security in the Internet of Things (IoT), hardware implementation of deep learning, deployment of convolution neural networks in the IoT edge devices, and hardware security issues due to adversarial learning. He received the Postdoctoral Fellowship Award from the Scholarship Regroupment Stratgique en Microsystmes du Québec, the SigmaXi Outstanding Research Award, the Faculty Research Award from Tennessee Tech University, the Kinslow Outstanding Research Paper Award from the College of Engineering, Tennessee Tech University, and the Summer Faculty Fellowship Award from the Air force Research Lab (AFRL). He has received research and teaching funding from NSF, ICT-funds UAE, AFRL, and Intel Inc. He has been part of the funded research projects, as a PI or a Co-PI, that worth more than $1.1 million. He has been the Session Chair and Technical Program Committee Member of several IEEE conferences including ISCAS, ICCD, MWSCAS, and NEWCAS, and a Regular Reviewer for several IEEE Transactions and other journals including TCAS-II, IEEE Access, *Integration, the VLSI Journal*, *IET Circuits, Devices & Systems*, and the IEEE Embedded System Letters.

FALAH AWWAD received the M.Sc. and Ph.D. degrees in electrical and computer engineering from Concordia University, Montreal, QC, Canada, in 2002 and 2006, respectively. He was a Postdoctoral Fellow with the Ecole Polytechnique de Montréal and also with Concordia University, Montreal, QC, Canada. From August 2007 to February 2013, he was an Assistant Professor with the College of Information Technology, United Arab Emirates University (UAE University). He is currently an Associate Professor with the Department of Electrical Engineering, College of Engineering, UAE University. He has published more than 80 research articles in peer-reviewed journals and international conferences. He is a member of the Editorial Board of several journals. He is the principle investigator and Co-PI of 20 research projects and supervised several postgraduate students. His scientific research interests include primarily sensors, circuits, and devices, in addition to hardware security, biomedical applications, and smart grids. He is the Session Chair and a Technical Program Committee Member of several international conferences including International Conference on Design and Technology of Integrated Systems in Nanoscale Era, ISCAS, IT Innovations Conference, MWSCAS, NEWCAS, ICM, ICREGA, EMBC, and the IEEE Sensors. He is also a Regular Reviewer for several journals such as *Microelectronics Journal*, the *Journal of Nanomaterials & Molecular Nanotechnology*, TVLSI, and *Biosensors and Bioelectronics*.