IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received May 25, 2019, accepted June 19, 2019, date of publication June 28, 2019, date of current version July 17, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2925787

Light-Weight Security and Blockchain Based Provenance for Advanced Metering Infrastructure

MOHSIN KAMAL, (Member, IEEE), AND MUHAMMAD TARIQ, (Senior Member, IEEE)
National University of Computer and Emerging Sciences at Peshawar, Peshawar 25000, Pakistan
Corresponding author: Muhammad Tariq (tariq.khan@nu.edu.pk)

ABSTRACT The protection of smart meters (SMs) from cyberattacks is of utmost importance because SMs in advanced metering infrastructure (AMI) are physically unprotected and produce a large amount of sensitive data. Due to scalability, the SMs are small-sized and low-cost devices having low computational capabilities. The algorithms that are designed to complete the security requirements of SMs should be lightweight. To address this issue, this paper proposes a lightweight security solution to address the man-in-the-middle attack, data tempering, and blockchain-based data provenance. Received signal strength indicator (RSSI) is used to generate link fingerprints, which are used along with pseudo-random nonce to secure AMI. The proposed algorithm detects the involvement of adversarial node or meter tempering by computing other values along with 0 and 1 as the average of consecutive RSSI and difference between the RSSI of connected static SMs. Pearson correlation coefficient (ρ) of 0.9102 is achieved when no adversarial node is present in between the connected SMs having mobility in one or both SMs. Negative or approximately equal to zero values of ρ are computed when the adversary is present in the AMI or any of the SM in the AMI is forged. For blockchain-based data provenance, all the hash values of the packet header are 100% matched with the hash functions present at the data concentrator unit (DCU), which shows no adversary's involvement in AMI. For cases when the adversary is in the AMI, hash functions show no match with the hash values present at the DCU.

INDEX TERMS Advanced metering infrastructure, link fingerprints, light-weight, provenance, security, blockchain.

I. INTRODUCTION

Electricity theft is a problem that every Electricity Service Provider (ESP) faces around the globe. It costs around 6 billion USD to ESP per year in USA [1]. In a developing country like Pakistan, approximately, 20.4% distribution losses are detected per year which is 18,919 GWh out of 92,480 GWh units [2]. Researchers are working on finding the ways to mitigate this problem. Internet of Things (IoT) has become an integral part in our daily life and has opened up many new ways of finding the solutions to our existing problems [3]. The smart grid is an application of IoT which is introduced in countries like USA, Japan and China to help the authorities in overcoming theft related issues. The traditional power grids are upgraded with the capabilities of communication and information systems as their main functions. These functions are important in analyzing, monitoring and controlling devices in the system. Goals of these upgraded grids are to save energy, reduce costs, and increase reliability and transparency [4] [5]. EPSs are installing Advanced Metering Infrastructure (AMI) which plays a key role in the detection, localization and prevention of any malicious activity in the network. The traditional electricity meters at users' end are changed to Smart Meters (SMs) which have full duplex transmission capabilities [6]. The authorities or ESPs can access user's data remotely and various algorithms are designed to detect any malicious activity at any level of the distributed network [7].

In most cases, SMs in the AMI are openly installed and are not physically protected due to which they are easily accessible and are prone to cyber attacks. Because of the small size and low computational capabilities of SM, the processes at SM to safeguard from these attacks need to be light-weight so that the memory does not overflow and the data is reliably routed to the DCU [8]. If any damage is done to the data at an SM, the DCU should be able to detect and localize the approximate location of the undesired activity in the AMI.

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

Due to the scalability of SMs in the AMI, the importance of data provenance cannot be neglected. Efficient and secure data provenance in AMI is used in improving the trust of user and data forensics [9]. The origin of the data helps to determine and explain the derivation history of the data starting from the original resource. However, the integrity of the data provider is a major issue. If the data is not adequately protected by implementing inefficient security protocols, it can be forged or tampered by unauthorized parties [10]. Blockchain has emerged as the possible solution for creating more secure distributed systems [11]. Blockchains contain specific and verifiable records of all transactions that have been made within a system. We can apply the concept in smart grids to secure the AMI.

Our main contributions are:

- proposing a light-weight solution for securing AMI,
- detecting real time adversarial node and,
- localizing and provenance through blockchain.

AMI does not require any additional hardware to perform these operations. The communication capabilities of SMs are used to detect any adversarial node or data tempering. The results are achieved by considering the cases when SMs are fixed and mobile.

Rest of the paper is organized as follows. Section II provides an overview on related work done on security of IoT and AMI networks. The system model, assumptions and threat model of our system are described in Section III. Methodology of our work is discussed in Section IV. Discussion on results generated by using MICAz motes and MATLAB are presented in Section V. The paper is concluded in Section VI.

II. LITERATURE REVIEW

The SMs in AMI are easily accessible due to which they are prone to cyber attacks. SMs are small in size and have less computational capabilities. The proposed solution should be light-weighted and should gain trust of the user and service providers [6]. Usually Zigbee module is used for communication of SMs in wireless meshed networks having little processing power, typically having only 4 to 12 kB of RAM and 64 to 256 kB of flash memory [4]. MICAz motes are used to secure IoT network considering the light-weight requirements for the network. RSSI values are used to generate link fingerprints. The results show that security and data provenance are achieved with very less computational cost [12]. This can be extended to work for AMI having SMs. A protocol called Integrated Authentication and Confidentiality (IAC) is derived to provide smart grid with efficient and secure AMI communication. In IAC, every time a new smart meter joins an AMI network, the AMI system can provide trust service, data privacy and integrity with mutual authentication [13]. Researchers have worked on the detection of a theft in AMI using machine learning techniques. The patterns of electricity usage are generated. If the patterns mismatch, the abnormal usage is detected. Both supervised and unsupervised machine learning algorithms come into play while detecting a theft [14]. Data mining algorithms for detection and localization of theft are proposed in [15]. The energy theft is detected using classifiers such as Support Vector Machine (SVM) and Decision Tree (DT). At the DT, the data is processed which is then given to SVM classifier as an input [16]. The algorithms based on stochastic Petri net formalism in grid-tied micro grids, as proposed in [5], are useful in theft detection and its localization but they need systems having large computational power. Due to which they are not suitable considering SMs. The comparison of energy consumption by various systems is very important when designing a protocol and finding a solution to secure AMI. This is done in [17] and [12]. The energy requirements for Graphics Processing Unit (GPU) and Field-Programmable Gate Array (FPGA) based systems are described in the first while various cryptographic and non-cryptographic protocols are compared in the later. The results show that FPGA based systems consumes less energy. In [6], the researchers have used the general radio capabilities of MICAz motes for a case when both SMs and adversary are stationary due to which adversary can get away with security algorithms applied by eavesdropping at a distance equal to the distance of SM_1 and SM_2.

For reliable communication, Amplify and Forward (AF), Decode and Forward (DF) and Decode Amplify and Forward (DAF) relaying protocols are used when the distance between the communicating devices are large. DAF outperforms all other relaying protocols for Signal to Noise Ratio (SNR) [18]. A security framework for IoT management called DeadBolt is introduced in [19]. DeadBolt hides all devices in the IoT deployment behind an access point, implementing a denying policy by default for both inbound and outbound traffic. The researchers in [20] have proposed a trusted Internet of Vehicles (IoV) network. Device-to-device vehicle-to-vehicle (D2D-V2V) based IoV networks are considered for quick delivery of contents by applying protocols to both physical and social layers. In [9], data provenance is achieved for body-worn devices. They have used MICAz motes to perform experiments for high and low activities which produces highly correlated link fingerprints. Further optimization techniques reduce the energy requirements to as low as 52.580 mJ. Gale-Shapley algorithm is used to improve the energy efficiency in IoT environment by matching D2D pair with user equipment (UE). Correlation among UEs are analyzed using mathematical models of strategic interaction between rational decision-makers [21], [22].

While blockchains have started as Bitcoin's core technology, their use cases have expanded to many other areas, including finance, IoT, and security etc. The technology of blockchain has the potential to take over the security challenges of IoT-enabled services, such as secure data sharing and data integrity [23]. Researchers in [24] have designed an IoT system using blockchains. The system has a private key stored in the IoT nodes while public keys are saved in Etherum.

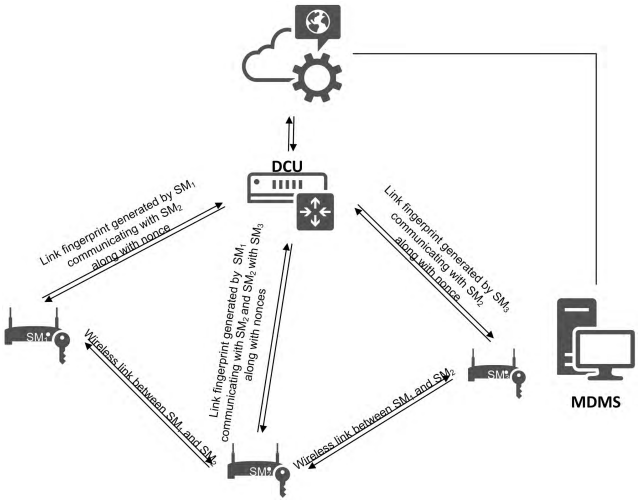

FIGURE 1. System Model.

III. NETWORK MODEL, ASSUMPTIONS AND THREAT MODEL

All the components in AMI communicate with each other by either a wired or a wireless link. The communication modules used are Zigbee and WIFI in case of wireless connectivity. In our case, we have used MICAz motes as communication module of SM, DCU and adversarial node. The system model is shown in Figure 1, which shows that in AMI, all SMs are connected to the DCU. SMs also communicate with each other. All SMs in AMI are physically unprotected and have very little computational capabilities because of its small size and scalability. The DCU is where all the computations are performed because of being dedicated, powerful and secure system. The DCU decides if any intrusion in AMI is detected or not. The Meter Data Management System (MDMS) is remotely connected to DCU via the Internet. The DCU transmits the data received from all SMs to the MDMS through the Internet, where the EPS generates bills, unit consumed and various other parameters for specific SM.

Securing AMI is of utmost importance in the Smart Grid. Most of the communication is carried out at the SMs which are not physically protected. If proper measurements are not taken, important data loss may occur. The attacker can get access to important information, can change the data or can change the control commands [25]. Threats include eavesdropping, man-in-the-middle attack, replay attacks and Distributed Denial of Service (DDoS). The following threats are considered in designing the methodology for AMI, which is light-weight and can effectively tackle these threats.

A. MAN-IN-THE-MIDDLE ATTACK

When an adversarial node comes in between the communicating path of SMs, it changes the communicating path. The SMs do not differentiate if a path is direct or via adversary. The adversary pretends to have the same node ID and there is no alteration in the original message. The adversary can be static or mobile.

B. JAMMING

It is a popular Denial of service (DoS) attack on physical layer of network. In AMI, adversaries interfere with the communication frequencies being used by the SMs. The communication of jammed SM with any other SM and DCU is not valid anymore.

C. TEMPERING

As SMs in AMI are not physically protected, it is easy for any attacker to temper the data of a legitimate SM. The attacker changes the data and sends it forward.

D. REPLAY ATTACK

In replay attack, an attacker sends the same data again and again to the DCU or any other SM. The receiving entity receives a wrong data and information loss occurs.

IV. METHODOLOGY

Methodology is designed for AMI in which SMs are both static and mobile. The RSSI values acquired from connected SMs have a linear relationship in terms of variations. The RSSI values received at each SM are in dBm which are converted to binary values by quantizing them. These binary streams are referred to as the link fingerprint. These link fingerprints are encoded and then transmitted to the DCU where the decision about the presence or absence of adversary is made. The data is sent to MDMS where the calculations regarding billing, data usage etc. are done. Figure 1 shows our system model.

A light-weight solution for the following scenarios is presented in our methodology:

1) detection of adversarial node between any two static SMs,
2) detection of adversarial node between any two mobile SMs,
3) detection of adversarial node between a static and mobile SM,
4) tempering of any SM in the AMI and,
5) data provenance.

The proposed scheme secures an AMI efficiently by consuming less energy and memory. Real-time experimental values are used to detect any intrusion in the AMI. MICAz motes are used as the communicating modules for SMs and an adversary. The results achieved from these motes are then simulated on MATLAB for the detection of any adversarial node in the network or any data tempering that is done at the node level. Figure 2 shows the orientation of experimental setup. MICAz motes are attached on the top the meters which are approximately 50 meters apart from each other in case of static SMs. MICAz motes are programmed to record RSSI values after every one minute. These RSSI values ranges from -55 dBm to 20 dBm.

A. OPERATIONS AT THE SMART METER

The Jakes fading model states that the radio channel is rapidly uncorrelated at approximately half the wavelength and that the channel is considered to be independent at distances

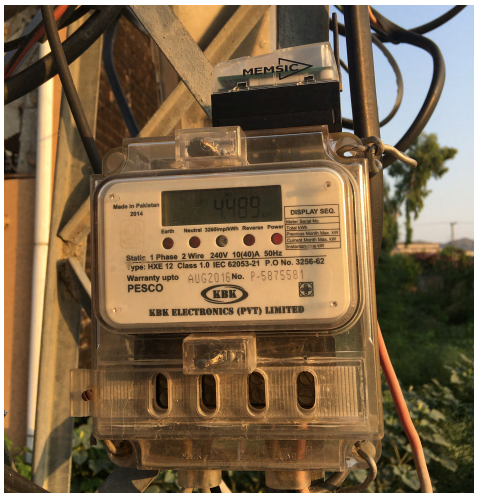

FIGURE 2. Experimental setup. MICAz mote attached on the top of meter provided by ESP.

beyond one wavelength. It means that the movement on the SM or the environment causes significant fluctuations in the time evolution of channel characteristics. In addition, if an adversary is at a distance greater than one wavelength, it is effectively measuring different spectrum and the access of shared measurements of the SM is not possible. Therefore, SM uses their channel measurements as a source of shared entropy for security benefits. Received power (P_r) is calculated by using the link-budget equation in which P_r is inversely proportional to path-loss (L_p).

$$L_p = (\frac{4\pi(di)}{\lambda})^2, \tag{1}$$

where di is the distance between two communicating SMs which is approximately 50 m. λ is representing the wavelength which is approximately 416 μ. According to Jakes model, if the adversary is half wavelength distance away, the variations in RSSI can be measured. If the distance is increased between communicating MICAz motes, the path loss will be increased and hence the received power will be decreased. As all the RSSI values are either negative or very small, a gain of 50 is given to make all the values positive. A link fingerprint (LF) of 8-bits is generated by quantizing the RSSI values. The link fingerprint is further encoded with a unique 8-bit secret key associated with the SM and a pseudo-random nonce (N_i). This nonce is clock synchronized at both the DCU and SM.

$$LF_{i(encoded)} = LF_i \oplus K_i \oplus N_i \quad \forall i \in \{1, \ldots, n\}, \tag{2}$$
$$N_{i(encoded)} = K_i \oplus N_i \quad \forall i \in \{1, \ldots, n\}. \tag{3}$$

In (2), $\oplus$ represents the logical exclusive-OR operator while in (3), $N_{i(encoded)}$ is the encoded nonce. The nonce being different at each time, the replay attack will not be possible as each time the $LF_{encoded}$ is different. The hash of 160 bits is generated by applying SHA-1 algorithm to the $LF_{encoded}$ and is added as a header to the data which is forwarded to the next SM. On receiving the packet containing both data and hash (as header), the SM adds its own hash as next header to the same packet and forwards it to the next SM if required. The hash values are used for data provenance, which is explained in section IV-D. $LF_{encoded}$ and $N_{i(encoded)}$ are sent to the DCU by each SM.

B. OPERATIONS AT THE DCU

The secret key associated with any SM is not shared with any other SM. Data is stored in DCU after the successful authentication. All keys defined for each SM are already present at the DCU which are K_1 and K_2 and so on. The first step at the DCU is to detach the encoded nonce and link fingerprint from the data it receives. It retrieves the nonce by using the key of SM from which the data is received. The decoding of the link fingerprint is performed by using nonce and secret key of the concerned SM.

$$N_i = K_i \oplus N_{i(encoded)} \quad \forall i \in \{1, \ldots, n\}. \tag{4}$$
$$LF_i = K_i \oplus LF_{i(encoded)} \oplus N_i \quad \forall i \in \{1, \ldots, n\}. \tag{5}$$

The binary codes of link fingerprints are converted to their respective values in dBm and various calculations are performed on the retrieved data from SMs.

1) MEASUREMENTS SUPPORTING STATIC SMS

A threshold is defined to check the fluctuations in RSSI values whether they are within limits or not. In (6), $P_{r(avg)}$ denotes the average of consecutive RSSI values which are checked against the predefined threshold.

$$P_{r(avg[i])} = \left|\frac{P_{r(i)} - P_{r(i+1)}}{2}\right| \quad \forall i \in \{1, \ldots, n\}. \tag{6}$$

$P_{r(i)}$ and $P_{r(i+1)}$ are the two consecutive RSSI values of SM. The synchronized difference (d) of two connected SMs are calculated using equation $P_{r[i](SM_1)} - P_{r[i](SM_2)}$. The average difference (d_{avg}) is calculated using the consecutive occurring values of d. The results computed are checked against predefined threshold. Mathematically,

$$d_{avg[i]} = \left|\frac{d_{(i)} - d_{(i+1)}}{2}\right| \quad \forall i \in \{1, \ldots, n\}. \tag{7}$$

2) MEASUREMENTS SUPPORTING MOBILE SMS

Besides the mechanism mentioned in section IV-B.1, the DCU also calculates the Pearson correlation coefficient (ρ) by using the information of link fingerprints which are converted to their respective values in dBm. Mathematically,

$$\rho_{X,Y} = \frac{C(X, Y)}{\sigma_X \sigma_Y}, \tag{8}$$

where, C represents the covariance and σ is the standard deviation. ρ returns a value between -1 and 1 in which 1 indicates perfect correlation, 0 indicates no correlation, and -1 indicates anti-correlation. Equation (8) can be further simplified as;

$$\rho = \frac{\sum_{i=1}^{n}(X_i - \bar{X})\sum_{i=1}^{n}(Y_i - \bar{Y})}{\sqrt{\sum_{i=1}^{n}(X_i - \bar{X})^2}\sqrt{\sum_{i=1}^{n}(Y_i - \bar{Y})^2}}. \tag{9}$$

X_i and Y_i are the RSSI values of the ith packet received at communicating SMs and $\bar{X}$ and $\bar{Y}$ are the respective mean RSSI values of a sequence of n packets.

C. ADVERSARY DETECTION

The detection of adversarial node or meter tempering is done at the DCU if the value of correlation coefficient is below a pre-defined threshold i.e., 0.8 without applying any filter and 0.9 by applying SavitzkyâĂ"Golay filter, or,

1) sudden rise in RSSI values of connected SMs, which remains there for a certain period of time,
2) besides 0 and 1, other values of $P_{r(avg)}$ are also computed,
3) the value of d does not remain constant,
4) variations in d_{avg}.

In the above mentioned matrices, if the value of ρ is above 0.8 but any among 1 to 4 is true then the malicious activity is observed and vice versa. This indicates that man-in-the middle attack or tempering of a SM has occurred.

The RSSI variations remain almost constant for stationary SMs when they are connected in Line of Sight (LoS) with each other. The communicating path is diverted if an eavesdropper comes in between the link and diverts the path. If a static SM is tempered, the DCU after decoding the link fingerprint gets random values of RSSI in dBm and hence gets variations in the difference. The average difference values have fluctuations as well. For SMs having mobility, the RSSI variations are linear with respect to the connected SM. The Pearson correlation coefficient (ρ) measures the correlation between the variations of two connected SMs. If the adversary has not affected the communication, then there is a linear relationship between the variations of SM with the one it is connected to. But if any adversary comes in between the communicating path of SMs or tempers the data at any SM, then the RSSI values will have non-linear relationship in terms of variations and hence a low value of ρ.

D. BLOCKCHAIN BASED DATA PROVENANCE

As discussed, the hash value is generated from the $LF_{encoded}$ of the concerned SM and is saved at the memory. Also, when data is transmitted from any SM to the next, hash value is inserted as a header to the data. The first header contains the hash of the last SM to which the data is received followed by the hash of the connected SM from which the packet is received and so on. As discussed previously, at DCU all the hash values are put in the hash table of each SM. If the origin of data needs to be confirmed, the headers are checked in order with all the saved hash values at the DCU. It is checked until all the header data is exhausted. The last matched hash value's table is observed and in whichever SM's table it is residing, is considered as the originating SM's data. If at any stage the hash value does not match, the malicious activity is assumed to be occurred in the AMI.

E. ALGORITHMS

The mechanism of generating link fingerprints and its encoding is represented in Algorithm 1. SMs read the RSSI values from its adjacent SMs. The RSSI values received are in negative dBm, whereas gain is given to make them positive. Each RSSI value is quantized and a link fingerprint is generated by assigning binary values to quantized levels. As these link fingerprints are the vital block to detect any adversary in the AMI, it is important to secure it by encoding it using the secret key and a pseudo-random nonce. SHA-1 algorithm is also applied on the link fingerprint which is inserted as a header to the payload and sent to the adjacent SM. The pseudo-random nonce is encoded as well with the secret key of the SM. The encoded link fingerprint along with encoded nonce is sent to the DCU.

The detection of any adversary activity is done at the DCU. Algorithm 2 shows the detection of adversary in the AMI. Firstly, the encoded nonce received at the DCU is decoded using the key associated with the SM from which it is received. This decoded nonce is used along with the same key to decode the link fingerprint. SHA-1 algorithm is applied to the link fingerprint and the hash value is stored at the memory of the DCU which is used for securing data provenance. The link fingerprints are converted to their respective decimal values in dBm. The averages of consecutive RSSI values received from the same SM are calculated. The same is done for all the SMs in AMI. The average of difference in RSSI values of connected SMs are measured as well. Besides that, pearson correlation coefficient is also calculated using RSSI values of two connected SMs. First of all, the value of pearson correlation coefficient is observed. If it is between 0.9 and 1 then this represents the absence of adversarial node in the AMI. But if its value is less than the empirical value 0.9 then the values of average RSSI and difference are observed. If the average of RSSI or the average of difference values is 0 or 1 then no adversarial node is detected in the AMI.

Algorithm 3 represents the data provenance using hash values described in Algorithms 1 and 2. A packet received at any SM contains all the hash values as header. These headers are detached in order and the hash values are compared with all the hash values present at the DCU memory. If a hash that matches with the hash of the last header then the table in which it is residing is the last SM from which the packet is received. But if any hash does not match then the data will be considered as forged in the previous link.

V. RESULTS AND DISCUSSIONS

This section is divided into four main subsections, i.e. Experimental results, simulation based results, comparisons with other security algorithms and energy requirements for securing AMI. They are as follows.

A. EXPERIMENTAL SETUP

The results are derived by achieving RSSI values using MICAz motes in an open environment. MICAz motes are tiny wireless measurement systems having the operating frequency of 2.4 GHz to 2.48 GHz with an IEEE 802.15.4 standard. They have embedded RAM of 4 KB, program flash memory of 128 KB and 512 KB of serial flash memory.

Algorithm 1 Generation of Link Fingerprints and Hash Functions at the Smart Meter

Input : Received RSSI values
Output: Encoded link fingerprints and nonce

Initialize the Smart Meter;
Reception of RSSI values from connected SMs;
$RSSI_{new}[i] \leftarrow RSSI[i] + 50$;
Quantize $RSSI_{new}[i]$;
LinkFingerprint[i] $\leftarrow$ *Assign binary code-word to Quantized* $RSSI_{new}[i]$;
$RSSI_{encoded}[i] \leftarrow XOR(LinkFingerprint[i], Key_{SM(a)}, N_i)$;
$PacketHeader[i] \leftarrow hash(RSSI_{encoded}[i])$;
$N_{i(encoded)} \leftarrow XOR(Key_{SM(a)}, N_i)$;
$RSSI_{encoded}[i]$ *bundled up with session identifiers*;
Send Packet to the connected SM;
Send $RSSI_{encoded}[i]$ *and* $N_{i(encoded)}$ *to the DCU*;

Algorithm 2 Adversary Detection at the DCU

Input : Encoded link fingerprints and nonce
Output: Pearson Correlation Coefficient, $RSSI_{avg}$, difference value

$N_i \leftarrow XOR(Key_{SM(a)}, N_{i(encoded)})$;
$LinkFingerprint[i] \leftarrow XOR(RSSI_{encoded[i]}, Key_{SM(a)}, N_i)$;
$Memory_{DCU}[i] \leftarrow hash(RSSI_{encoded}[i])$;
$RSSI_{new[i]} \leftarrow$ *bin-dec conversion(LinkFingerprint[i])*;
$N_j \leftarrow XOR(Key_{SM(b)}, N_{j(encoded)})$;
$LinkFingerprint[j] \leftarrow XOR(RSSI_{encoded[j]}, Key_{SM(b)})$;
$Memory_{DCU}[j] \leftarrow hash(RSSI_{encoded}[j])$;
$RSSI_{new[j]} \leftarrow$ *bin-dec conversion(LinkFingerprint[j])*;
$RSSI_{avg_i} \leftarrow abs[(RSSI_{new[i]}\text{-}RSSI_{new[i+1]})/2]$;
$RSSI_{avg_j} \leftarrow abs[(RSSI_{new[j]}\text{-}RSSI_{new[j+1]})/2]$;
$difference_{[k]} \leftarrow abs[RSSI_{new[i]}\text{-}RSSI_{new[j]}]$;
$difference_{avg[k]} \leftarrow abs[difference_{[k]}\text{-}difference_{[k+1]}/2]$;
Correlate $RSSI_{new[i]}$ *and* $RSSI_{new[j]}$;
if $0.9 < correlation \leq 1$ **then**
    No adversarial node is detected;
    else if $0 \leq RSSI_{avg_i}$ *or* $RSSI_{avg_j} \leq 1 \;||\; 0 \leq difference_{new[k]} \leq 1$ **then** ;
    No adversarial node is detected;
    else Adversarial node is detected;

Algorithm 3 Data Provenance Using the Hash Functions Present as a Packet Header. i and j Represent the Headers Attached to the Payload Received at the Last SM and All the Hash Values Present at the DCU, Respectively

Input : Packet headers
Output: Origin of the packet or intrusion detection

for $i \leftarrow n$ **to** 1 **do**
    // n is the last IoT node the packet is received at;
    for $j \leftarrow n$ **to** 1 **do**
        if $Header_i == Hash\_Memory_{DCU}[j]$ **then**
            $origin = SM[i-1]$;
            else adversary between $SM[i]$ and $SM[i-1]$;

They are designed specifically for embedded wireless sensor networks. A single MICAz mote provides a data rate of 250 kbps. They have wireless communication capabilities with each other and each node also provides a routing path to communicate with any other mote if two motes are out of range in the same network. MICAz mote can be expanded by attaching a sensor chip to sense various factors e.g., light, temperature, barometric pressure, acceleration etc. from the environment. MPR2400CA is the radio and processor platform of MICAz mote. The MPR2400 is based on the Atmel ATmega128L which is a low power microcontroller. The expansion of 51 pins connector supports analog inputs, digital inputs and outputs and UART interfaces. Because of these interfaces it becomes easy to connect it to a wide variety of external peripherals. The MICAz radio offers both high speed data rate and an AES-128 hardware security. MICAz motes have a range of 20 m to 30 m in indoor environment while 70 m to 90 m in outdoor for communication purposes [26].

In order to perform the experiment, the infrastructure of electricity distribution provided by Peshawar Electric Supply Company (PESCO) is used. The conventional meters are attached to the electricity poles approximately 50 meters apart. The MICAz mote are placed on the top of these meters as shown in Figure 2 and are considered as SMs because of the communication capabilities added up by MICAz motes. The SMs are initially considered static and then mobile (depends on the nature of experiment carried out). In case when both SMs and adversary are stationary, we have considered directional beam instead of omni-directional beam. RSSI values are recorded at the base station placed in the middle of two communicating SMs. The base station is attached to a computer having LINUX operating system and TinyOS is programmed to record RSSI values. MICAz mote is also used as an adversary and is placed in between the established communicating path. As most of the values that we received were in negative dBm, a gain of 50 was given to bring them in positive range so they become more easily readable.

B. SIMULATION RESULTS

The RSSI values received from SMs and adversarial nodes are simulated on MATLAB 2017Ra. Simulations are performed for various scenarios which are discussed below.

1) NO ADVERSARIAL NODE IN THE AMI

Two cases are considered and results are achieved. The cases are as under:

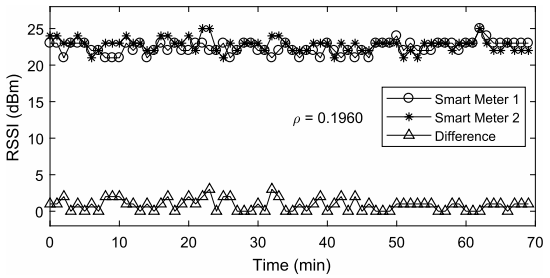

FIGURE 3. No adversarial node in between two connected static SMs. The RSSI values remain almost constant.

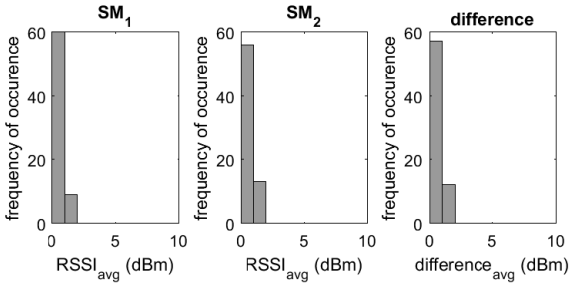

FIGURE 4. The frequency of occurrence of average RSSI and difference showing all values either 0 or 1.

Case 1 (Static SMs in the AMI): When all SMs are static and there is no mobility then the RSSI values recorded show a very constant pattern of RSSI as shown in Figure 3. No or very little fluctuations are observed. The difference between RSSI values of both SMs are also almost constant throughout. Either 0 or 1 is computed as the average of consecutive RSSI and consecutive difference values because there is no or very little change in RSSI values at each SM. No abrupt change or fluctuations in RSSI and difference values are observed. Figure 4 shows that no other values are computed other than 0 and 1 as average RSSI and difference in 70 samples taken. The value of ρ is 0.1960, which shows the uncorrelation in the RSSI patterns of both SMs but the previous results prove that there is no involvement of the adversary in the AMI.

Case 2 (At Least One SM Is Mobile in the AMI): There are cases in AMI in which one of the two connected SMs or both SMs are mobile. Our experiment is performed when a static SM is connected to a mobile SM (mobile node). The result achieved is shown in Figure 5. The relations in variations of RSSI show linear relationship. The ρ value computed is 0.8378 which shows the linear relationship between the RSSI of two connected SMs. The results are further improved by applying Savitzky-Golay filter which smooths the data and increases the precision of the data without distorting the signal tendency as shown in Figure 6. The value of ρ is improved to 0.9102. Though we do not get the desired results for average RSSI and difference which is presented in Figure 7. Our proposed protocol defines that if ρ is below the threshold then we look for other values to decide whether or not any adversary is present in the AMI. Hence, the results show that there is no adversary present in the AMI.

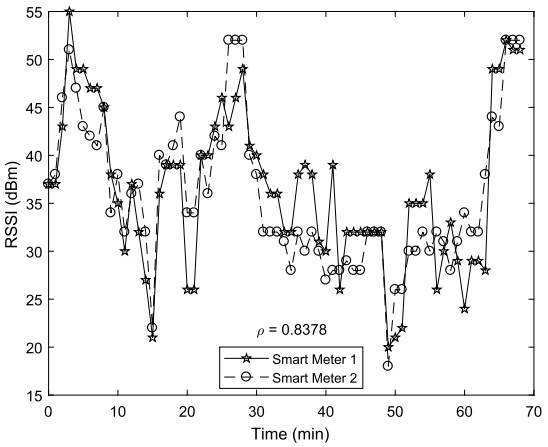

FIGURE 5. No adversarial node in between two connected mobile SMs. The RSSI values show linear relationship with each other and hence a high value of ρ.

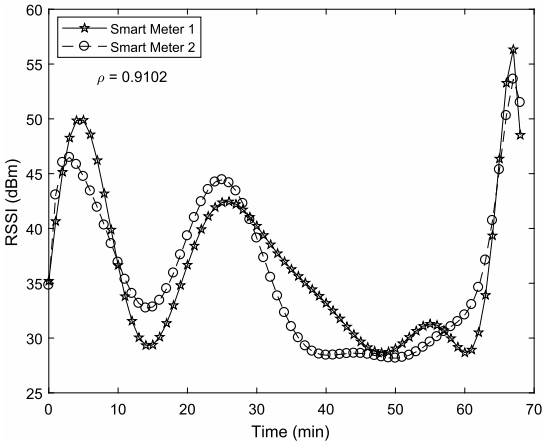

FIGURE 6. Savitzky-Golay filter applied to further smooth out the RSSI variations of two connected mobile SMs presented in Figure 5.

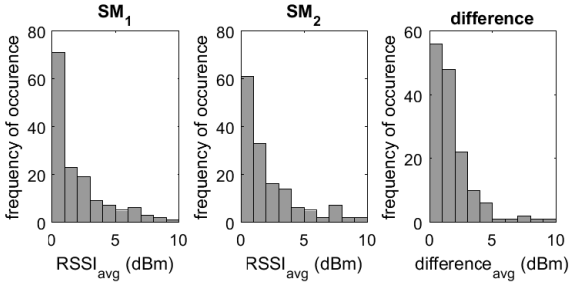

FIGURE 7. Frequency of occurrence of average RSSI and difference showing that along with 0 and 1, other values are also computed.

2) ADVERSARIAL NODE IN AMI

Case 1 (A Static Adversarial Node in Between Two Static SMs): When two static SMs communicate with each other the RSSI values remain almost constant. But when an adversarial node comes in between the communication path, now the link changes to $SM_1 \rightarrow$ adversarial node $\rightarrow SM_2$. We have used directional beams in this particular case because if man-in-the-middle attack is carried out at the distance equals to the original distance of SM_1 and SM_2 then adversary will not be detected. When adversary changes the

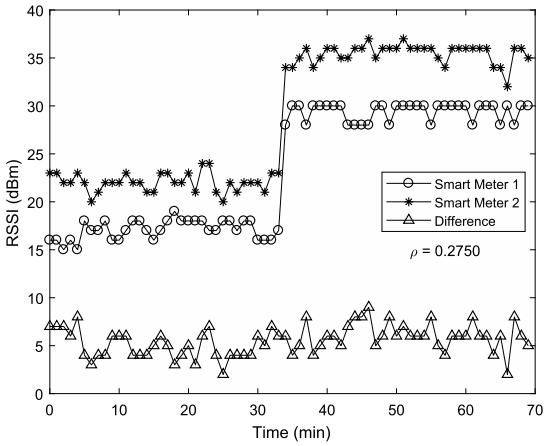

FIGURE 8. Static adversary in between two communicating SMs. An abrupt change in RSSI values are observed showing that communication link is now via adversarial node.

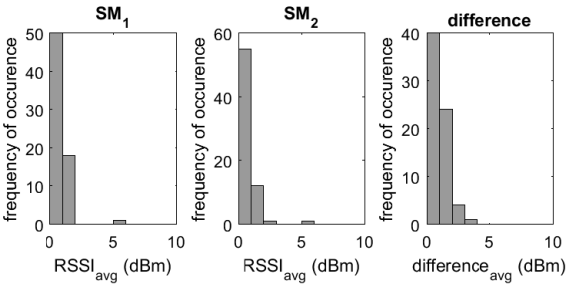

FIGURE 9. Adversarial node in between two communicating node causing the computation of other values other than 0 and 1 as well.

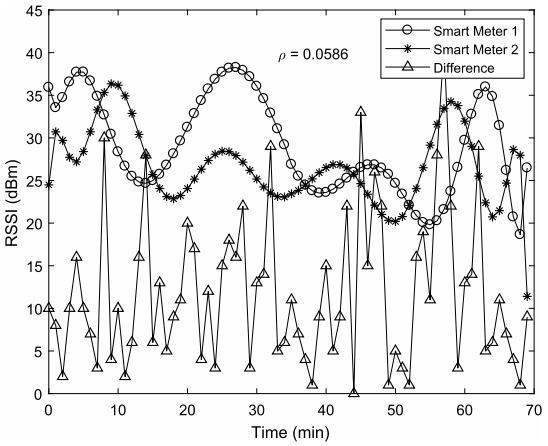

FIGURE 10. Mobile adversary in between two communicating SMs. A non-linear relation between RSSI values of both SMs are observed and hence a low value of ρ.

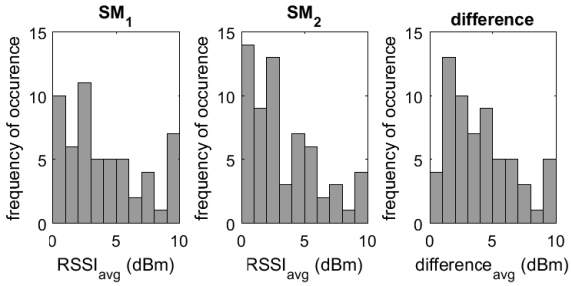

FIGURE 11. Computed values of average RSSI and difference showing a spread in values.

communication path, a sudden up rise in RSSI values are observed as shown in Figure 8. This jump at a certain point witnesses the presence of adversarial node in between. The difference values do not witness much change as RSSI values of both communicating SMs go higher at the same time. As discussed in Section IV, the adversarial node is present in AMI if the average of two consecutive RSSI values is computed other than 0 or 1. In Figure 9, it is observed that a value "5" has also occurred once in 70 samples showing the disturbance in AMI and hence the occurrence of adversarial node in the AMI. The value of ρ is also below the threshold i.e. 0.2750.

Case 2 (A Mobile Adversarial Node in Between Two Static SMs): The experiment is simulated for a mobile adversarial node in between two communicating static SMs. As the adversarial node moves in between the static path, the RSSI variations are not constant and hence the linear relationship between the RSSI variations are not achieved as shown in Figure 10. This results in a nearly equal to zero value of ρ which is 0.0586. Besides that, Figure 11 clearly shows that the average of RSSI values of each SM and average difference of the RSSI have the repeated other values along with zeros and ones.

3) DATA TEMPERING AT SM

As SMs in AMI are physically unprotected, that is why the cases of data tempering cannot be neglected. The experiments performed and results achieved in this regard are presented below as cases.

Case 1 (Data Is Tempered When Both SMs Are Static): The packet data is changed when an SM is tempered. The data contains the information of RSSI value which is changed. In this case, we have tempered the data at SM_1 and is sent to DCU for computations. The results show that the DCU retrieves the random values of RSSI. In Figure 12, it can be seen that the variations in RSSI values of SM_1 are observed while the untempered SM_2 shows almost constant values of RSSI. This represents that the tempering is performed at SM_1. By looking at the bar graphs in Figure 12, it is observed that $RSSI_{avg}$ of SM_1 and difference values are spread all over witnessing the involvement of adversary in the AMI. Besides that, ρ is achieved as -0.0117 which shows the uncorrelation in RSSI variations.

Case 2 (Data Is Tempered at Static SM While Communicating With a Mobile SM): When data is tempered at the static SM node which is communicating with a mobile SM, link fingerprint is also altered which is sent to the DCU. In our case, SM_1 is tempered and can be seen in Figure 13 that instead of getting static RSSI values, we get variations in RSSI. Comparing the relationship of variations of SM_1 and SM_2, it is obvious that both SMs show nonlinear RSSI pattern. Due to the mentioned reason ρ = -0.1267 is computed.

TABLE 1. All hash functions stored as log files at the DCU. The hash functions present as the head of packet for both cases (with and without adversary's involvement) are presented as well.

SM1 → SM2	SM2 → SM1	SM2 → SM3	SM3 → SM2
25087409E0AADBAB4FE8AC81D90D0EC0895C0F58	FB8967D0546886146E9088F8CC60DEC9995AAE20	540C52FD83B686BF98B5EF71D20477A10FABCE02	03A57989041B385D6DB81D9A0A9AEFF3477DA1DF
AB83E8598A30DA4D5D1C98B9C4B4824F8A2AEF62	1BA130A1EE0151BCE772F3D86294A80366DD1CDE	4D761D700F03AAEE5B49EC95CA445DB62E787FC2	FCEB722DB2E571631O4E3BA948EF27A95867CA85
FC4479DCF2ADB329EDFBA7B0CAE61D39AA5FC3BF	4BBA6AB29B156A3419C65832BB06808ABC32E435	46E997961D341AF79D95374CE9B207CA2BCC9FD7	18653C7EBE69E719360C6E7A3BF3D8D8932C6F5E
4DBF9734C332F43E1C7930EED9EDC5E2E6124922	1991AA67A6F0BC2448733C22D9966073F81AC85D	3FFA8D1C96C867223D73BAD6398D20D632AEDCB5	94E18CFE6DE9DEE7CA92BB91BD7D6DE0B8EA7AF5
Packet Header at SM3: 18653C7EBE69E719360C6E7A3BF3D8D8932C6F5E4D761D700F03AAEE5B49EC95CA445DB62E787FC21BA130A1EE0151BCE772F3D86294A80366DD1CDE25087409E0AADBAB4FE8AC81D90D0EC0895C0F58			
Packet Header at SM3: 18653C7EBE69E719360C6E7A3BF3D8D8932C6F5E4D761D700F03AAEE5B49EC95CA445DB62E787FC21BA130A1EE0151BCE772F3D86294A80366DD1CDE~~FB8F0211382DDFF69E7000BD8664B69195619650~~			

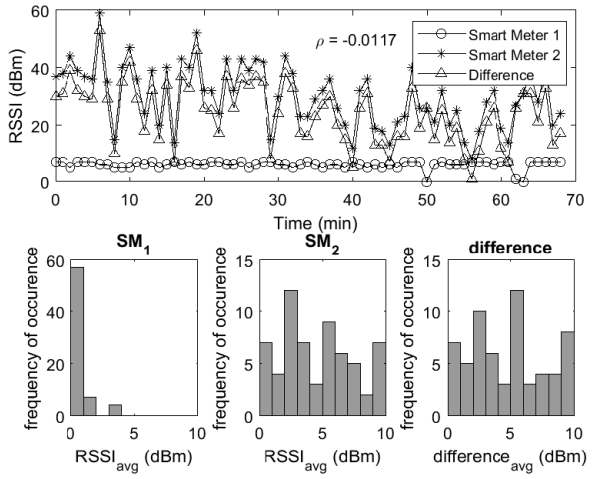

FIGURE 12. When data at one of the two communicating static SMs is forged. The resulting plots show variations in RSSI of forged SM. Besides 0 and 1, other values are also computed as $RSSI_{avg}$ and $difference_{avg}$ which shows the involvement of adversary in the AMI.

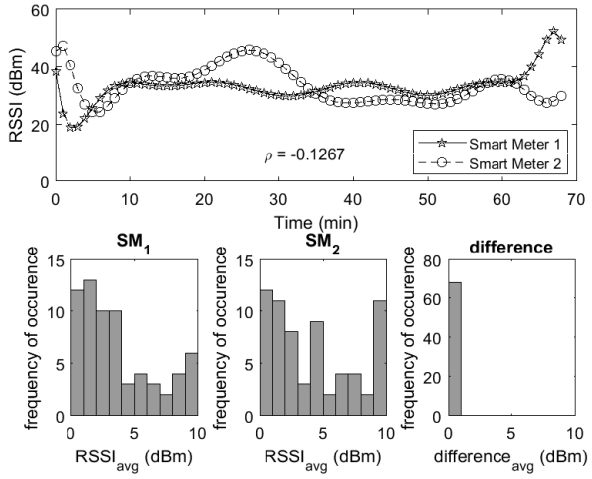

FIGURE 13. When data at one of the two communicating mobile SMs is forged. The resulting plots show variations in RSSI of forged SM. Besides 0 and 1, other values are also computed as $RSSI_{avg}$ which shows the involvement of adversary in the AMI.

Also, the values are spread all over in bar graphs for average RSSI values of SM_1 and SM_2.

4) DATA PROVENANCE USING BLOCKCHAIN

Data provenance is achieved and presented for two cases.

Case 1 (No Tempering of Data): In this case, data is initiated from SM_1 and receives at SM_3 via SM_2 as shown in Figure 1. SM_1 adds 160 bits hash value that it generates using secret key, link fingerprint and pseudo-random nonce as the header of the packet and sends it to SM_2. SM_2 upon receiving it adds two hash values to the header, one of link $SM_2 \rightarrow SM_1$ and the other one of link $SM_2 \rightarrow SM_3$. The total header of 320 bits are added by SM_2 to the packet and forward it to SM_3. SM_3 adds the hash of its link with SM_2. The packet is checked for the origin of data. As discussed in IV that the DCU keeps all the hash values in its memory in a table for each link. Table 1 shows the first ten hash values stored for each link. The packet received at SM_3 is also shown in Table 1. In case of hexadecimal values, first 40 digits are retrieved first and compared with the hash values of the tables present at the memory of DCU. It can be seen in blue color of the table that it has 100% match in column representing all the hash values of link $SM_3 \rightarrow SM_2$. The next 40 hexadecimal digits are checked against the tables containing SM_2 which are $SM_2 \rightarrow SM_3$ and afterwards $SM_2 \rightarrow SM_1$ whose match is represented in red and violet colors, respectively. Finally, the hash values at $SM_1 \rightarrow SM_2$ table at the DCU is scanned and a match is found which is shown in purple color. All the header digits are exhausted which shows that the origin of this packet is SM1. It is observed that in all the cases we get 100% match of hash values of the header with the hash values present at the DCU. The interesting fact is that all the hash values are unique because of the involvement of pseudo-random nonce.

Case 2 (Packet Is Tempered at SM): In this case, the data is forged at SM_1. The forged data is sent to SM_2 which forwards it to SM_3. The process is the same as described in case 1 but as shown in Table 1, the last 40 hexadecimal header digits do not correlate with any hash value in the table. It is represented in bold with strikethrough hash value. As the rest of hash values correlated except the last one, which is supposed to be present at $SM_1 \rightarrow SM_2$ table at the DCU, shows that data is tempered at SM_1.

C. COMPARISON WITH EXISTING SECURITY ALGORITHMS

The proposed security algorithms presented in this paper are compared with other state-of-the-art protocols shown in Table 2. The comparisons are made for various cyber attacks and data provenance. All other algorithms do not provide data provenance along with defense against security threats except protocol proposed in [9]. Gope and Sikdar [27] provides a lightweight security algorithm but the proposed algorithm is not resilient against data tempering and location proximity. Dong *et al.* [28] has used the round trip time (RTT)

TABLE 2. Comparison with existing security algorithms.

Security Requirements	Gope et al. [27]	Dong et al. [28]	Ali et al. [9]	Kamal et al. [6]	Hussain et al. [29]	Proposed Protocol
MITM attack	✓	✓	✓	✓	✓	✓
Jamming	✓	✗	✓	✓	✗	✓
Data Tempering	✗	✗	✓	✓	✓	✓
Replay attack	✓	✗	✗	✓	✓	✓
Location Proximity	✗	✗	✗	✓	✗	✓
Data Provenance	✗	✗	✓	✗	✗	✓

TABLE 3. Energy consumption on forwarding data to DCU considering three SMs in AMI.

SM (1,2,3)	Fingerprint (bytes)	Nonce (bytes)	Transmission Cost (μJ)	AES-128 (μJ)	SHA-1 (μJ)	ECDSA-160 (mJ)	Total (mJ)
1	16	1	81.6	1.944	327.25	52	52.410
2	32	2	163.2	3.888	654.5	104	104.82
3	16	1	81.6	1.944	327.25	52	52.410
Total Energy dissipated when forwarding data to the DCU							209.64

TABLE 4. Energy consumption on forwarding hash as packet header to connected SM considering three SMs in AMI.

SM (1,2,3)	Hash (bytes)	Transmission Cost (μJ)	AES-128 (μJ)	SHA-1 (μJ)	ECDSA-160 (mJ)	Total (mJ)
1	20	96	2.287	385	52	52.483
2	60	288	6.861	1155	52	53.449
3	80	384	9.148	1540	52	53.933
Total Energy dissipated when packet has reached to SM_3						159.865

to detect man-in-the-middle attack. The localization of adversarial node and defense against reply attack is not provided by Ali et al. in [9]. They have considered a single hop network. If the network is extended then the proposed algorithm will overcome these problems. In [6], a lightweight solution is provided but the provenance is not added to the system. The origin of the packet received at any SM cannot be determined. The algorithm is proposed to counter node replication, replacement, and man-in-the-middle attacks in [29]. They have used MICAz and TelosB sensor nodes to carry out their experiments.

D. ENERGY REQUIREMENTS

SMs require energy to transfer the security related information to the DCU and also to forward data packet, which includes the hash values as header to the next SM. As MICAz motes are used as the module for communication in the AMI, energy requirements are calculated by looking at the data sheet of MICAz motes. For MICAz motes, the transmission cost of one bit is 0.6 μJ, energy dissipation by applying Elliptic Curve Digital Signature Algorithm (ECDSA) having public key of 160 bits is 52 mJ [30]. Also, the energy requirements for Advanced Encryption Standard (AES-128) and generating hash of 64 bits are 1.83 μJ [31] and 154 μJ [32], respectively.

1) FORWARDING DATA TO THE DCU

The energy calculations are performed by considering three SMs in the AMI. SM_1 sends the encoded link fingerprint of 128-bits along with an 8-bit long nonce to the DCU. The total energy consumed by SM_1 is calculated as 52.410 mJ as shown in Table 3. As SM_2 is connected to SM_1 and SM_3, it has to do double effort by sending the information related to both links to the DCU. Hence the dissipated energy is double compared to SM_1 i.e., 104.82 mJ. SM_3 works just like SM_1 as it is connected to SM_2 only and the same amount of energy is consumed. The total energy required for the AMI having three SMs is 209.64 mJ. Table 3 shows energy consumed at each step.

2) FORWARDING PACKET TO THE SM

Table 4 shows the energy requirements by each SM when forwarding the packet to the next SM. It also shows the total energy requirements of the AMI having three SMs. As discussed in section IV, SM only attaches the hash values it computes to the packet as header and forwards it to the next SM. SM_1 is connected to SM_2 only, a 160 bits long header is attached to the packet as header and is sent to SM_2. SM_2 on receiving it attached two headers having hash values of link SM_2 to SM_1 and SM_2 to SM_3, which makes the total length of header equals 480 bits. SM_3 adds its own header of link SM_3 to SM_2 of 160 bits, which makes a total header size of 640 bits. The energy required are 52.483 mJ, 53.449 mJ and 53.933 mJ for SM_1, SM_2 and SM_3, respectively. The total energy required for the AMI having three SMs for this case is 159.865 mJ.

3) TOTAL ENERGY REQUIREMENTS FOR AMI

The total energy required for AMI to carry out operations at the SMs is 370.955 mJ as shown in Table 5. All the energy values are added together to compute the energy requirements of the SMs.

4) COMPARISON WITH OTHER PROTOCOLS

The algorithm proposed in this paper is compared with the various protocols proposed by Ali *et al.* [9]. It can be seen that the energy requirements of our system is less, comparatively. The reason is that the link-fingerprint has less size and

TABLE 5. Energy dissipated by each SM and by the whole AMI.

SM (1,2,3)	Energy Dissipated (mJ)
1	104.893
2	158.269
3	106.343
AMI Network	370.955

TABLE 6. Energy dissipation comparison.

System	Packet (Bytes)	Energy Dissipated (mJ)
Our (forwarding to DCU)	17	52.410
Our (forwarding to SM)	80	53.933
Level Crossing [9]	52	53.305
Ranking [9]	598	66.450
Raw RSSI [9]	2292	109.801

provides maximum security for AMI. The comparisons are shown in Table 6.

VI. CONCLUSION

Due to scalability and low computational capabilities of SMs, light-weight security algorithms are the requirements for AMI. In this paper, the RSSI variation pattern of two connected SMs are used to generate link fingerprints. These RSSI values are highly correlated when any of the two SMs or both are in motion. In case of static SMs, the RSSI values at both communicating SMs remain almost constant. Introducing an adversarial node in between two connected SMs produces different results for both static or mobile SMs. A pseudo-random nonce is used to generate hash values, which are unique. Hash values are inserted as packet header and forwarded to the next SM. The data provenance is achieved using blockchain technology by comparing all hash values with hash functions present at the DCU. We get light-weight solution for the security of AMI by introducing only 20 byte header for data provenance and a 16 byte of link fingerprint. The header size can be increased or decreased depending on the computational capabilities of the SM. Simple encryption technique is used instead of any cryptographic solution for the information sent to the DCU for intrusion detection. SHA-1 algorithm is applied for data provenance, which is one of the most light-weighted cryptographic hash functions. In addition, no extra hardware needs to be added to the system in order to detect the adversarial node or data forging. Only changes in the program are required in the already existing SMs and DCUs to detect any unethical activity.

REFERENCES

[1] M. Ismail, M. Shahin, M. F. Shaaban, E. Serpedin, and K. Qaraqe, "Efficient detection of electricity theft cyber attacks in AMI networks," in *Proc. IEEE Wireless Commun. Netw. Conf. (WCNC)*, Apr. 2018, pp. 1–6.

[2] F. Jamil and E. Ahmad, "An empirical study of electricity theft from electricity distribution companies in Pakistan," *Pakistan Develop. Rev.*, vol. 53, no. 3, pp. 239–254, 2014.

[3] C.-H. Ke, S.-Y. Hsieh, T.-C. Lin, and T.-H. Ho, "Efficiency network construction of advanced metering infrastructure using zigbee," *IEEE Trans. Mobile Comput.*, vol. 18, no. 4, pp. 801–813, Apr. 2019.

[4] S. Das, Y. Ohba, M. Kanda, D. Famolari, and S. K. Das, "A key management framework for AMI networks in smart grid," *IEEE Commun. Mag.*, vol. 50, no. 8, pp. 30–37, Aug. 2012.

[5] M. Tariq and H. V. Poor, "Electricity theft detection and localization in grid-tied microgrids," *IEEE Trans. Smart Grid*, vol. 9, no. 3, pp. 1920–1929, May 2018.

[6] M. Kamal and M. Tariq, "Light-weight security for advanced metering infrastructure," in *Proc. 89th Veh. Technol. Conf. (VTC-Spring)*, 2019, pp. 110–115.

[7] X. Xia, Y. Xiao, and W. Liang, "ABSI: An adaptive binary splitting algorithm for malicious meter inspection in smart grid," *IEEE Trans. Inf. Forensics Security*, vol. 14, no. 2, pp. 445–458, Feb. 2019.

[8] J. Wu, M. Dong, K. Ota, M. Tariq, and L. Guo, "Cross-domain fine-grained data usage control service for industrial wireless sensor networks," *IEEE Access*, vol. 3, pp. 2939–2949, 2015.

[9] S. T. Ali, V. Sivaraman, D. Ostry, G. Tsudik, and S. Jha, "Securing first-hop data provenance for bodyworn devices using wireless link fingerprints," *IEEE Trans. Inf. Forensics Security*, vol. 9, no. 12, pp. 2193–2204, Dec. 2014.

[10] M. N. Aman, K. C. Chua, and B. Sikdar, "Secure data provenance for the Internet of Things," in *Proc. 3rd ACM Int. Workshop IoT Privacy, Trust, Secur.*, 2017, pp. 11–14.

[11] M. Singh, A. Singh, and S. Kim, "Blockchain: A game changer for securing IoT data," in *Proc. IEEE 4th World Forum Internet Things (WF-IoT)*, Feb. 2018, pp. 51–55.

[12] M. Kamal and M. Tariq, "Light-weight security and data provenance for multi-hop Internet of Things," *IEEE Access*, vol. 6, pp. 34439–34448, 2018.

[13] Y. Yan, R. Q. Hu, S. K. Das, H. Sharif, and Y. Qian, "An efficient security protocol for advanced metering infrastructure in smart grid," *IEEE Netw.*, vol. 27, no. 4, pp. 64–71, Jul. 2013.

[14] P. Jokar, N. Arianpoo, and V. C. M. Leung, "Electricity theft detection in AMI using customers' consumption patterns," *IEEE Trans. Smart Grid*, vol. 7, no. 1, pp. 216–226, Jan. 2016.

[15] A. H. Nizar, Z. Y. Dong, and Y. Wang, "Power utility nontechnical loss analysis with extreme learning machine method," *IEEE Trans. Power Syst.*, vol. 23, no. 3, pp. 946–955, Aug. 2008.

[16] A. Jindal, A. Dua, K. Kaur, M. Singh, N. Kumar, and S. Mishra, "Decision tree and SVM-based data analytics for theft detection in smart grid," *IEEE Trans. Ind. Informat.*, vol. 12, no. 3, pp. 1005–1016, Jun. 2016.

[17] M. Ibrahim, M. Kamal, O. Khan, and K. Ullah, "Analysis of radix-2 decimation in time algorithm for FPGA co-processors," in *Proc. Int. Conf. Comput., Electron. Elect. Eng. (ICE Cube)*, Apr. 2016, pp. 154–157.

[18] M. Kamal, M. Ibrahim, S. Mir, and M. N. Aman, "Comparison of multihop relaying protocols in cognitive radio networks," in *Proc. 6th Int. Conf. Innov. Comput. Technol. (INTECH)*, Aug. 2016, pp. 611–616.

[19] R. Ko and J. Mickens, "DeadBolt: Securing IoT deployments," in *Proc. Appl. Netw. Res. Workshop*, 2018, pp. 50–57.

[20] Z. Zhou, C. Gao, C. Xu, Y. Zhang, S. Mumtaz, and J. Rodriguez, "Social big-data-based content dissemination in Internet of vehicles," *IEEE Trans. Ind. Informat.*, vol. 14, no. 2, pp. 768–777, Feb. 2018.

[21] Z. Zhou, K. Ota, M. Dong, and C. Xu, "Energy-efficient matching for resource allocation in D2D enabled cellular networks," *IEEE Trans. Veh. Technol.*, vol. 66, no. 6, pp. 5256–5268, Jun. 2017.

[22] Z. Zhou, H. Yu, C. Xu, Y. Zhang, S. Mumtaz, and J. Rodriguez, "Dependable content distribution in D2D-based cooperative vehicular networks: A big data-integrated coalition game approach," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 3, pp. 953–964, Mar. 2018.

[23] A. Alketbi, Q. Nasir, and M. A. Talib, "Blockchain for government services—Use cases, security benefits and challenges," in *Proc. 15th Learn. Technol. Conf. (L&T)*, Feb. 2018, pp. 112–119.

[24] S. Huh, S. Cho, and S. Kim, "Managing IoT devices using blockchain platform," in *Proc. 19th Int. Conf. Adv. Commun. Technol. (ICACT)*, Feb. 2017, pp. 464–467.

[25] R. K. Bhatia and V. Bodade, "Defining the framework for wireless-AMI security in smart grid," in *Proc. Int. Conf. Green Comput. Commun. Elect. Eng. (ICGCCEE)*, Mar. 2014, pp. 1–5.

[26] *MICAz Wireless Measurement System*, Crossbow Technology, Milpitas, CA, USA, 2007.

[27] P. Gope and B. Sikdar, "Lightweight and privacy-preserving two-factor authentication scheme for IoT devices," *IEEE Internet Things J.*, vol. 6, no. 1, pp. 580–589, Feb. 2019.

[28] Z. C. Dong, R. Espejo, Y. Wan, and W. Zhuang, "Detecting and locating man-in-the-middle attacks in fixed wireless networks," *J. Comput. Inf. Technol.*, vol. 23, no. 4, pp. 283–293, 2015.

[29] S. Hussain and M. S. Rahman, "Using received signal strength indicator to detect node replacement and replication attacks in wireless sensor networks," *Proc. SPIE*, vol. 7344, Apr. 2009, Art. no. 73440G.
[30] G. de Meulenaer, F. Gosset, F.-X. Standaert, and O. Pereira, "On the energy cost of communication and cryptography in wireless sensor networks," in *Proc. IEEE Int. Conf. Wireless Mobile Comput., Netw. Commun.*, Oct. 2008, pp. 580–585.
[31] J. Lee, K. Kapitanova, and S. H. Son, "The price of security in wireless sensor networks," *Comput. Netw.*, vol. 54, no. 17, pp. 2967–2978, 2010.
[32] C.-C. Chang, S. Muftic, and D. J. Nagel, "Measurement of energy costs of security in wireless sensor nodes," in *Proc. 16th Int. Conf. Comput. Commun. Netw.*, Aug. 2007, pp. 95–102.

MOHSIN KAMAL (M'16) received the B.S. degree in telecommunication engineering from the National University of Computer and Emerging Sciences at Peshawar, Peshawar, Pakistan, in 2008, and the M.S. degree in electrical engineering from Blekinge Tekniska Högskola, Karlskrona, Sweden, in 2012. He is currently pursuing the Ph.D. degree in electrical engineering with the National University of Computer and Emerging Sciences at Peshawar.

Since 2013, he has been an Assistant Professor with the National University of Computer and Emerging Sciences at Peshawar, where he has been the IEEE Student Branch Counselor, since 2016. His research interests include the development of lightweight solutions for the various IoT applications, wireless sensor networks, cooperative communication, and cognitive radio networks. He has also been elected as the Secretary of the IEEE Peshawar Sub-Section, since 2019.

MUHAMMAD TARIQ (S'08–M'12–SM'17) received the M.S. degree from Hanyang University, South Korea, as an HEC Scholar, and the Ph.D. degree, as a Japanese Government (MEXT) Scholar, from Waseda University, Japan, in 2012. He completed his postdoctoral research at Princeton University as a Fulbright Scholar under the supervision of Prof. H. V. Poor, in 2016. He was the Head of the Department of Electrical Engineering, FAST National University of Computer and Emerging Sciences (NUCES) at Peshawar, where he is currently the Director. He has authored/coauthored over 50 research articles. He received many awards for his work. He has coauthored a book on smart grids with leading researchers from Europe, China, Japan, and USA, which was published by John Wiley and Sons, in 2015. He has presented his research work in various IEEE flagship conferences held around the world. He rendered his technical committee services in various IEEE flagship conferences and transactions. In 2017, Chinese Government selected him as a High-End Foreign Expert through the International Cooperation Project funded by the State Administration of Foreign Experts Affairs China. He has delivered research talks as a Guest/Invited/Keynote Speaker at various forums and universities in Pakistan, China, Saudi Arabia, and USA. He is the Program Evaluator (PEV) of the Pakistan Engineering Council.