Journal of Computers Vol. 29 No. 6, 2018, pp. 158-167
doi:10.3966/199115992018122906015

A Safe and Efficient Storage Scheme Based on BlockChain and IPFS for Agricultural Products Tracking

JinTao Hao*, Yan Sun, Hong Luo

Beijing Key Laboratory of Intelligent Telecommunications Software and Multimedia, Beijing University of Posts and Telecommunications, Beijing 100876, China
{haojintao, Sunyan, luoh}@bupt.edu.cn

Received 9 January 2018; Revised 22 January 2018; Accepted 22 January 2018

Abstract. The adoption of agricultural products traceability management based on Internet of Things (IoT) technology provides excellent benefits for the current food safety issues. The provenance data can demonstrate agricultural products movement process from the countryside to the dining table. However, the massive provenance data incurs an inefficient query. Meanwhile, the provenance data can be tampered deliberately which affect food safety. There are seldom reported approaches that can solve the above problem effectively. In this paper, we propose a data storage model based on Inter-Planetary File System (IPFS) and blockchain. First, IPFS is used to store video, images, and real-time monitoring data reported from the sensors. Then, in order to avoid a malicious user in case of data faking attack, we exploit the blockchain to store the IPFS hash address of the provenance data. Based on that, we design an authentication mechanism based on blockchain. It can verify the data and ensures effective data security. The experimental results show that the proposed approach can outperforms the existing methods.

Keywords: agricultural products tracking, blockchain, IPFS, storage scheme

1 Introduction

For agricultural products movement process from the countryside to the dining table, it needs to go through the cultivation, processing, transportation and sales. Any one of the above mentioned links can produce serious food safety hazards if there is artificial fraud. Therefore, with the development of IoT technology, many logistics management systems have emerged. Although these systems can automatically trace the entire process, they can not avoid the food safety problem caused by modifying the data artificially. The reason is that some people deliberately tamper and destroy the data in the traditional data storage process. In order to solve mentioned-above problems, researchers began to try to use blockchain technology to store data and protect data security [1-2].

[3] designs a provenance data storage scheme for agricultural products based on blockchain. After the agricultural product is bound to the IoT sensors, they will upload the collected data to the server in real time. Then, the server will automatically store the data in the blockchain after data processing. In this case, blockchain technology is used for implementing provenance data secure storage. Therefore, it can guarantee provenance data authenticity effectively. Massive real-time monitoring data would be generated, when a large number of agricultural products join the quality tracing platform. Blockchain was originally created for digital currency transactions, the amount of data generated is much smaller than real-time monitoring data. For this reason, the speed of block generation is very difficult to keep up with the storage of traceability data. Moreover, storage security of the raw data such as monitoring video in each process is needed to be guaranteed. So, blockchain technology can not be directly applied.

In this paper, we combine the Inter-Planetary File System (IPFS) and blockchain to present a data storage and query mechanism based on agricultural products provenance platform. IPFS is a global, peer-

* Corresponding Author

to-peer distributed file system that seeks to connect all computing devices with the same system of file for a large amount of data storage. First, we propose a data storage model based on IPFS and blockchain. This model encapsulates and parses the uploaded video, image and sensor data automatically. Then, the above data is written to IPFS and the corresponding hash addresses are stored in the blockchain. Next, the hash values of the blockchain transaction are stored in the database. When users query the provenance data of a product, they can retrieve the data by exploiting the transaction content from the blockchain (e.g., the provenance data hash address of in IPFS).

The rest of the paper is organized as follows. Section 2 presents related work. Section 3 overviews the architecture of the product tracking system. Section 4 address the key algorithms including Writing data to IPFS algorithm (WDT_IPFS), query transaction and Verify trace data algorithm (QTAVTD), Data storage algorithm, and Data query algorithm, respectively. Experimental results are discussed in Section 5, while conclusion of this paper is given in Section 6.

2 Related Work

Some works in blockchain and IPFS have emerged in very recent years. As comprehensive surveys are already presented in [4] and [5], a short summary of related works are given in this section.

The concept of Bitcoin is first introduced by Nakamoto in [6]. They build the Bitcoin system which is used for electronic cash. Its key technology is the blockchain which keeps ordered records called blocks. Subsequently, many groundbreaking researches are proposed on the base of this technology. Wan et al. [7] analysis the financing model of Chinese Internet financial enterprises and employ the blockchain technology to reduce the ABS threshold for small and medium enterprises. Cui et al. [8] apply the blockchain principle to solve the problem of multi-level demand response reliable mechanism between users, load aggregators and power grids. A new algorithms and hardware mechanism for Bitcoin systems is provided by Taylor [9]. They represent the computer hardware from the view of mining and think customized hardware may be better than the generalized hardware facilities on the performance. Göbel et al. [10] focus on the importance of communication delay in blockchain system. A customized Markov model is used for tracking the blockchain state. Moreover, they study a discrete event simulation to predict the behavior of Bitcoin miners. Wright et al. [11] summarize the strengths and weaknesses, potential social effects, application risks in government and lay of this decentralize technology. Huckle et al. [12] offer the solution of employing IoT and blockchain technology to improve shared economy. For example, the automatic payment mechanism, forex platform, digital rights management and cultural heritage management would reap the benefits from the blockchain technology based distributed IoT architecture. Eyal et al. [13] present the Bitcoin-NG (Next Generation), a new blockchain protocol designed to scale. It is a Byzantine fault tolerant blockchain protocol that is robust to extreme churn and shares the same trust model as Bitcoin. And they introduce some novel metrics of interest in quantifying the security and efficiency of Bitcoin-like blockchain protocols.

As for InterPlanetary File System (IPFS), it is proposed by Benet [14] to connect all related devices with the same system of files. In other words, IPFS is a peer-to-peer distribute file system which can furnish a high throughout block storage with a content-addressed method. Actually, there are many other attempts aiming to build a global file system. Howard et al. [15] establish the AFS system which can improve the ability of cache validation, sever process structure and so on. What's more, some large media file-sharing system, such as Napster, KaZaA, and BitTorrent, are introduced to store massive data. Alam et al. [16] present the InterPlanetary Wayback as a permanent Web archive to distribute data files into IPFS network. Header and payload are splitted for every response records, then they are disseminated into IPFS, thus a CDXJ index is constructed. The average indexing rate can be boosted by the method.

3 System Overview

3.1 System Structure

We designed the traceability and tracking system of agricultural products based on IPFS, blockchain and Internet of Things technology, as shown in Fig. 1. Firstly, obtain real-time data of the product quality through sensors, and collect video and picture data in the process of production, processing and logistics, and then encapsulate and process the above data and store them into IPFS. In order to ensure the authenticity of data stored in IPFS, it will write the hash generated by the IPFS to the blockchain, so that the user can verify the authenticity of the product data.

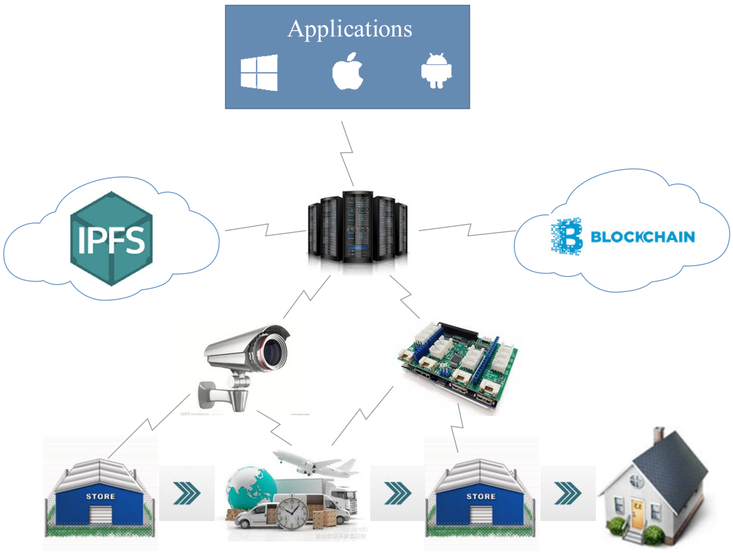


Fig. 1. Agricultural products tracking system architecture

In the process of production, processing and transportation, agricultural products need to collect necessary video, image and environmental data in order to ensure the authenticity and traceability of the whole process. The temperature and humidity sensor, position sensor, image sensor, and the camera are placed on the working room, the warehouse or the transport vehicle to upload data automatically. After receiving the data, the server parses and encapsulates the data and writes it to the IPFS, and then the hash address is stored in the blockchain to complete the data storage. We can design all kinds of applications in the upper system to achieve agricultural safety tracing function based on these data.

3.2 Data Storage Model

We design a data storage model based on open source IPFS and blockchain framework ethereum, as shown in Fig. 2. The module of the system mainly includes data encapsulation, data analysis and data management system module. The data encapsulation module mainly obtains uploaded video, picture and sensor data and then encapsulates them. Data management system module is to interact with IPFS, database and blockchain. After obtaining the encapsulated data, the data management system queries the transaction hash of the agricultural product in the blockchain from the database, and get the transaction content (IPFS hash address of the tracing data) from the blockchain. Then the previous provenance data packet is obtained. The new data will be stored in the provenance packet to generate a new provenance packet. When extracting data, the system uses the blockchain transaction hash to query transaction and obtains the IPFS hash address. Then get the data from IPFS. The data analysis module analyzes the data taken from the data management system and returns it to the application layer to build the application.

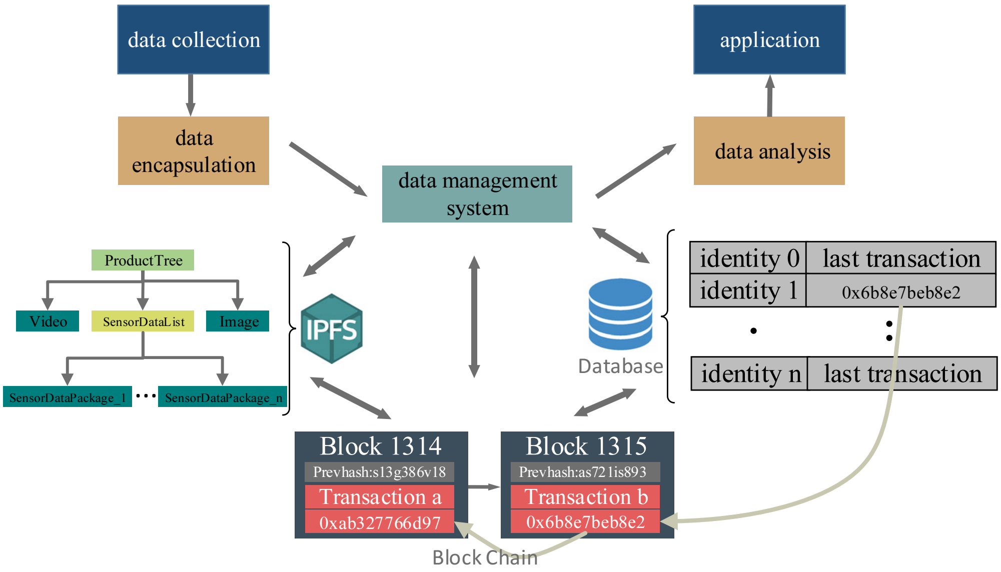


Fig. 2. Storage model

In storage scheme, the tree object in IPFS is used to store the provenance data for each product. ProductTree is the root object for a product. In the tree structure of the product, the video data, the image data, and the real-time data reported by the sensor are sequentially stored. Since each sensor senses a relatively small amount of data, we encapsulate the amount of data reported in one hour into a sensor DataPackage based on the amount of time and amount of data reported by the sensor in real time. And the sensor DataPackage generated in a day are encapsulated into a sensor DataList to solve the problem that the amount of perceptual data is too small to cause space wasted in IPFS. Therefore, reduce the block security verification time.

4 Data Storage Scheme

4.1 IPFS Storage Method

This article uses the DAG in IPFS to manage all provenance data for agricultural products. For example, store the relevant product and logistics data of rice in IPFS. The ProductTree object is used to store the provenance data for each farmer in the following format. Use the data property to store the type and productID. Use the links attribute to store links to traceability data, where hash represents the hash address of the data on IPFS.

```
{
  "data": {"type": "ProductTree", "productID": "Rice_3067"},
  "links": [
      {"hash": "QmQZzTMN2X54SxC2jMuAug6Qcz1K-YS5ZB12i3gGusvBnkn",
      "name": "Image_20170910_package", "size":"7987"},
      {"hash":"QmUgeQaCjhZ8V42DKBLtTCvzxojUt-LpaX6QmXX9rTLNnxF",
      "name": "SensorDataList_20170911", "size":"5765"}
      ]
}
```

Images and videos which are large and one-time upload data directly are stored on the IPFS, while sensor data which is small and frequently uploaded data is stored using custom object. SensorDataList is used to store one day of the sensor package links, the structure is as follows:

```
{
  "data": {"type": "SensorDataList"},
  "links": [{"hash": "QmQduvUQBdnNtuhBB6fUuf4NZ-CGhxDo3GwT3gjgZMX14o6",
  "name":"2017091112","size":"519"}]
}
```

Data attribute stores type. links attribute stores sensor DataPackage links, and hash is the IPFS hash address of the sensor DataPackage. Sensor data is packaged into a sensor DataPackage in hours in the following format:

```
"860719023995818;32;251;102451;0.25;1482799635;3957.57214N;11620.99005E"
```

These data represent: sensor ID, humidity, temperature, pressure, Acceleration, date time, latitude, longitude. The specific process for storing the data into IPFS is described in Algorithm 1.

Algorithm 1. *Writing data to IPFS algorithm (WDT_IPFS)*

```
Input: data , old ProductTree IPFS hash (OPT_hash)
Out: data IPFS hash , new ProductTree IPFS hash (NPT_hash)
OPT_Data ← Get oldProductTree object according to OPT_hash;
if Data is video or image then
    Data_hash ← Store the data directly to IPFS, and obtain the data IPFS hash address;
    NPT_Data_hash ← Link the data_hash into OPT_Data, generate new ProductTree object, and obtain NPT_Data_hash;
else
    Data_hash ← Store the data to IPFS by creating a custom object, and obtain the data IPFS hash address;
    NPT_hash ← Link the data into the sensorDataList object and obtain new ProductTree IFPS hash;
end if
return Data_hash, NPT_ hash;
```

4.2 Double Chain Verification Mechanism Based on Blockchain

There is a risk of storing the ProductTree IPFS hash address directly into the blockchain. The tamper only need to modify the latest blockchain transaction hash, and links the contents of the transaction to the spoofed data IPFS hash address to achieve the purpose of tampering with the data. To deal with this problem, we store the hash of the newly stored data besides the IPFS hash of ProductTree. We can match the hash address of the provenance data in ProductTree while querying.

Each time the data is stored in IPFS, the updated ProductTree hash address and the hash address of the corresponding data block are reported to the management system. The management system encapsulates these addresses into a transaction and write it into blockchain. The transaction is as follows:

```
{
    "data":"QmUgeQaCjhZ8V42DKBLtTCvzxojUtLpaX6Q-mXX9rTLNnxF",
    "productTree": "QmYc13LpsJJ7q-En6MRUR4asynv5c8ykQ86XWYD9a3fQNVQ"
}
```

Encapsulated as blockchain transaction data:

```
{
    "jsonrpc": "2.0", "method": "eth_sendTransaction", "id": 1,
    "params": [{
      "from": "0xb60e8dd61c5d32be8058bb8eb-970870f07233155",
      "to": "0xd46e8dd67c5d32be8058bb8eb9-70870f07244567",
      "gas": "0x76c0",
      "gasPrice": "0x9184e-72a000",
      "value": "0x9184e72a",
        "data":"0x383630373139303233393935383138 3b33323b3235313b31303234353
        13b302e32353b31343832373939363335 3b333935372e35373231343b4e3b31
        313632302e39393030353b45"
      }]
  }
```

The data is sent to the blockchain node in json format. The blockchain node initiates the transaction and writes to the block after receiving the transaction request. All the blockchain nodes will be synchronized block data, the data will persist in the block after the consensus is completed.

In the blockchain, each block contains information of the last block and constitutes a chain storage structure. However, there is no association between the transactions stored in the block. So we use the preliminary research results of our task group. We store the last transaction hash for each productID in the secondary database. When we need to write new data, we will find the last transaction hash and write as parent hash in this blockchain transaction. The parent transaction of each transaction can be obtained easily, so that we can move forward to find all the transactions. The productID of the i-th blockchain transaction data is defined as Tx_i:

$$Tx_i = \begin{cases} f(I_i, P_i, h_{i-1}).\, if\ i > 0 \\ g(I_i, P_i).\, if\ i = 0 \end{cases} \tag{1}$$

Where I_i represents the IPFS hash address of the stored provenance data for this time. P_i is the latest IPFS hash address of ProductTree after storing this provenance data, which links to all provenance data for that product. h_i is the hash value of the i-th blockchain transaction. The function f reassembles I_i, P_i, and h_{i-1} into new data content and The function g reassembles I_i and P_i into new data content. The process of querying and verifying provenance data is described in Algorithm 2.

Algorithm 2. *Query transaction and verify trace data algorithm (QTAVTD)*

```
Input: productID
Output: result (success or fail)
LTx_hash ← get the last transaction in the database;
while LTx_hash is not null do
    LTx ← get transaction from blockchain by LTx_hash;
    if LTx is not null then
        Data ← get data from LTx;
        Data_hashlist ← get trace data IPFS hash from Data;
        LTx_hash ← get parent transaction hash from Data;
    end if
end while
PT_Data ← Get ProductTree object according to the newest ProductTree hash
if match(P T Data; Data hashlist) == true then
    return success;
else
    return fail;
end if
```

4.3 Data Storage Scheme Based On IPFS and BlockChain

A secure tracing system based on IPFS and blockchain is presented in this paper. We use IPFS to store a large amount of provenance data and use the blockchain to store the IPFS hash address of the data. When reading provenance data from IPFS, it can be used as valid validation data once it is found to ensure the security of data in IPFS and avoid malicious tampering.

To ensure the security of data stored in IPFS, we designed a blockchain-based double-chain storage structure for recording the hash values in the above data structures. So that verify the authenticity of the data when the user doubts the data. Provenance data storage and query verification algorithm is as follows:

Algorithm 3. *Data storage algorithm*

```
Input:  trace data, productID
Output: result (success or fail)
Tx_hash ← Get transaction hash from database by productID;
OPT_hash ← Get old ProductTree IPFS hash from blockchain by Tx_hash;
Data_hash, NPT_hash← WDT_IPFS(data, OPT_hash);
Transaction← Package(Data_hash, NPT_hash);
Send blockchain transaction and store the transaction hash to database by productID;
return result (if no error then success, or fail);
```

Algorithm 4. *Data query algorithm*

```
Input:  productID
Output:  data, result (success or fail)
Tx_hash ← Get transaction hash from database by productID;
PT_hash ← Get ProductTree IPFS hash from blockchain by Tx_hash;
data ← Get the ProductTree object and then obtain the data linked in ProductTree;
if QTAVTD(productID) == success then
    return data, success;
else
    return data, fail;
end if
```

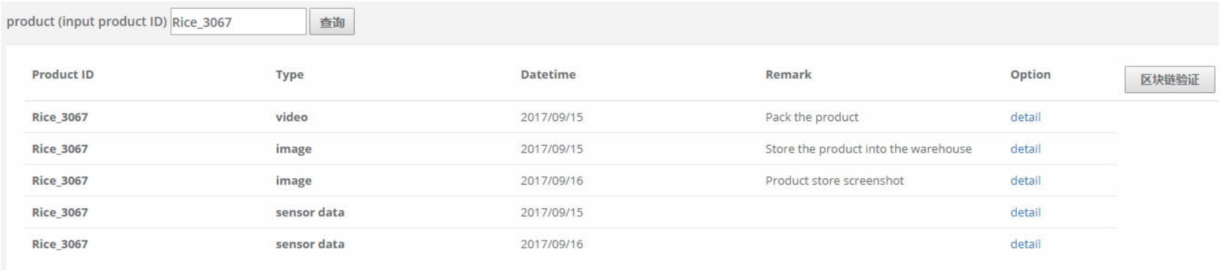
Each block in the blockchain constitutes a non-tamperable chain, and each transaction stores the hash of the last transaction as a forward-looking chain. Such a double-chain structure ensures traceability and usability of data.

5 Experimental Evaluation

This experiment uses go-ethereum 1.9 as the blockchain platform and the IPFS version is go-ipfs v0.4.11. Then we build the management system with jdk-8u101. We deploy the blockchain node in five machines, and each machine possesses a 3.4 GHz core Intel processor with 8GB memory. All the nodes are deployed in Ubuntu 14.04 OS. One of the machines deploys a storage system for receiving traceability data, and other machines are for create blocks.

5.1 Case Study

We designed the application system for traceability of agricultural products. The IOT sensors are binding with agricultural products and upload real-time data every ten minutes. The system captures video and image data during transport to the storage system. The user can query the traceability data in the system according to the ID of the product. As shown in Fig. 3, after entering the product ID in the system, user can query the trace data stored in the IPFS corresponding to the ID.



product (input product ID) Rice_3067

Product ID	Type	Datetime	Remark	Option
Rice_3067	video	2017/09/15	Pack the product	detail
Rice_3067	image	2017/09/15	Store the product into the warehouse	detail
Rice_3067	image	2017/09/16	Product store screenshot	detail
Rice_3067	sensor data	2017/09/15		detail
Rice_3067	sensor data	2017/09/16		detail

Fig. 3. Traceability data of product Rice_3067

5.2 Efficiency Comparison

This experiment compares the storage scheme based on blockchain and IPFS and the scheme proposed in [3]. The experimental results are shown in Fig. 4.

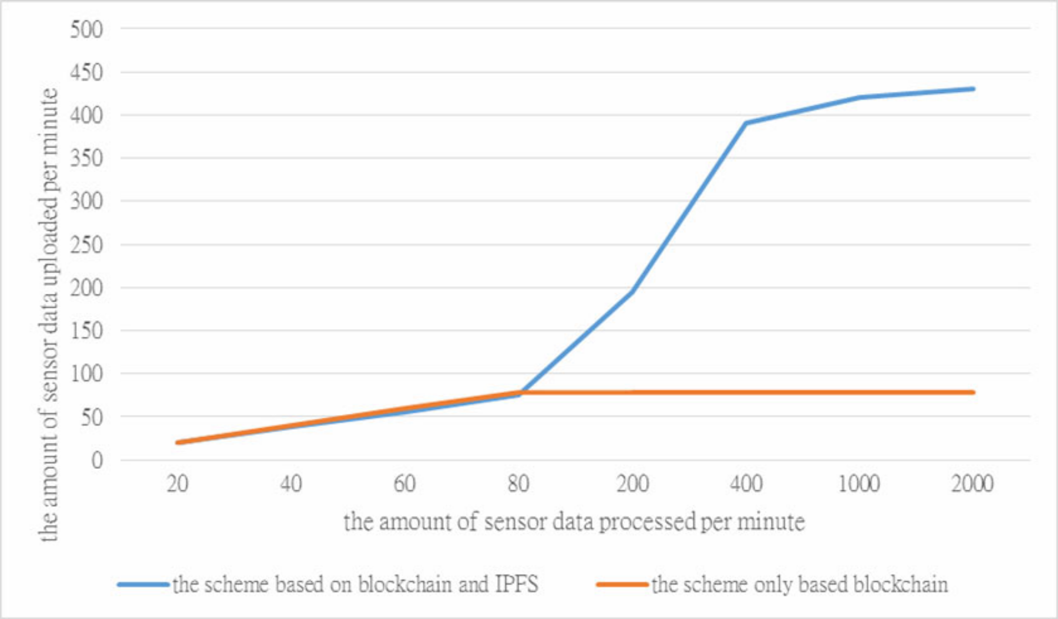


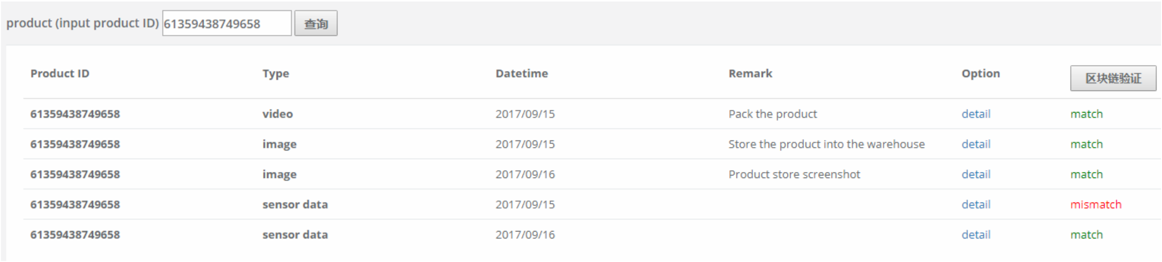
Fig. 4. Storage rate comparison

The horizontal axis is the amount of sensor data uploaded per minute, and the vertical axis is the amount of sensor data processed per minute. When the amount of data is very small, this program's storage speed is less than [3] due to the addition of IPFS storage time. But with the increase in the amount of data, our program continues to improve the amount of data processed. While the program of [3] soon encounters a bottleneck. Although in the end our program will eventually encounter bottlenecks, but this is inevitable. Our scheme is superior to [3] in dealing with large numbers of cases.

Due to the size of the transaction in each block, the scheme only based on blockchain can handle about 70 sensor data per minute. The storage efficiency of the scheme propose by us has been significantly improved. It can handle about 450 sensor data per minute. This is because our method packages the data of a hour into a packet, and the sensor upload 6 times per hour. Then the final efficiency increased by 6 times. It is foreseeable that expanding the capacity of packet will further improve storage efficiency. But this will reduce the real-time characteristics of provenance data. So we need to make a trade-off between the them according to different scenes in practical application. We will do further research in our future work.

5.3 Data Verification

If the user is skeptical about the traceability of the product, the data can be verified by the blockchain. At this point the system will return the provenance data validation results which use blockchain validation, its results are depicted in Fig. 5.



Product ID	Type	Datetime	Remark	Option	
61359438749658	video	2017/09/15	Pack the product	detail	match
61359438749658	image	2017/09/15	Store the product into the warehouse	detail	match
61359438749658	image	2017/09/16	Product store screenshot	detail	match
61359438749658	sensor data	2017/09/15		detail	mismatch
61359438749658	sensor data	2017/09/16		detail	match

Fig. 5. Trace data validation results

6 Conclusion

In order to realize agricultural products data authenticity in open environments, researchers adopt blockchain technology to store these data. In this case, the stored data is tamper-resistant. However, blockchain was originally created for digital currency transactions. For this reason, block generation speed and storage efficiency are very difficult to keep up with the provenance data generation speed. To solve the problem, we combine the IoT, IPFS and blockchain to design an agricultural products provenance platform. First, for the massive data generated from products movement process, we put forward a data storage model based on IPFS and blockchain. In addition, storage and query algorithms based on IPFS are proposed. Finally, in order to avoid a malicious user in case of data faking attack in IPFS, we present an authentication mechanism based on blockchain. It writes the provenance data into IPFS and the corresponding hash addresses are stored in the blockchain, which ensures effective provenance data security. The experimental results show that the storage efficiency of proposed approach can outperforms the existing methods when storing large amounts of data.

Acknowledgments

This work is supported by the National Natural Science Foundation of China under Grant No. 61672109, No. 61772085 and No. 61370196.

References

[1] F. Tian, An agri-food supply chain traceability system for China based on RFID & blockchain technology, in: Proc. 13th International Conference on Service Systems and Service Management (ICSSSM), 2016.

[2] F. Tian, A supply chain traceability system for food safety based on HACCP, blockchain & Internet of things, in: Proc. International Conference on Service Systems and Service Management (ICSSSM), 2017.

[3] C. Xie, Y. Sun, H. Luo, Secured data storage scheme based on block chain for agricultural products tracking, in: Proc. 3rd International Conference on Big Data Computing and Communications (BIGCOM), 2017.

[4] F. Tschorsch, B. Scheuermann, Bitcoin and beyond: a technical survey on decentralized digital currencies, IEEE Communications Surveys & Tutorials 18(3)(2016) 2084-2123.

[5] M. Tsukerman, The block is hot: a survey of the state of Bitcoin regulation and suggestions for the future, Berkeley Tech. LJ 30(2015) 1127.

[6] S. Nakamoto, Bitcoin: a peer-to-peer electronic cash system. <http://bitcoin.org/en/bitcoin-paper>, 2008.

[7] X. Wan, Z. Lu, Q. Hu, M. Yu, Application of asset securitization and block chain of Internet financial firms: take Jingdong as an example, in: Proc. International Conference on Service Systems and Service Management (ICSSSM), 2017.

[8] G. Cui, K. Shi, Y. Qin, L. Liu, B. Qi, B. Li, Application of block chain in multi-level demand response reliable mechanism, in: Proc. 3rd International Conference on Information Management (ICIM), 2017.

[9] M.B. Taylor, Bitcoin and the age of bespoke silicon, in: Proc. the 2013 International Conference on Compilers, Architectures and Synthesis for Embedded Systems, 2013.

[10] J. Göbel, H.P. Keeler, A.E. Krzesinski, P.G. Taylor, Bitcoin blockchain dynamics: the selfish-mine strategy in the presence of propagation delay, Performance Evaluation 104(2016) 23-41.

[11] A. Wright, P. De Filippi, Decentralized blockchain technology and the rise of lex cryptographia. <http://ssrn.com/abstract:258664>, 2015.

[12] S. Huckle, R. Bhattacharya, M. White, N. Beloff, Internet of things, blockchain and shared economy applications, Procedia

Computer Science 98(2016) 461-466.

[13] I. Eyal, A.E. Gencer, E.G. Sirer, R. van Renesse, Bitcoin-NG: a scalable blockchain protocol, in: Proc. the 13th USENIX Symposium on Networked Systems Design and Implementation, 2016.

[14] J. Benet, Ipfs-content addressed, versioned, p2p file system. <https://arxiv.org/abs/1407.3561>, 2014.

[15] J.H. Howard, M.L. Kazar, S.G. Menees, D.A. Nichols, M. Satyanarayanan, R.N. Sidebotham, M.J. West, Scale and performance in a distributed file system, ACM Transactions on Computer Systems (TOCS) 6(1)(1988) 51-81.

[16] S. Alam, M. Kelly, M.L. Nelson, Interplanetary wayback: the permanent web archive, in: Proc. ACM Joint Conference on Digital Libraries (JCDL), 2016.