This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TII.2019.2894573, IEEE Transactions on Industrial Informatics

A Blockchain-Based Solution for Enhancing Security and Privacy in Smart Factory

Jiafu Wan, Jiapeng Li, Muhammad Imran, Di Li, Fazal-e-Amin

***Abstract*—Through the Industrial Internet of Things (IIoT), a smart factory has entered the booming period. However, as the number of nodes and network size become larger, the traditional IIoT architecture can no longer provide effective support for such enormous system. Therefore, we introduce the Blockchain architecture, which is an emerging scheme for constructing the distributed networks, to reshape the traditional IIoT architecture. First, the major problems of the traditional IIoT architecture are analyzed, and the existing improvements are summarized. Second, we introduce a security and privacy model to help design the Blockchain-based architecture. On this basis, we decompose and reorganize the original IIoT architecture to form a new, multi-center, partially decentralized architecture. Then, we introduce some relative security technologies to improve and optimize the new architecture. After that, we design the data interaction process and the algorithms of the architecture. Finally, we use an automatic production platform to discuss the specific implementation. The experimental results show that the proposed architecture provides better security and privacy protection than the traditional architecture. Thus, the proposed architecture represents a significant improvement of the original architecture, which provides a new direction for the IIoT development.**

***Index Terms*— Blockchain, Industrial Internet of Things (IIoT), Security and Privacy, Smart Factory**

I. Introduction

With the development of various technologies, and the rise of the innovative concepts such as Big Data, Cloud Computing, Cyber Physical System (CPS), etc., the modern industry has been promoted to a new level [1]. Consequently, a smart factory, with the Industrial Internet of Things (IIoT) as the core, aiming to realize the intelligent manufacturing, has emerged. Through the IIoT, different types of industrial equipment in a smart factory realize the cluster interaction, in which the data is no longer independent. Via the collision and fusion of the data, the transformation and upgrading of the intelligent and networked manufacturing process can be promoted [2], [3]. In fact, current industry trends and initiatives aim to use the Internet to connect the unconnected things in the industry and realize the fourth industrial revolution [4].

Manuscript received November 11, 2018; revised September 06, 2018; accepted January 12, 2019. Date of publication xx, 2019; date of current version xx, 2019. The corresponding author: M. Imran

J. Wan, J. Li and D. Li are with the School of Mechanical and Automotive Engineering, South China University of Technology, Guangzhou, China (e-mails: mejwan@scut.edu.cn, 201720100821@mail.scut.edu.cn, itdili@scut.edu.cn).

M. Imran and F. Amin is with the College of Computer and Information Sciences, King Saud University, Saudi Arabia (e-mail: dr.m.imran@ieee.org, famin@ksu.edu.sa).

Thanks to the attention and promotion from all areas of life, there are already many mature industrial IoT platforms serving the smart factories, such as Simens MindSphere, Schneider EcoStruxure, GE Predix, etc. [5]. These industrial IoT platforms accomplish the vertical integration, horizontal integration, and end-to-end integration, breaking the information isolated island problem of the equipment and realizing the integration of various equipment in a smart factory. However, due to the limitations of the IIoT architecture and vulnerabilities of the underlying equipment, a large amount of critical security and private data is very vulnerable to the attacks. Several cases of the attack on the data acquisition systems under the normal IoT environment were shown in [6]. On the other hand, in [7], through the access, detection, and long-term monitoring of various types of embedded equipment on the Internet, a large number of vulnerable nodes were discovered, revealing many different loopholes at the equipment level. Besides, in [8], the authors revealed that malware, malicious scripts, etc., could be easily sneaked into various equipment at the application level, which could violate users' private data without users' permission. Such threats and attacks would cause not only sensitive data leakage but also would bring huge social and economic problems, even endanger the national security. Therefore, there is an urgent need to solve the security and privacy shortcomings of the current IIoT architecture.

In 2008, based on the Blockchain technology, Nakamoto proposed a peer-to-peer digital currency system named the Bitcoin [9]. Since now, the system has been running steadily. There are many similarities between the IIoT system and the Bitcoin system, such as a tremendous number of different nodes, frequent and volatile heterogeneous data exchange, high security and privacy requirements, etc. Hence, it is feasible to apply the Blockchain technology to the IIoT system to improve the underlying architecture and solve the above-mentioned shortcomings. In [10], the authors proposed a new two-factor authentication scheme based on the Blockchain technology to ensure data security. Further, in [11], IoT was combined with the open source Blockchain platform to realize data exchange, and a distributed method was created to improve the security on the equipment level. Based on the Blockchain and Smart Contract, a point-to-point IIoT platform called the BPIIoT was created in [12] to realize data exchange without trusted intermediaries. The mentioned works are very enlightening. The shortcomings of the current IIoT are discussed and analyzed, and corresponding improvement methods are proposed. However, in addition to the high security and privacy, an excellent IIoT architecture also needs to have

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TII.2019.2894573, IEEE Transactions on Industrial Informatics

decent performance. The existing research has three main drawbacks. First, the introduction of the Blockchain technology increases the transmission and computing burden of the IIoT architecture, but the impact on the real-time capability of the industrial environment is not taken into deep consideration. Second, the scope of the majority of the relative studies is small, which means that a completely independent system is not constructed. Finally, since most of the research is based on the open source platforms, such as Multichain, Ethereum, Hyperledger, etc. [13], high coupling with the third-party platforms may cause unpredictable problems.

Therefore, in this work, we aim to realize the profound integration of the Blockchain and IIoT. The novelty of this work is as follows: 1) we combine the Blockchain technology and introduce the Bitcoin design to build a privatized, lightweight, easily expanded, and partially decentralized IIoT architecture for a smart factory; 2) a security and privacy model is introduced to help analyze the key aspects of the architecture; 3) based on the rigorous analysis and demonstration, we introduce the whitelist mechanism, asymmetric encryption mechanism, and other methods to improve the security and privacy of the IIoT architecture; 4) we transform an automatic production platform according to the proposed architecture, and take the platform as an example to discuss the specific defense measures.

The paper is organized as follows. In Section II, the modified IIoT architecture based on the Blockchain technology is presented, the key aspects are enumerated and the relevant key technologies of the IIoT architecture are discussed. In Section III, we introduce a model to help analyze the key aspects of the architecture and guide the defense mechanism design; In Section IV, the detailed flowcharts are presented to describe the data flow, the specific algorithms to control data interaction are given, and the defensive mechanisms of the proposed architecture are analyzed. In Section V, we transform an automatic production platform with more specific settings, and analyze the security and privacy of the proposed architecture in practice. Lastly, a brief conclusion and future research direction are provided in Section V.

II. Blockchain-Based IIoT Architecture For Smart Factory

Currently, most of the smart factories are based on the Cloud-Based Manufacturing (CBM) architecture [14]. Such an architecture enables users to access the shared pool of manufacturing resources anytime and anywhere on demand, and rapid configuration and management of resources can be realized with the minimal work and third-party interaction [15]. However, the centralized architecture is very fragile. Namely, as long as the central node is damaged, all services will be suspended. Therefore, we aim to build a decentralized system with nodes supervising each other mutually. The proposed Blockchain-based IIoT architecture for a smart factory is shown in Fig. 1. The proposed architecture has five layers, the sensing layer, the management hub layer, the storage layer, the firmware layer, and the application layer.

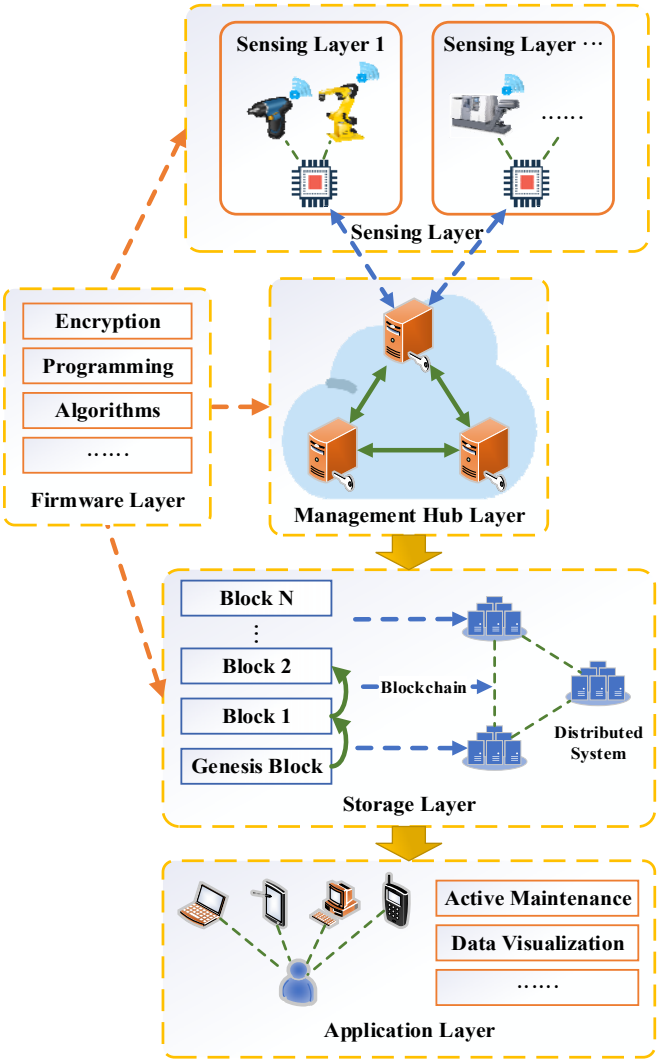

Fig. 1. The Blockchain-based IIoT architecture for a smart factory.

The sensing layer includes various types of sensors and at least one microcomputer with a certain computing power, which can obtain information on various equipment, and preprocesses the collected data. On the one hand, the management hub layer parses the uploaded data, encrypts the data, packages the data to generate blocks, and stores it in the database. On the other hand, management hub layer needs to integrate and manipulate different equipment based on the production scheduling strategy and respond to the users' requests in real time to provide the customized services. The storage layer plays the role of a data center, keeping the encrypted, tamper-resistant data and blockchain records, which are stored in a distributed form and synchronized at a certain interval. To make the sensing layer, the management hub layer and the storage layer complement each other effectively, we propose the firmware layer, which involves the underlying implementation technologies to connect each layer, including the data acquisition, distributed algorithms, data storage technology, etc. The application layer provides users with different kinds of services such as real-time monitoring, failure prediction, etc.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TII.2019.2894573, IEEE Transactions on Industrial Informatics

A. Division of the architecture

The proposed architecture is divided into the intranet and the extranet. The intranet aims to collect and store data, while the extranet aims to utilize the data to provide different users different services.

The intranet consists of the sensing layer, the management hub layer, and the storage layer. Due to the limitation on the computing power, a peer-to-peer (P2P) network is not considered. The data of each equipment node need to be uploaded to the designated management hub and managed by the management hub. Every equipment node needs to request permission from the management hub for different operations.

Bitcoin uses the Unspent Transaction Output (UTXO) format to ensure good anonymity and security. Since the number and authority of participating equipment nodes in the private blockchain are strictly limited, instead of the UTXO, we record the state of sensors directly. Moreover, considering the diversity, complexity, etc., in the IIoT system [16], the sensing layer is equipped with microcomputers, such as STM32, Raspberry Pi, and so on, to preprocess the data, which help effectively reduce the overhead of the upper system.

The extranet consists of the management hub layer and the application layer. The main difference between the intranet and extranet is that the intranet is oriented to the equipment, while the extranet is oriented to the users. Therefore, the extranet needs to connect to the Internet and consider connection, algorithms, tools, etc. to utilize the data, and create a reasonable access method to provide services for the users. That means users can customize diverse management hubs according to their own needs, and high quality of service (QOS) can be ensured.

In order to ensure data security and privacy, both intranet and extranet have the whitelist and dynamic authentication mechanisms to restrict nodes. In the whitelist mechanism, a whitelist is usually used together with a blacklist, which determines the right to access or deny. Such a mechanism can quickly verify the access traffic and filter the malicious traffic, providing fast and convenient security and privacy [17]. Considering the equipment nodes and user nodes in the architecture, the whitelists and blacklists should be created respectively for the intranet and extranet. Furthermore, the dynamic verification mechanism is also an effective way to guarantee good security and privacy of data [18]. The permission acquired by equipment nodes and user nodes is time-limited. When the time limit is reached, the permission and the Proof of Work (PoW) need to be re-verified. What's more, the management hub layer and the user layer are also provided with a self-running algorithm to generate a paired verification code. Users need to provide the code to maintain access permission when re-verifying.

B. Management hub layer

In the blockchain system, besides the transaction node, there is a special node responsible for recording blocks. In the proposed architecture, that is the management hub, constituting the management hub layer.

In order to ensure that all management hubs are trusted, some consensus algorithms must be applied to the architecture, such as Proof of Work (PoW), Proof of Stake (PoS), Practical Byzantine Fault Tolerance (PBFT), etc. [19]. By raising a mathematical question with a reward as the PoW, the Bitcoin system encourages the recording nodes to compete. The reward guarantees the participation of the recording nodes, and the computational cost brought by the mathematical problem solving makes the malicious operation costly. However, this solution not only wastes a lot of computational resources but also brings the problem of scalability [20].

In the IIoT system, we pay more attention to the utilization of resources and the efficiency of data interaction. Moreover, unlike the untrusted system in the Bitcoin, the proposed system is a private blockchain system with all nodes trusted initially. Therefore, we abandon the competition and reward mechanism. Each workshop in a smart factory is equipped with one or more specialized management hubs for data management. Then, the Statistical Process Control (SPC) [21] or other comparison algorithms are introduced to complete the PoW. For SPC, according to the specific equipment, we first set eigenvalues such as control limits, average values, etc. Then, some statistical analysis of the uploaded data is carried out to compare it with the set values and authenticate the transaction. Each time a new block record is generated, the PoW should be carried out once. Such a design can improve the fault tolerance, scalability, and real-time capability of the architecture.

Instead of setting up the cloud, we consider multiple management hubs to form a private cloud, which allows users to connect and access the data. It can be considered that the management hub is equivalent to the fusion of an intelligent edge gateway and a cloud, which is a simplification and improvement of the edge computing architecture [22]. Although the sensing layer and management hub layer are a centralized LAN system, from the global perspective, multiple management hubs constitute a partially decentralized system. Multiple distributed nodes can reduce the pressure of a single central node and avoid system failure caused by a single central node crash. Such design is a suitable solution for the Blockchain technology in combination with the actual industrial environment.

C. Private blockchain

We design a private, unique block structure to record data in the storage layer. As shown in Fig. 2, the block contains two parts: block header and block body. Structured data are stored in the block header, recording multiple characteristic values of the current block, including the hash value of the previous block, the timestamp of the current block, and the Merkel root generated by specific algorithm on the collected data [23]. Generally speaking, a block is more like an index file. The specific equipment data is still stored by the database, and the access record (storage, reading, and control) of each node is stored in the block. Through this setting, each operation will be strictly supervised via blocks, while the advantages of traditional databases can be maintained. Therefore, various operations of each node can be traced and the data interaction can be highly protected.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TII.2019.2894573, IEEE Transactions on Industrial Informatics

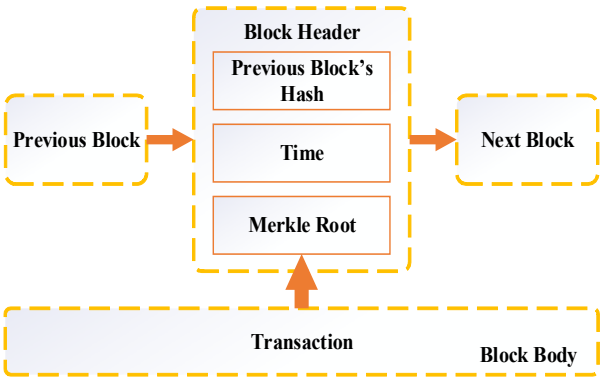

Fig. 2. Block structure.

In Table 1, the examples of the information recorded in the block body, including the request address, the request content, the Merkel root, and the response from the management hub, are given. The block limits the number of data records, which means a new block will be generated after a certain number of insertions. In fact, the storage layer is the physical storage device in the management hub, such as SSD, HDD, and so on, which constitutes a distributed storage system.

TABLE I
TRANSACTION IN THE BLOCK BODY

Address	Request	Merkle Root	Action
1	Store	1	Deny
2	Read	2	Deny
3	Control	3	Deny

In order to achieve good privacy as well, we introduce the hash algorithm and asymmetric encryption algorithm. For our architecture, we choose the SHA256 [24] and Elliptic Curve Cryptography (ECC) [25]. By using the SHA256, we can generate the Merkle root, which can effectively compress the volume of data to link each block. When using the encrypted data, users can perform the same hash calculation and compare the hash codes to verify the data. The ECC algorithm is used to generate the public and private keys to encrypt the data. Data in the database will be encrypted into ciphertext via the public key. Users should provide the private key to decrypt the encrypted message for customized services. These two algorithms can ensure good privacy of the data and prevent illegal manipulation.

III. DATA SECURITY AND PRIVACY MODEL

We consider that the architecture needs to address three major requirements: confidentiality, integrity, and availability (CIA), which can be referred to as the CIA requirement [26]. The confidentiality ensures that only authorized users can read the message; the integrity ensures that no changes are made to the received and sent messages; the availability means that each service and data are available [27]. In this section, we consider combining Bell-La Padula (BLP) model with Biba model [28], which are suitable for CIA requirements, to help perfect our architecture. Based on these two models, we make some simplification and introduce some attributes:

$$S = \{s_1, s_2, s_3, \cdots, s_n\}$$
$$O = \{o_1, o_2, o_3, \cdots, o_n\}$$
$$\mu = \{M_1, M_2, M_3, \cdots, M_n\}$$
$$A = \{w, r, c\}$$
$$L = \{l_1, l_2\}$$

S is the set of subjects. O is the set of objects. μ is the set of access matrixes, which indicates the access privileges that the subjects have to the objects. A is the set of access attributes, in which w stands for storing, r stands for reading, and c stands for controlling. L means different privilege levels, in which $l_1 < l_2$. In the proposed architecture, we can easily combine security level with integrity level to get comprehensive privilege levels of different subjects or objects. Based on the definition above, as shown in Table 2, we design an Access Control List (ACL):

TABLE II
ACCESS CONTROL LIST

Subject \ Object	l_1	l_2
l_1	$\{w, r, c\}$	$\{r, c\}$
l_2	$\emptyset$	$\{w, r, c\}$

It can be concluded that the same level of subjects to have all permissions to objects, high-level subjects have read and control access to low-level objects, and low-level subjects have no permissions to objects above their own level. On the one hand, data should be restricted to flow from low level to high level; on the other hand, data from low level cannot be tampered by high level. These two rules help maintain fine CIA requirements. Furthermore, we consider that there are three entities in the architecture: equipment nodes, management hubs and user nodes. Equipment nodes and management hubs should belong to l_2, while user nodes should belong to l_1. The data can just flow from equipment nodes and management hubs to user nodes; and user nodes have no authorization to write or modify the data. However, equipment nodes and management hubs have all permissions to each other, so that effective data interaction can be implemented.

Nevertheless, we also define a formula to determine if the current state is safe:

$$V = S \times O \times A \times \mu \times L \quad (1)$$

In this formula, $S \times O \times A$ indicates a subject uses some method to access an object; μ indicates the access matrix; and L indicates the privilege levels. Once all the elements are secure and trusted, a secure state can be ensured. Since we have divided the architecture into different levels and made some definitions, which means $S \times O \times A$ and L have been strictly restricted and abided by, μ is the last element that should be considered. Therefore, as shown in Table 3, we have designed some defensive mechanisms to help control access matrix.

Actually, from Table 3 we can see that management hubs play the role of a data transmission intermediary. Except for the rigorous hierarchy, different means of verification are set to help keep the architecture secure. Together with Blockchain technology, not only can they provide effective methods to control all kinds of access which help prevent from malicious

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TII.2019.2894573, IEEE Transactions on Industrial Informatics

TABLE III
DEFENSIVE MECHANISMS

Subject \ Object	Equipment nodes	Management hubs	User nodes
Equipment nodes		Whitelist, PoW, Dynamic verification, Merkle Root	
Management hubs			
User nodes		Whitelist, Dynamic verification, Asymmetric encryption, Merkle Root	

activities, but also can enhance the security and privacy of the architecture, forming a more mature and stable system.

IV. DATA INTERACTION PROCESS DESIGN

However, in essence, the proposed architecture is derived from the IoT. Therefore, in this section, we refer to [29], taking the temperature collection as an example to design the process of data interaction in the architecture to prevent the possible attacks and threats: leakage of permissions, DoS or DDoS attacks, network sniffer, compromised-key attack and invasion. We give specific figures and algorithms to depict our design, and the main notations of this section are given in Table 3.

TABLE IV
NOTATION

Notation	Definition
whitelist[1…*a*]	Record trusted ID. There are backups in each management hub
mComputer[1…*b*]	Record all microcomputers in the system
mHub[1…*c*]	Record all management hubs in the system
requestReceived	Indicate if data arrives
users[1…*d*]	Record all users in the system
time	Record the running time in the system

The proposed architecture should achieve data acquisition first, which depends on the microcomputers. Generally, a microcomputer can manage one or more sensors and connect to one management hub. After the data is acquired, the microcomputers are required to register a unique ID, which will be put in the whitelist in the connected management hub. The whitelist has backups in each management hub. If one management hub crashes, the connected microcomputer can be standby or choose to change its network settings to switch to other management hubs. The flowchart of preparation is shown in Fig.3

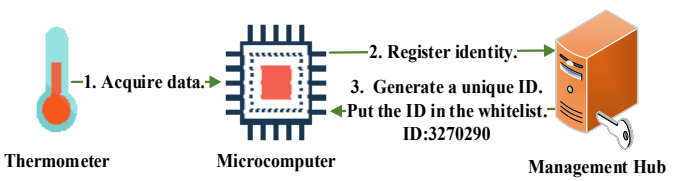

Fig. 3. Data preparation flowchart.

In this process, we consider the attacks of stealing and abusing node permissions (primarily the sensing layer and the management hub layer). Therefore, we design two defense mechanisms. On the one hand, in the sensing layer, we integrate the whitelist mechanism, the dynamic verification mechanism, and the PoW consensus algorithm to prevent malicious traffic and injection of erroneous data. On the other hand, as a multi-center system is created, under the supervision of the other management hubs, the invaded management hub can be discovered, excluded and replaced quickly. These two defense mechanisms can guarantee the stable operation of the underlying system.

Algorithm1: Data interaction in the intranet

```
01 begin
02   for i←1 to mComputer[1…a]
03     find the connected mComputer[j] for mComputer[i]
04     register ID
05   end for
06   wait()   //wait for application
07   if(requestReceived == true)
08     if(compare the mComputer with whitelist[1…a] == true)
09       tick()   //record the running time
10       wait for enough insertions in the buffer for the PoW
11       if(execute PoW == true)
12         generate and broadcast a block record
13         subsequent data is uploaded to the database directly
14       else
15         deny and generate a block record
16         discard the data in the buffer
17       end if
18       if(time == set value)  close the connection
19       end if
20     else
21       deny, generate and broadcast a block record
22     end if
23   end if
24 end
```

The detailed process of applying for the storage permission by the equipment node is shown in Fig. 4, and it can be concluded as Algorithm 1. After obtaining the permission through the whitelist verification, the data is first put into the buffer pool. When reaching a certain amount of data, the management hub will use built-in comparison algorithms to calculate the characteristic values and compare them with the set values, through which PoW is accomplished. If the requirements are satisfied, the data in the buffer pool will be put into the database; meanwhile, the uploaded data can be directly transmitted to the database for a period, and all operations within the permission of the equipment node will be allowed; otherwise, the data collected in the buffer pool will be discarded, and the permission request will be rejected. It should be noted that all data being transmitted to the database need to

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TII.2019.2894573, IEEE Transactions on Industrial Informatics

be converted into ciphertext by the public key. The management

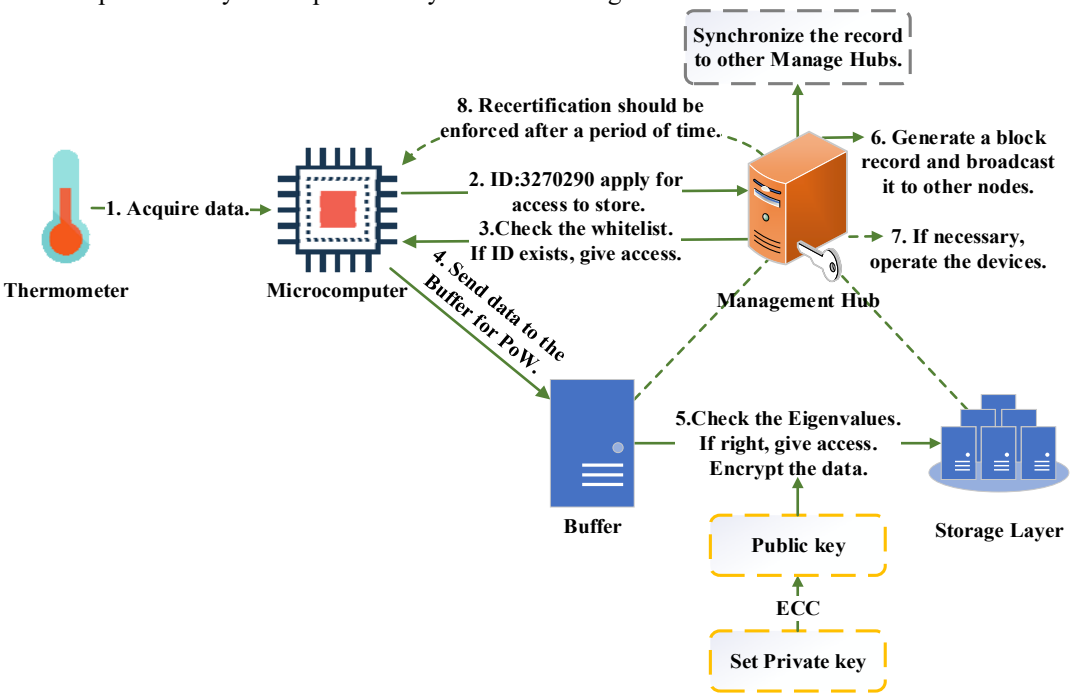

Fig. 4. The intranet flowchart (equipment interaction).

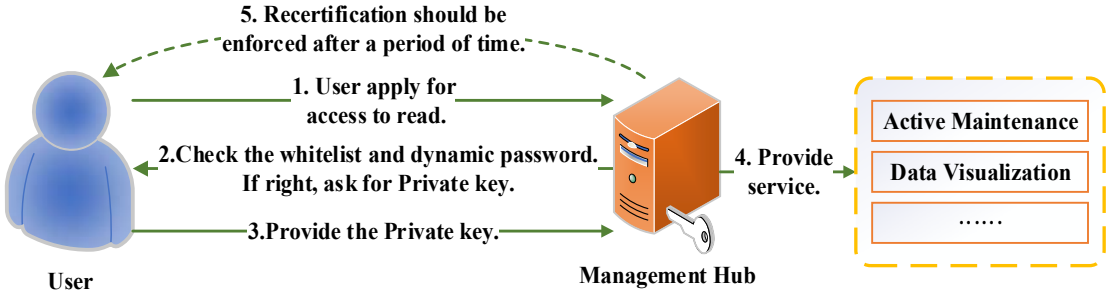

Fig. 5. The extranet flowchart (users apply for service)

hub will generate a new block record for each permission request. The block record is then broadcasted to the other management hubs, which will verify the block record again and record it. However, after a certain period, the dynamic verification mechanism requires the re-authentication, so the system needs to repeat the process of 1-6 in Fig. 4. The flowchart is an example of applying for the storage permission, and the process of reading and control request are the same.

As discussed above, we implement data interaction in the intranet. As shown in Fig. 5, the flowchart of the extranet interaction is similar to that of the intranet. Since the management hub is connected to the Internet, the Dos attacks or DDoS attacks [30] will be very frequent. Therefore, the whitelist mechanism, the dynamic verification mechanism, and the asymmetric encryption mechanism are set up for the extranet. The whitelist mechanism and the dynamic verification mechanism are the same as that of the intranet, performing the screening and eliminating malicious traffic on the Internet. On the other hand, the asymmetric encryption technology is designed specifically for the extranet to prevent unauthorized access. The process is also concluded as Algorithm 2.

Algorithm2: Data interaction in the extranet

```
begin
  for i←1 to user[1…d]
    find the connected mComputer[j] for users[i]
    register ID
  end for
  wait()   //wait for application
  if(requestReceived == true)
    if(compare with the whitelist[1…a] and password == true)
        tick()   //record the running time
        user verify the data, provide private key to get service
        generate and broadcast a block record
          if(time == set value)  close the connection
          end if
    else  deny, generate and broadcast a block record
    end if
```

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TII.2019.2894573, IEEE Transactions on Industrial Informatics

16 **end if**
17 **end**

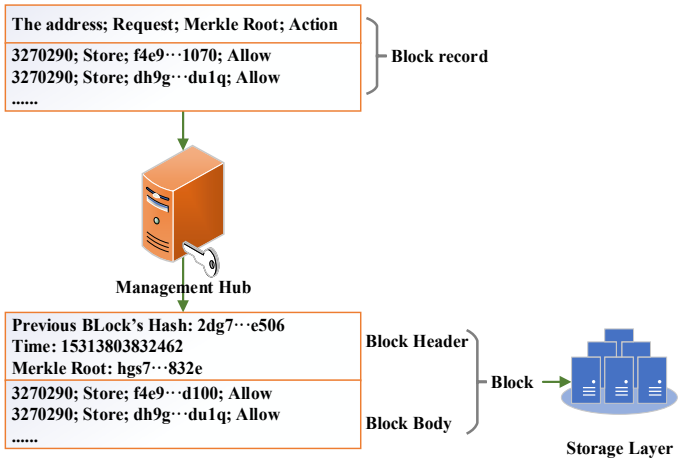

Fig. 6. Block generation.

In the proposed architecture, the management hub is also a record node. Each block and equipment data have a copy in each management hub. When block records reach the limit, the management hub with a low overhead will generate a block to record all the access applications during that period. As shown in Fig. 6, after the block is generated, it will be placed in the storage layer and synchronized with the other management hubs. In addition to the above defense technologies, we introduce double Merkel roots in the block record to protect the data. The first one is applied on the data in the buffer pool existing in the block record; the second one is applied on the data in the block body existing in the block header. Such kind of nesting guarantees that the data will not be sniffed and the malicious invasion is difficult to achieve.

V. A Case Study: A Blockchain-Based Automatic Production Platform

In [31], a cloud-based manufacturing big data solution for active preventive maintenance is proposed; in [32], a study of Context-Aware Cloud Robotics and its applications are discussed. In fact, for the IIoT, many researchers focus on higher real-time capability, stronger computing power, more intelligent applications, etc., among which security and privacy are rarely noticed. Therefore, in this section, we transform an automatic production platform to discuss the improvement of the security and privacy for the IIoT application with the Blockchain technology.

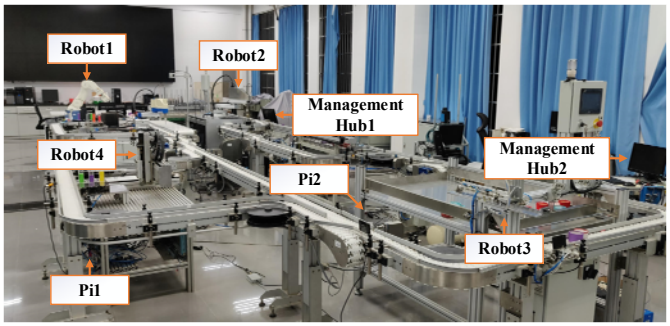

Fig. 7. Automatic production platform.

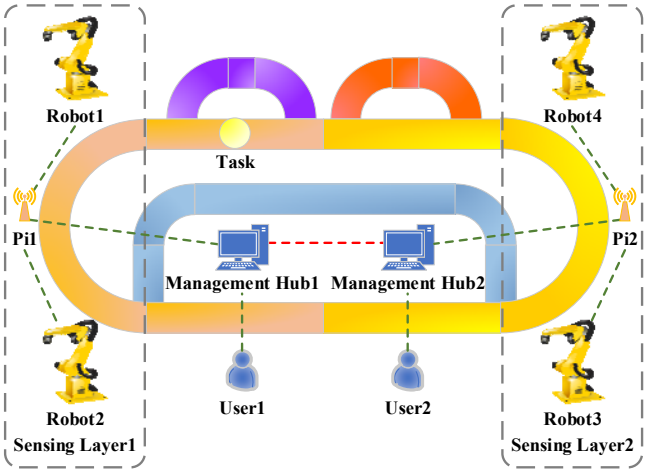

Fig. 8. The architecture of the automatic production platform.

As shown in Fig. 7 and Fig. 8, an automatic production platform according to the proposed architecture are realized. Four industrial robots were designed on the platform to perform different tasks. Then, we set up two sensing layers equipped with 3B Raspberry Pis to collect data, such as operating temperature, operating time and so on. Finally, we set up two management hubs equipped with Intel I5 platform, and relevant algorithms and software would be programmed in the management hubs. After that, the equipment data were collected and sent to the Raspberry Pis (sensing layer) via the sensors. After certain pre-processing, the data were uploaded to the management hub (management hub layer) for further processing (storage layer and firmware layer). Users accessed the management hubs through independent computers outside the working area to obtain different services (application layer).

We consider the system we construct in Fig. 8. Four industrial robots are divided into two groups, and data will be collected and pre-processed by different Raspberry Pis. After the collection, all data will be uploaded for further processing, usually are failure prediction, data visualization and so on, and users can require wanted services. As shown in Fig. 9, we depict the data flow in the system. *N* means any node in the architecture, and the direction of arrows indicates the flow direction of data, which is protected. This setting accomplishes the collection and processing of the data, and conforms to the security and privacy model we discussed above.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TII.2019.2894573, IEEE Transactions on Industrial Informatics

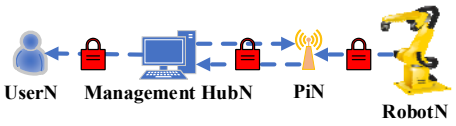

Fig. 9. Data interaction in the transformed platform.

Actually, based on the context of the traditional IIoT system, there is no any verification mechanism in the proposed system to keep the sensors trusted all the time. All sensors are designed to be attached to the Raspberry Pi or the management hubs, which means they are a tight coupling relation. Once the verification is completed at the beginning, sensors only need to upload data constantly. This problem is the same as user nodes. Therefore, to prevent malicious operations and invalid invasion, different defensive methods are set. We consider the Raspberry Pis and sensors as equipment nodes, together with the user nodes and the management hubs to construct the defense measures, which is shown in Table 4.

TABLE V
DEFENSIVE MECHANISMS FOR THE PLATFORM

	Equipment nodes to Management hubs	User nodes to Management hubs
Whitelist	Connected through the Ethernet and assigned a fixed IP address.	Connected through the Internet. The malicious access IP will be banned for a period.
PoW	Setting control value by SPC theory.	
Dynamic Verification	PoW should be re-accomplished for access authorization after a period.	The dynamic password should be re-provided for access authorization after a period.
Merkle Root	Generated from the data in the buffer.	Generated from the data in the block body.
Asymmetric Encryption	Using public key to encrypt the uploaded data.	Using private key to decrypt the return data for services.

TABLE VI
COMPARISON OF THE ARCHITECTURES

Properties	Cloud-based IIoT architecture	Blockchain-based IIoT architecture
Identity and Authentication	Specific accounts	Specific IP address or accounts
Access control	Static password	PoW, dynamic password
Storage	Plaintext	Ciphertext
Protocol and network security	Pre-defined with static password	Pre-defined with Whitelist, PoW, Dynamic Verification
Privacy and Non-Reputation		Asymmetric Encryption, Hash, Merkle Root
Real-time capability	High	High
Resource overhead	Medium	Medium
Fault tolerance	Medium	High
Scalability	High	Medium

In general, as shown in Table 5, we compare the proposed architecture and the traditional cloud-based IIoT architecture in a concise manner. With the Blockchain technology, we can see that all the stages have been considered. The data for different services can be ensured good confidentiality and integrity since it has been encrypted and verifiable; and fine availability can be guaranteed since malicious traffic can be eliminated. What's more, the system still maintains low overhead.

VI. CONCLUSIONS

In this paper, we propose an innovative blockchain-based IIoT architecture to help build a more secure and reliable IIoT system. By analyzing the shortcomings of the existing IIoT architecture and the advantages of the Blockchain technology, we introduce a new IIoT architecture and give a detailed analysis of all architecture layers. Also, we introduce BLP model as well as Biba model to design secure assurance in theory. On this basis, we describe the key technologies, the flow, and the defense mechanisms of the proposed architecture. Finally, we retrofit the existing automatic production platform to discuss the improvement compared with the traditional IIoT architecture. It shows that the proposed architecture can enhance the CIA requirements greatly. Therefore, the proposed architecture can achieve more expansion in the future, such as through the integration of smart contract, achieving the automatic configuration of resources; through the distributed system, online remote upgrade of all equipment can be realized.

ACKNOWLEDGMENT

This work was supported in part by the National Key Research and Development Project of China (No. 2017YFE0101000), the Joint Fund of the National Natural Science Foundation of China and Guangdong Province (No. U1801264), and the Key Program of Natural Science Foundation of Guangdong Province (No. 2017B030311008). This work is also supported by the Deanship of Scientific Research, King Saud University through Research Group No. RG-1435-051.

REFERENCES

[1] M. Wollschlaeger, T. Sauter, and J. Jasperneite, "The Future of Industrial Communication: Automation Networks in the Era of the Internet of Things and Industry 4.0," *IEEE Industrial Electronics Magazine,* vol. 11, no. 1, pp. 17-27, 2017.

[2] J. Wan, B. Yin, D. Li, A. Celesti, F. Tao, and Q. Hua, "An Ontology-based Resource Reconfiguration Method for Manufacturing Cyber-Physical Systems," *IEEE/ASME Transactions on Mechatronics*, vol. 23, no. 6, pp. 2537-2546, 2018.

[3] J. Wan, S. Tang, D. Li, M. Imran, C. Zhang, C. Liu, and Z. Pang, "Reconfigurable Smart Factory for Drug Packing in Healthcare Industry 4.0," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 1, pp. 507-516, 2019.

[4] A. Sadeghi, C. Wachsmann, and M. Waidner, "Security and privacy challenges in industrial Internet of Things," in *2015 52nd ACM/EDAC/IEEE Design Automation Conference (DAC)*, 2015, pp. 1-6.

[5] B. Chen, J. Wan, L. Shu, P. Li, M. Mukherjee, and B. Yin, "Smart Factory of Industry 4.0: Key Technologies, Application Case, and Challenges," *IEEE Access,* vol. 6, pp. 6505-6519, 2018.

[6] B. Miller and D. Rowe, "A survey SCADA of and critical infrastructure incidents," presented at the *Proceedings of the 1st Annual conference on Research in information technology*, Calgary, Alberta, Canada, 2012.

[7] A. Cui and S. J. Stolfo, "A quantitative analysis of the insecurity of embedded network devices: results of a wide-area scan," presented at the *Proceedings of the 26th Annual Computer Security Applications Conference*, Austin, Texas, USA, 2010.

[8] S. Sharmeen, S. Huda, J. H. Abawajy, W. N. Ismail, and M. M. Hassan, "Malware Threats and Detection for Industrial Mobile-IoT Networks," *IEEE Access,* vol. 6, pp. 15941-15957, 2018.

[9] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," *Consulted,* 2008.

[10] L. Wu, X. Du, W. Wang, and B. Lin, "An Out-of-band Authentication Scheme for Internet of Things Using Blockchain Technology," in *2018 International Conference on Computing, Networking and Communications (ICNC)*, 2018, pp. 769-773.

[11] M. Samaniego and R. Deters, "Internet of Smart Things - IoST: Using Blockchain and CLIPS to Make Things Autonomous," in *2017 IEEE International Conference on Cognitive Computing (ICCC)*, 2017, pp. 9-16.

[12] A. Bahga and V. K. Madisetti, "Blockchain Platform for Industrial Internet of Things," *Journal of Software Engineering & Applications,* vol. 09, no. 10, pp. 533-546, 2016.

[13] C. Pahl, N. E. Ioini, and S. Helmer, "A Decision Framework for Blockchain Platforms for IoT and Edge Computing," in *International Confernce on Internet of Things, Big Data and Security*, 2018.

[14] D. Wu, D. W. Rosen, L. Wang, and D. Schaefer, "Cloud-based design and manufacturing: A new paradigm in digital manufacturing and design innovation," *Computer-Aided Design,* vol. 59, pp. 1-14, 2015.

[15] A. W. Colombo, T. Bangemann, S. Karnouskos, J. Delsing, P. Stluka, R. Harrison, F. Jammes, and J. Lastra, "Industrial cloud-based cyber-physical systems: The IMC-AESOP approach," *Springer Ebooks,* pp. 15-16, 2014.

[16] P. O'Donovan, K. Leahy, K. Bruton, and D. T. J. O'Sullivan, "An industrial big data pipeline for data-driven analytics maintenance applications in large-scale smart manufacturing facilities," *Journal of Big Data,* vol. 2, no. 1, p. 25, 2015.

[17] L. Bo, I. Foster, F. Siebenlist, R. Ananthakrishnan, and T. Freeman, "A Multipolicy Authorization Framework for Grid Security," in *Fifth IEEE International Symposium on Network Computing and Applications (NCA'06)*, 2006, pp. 269-272.

[18] C. Y. Liu, J. Lin, and B. Tang, "Dynamic trustworthiness verification mechanism for trusted cloud execution environment," *Journal of Software*, vol. 25, no. 3, pp. 662-674, 2014.

[19] Z. Zheng, S. Xie, H. Dai, X. Chen, and H. Wang, "Blockchain Challenges and Opportunities: A Survey," *International Journal of Web & Grid Services,* Vol. 14, No. 4, pp. 352-375, 2018

[20] A. Gervais, G. O. Karame, K. W, #252, st, V. Glykantzis, H. Ritzdorf, and S. Capkun, "On the Security and Performance of Proof of Work Blockchains," presented at the *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, Vienna, Austria, 2016.

[21] M. Colledani and T. Tolio, "Impact of Quality Control on Production System Performance," *CIRP Annals - Manufacturing Technology,* vol. 55, no. 1, pp. 453-456, 2006.

[22] J. Wan, B. Chen, S. Wang, M. Xia, D. Li, and C. Liu, "Fog Computing for Energy-aware Load Balancing and Scheduling in Smart Factory," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 10, pp. 4548-4556, 2018.

[23] Z. Zheng, S. Xie, H. Dai, X. Chen, and H. Wang, "An Overview of Blockchain Technology: Architecture, Consensus, and Future Trends," in *2017 IEEE International Congress on Big Data (BigData Congress)*, 2017, pp. 557-564.

[24] H. Gilbert and H. Handschuh, "Security Analysis of SHA-256 and Sisters," in *Selected Areas in Cryptography*, Berlin, Heidelberg, 2004, pp. 175-193.

[25] M. Bafandehkar, S. M. Yasin, R. Mahmod, and Z. M. Hanapi, "Comparison of ECC and RSA Algorithm in Resource Constrained Devices," in *2013 International Conference on IT Convergence and Security (ICITCS)*, 2013, pp. 1-3.

[26] N. Komninos, E. Philippou, and A. Pitsillides, "Survey in Smart Grid and Smart Home Security: Issues, Challenges and Countermeasures," *IEEE Communications Surveys & Tutorials,* vol. 16, no. 4, pp. 1933-1954, 2014.

[27] A. Dorri, S. S. Kanhere, R. Jurdak, and P. Gauravaram, "Blockchain for IoT security and privacy: The case study of a smart home," in *2017 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops)*, 2017, pp. 618-623.

[28] G. Y. Lin, H. E. Shan, H. Hao, W. U. Ji-Yi, and C. Wei, "Access control security model based on behavior in cloud computing environment," *Journal on Communications*, vol. 33, no. 3, pp. 59-66, 2012.

[29] M. Abomhara and G. M. Køien, "Security and privacy in the Internet of Things: Current status and open issues," in *2014 International Conference on Privacy and Security in Mobile Systems (PRISMS)*, 2014, pp. 1-8.

[30] X. Yang, T. Ma, and Y. Shi, "Typical DoS/DDoS Threats under IPv6," in *2007 International Multi-Conference on Computing in the Global Information Technology (ICCGI'07)*, 2007, pp. 55-55.

[31] J. Wan, S. Tang, D. Li, S. Wang, C. Liu, H. Abbas, and A. V. Vasilakos, "A Manufacturing Big Data Solution for Active Preventive Maintenance," *IEEE Transactions on Industrial Informatics*, vol. 13, no. 4, pp. 2039-2047, 2017.

[32] J. Wan, S. Tang, Q. Hua, D. Li, C. Liu, and J. Lloret, "Context-Aware Cloud Robotics for Material Handling in Cognitive Industrial Internet of Things," *IEEE Internet of Things Journal*, vol. 5, no. 4, pp. 2272-2281, 2018.