Computers, Materials & Continua
DOI:10.32604/cmc.2021.014035
Article

Tech Science Press

Authenblue: A New Authentication Protocol for the Industrial Internet of Things

Rachid Zagrouba[1,*], Asayel AlAbdullatif[1], Kholood AlAjaji[1], Norah Al-Serhani[1], Fahd Alhaidari[1], Abdullah Almuhaideb[2] and Atta-ur-Rahman[2]

[1]Department of Computer Information System, College of Computer Science and Information Technology (CCSIT), Imam Abdulrahman Bin Faisal University, Dammam, 31441, Saudi Arabia
[2]Department of Computer Science, College of Computer Science and Information Technology (CCSIT), Imam Abdulrahman Bin Faisal University, Dammam, 31441, Saudi Arabia
[*]Corresponding Author: Rachid Zagrouba. Email: rmzagrouba@iau.edu.sa
Received: 30 August 2020; Accepted: 22 November 2020

Abstract: The Internet of Things (IoT) is where almost anything can be controlled and managed remotely by means of sensors. Although the IoT evolution led to quality of life enhancement, many of its devices are insecure. The lack of robust key management systems, efficient identity authentication, low fault tolerance, and many other issues lead to IoT devices being easily targeted by attackers. In this paper we propose a new authentication protocol called Authenblue that improve the authentication process of IoT devices and Coordinators of Personal Area Network (CPANs) in an Industrial IoT (IIoT) environment. This study proposed Authenblue protocol as a new Blockchain-based authentication protocol. To enhance the authentication process and make it more secure, Authenblue modified the way of generating IIoT identifiers and the shared secret keys used by the IIoT devices to raise the efficiency of the authentication protocol. Authenblue enhance the authentication protocol that other models rely on by enhancing the approach used to generate the User Identifier (UI). The UI values changed from being static values, sensors MAC addresses, to be generated values in the inception phase. This approach makes the process of renewing the sensor keys more secure by renewing their UI values instead of changing the secret key. In this study, Authenblue has been simulated in the Network Simulator 3 (NS3). Simulation results show an improved performance compared to the related work.

Keywords: Authentication; industrial internet of things; security; Authenblue; blockchain; NS3

1 Introduction

The Internet of Things (IoT) is where almost anything can be controlled and managed remotely. The IoT evolution led to making life much easier, enhancing devices' functionalities and features [1]. For example, during a rainy day, a person can close the home windows and turn on the heaters remotely from his/her office. IoT devices consist of physical components,

such as microcontrollers, transceivers, and memory. In addition to that, they are integrated with a set of simple protocols, which establish communication between IoT devices and their users, and written codes for managing and controlling the IoT devices [2]. IoT helps in making the internet universal and immersive by allowing simple broad communications with many devices, such as house devices, automobile, monitoring cameras, and sensors [2,3]. Additionally, IoT assists in the development of many applications that make use of the excessive amount of data, that is generated by these devices for giving new services to people, governments, and organizations [2]. Furthermore, it is noticed that many industries have started manufacturing IoT devices where there are many IoT products to be used for smart homes, medical support, vehicles manufacturers, and in a variety of other domains [2,4–6]. Regardless of IoT advanced functionalities, the IoT devices themselves are insecure. The lack of a robust key management systems, efficient identity authentication, low fault tolerance and many other issues lead IoT devices to being easily targeted by attackers [7–12].

To overcome these root issues, many research works have urged to utilize the Blockchain technology, since its features can provide promising solutions for a variety of IoT security issues. Blockchain has many advanced features that distinguish it from any other technology [13]. Initially, blockchain was linked with bitcoin, which is mostly known for proof-of-work and hash-based-mechanisms. Nowadays, blockchain is known for providing security and functional assurances [14,15]. Blockchain can be used by many industries in different applications to enhance both, the functionality and security [16,17]. The robust authentication systems used in the cryptocurrency for authenticating the transactions made the cryptocurrency field protected against a variety of attacks [18]. Having a robust identity authentication management system that authenticates devices is what IIoT security needs. Moreover, considering the limited capabilities of IIoT devices by reducing the resource consumption as much as possible is crucial for the sustainability of IIoT field.

Currently, the applied identity authentication management systems in IIoT have two main challenges that hinder them from being widely adopted. These challenges are the low speed and storage of its devices [19]. Developing a strong lightweight authentication protocol for mutually authenticating the identities of IIoT devices and coordinators along with their messages is the main problem to be tackled in this paper. Having such protocol protects against various types of attacks, such as identity spoofing, and modification and fabrication of messages. This work aims to answer the following questions:

- How to develop an effective authentication protocol for authenticating IIoT identities and messages yet it is light enough to suit IIoT limited capabilities?
- How can this protocol enhance IIoT security and protects it against many attacks?

The rest of this paper is organized as follows. Section 2 presents background for IoT and blockchain technologies. In Section 3, we present the related literature review along with our findings and gap analysis. Section 4 depicts the proposed solution and Section 6 shows the simulation work conducted to validate the prosed work. Finally, Section 7 gives the conclusion of this research paper.

2 Literature Review

Although the use of blockchain technology in IoT is an emerging field, many research have shown its effectiveness in increasing IoT overall functionalities [16,20,21]. However, the focus of this paper is on the security perspective. This section consists of six subsections. The first three

subsections address recent research works on IoT security. The related works are discussed based on their proposed solutions. The first subsection includes research works related to the security in IoT communications. As for the second subsection, papers related to the utilization of blockchain for trust management and authentication in the IoT field are discussed. The third subsection covers some papers related to the utilization of blockchain for controlling IoT devices. Lastly, after discussing the related works, comparisons, analysis, and findings are addressed in Subsections 4, 5, and 6 respectively.

2.1 Security in IoT Communications

Wireless sensors network is a pivotal part of the IoT domain. Many research works have targeted the wireless sensors networks (WSN) for enhancing their security [22,23]. In 2013, Li et al. [24] proposed a heterogeneous signcryption scheme for WSN in the paper entitled by "Practical Secure Communication for Integrating Wireless Sensor Networks into the Internet of Things". The proposed scheme algorithms are applied in two stages, offline and online. The scheme aims to secure the communication between the wireless sensors and the internet hosts by providing confidentiality, integrity, authenticity, and nonrepudiation [24]. The authors used the Identity-Based Cryptography (IBC) for the sensors where there are no certificates as in the Public Key Infrastructure (PKI) which can cause an overhead for managing their validity [25]. The main feature of this scheme is the ability of the wireless sensors, which apply IBC, to communicate with the internet hosts, that apply PKI, with high confidentiality, integrity, authenticity, and nonrepudiation. For measuring the scheme's security, the authors proved that their scheme satisfies IND-CCA2, for measuring the encryption security, and EUF-CMA, which measures the signature scheme security [25]. What distinguishes Li and Xiong scheme is its heterogeneous nature, the ability of devices applying IBC to communicate with others that use PKI. Although the dominant of the signcryption schemes are homogeneous, however, heterogeneous schemes suite in many IoT domains, where Internet hosts must communicate directly with servers, and other internet hosts that use different cryptography paradigm [8].

As an enhancement on this scheme, Ting et al. [8] have proposed a scheme with lower computation costs yet has higher security and efficiency [7,26]. The scheme aims to provide a holistic approach for enhancing the four aforementioned main security aspects, which are confidentiality, integrity, authenticity, and nonrepudiation. As in Li and Xiong scheme, this scheme has two stages, offline and online where most of the computations are done in the offline stage.

2.2 Utilization of Blockchain for Trust Management and Authentication in the IoT

In 2017, the paper entitled by "Blockchain Based Trust & Authentication for Decentralized Sensor Networks" posed a security model that uses blockchain data structure to save the sensors decentralized authentication and trust data for achieving integrity and validity, to have cryptographic authenticated data, and trust in peer to peer wireless sensors network, which is heavily used in IoT environment [9]. Additionally, handling of security and privacy in WSN cause problems like, low resource on computation, constraints in energy consumption, and hardware functionality [9]. Moreover, the paper focused on two subjects, security and privacy of data, node authentication and trust management [9]. First, authentication and trust management; WSN has security constraints on node authentication to confirm validity and confidentiality of data [9]. Second, trust management, which is considered upon authentication mechanism to recognize the trustee and trustor [9]. Furthermore, the paper mentioned briefly about the blockchain and the usage of it in financial transactions, where it uses blocks of cryptographic hashes in a linear order that have the previous and next block hash to ensure continuity [9]. The framework is a

service-oriented architecture that handles the data in a decentralized network, which consists of resource constrained nodes that uses embedded system in them.

The proposed module is Blockchain Authentication and Trust Module (BATM), where it uses public key infrastructure for achieving confidentiality by encryption, digital signature authentication, and trust by Peer's identity validation [9]. BATM authentication uses a master key to generate secondary keys for encryption and digital signature, so the key management has a great importance in the module [9]. BATM block mining, where the data payload contains the information of Network Node (NN) condition and cryptographic data. In the case of authentication, the node gives its credentials including the master key, secondary keys [9]. Also, to reduce the number of attacks, the key renewal is done by having key validity timeouts, where the key is renewed when the timer timeouts [9]. Moreover, the privacy of network security depends on the blockchain data, so BATM prevents adding new block unless it's from an authenticated node that did not create any payload in the block, where the choice of the payload to be included in the block is done by the miner. Additionally, to have a valid block it must resolve a problem and include the miner's approved valid payload (include digital signature of random value from the previous authenticated block), that both are created by the miner. As demonstrated in the algorithm below [9].

Algorithm 1: Block Validity Check

Require: currentblock, previousblock **Ensure:** block validity

```
if not (HashCurrentBlock resolves problem) then
return false
end if
if not (MinerApproval payload valid) then
return false
end if
if CurrentBlock has event payload for miner NN then
return false
end if
if not (all payloads in block valid) then
return false
end if
return true
```

BATM trust management, which is accomplished by maintaining the reputation level of nodes, where the reputation is made up of mutual surveillance of all the nodes in the network, that can be known from the node payload where it contains the node behavior that came from its actions, and it is collected over time to ensure its credibility [9]. Additionally, the trust level is calculated from the number of authenticated nodes to the node. This way of trust management made it unfeasible for attackers to overload the network of validated nodes by having, timers, key validity timeouts, and event reputation [9]. In conclusion, this paper introduces a new module that uses blockchain in decentralized sensor networks; which is one of the main components of IoT, that ensures trust management and authentication, security and privacy of data for the goal of a better handling of users' information. Another mechanism that discusses IoT devices authentication was proposed in 2018 named BCTrust [27]. BCTrust was proposed by Hammi et al. [27]. It is based on the blockchain technology and it targets the IoT field since it does not overload its devices [27]. The mechanism has been implemented through Ethereum platform, with

an extra layer for making the blockchain network private [27]. Moreover, certain nodes were given high privileges as stated in the smart contract. These nodes are named Coordinator of Personal Area Network (CPAN) [27]. CPAN nodes are the only ones who can make transactions, and to make this securer, each node has its own pair of keys [27].

Each CPAN node manages a set of nodes under it. In BCTrust mechanism, the principle of "The friend of my friend is my friend" is what the mechanism relies on [27]. By that, if a node named n is managed by the CPAN of the name N, then N initially authenticates n. Once n is authenticated, a transaction is sent to the blockchain to be validated by the CPANs [27]. This transaction shows that N authenticates and manages n, and as a result, N has n's shared symmetric keys to be used for exchanging data securely [27]. This whole process is done by exchanging four messages. If n wants to change its CPAN, to be within a different set of nodes, exchanging two messages for this process is enough. Firstly, the new CPAN checks for the aforementioned transaction in the blockchain. Secondly, if this new CPAN found that n is already authenticated by N, it asks for n's key through a secure channel that uses a key and an initialization vector [27]. Now that the new CPAN has n's key, n officially is considered to be managed by this CPAN, which therefore has to send a transaction just as the one before [27]. The work in [27] have showed how their mechanism has less time and power consumption compared with previous mechanisms. The main reason behind this would be the reduction of the needed messages to be exchanged when associating a node to a new CPAN [27].

2.3 Utilization of Blockchain for Controlling IoT Devices

This section discusses two related works on the use of blockchain technology for controlling IoT devices in terms of resources consumption. Both papers used Ethereum platform for controlling the devices. In the first paper, the proposed approach showed its effectiveness for limiting power consumption. For the second paper, it intended to control the IoT network traffic for enhancing security.

Huh et al. [19] have proposed a new approach for managing IoT devices and securing them [16]. What differentiates their approach than others are their adoption of blockchain technology. Nevertheless, the approach considered the limitations of IoT capabilities and proposes an energy-saving mode. The adopted blockchain platform here is Ethereum, and the used cryptosystem is RSA. The public keys are stored through Ethereum, while the private keys are kept on the devices themselves [19]. Ethereum uses smart contracts, in this approach, the contracts are used to include codes for controlling the IoT devices.

Javid et al. [7] have proposed an integration of blockchain with IoT using a blockchain-based decentralized platform. Their work aimed to prevent unauthorized access to the network by using Ethereum's smart contract functionality. They have also proposed a method of resource allocation that can tackle the issue of turning IoT devices into zombies for performing DDoS attacks. The proposed solution of integrating Ethereum with the IoT device-to-server communication architecture has three security and architectural properties, a blockchain-based framework to detect and prevent IoT DDoS attacks; a distributed framework to control and enable trust-free IoT operations; and the integration of legacy IoT devices with low computational capabilities [7].

As [13] mentioned the single-point-of-failure issues that the IoT centralized-server introduces; the IoT-Ethereum framework proposed in [7] utilizes the smart contract functionality to avoid such issues, as well as other issues related to authentication and trust. The single-point-of-failure can be eliminated through distributing control and trust among multiple participant nodes; where the computational requirements for running the blockchain are distributed among the nodes, and

trust is established through a consensus protocol instead of a third party; the framework can be considered decentralized by the previous ways [7].

2.4 Comparison

To analyze the current solutions for enhancing IoT security, and to decide about the possible contribution to be proposed, six of the aforementioned research works proposed solutions are used as references. As shown in Tab. 1, the solutions are BATM [9], BCTrust [27], Li et al. [24], Ting et al. [8], Huh et al. [19], and IoT-Ethereum Framework [7]. The comparison is done based on the security aspects they target, these aspects are Confidentiality (C), integrity (I), Availability (A), Authentication (AN), authorization (AR), Non-Repudiation (NR), and an efficient management of keys (KM). Beside the security aspects, further factors are chosen, which are BloCkchain utilization (BC), whether the solution uses blockchain or not, and the Resources Consumption (RC), whether the proposed solution highly considers the low capabilities of IoT devices and uses their resources efficiently with a relatively low consumption or not.

Table 1: Comparison between the related works' proposed solutions

Factor	Description	Solution					
		BATM [9]	BCTrust [27]	LX [24]	TTW [8]	HCK [19]	IoT-Ethereum framework [7]
C	The secrecy of the transmitted and stored data	✓	✓	✓	✓	✓	×
I	The accuracy and non-alteration of the transmitted and stored data	✓	✓	✓	✓	✓	×
A	The timely service and information accessibility for IoT devices	×	✓	×	×	×	✓
AN	The verification of IoT device's identity	✓	✓	✓	✓	✓	✓
AR	The granting of privileges to the authorized IoT device	×	✓	×	×	×	✓
NR	The protection against deniability of actions	✓	✓	✓	✓	✓	✓
RC	The consumption of IoT devices' resources is within an acceptable range	✓	✓	×	✓	×	✓
KM	The use of an efficient key management mechanism	✓	×	×	×	✓	×

2.5 Analysis

This section provides analysis for the compared solutions in Tab. 1. It analyzes each solution based on the addressed factors. In BATM model, it utilizes blockchain to ensure two of the main information security model components, which are confidentiality and integrity via encryption.

Moreover, it provides authentication by using public key infrastructure and digital signature while it ensures non-repudiation by using digital signature. Furthermore, it consumes less power and takes less time according to RESTful Model [28], additionally, it consumes less resources since it is low in resource wastage referring to Service-Oriented Architecture, that BATM is based on [29].

For the second solution, BCTrust mechanism, all security aspects are considered. It uses a customized private Ethereum platform, this made it satisfies both, the security aspects applied in Ethereum [27]. Moreover, the use of symmetric and asymmetric keys and the procedure it follows for authenticating IoT devices satisfy the confidentiality, integrity and authentication aspects [27]. As for resources consumption, BCTrust has less power consumption compared with previous mechanisms [27]. The main reason behind this would be the reduction of the needed messages to be exchanged when associating a node to a new CPAN, as clarified previously [27].

For LX and TTW schemes, both consider the same security aspects, which are confidentiality, integrity, authentication, and non-repudiation. They do not consider the authorization aspect as in BATM. Moreover, these schemes do not utilize blockchain technology. LX and TTW varies in their resources' consumption ranges. TTW scheme consumes less memory and energy compared to LX [8,24]. Nevertheless, TTW scheme has better utilization for the microcontrollers in which it made it faster than LX by approximately 30%. Unlike LX scheme which highly consumes the microcontrollers during one of its phases, which is the unsigncrypt phase [8].

As for HCK approach, it adopts the blockchain technology, unlike LX and TTW. As for the considered security aspects, it covers the same as LX and TTW schemes. For the resources' consumption, HCK has an energy-saving mode [19]. This assists in saving IoT devices energy. As for the memory consumption, HCK requires a high storage medium, which is not applicable in IoT devices [19]. Therefore, a solution for this weakness must be addressed in future works.

Lastly, as for the IoT-Ethereum framework, it targets and considers the availability, authentication, authorization, and non-repudiation security aspects. Additionally, it utilizes the blockchain technology, as it uses Ethereum, a blockchain variant. Furthermore, the transactions and data exchange are verified in this framework using high computational and processing capabilities [7].

After comparing and analyzing the proposed solutions in the related works, it is found that BCTrust is the only solution that considered the six specified security aspects. In addition to that, BCTrust mechanism had the least overhead on the IoT devices where it does not exhaust their limited capabilities. Based on this, working on further enhancements on this mechanism may lead into having a powerful mechanism for authenticating and managing IoT devices.

3 The Proposed Solution

Authenblue protocol aims to improve the authentication mechanism in BCTruct protocol. BCTrust has an authentication mechanism for authenticating IIoT devices and CPANs in OCARI networks. Furthermore, it utilizes the blockchain technology for enhancing the association feature. As for Authenblue, it is to be applied in Zigbee-based WSN environment. The focus is on the personalization, association, authentication, and the encryption/decryption functions. Authenblue authenticates IIoT identities and messages in a better way. Furthermore, it has better key management than the one in BCTrust. Authenblue aims to provide a high authentication of IIoT devices identities and packets, along with a good encryption and integrity. Meanwhile, ensure its lightness and suitability for the limited capabilities in the IIoT environments Authenblue provides a mutual lightweight authentication and a key management method for the IIoT devices and their CPANs in the WSN. In a WSN, there are different clusters. Each cluster has IIoT devices and is

coordinated by a CPAN. All CPANs and nodes should have their own unique keys. These keys and other unique identifiers (UI) are set by a trusted authority in the network, known as Provider. With these unique values, IIoT devices and the CPANs start association and authentication procedures. When the association and authentication are done successfully, secure channels between the IIoT nodes and their CPANs are established. Through these channels, CPANs and IIoT nodes can exchange packets securely. All these functions are addressed below in phases followed by their functional requirements.

3.1 Preparation Phases

Initially, CPANs and IIoT devices need to be configured to have unique keys and values. The values generated in the preparation phase give each device a distinctive identity, Tab. 2 illustrates this phase.

Tab. 2. Preparation phase in Authenblue protocol.

Table 2: Preparation phases

Actor	Provider and administrator.
Description	CPANs need to have initial keys, and IIoT devices need to have unique identifiers along with devices keys. The Provider must generate these values, and the administrator must set them on the CPANs and the IIoT devices. This process is to be done once a new CPAN/IIoT device is brought. After that, the administrator has the choice whether to renew these values such as annually or every five years.
Priority	This phase must be at first. Without this preparation phase, Authenblue cannot function.
Process	Generating CPAN keys: • Administrator inputs the MAC addresses of the CPANs to the Provider. • The provider generates a secret key for each CPAN. • The administrator set each key in its CPAN. Generating IIoT device keys: • Administrator inputs the MAC addresses of the device into the Provider. • The Administrator specifies a CPAN that would coordinate this IIoT device. • The Provider generates a unique identifier for the IIoT device. • The Provider generates a device key derived from the UI and CPAN's initial key. • The administrator set each value and key in the IIoT device.

3.2 Authenticated Encryption\|Decryption

IIoT nodes associated with their CPANs can communicate through a secured channel resulted from the association and authentication phases. CPANs need to have initial keys, and IIoT devices need to have unique identifiers along with devices keys. The Provider must generate these values, and the administrator must set them on the CPANs and the IIoT devices. This process is to be done once a new CPAN/IIoT device is brought. The packets sent by CPANs and IIoT nodes can be authenticated by encrypting them and sending them with a tag. Through this, confidentiality,

integrity, and authentication of both, the identity and the message are accomplished. Tab. 3 illustrates this phase.

Table 3: Authenticated encryption/decryption functions

Actor	IIoT devices and CPANs.
Description	The packets sent by CPANs and IIoT nodes can be authenticated by encrypting them and sending them with a tag. Through this, confidentiality, integrity, and authentication of both, the identity, and the message, are accomplished.
Priority	Authenticated encryption/decryption can be performed after a successful association with a CPAN.
Process	Authenticated encryption: • The sender encrypts the data to be sent using the authenticated encryption function in AES-GCM. • The sender generates a tag for the encrypted data. Authenticated decryption: • The receiver checks the received tag, if it is found to be incorrect, it drops the packet, otherwise, it proceeds to the next step. • The receiver decrypts the ciphertext based on the authenticated decryption function in AES-GCM.

3.3 Personalization

The Fig. 1 illustrates how the initial key Ki is generated for the CPAN. At first, the administrator has to manually input the MAC address and the name of the CPAN into the Provider, which in turn generates the key [Ki = HMAC (MAC, random)] and stores it in the local database. The generated key will be received and manually inserted into the CPAN by the administrator.

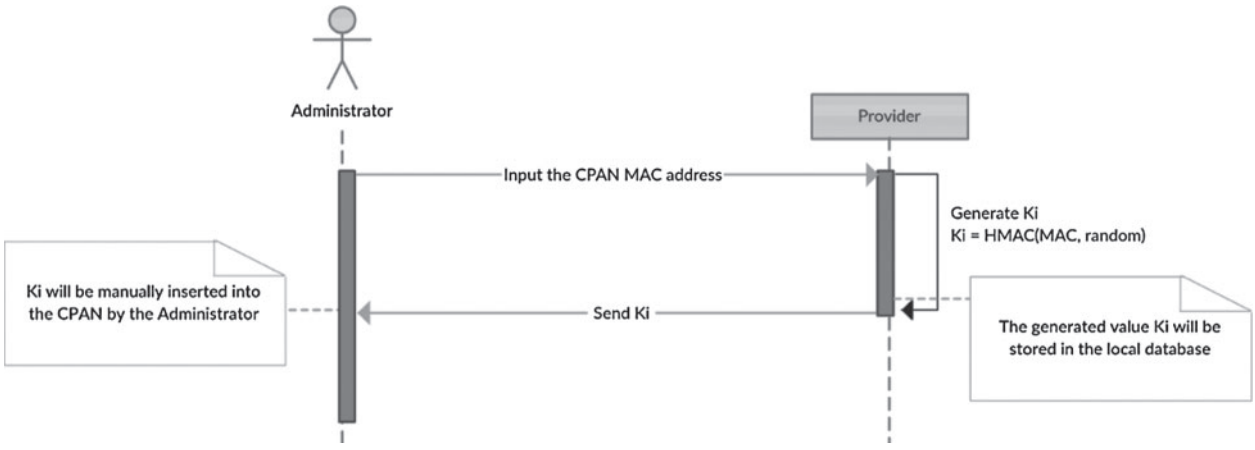

Figure 1: Personalization of the CPAN

The Fig. 2 illustrates how the UI and Kd are generated for a device. At first, the administrator has to manually input the MAC address of the device into the Provider, which in turn generates UI, the 8 byes address, and the derived key Kd [UI = Func (MAC, random), Kd = HMAC (Ki, UI)] and stores them in the local database. The generated values will be received and manually inserted into the device by the administrator.

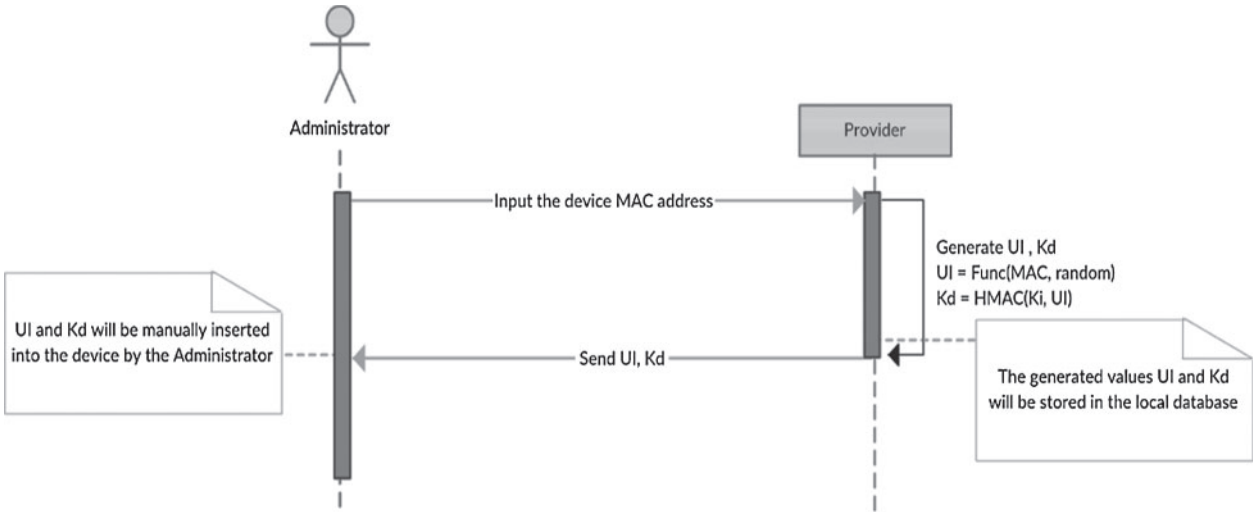

Figure 2: Personalization of IoT devices

3.4 Associating a Device to a CPAN

The process of associating a device to a CPAN in the mutual authentication protocol the BCTrust [27] is illustrated in the diagram below. At first, the device sends an association request that contains its UI to the CPAN, which in turn generates a challenge (a random number) and sends it to the device as an authentication request. The device then computes otp1 using its derived key Kd and the received challenge [otp1 = HOTP (Kd, challenge)], then sends the computed otp1 to the CPAN as an authentication response. The CPAN computes Kd of the device through the personalization function, where it inserts its initial key Ki and the device's UI [Kd = HMAC (Ki, UI)], then generates otp1' using the device's Kd and the challenge [otp1' = HOTP (Kd, challenge)].

The CPAN then compares between the received otp1 and the computed otp1'; if they differ, the device authentication fails and its association_req_count (failed association request counter) will be compared to association_req_max (maximum number of failed association request attempts), if they are equal, the device's UI will be blacklisted and the association operation stops, if they are different, the association_req_count for the device will be incremented and the association operation stops. Otherwise, if the computed otp1' is equal to the received otp1, the device is authenticated successfully, and the CPAN generates a symmetric key Ku (unicast mode), signature, hiddenKeyBroadcast and otp2 [Ku = PRF (Kd, challenge), signature = HMAC (Ku, otp1), hiddenKeyBroadcast = signature $\oplus$ Kb, otp2 = HOTP (Ku, hiddenKeyBroadcast)].

The CPAN sends the device an association response that contains otp2 and hiddenKeyBroadcast. The device in turn computes Ku, signature, Kb and otp2' [Ku = PRF (Kd, challenge),

signature = HMAC (Ku, otp1), Kb = signature ⊕ hiddenKeyBroadcast, otp2' = HOTP (Ku, hiddenKeyBroadcast)]. If the computed otp2' equals the received otp2, the CPAN is authenticated successfully, a mutual authentication is successful, and a secured channel is created. Otherwise, the CPAN authentication fails and the association operation stops.

Fig. 3 also demonstrates how an attacker can leverage the blacklisting mechanism to blacklist the UI of innocent legitimate devices; where an attacker sends the CPAN an association request that contains the spoofed UI of device a2, then receives an authentication request from the CPAN that contains a challenge. Upon computing otp1 using an incorrect Kd [otp1 = HOTP (Kd, challenge)] and sending it to the CPAN as an authentication response, the CPAN compares it to the computed otp1', which will turn to be different, causing device authentication failure. The CPAN then will either blacklist the UI of device a2 if its association_req_count is equal to the association_req_max, or increment the device's association_req_count, then the association operation stops. This UI-based blacklisting mechanism is a weakness in BCTrust 27, as an attacker can repeat the illustrated process causing innocent devices to be blacklisted in the network.

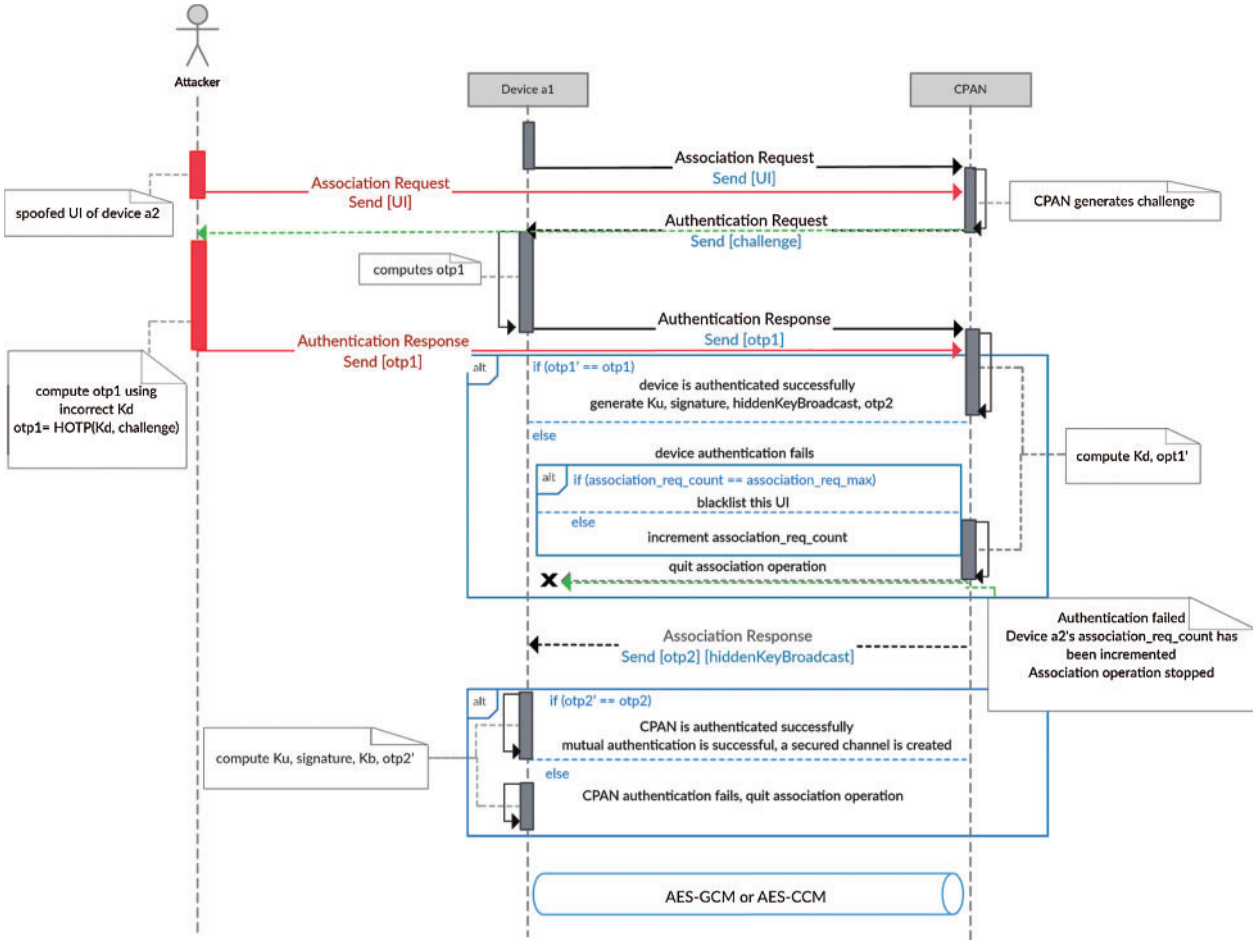

Figure 3: Associating a device to a CPAN

As opposed to what is illustrated in Fig. 3, where the blacklisting mechanism that is based on blacklisting devices' UI can be leveraged by attackers to cause legitimate devices to be blacklisted, eliminating this mechanism from the protocol would prevent the occurrence of such attacks. Fig. 4 illustrates how the protocol would operate after eliminating the blacklisting mechanism.

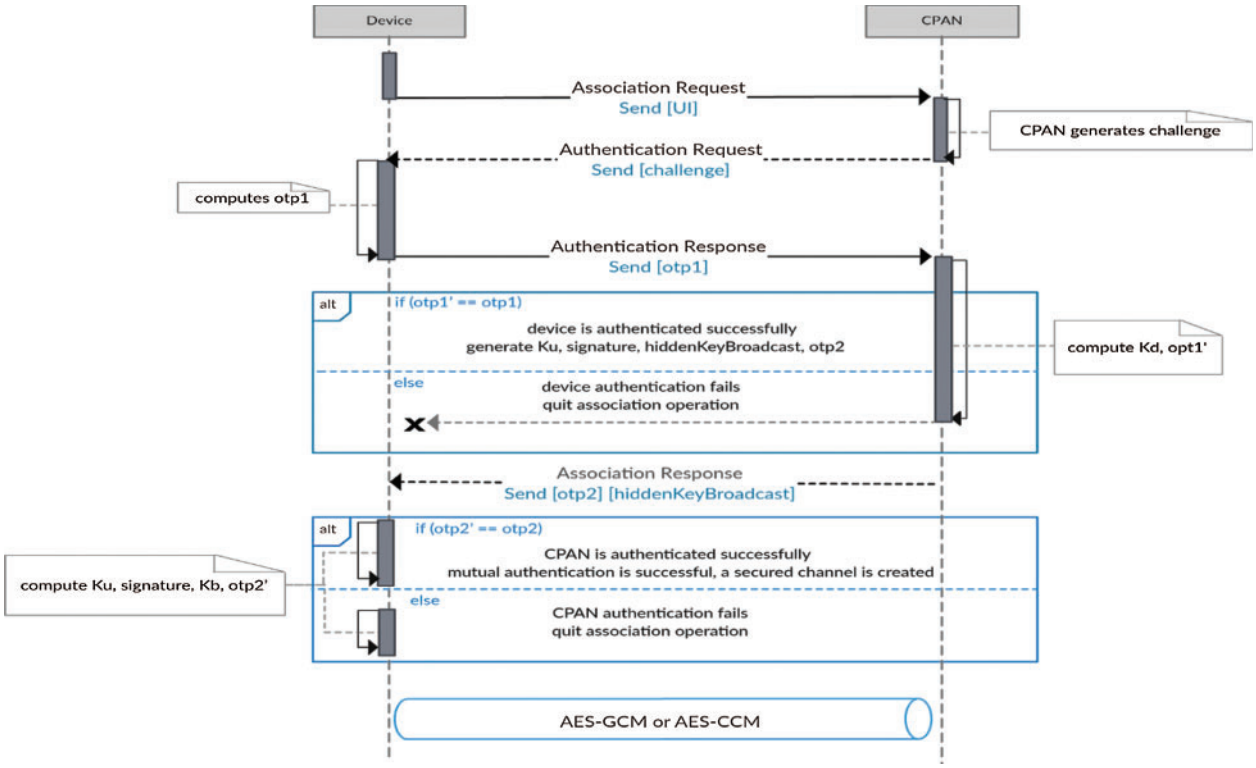

Figure 4: Eliminating the blacklisting mechanism in associating a device to a CPAN

3.5 Authenticated Encryption and Decryption

Now that the IIoT devices are associated to the CPANs and each IIoT node has the Ku and Kb, both IIoT nodes and CPANs can encrypt and decrypt their packets. The encryption and decryption functions are the authenticated encryption/decryption functions in AES-GCM, which are adopted in BCTrust [24].

Fig. 5 shows a sequence diagram that illustrates how an IIoT device can encrypt data (P) and send them to the CPAN as n blocks of ciphertext (C) concatenated with a tag (T). Firstly, the device generates an IV based on its key (K), Ku if the packet is to be sent to the CPAN and a Kb if the packet is to be sent to all the nodes in the cluster, and a counter. After that, the device uses an additional authenticated data (A) that is preconfigured and known for both sides. The device then computes H by encrypting 128 zeros with the K. After that, it set a counter (Y) that is initialized to the value of the IV concatenated with 31 zeros and a1. The counter keeps incrementing until it reaches (n), the number of plaintext blocks to be encrypted. These blocks are encrypted by computing the Exclusive OR of their values along with the counter Y after it is encrypted by K. Lastly, a tag is generated and concatenated with the C to be sent to the CPAN.

The tag is computed as follows: $T = MSBt\ (GHASH(H, A, C, len(A), len(C)) \oplus E(K, Y0))$, where MSBt refers to taking the left t bits of the result.

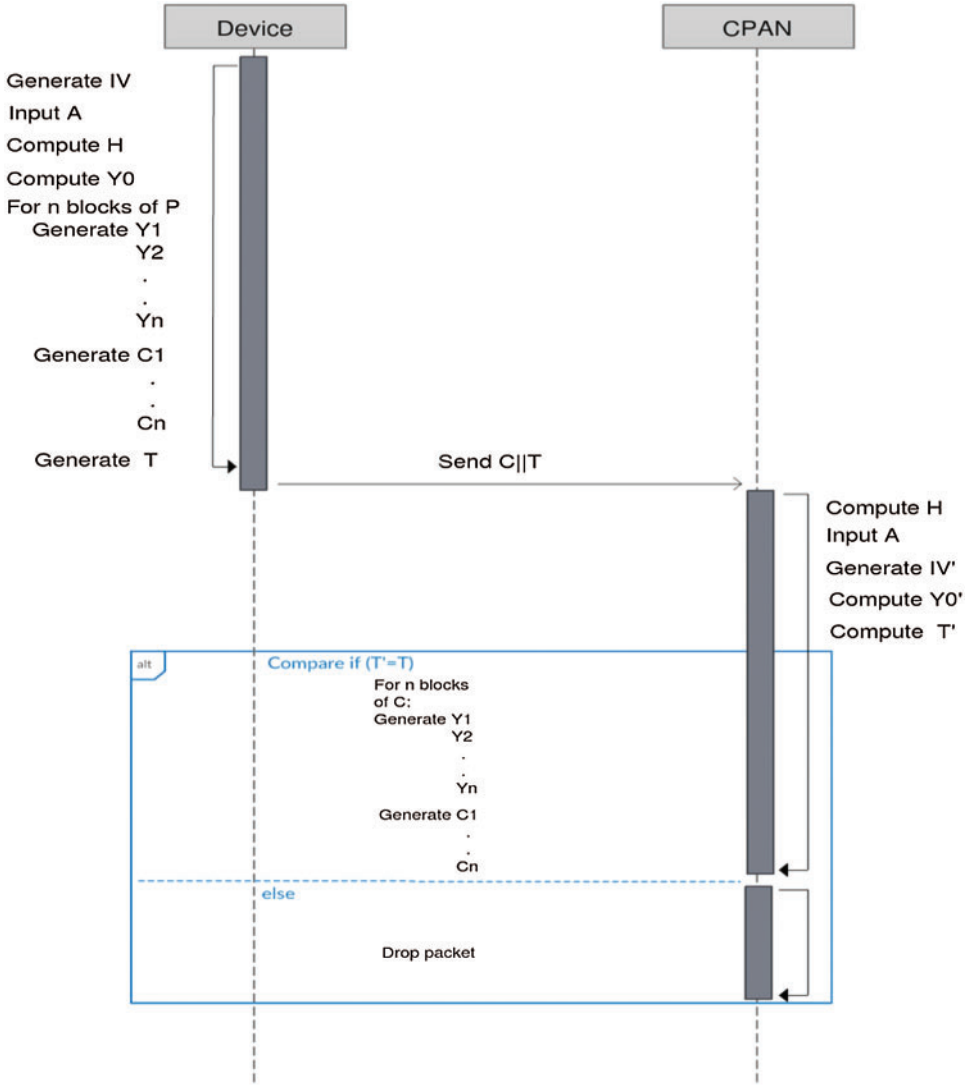

Figure 5: Authenticated encryption and decryption [27]

The same process is to be done on the CPAN side, however, the order is different where the CPAN has to compute the tag and compare it with the received one, if they are matched, then it decrypts the packet by computing the Exclusive OR of their values along with the counter Y after it is encrypted by K. Otherwise, it drops the received packets.

4 Simulation Work

4.1 Simulation Architecture

Authenblue protocol consists of different types of devices and goes through multiple phases, Fig. 6 summarizes Authenblue general architecture. The main phases that need to be simulated

are the association request and response, and the authentication request and response. To simulate these phases, the simulation environment must be fully prepared. Moreover, the functions to be used in these phases must be finalized and programmed.

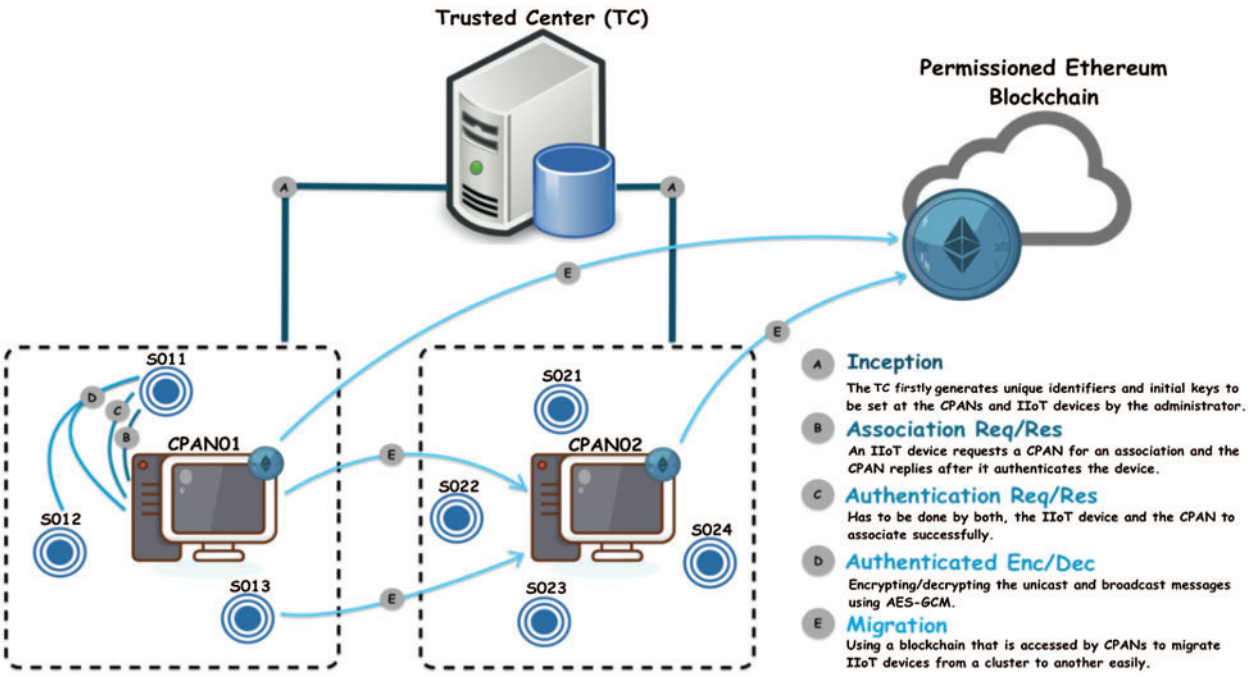

Figure 6: The general architecture of Authenblue

This section covers the work done on NS3 simulator for preparing Authenblue environment and the inception phase where a sample of keys are generated. The section also presents a comprehensive simulation of the Association Request phase. As for the rest of the phases, the used functions and values are demonstrated along with the expected simulation results.

4.2 Performance Testing

Performance testing has been conducted with a focus on measuring the time consumption. The test is conducted by calculating the association request packet received time. After measuring the time it takes an association request packet to be received by the CPAN, it is found that the association request phase takes 0.008536 s as shown in Tab. 4. As for the overall time consumption, it is required to have a comprehensive simulation of Authenblue to have an estimated time.

4.3 Analysis

After coding and simulating Authenblue, this section compares it with the solutions discussed previously in the literature review, Tab. 5 shows this comparison. The main two aspects that differentiate Authenblue than the other solutions are its management of keys.

The Tab. 3 shows how Authenblue has an efficient key management mechanism. This is due to the way of generating the unique identifiers (UI) of the sensors. In Authenblue, UIs are generated

based on a random generator to produce a 128-bit length. Unlike the static value which was presented in BCTrust solution. This makes it easier for renewing the identifiers and the secret keys of the sensors themselves.

Table 4: Transmission time

Device	Sent time	Transmit time
Sensor 1	0.0	0.003296
Sensor 2	1.0	0.0015
Sensor 3	2.0	0.00374

Table 5: Comparing Authenblue with other solutions

Factor	Description	Solution						Authenblue
		BATM	BCTrust	LX	TTW	HCK	IoT-Ethereum Framework	
C	The secrecy of the transmitted and stored data	✓	✓	✓	✓	✓	×	✓
I	The accuracy and non-alteration of the transmitted and stored data	✓	✓	✓	✓	✓	×	✓
A	The timely service and information accessibility for IoT devices	×	✓	×	×	×	✓	✓
AN	The verification of IoT device's identity	✓	✓	✓	✓	✓	✓	✓
AR	The granting of privileges to the authorized IoT device	×	✓	×	×	×	✓	✓
NR	The protection against deniability of actions	✓	✓	✓	✓	✓	✓	✓
RC	The consumption of IoT devices' resources is within an acceptable range	✓	✓	×	✓	×	✓	✓
KM	The use of an efficient key management mechanism	✓	×	×	×	✓	×	✓

5 Conclusion

This paper covered the blockchain based solutions on IoT security, especially the security in IoT communications, the utilization of blockchain for trust management and authentication in the IoT field, and the utilization of blockchain for controlling IoT devices.

Additionally, several solutions regarding the IoT security were compared and analyzed, with the conclusion that working on the enhancements of BCTrust mechanism would lead to having a powerful blockchain-based identity authentication system for managing. We proposed a new authentication protocol named Authenblue that helps in the authentication process of sensors, IIoT nodes, and coordinators in an IIoT environment. It is a security solution that aims to enhance the authentication process in san IIoT environment, by assisting in the mitigation of the occurrence of some cyber-attacks. Authenblue enhance the authentication protocol that BCTrust and other models rely on by enhancing the way of generating the UIs. The unique identifiers (UI) values changed from being static values, sensors MAC addresses, to be generated values in the inception phase. Such modification is crucial for the key managed process. It makes the process of renewing the sensor keys more efficient by renewing their UI values instead of changing the secret key of the CPAN. Furthermore, this paper has simulated parts of the protocol through NS3. Such simulation contributes to the present NS3 authentication models. The simulation result show that Authenblue has an efficient key management mechanism. This is due to the way of generating the UI of the sensors. In Authenblue, UIs are generated based on a random generator to produce a 128-bit length. Unlike the static value which was presented in BCTrust solution. This makes it easier for renewing the identifiers and the secret keys of the sensors themselves.

Funding Statement: The author(s) received no specific funding for this study.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] B. Marsh and P. Piscioneri, "The Internet of Postal Things," in *Proc. Int. Conf. on Collaboration Technologies and Systems*, Atlanta, USA, pp. 3–4, 2015.

[2] R. Alhajri, R. Zagrouba and F. Alhaidari, "Survey for anomaly detection of IoT botnets using machine learning auto-encoders," *International Journal of Applied Engineering Research*, vol. 14, no. 10, pp. 2417–2242, 2019.

[3] A. Kardi and R. Zagrouba, "Attacks classification and security mechanisms in wireless sensor networks Advances in Science," *Technology and Engineering Systems Journal*, vol. 4, no. 6, pp. 229–243, 2019.

[4] S. Jha, L. Nkenyereye, G. Prasad Joshi and E. Yang, "Mitigating and monitoring smart city using Internet of Things," *Computers, Materials & Continua*, vol. 65, no. 2, pp. 1059–1079, 2020.

[5] S. Abbas, M. A. Khan, L. E. Falcon-Morales, A. Rehman, Y. Saeed *et al.,* "Modelling, simulation and optimization of power plan energy sustainability for IoT enabled smart cities empowered with deep extreme leaning machine," *IEEE ACCESS*, vol. 8, no. 1, pp. 39982–39997, 2020.

[6] A. Ata, M. A. Khan, S. Abbas, M. S. Khan and G. Ahmad, "Adaptive IoT empowered smart road traffic congestion control system using supervised machine learning algorithm," *Computer Journal*, vol. 3, pp. 1, 2020.

[7] U. Javaid, A. K. Siang, M. N. Aman and B. Sikdar, "Mitigating IoT device based DDoS attacks using blockchain," in *Proc. 1st Workshop on Cryptocurrencies and Blockchains for Distributed Systems*, Munich, Germany, pp. 71–76, 2018.

[8] P. Y. Ting, J. L. Tsai and T. S. Wu, "Signcryption method suitable for low-power IoT devices in a wireless sensor network," *IEEE Systems Journal*, vol. 12, no. 3, pp. 2385–2394, 2018.

[9] A. Moinet, B. Darties and J. L. Baril, "Blockchain based trust & authentication for decentralized sensor networks," vol. 1, pp. 1–6, 2017.
[10] M. Sohail, M. A. Khan, I. Ahmad and O. Sohail, "Intelligent data encryption scheme for light weighted AIoT enabled devices," *Journal of Information Assurance and Security*, vol. 15, no. 1, pp. 17–25, 2020.
[11] A. Afzal, M. A. Khan and S. Abbas, "Secure communication of IoT based devices using EPEB algorithm," *Journal of Information Assurance and Security*, vol. 13, no. 3, pp. 91–97, 2020.
[12] F. Alhaidari, A. Rahman and R. Zagrouba, "Cloud of things: Architecture, applications and challenges," *Journal of Ambient Intelligence and Humanized Computing*, vol. 3, no. 6, pp. 1099, 2020.
[13] H. Atlam, A. Alenezi, M. Alassafi and G. Wills, "Blockchain with Internet of Things: Benefits, challenges, and future directions," *International Journal of Intelligent Systems and Applications*, vol. 10, no. 6, pp. 40–48, 2018.
[14] G. Karame and S. Capkun, "Blockchain security and privacy," *IEEE Security & Privacy*, vol. 16, no. 4, pp. 11–12, 2018.
[15] M. T. Hammi, B. Hammi, P. Bellot and A. Serhrouchni, "Bubbles of trust: A decentralized blockchain-based authentication system for IoT," *Computers & Security*, vol. 78, pp. 126–142, 2018.
[16] J. Shen, C. Wang, T. Li, X. Chen, X. Huang *et al.,* "Secure data uploading scheme for a smart home system," *Information Sciences*, vol. 453, pp. 186–197, 2018.
[17] M. Maqsood, A. Rahman, L. Malrey and J. Y. Choi, "Evolutionary-based image encryption using RNA codons truth table," *Optics & Laser Technology*, vol. 121, pp. 105818, 2020.
[18] F. H. Al-Naji and R. Zagrouba, "A survey on continuous authentication methods in Internet of Things environment," *Computer Communications Journal*, vol. 163, pp. 109–133, 2020.
[19] S. Huh, S. Cho and S. Kim, "Managing IoT devices using blockchain platform," in *Proc. 19th Int. Conf. on Advanced Communication Technology*, Bongpyeong, pp. 464–467, 2017.
[20] M. A. Khan and K. Salah, "IoT security: Review, blockchain solutions, and open challenges," *Future Generation Computer Systems*, vol. 82, pp. 395–411, 2018.
[21] F. H. Al-Naji and R. Zagrouba, "Secure IoT based on blockchain: Quantitative evaluation and analysis of the correlation between block mining time and blockchain efficiency," *International Journal of Applied Engineering Research*, vol. 15, no. 4, pp. 377–384, 2019.
[22] R. Zagrouba, "AMIS: Authentication mechanism for IoT security," *ACTA Scientific Computer Sciences*, vol. 2, no. 7, pp. 32–37, 2020.
[23] A. Kardi and R. Zagrouba, "Attacks classification and security mechanisms in wireless sensor networks, Advances in Science," *Technology and Engineering Systems Journal*, vol. 4, no. 6, pp. 229–243, 2019.
[24] F. Li and P. Xiong, "Practical secure communication for integrating wireless sensor networks into the internet of things," *IEEE Sensors Journal*, vol. 13, no. 10, pp. 3677–3684, 2013.
[25] M. Masdari, S. Jabbehdari and J. Bagherzadeh, "Improving OCSP-based certificate validations in wireless ad hoc networks," *Wireless Personal Communications*, vol. 82, no. 1, pp. 377–400, 2015.
[26] A. Al-Mousa, M. Al-Qomri, S. Al-Hajri, R. Zagrouba and S. Chaabani, "Environment based IoT security risks and vulnerabilities management," in *Proc. IEEE 2020 Int. Conf. on Computing and Information Technology*, Al-Jouf, KSA, 2020.
[27] M. T. Hammi, P. Bellot and A. Serhrouchni, "BCTrust: A decentralized authentication blockchain-based mechanism," in *Proc. IEEE Wireless Communications and Networking Conf.*, Barcelona, 2018.
[28] L. H. Nunes, L. H. V. Nakamura, H. F. Vieira, R. M. O. Libardi, E. M. Olivera *et al.,* "Performance and energy evaluation of RESTful web services in Raspberry Pi," in *Proc. IEEE 33rd Int. Performance Computing and Communications Conf.*, Austin, TX, pp. 1–9, 2014.
[29] H. A. Moniem and H. H. Ammar, "Performance prediction of service-oriented architecture, a survey," *International Journal of Computer Applications Technology and Research*, vol. 3, no. 12, pp. 831–835, 2014.