Proof of Video Integrity Based on Blockchain

Vasyl Yatskiv
Department of Cyber Security
Ternopil National Economic University
Ternopil, Ukraine
vy@tneu.edu.ua

Nataliya Yatskiv
Department for Information Computer Systems and Control
Ternopil National Economic University
Ternopil, Ukraine
jatskiv@ukr.net

Oleh Bandrivskyi
Department for Information Computer Systems and Control
Ternopil National Economic University
Ternopil, Ukraine
o.bandrivskyi@gmail.com

***Abstract* — In the paper a new approach for protecting video files from unauthorized changes using the Blockchain technology and cloud services is proposed. The essence of our approach consists in hash function calculating for each frame and forming the blocks sequence of calculated hash functions for all frames of video file. The hash function blocks are sent to the Blockchain cloud storage, which is trusted by all interested parties.**

Keywords — data integrity, blockchain, hash function, video files.

I. INTRODUCTION

The video surveillance systems are widely used in various fields, in particular, in road safety systems, traffic monitoring systems, access control systems, etc. However, the use of Closed Circuit TeleVision (CCTV) results by various stakeholders needs special tools for protection of video data from unauthorized changes in the process of receiving, storing and transmitting them by any users (authorized, unauthorized), that means ensure their integrity.

Different cryptographic algorithms and functions such as BLAKE, HMAC, Keccak, MD6, RIPEMD-160, SHA-1, SHA-2, SHA-3, Skein , etc. are used to ensure the data integrity [1].

Blockchain is a new information technology that has wide variety of uses in many industries, in particular, for data integrity protection. The first and most famous example of the use of the blockchain technology is the cryptocurrency Bitcoin [2]. Due to the decentralized structure, high reliability and fault-tolerance of block-chain technology can be used in intelligent transportation systems, logistics, warehouse systems, cloud computing, as well as in the Internet of Things and cyber-physics systems [3, 4].

Blockchain is a relatively new concept with high potential, therefore, it requires additional research for its effective application in new industries such as cyberphysics systems and Internet of Things. Blockchain integration in Internet of Things technology allows to create a new computing segment in which data can be safely processed and analyzed, while remaining private, which will increase the security and privacy of using devices connected to the Internet.

IBM studies the Blockchain technology and develops software for smart contracts which automatically execute transactions following predetermined rules, and the encrypted records of those transactions are shared across participants. IBM jointly developed with Samsung Electronics the Autonomous Decentralized Peer-to-Peer Telemetry (ADEPT) proof-of-concept (PoC) aimed to build a distributed network of devices – a decentralized Internet of Things and to track connected to the network devices [5, 6, 7].

In [8] the blockchain based video surveillance system BlockSee is proposed. The blockchain technology is employed for protection of camera settings, the positions and orientation of cameras and for providing of the video flows integrity.

The authors in [9] present an application that converts a smartphone videocamera into tamperproof dashboard camera. If the built-in sensors of the phone detect a collision, the program automatically generates a hash function for the relevant video. This hash is immediately transmitted to the OriginStamp service, which includes a hash in a transaction executed in the Bitcoin network which verifies the transaction, the hash of the video file is permanently secured in the decentralized public ledger that is blockchain.

Knirsch, F et al. in [10] describe the implementation of the image protection system based on the proof-of-possession consistency algorithm and Ethereum blockchain. The paper investigates the influence of image size on system performance and costs.

The work in [11] considers a multimedia blockchain framework based on a self-embedding watermarking algorithm that uses compressive sensing to detect any tampering and to retrieve the original content. According to the research unique information in the watermark contains two parts of the information: a) a cryptographic hash containing history of transactions; b) an image hash that stores the original content. After the watermark is extracted the first part of the watermark is passed to the distributed ledger in order to obtain the historical trail of the transaction, and the last part is used to identify the edited / tampered parts.

In the considered papers the hash is computed only at the moment of accident [9] for the increasing of the system efficiency of video files integrity protection based on the blockchain technology and the dependence of the cost on the size of the images is examined [10]. In our paper, we suggest to compute the hash function for a fixed number of frames, and use the time spent for hash functions generating as a criterion of system efficiency.

II. VIDEO DATA INTEGRITY PROTECTION

In this paper we propose new approach based on blockchain technology to protect video data integrity from unauthorized changes. Blockchain is a distributed data structure which consists of the blocks sequence, each block typically contains a hash pointer as a link to a previous block, thus forming a chain of blocks. Blockchain works as a distributed database that records all transactions on the network. Operations have a timestamp and are stored in blocks where each block is identified by its cryptographic hash [1].

978-1-7281-0450-8/19/$31.00 ©2019 IEEE

The essence of the proposed approach is as follows. Hash function of each video frame is computed. The next block consists of the hash - sum of the first video frame hash and the second video frame from which the hash function is calculated again and so on (Fig. 1). The sequence of blocks that uniquely match the video file is formed as a result [2].

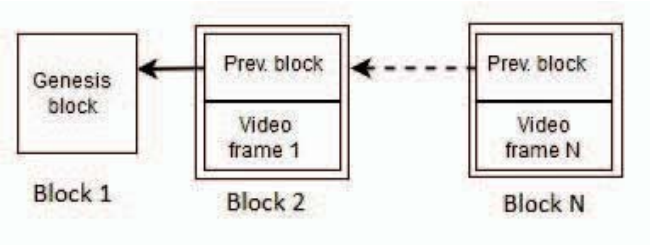


Fig. 1. Sequence and structure of blocks.

The first block in the chain (parent block, genesis block) is considered as a separate case, since it does not have a previous. Prev.block is a hash-sum of previous block.

Since the hash sum of each video frame is a part of the computation of next frame hash function , then the change (modification) of any frame will change the hash sum of the video file.

The public cloud services or cloud blockchain services trusted by all stake-holders can be employed to store a computed sequence of blocks. At the same time, only the result of hash function computation, the size of which is from 128 to 512 bits, depending on the used hash function requires transmitting and storing in cloud services.

III. The Structure of the System

The general structure of the system is shown in Fig. 2. The software implementation was fully implemented on the platform Node.js, based on engine V8, which converts JavaScript from a specialized language into a general-purpose language.

At the first stage of the system development a flexible tool for video files processing was required. For this purpose Ffmpeg was chosen. It is a complex of home-made computer programs and software libraries for manipulating digital video and audio materials such as recording, converting and packaging into various container formats.

FFmpeg provides large amount of information about the file and the ability to split into frames video stream. It supports more than 340 different file formats and works on all generally used platforms.

The 'fluent-ffmgep' library is used for interaction with FFmpeg that provides better integration with system. It transforms the complex use of command line for interaction with ffmpeg into a flexible and easy-to-use node.js module.

The modules 'fs' and 'pach' provide physical interaction with selected by FFmpeg frames. These modules determine the location of the video frames and delete the them after processing. The standard 'crypto' node module is used for processing of encrypted data. It allows to generate the hash functions for the received frames and further to add them. SHA-2 is selected as the hash algorithm.

The received hash sum are transmitted for further processing in Naivechain by the use of 'request' (Simplified HTTP client) library.

The next step is the direct implementation of Blockchain technology for the storage of video frames hash sum blocks. Naivechain (Fig. 3) is responsible for this part of the functional.

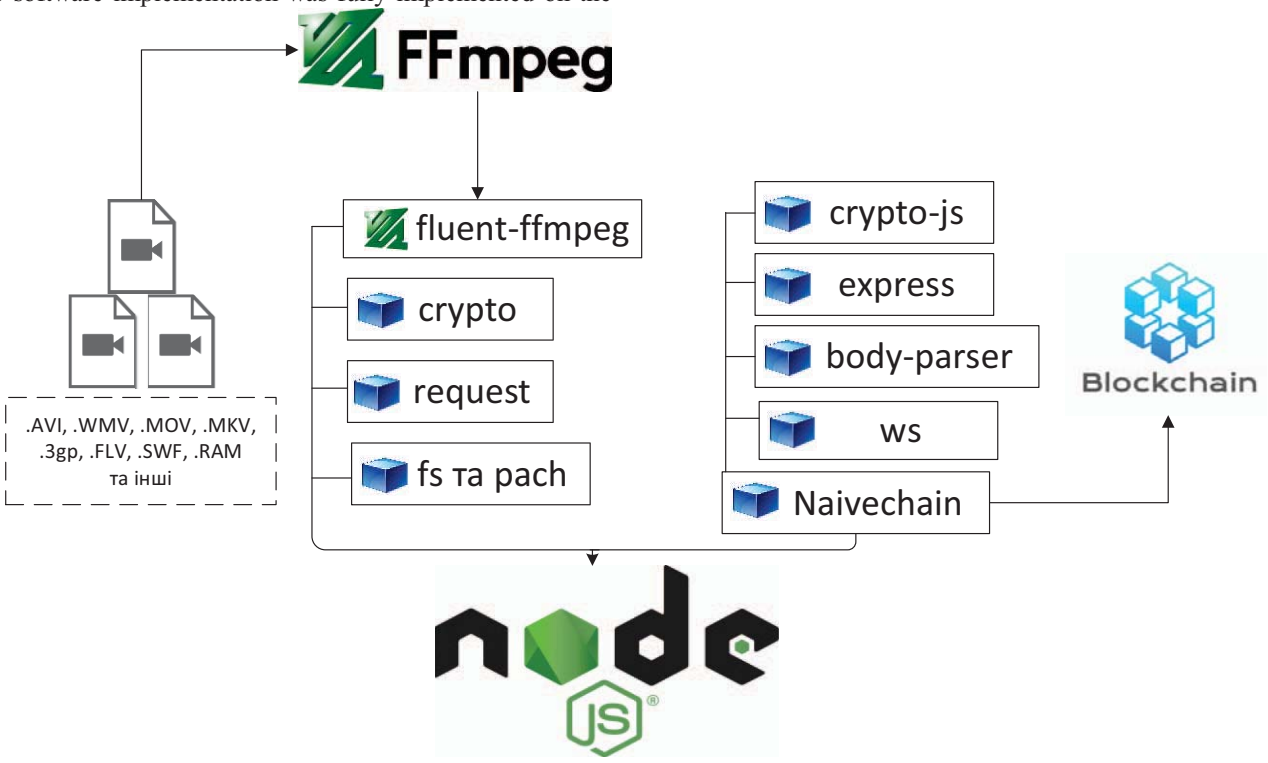


Fig. 2. The structure of the system

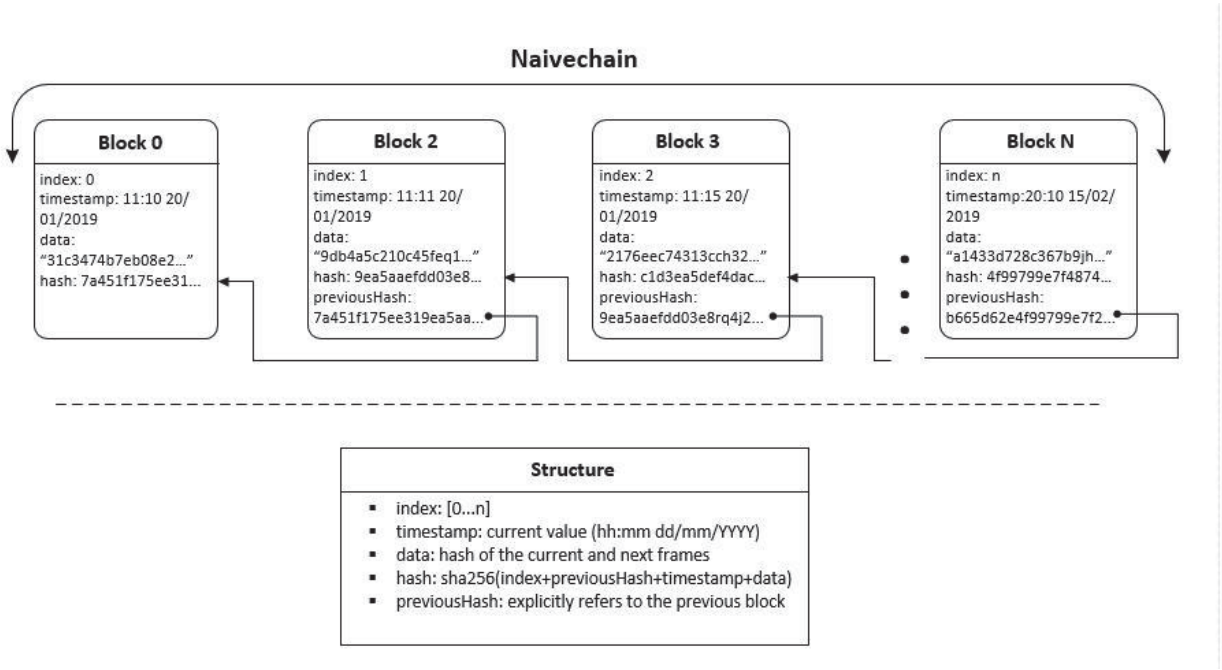


Fig. 3. Block structure in the Naivechaine

The main concepts of Naivechain [12]:

HTTP interface to control the node;

Use Websockets to communicate with other nodes (P2P);

Super simple "protocols" in P2P communication;

Data is not persisted in nodes.

No proof-of-work or proof-of-stake: a block can be added to the blockchain without competition.

For implementation web application framework Express.js is used. The data received from 'request' module are processed by 'BodyParser' and then they will be a part of the blocks that form the chain (Fig. 3).

The minimalist structure of Naivechain is compensated by the modules effectiveness and the simplicity of the modification.

A. An Algorithm

In general case the protecting video data algorithm based on the blockchain technology consists of the following steps:

1. The input video is split into a stream of video frames.

2. The hash sum is calculated for the first frame.

3. The hash sum of the previous frame is added to the current frame and then the hash sum of current frame is calculated.

4. Each hash sum is written in the block which will be a part of blockchain.

This flowchart represents an algorithm for ensuring the integrity of the video file. It doesn't consider peculiarities of programming for chosen software platform and doesn't include the intermediate technical blocks (pre-launch of the web server, connection of nodes, etc.).

B. Research

The stability and rate of the developed system depend on the characteristics of the hardware. The highest load occurs during splitting video stream into frames. The frames number optimal choice defines the performance of the system as a whole and the reliability of applied approach for video data protection.

The research was conducted on the Intel (R) Core (TM) CPU i7-7700HQ CPU, respectively, for another type of CPU the time characteristics will deviate.

Table 1 describes the relation between time spent on creating screenshots (generating of frames) and the amount of frames for which the hash is being computed. The video file with resolution 720p and a 58 minutes and 15 seconds duration was used for the research.

TABLE I. THE TIME SPENT ON CREATING SCREENSHOTS

Number of frames per time unit	Number of frames	Time(seconds)
1 frame every 10s	352	99,418
1 frame every 5s	703	200,284
1 frame every second	3515	975,39
5 frames every second	17575	4945,258

Depending on the video stream changes dynamics we can define the corresponding number of frames per time unit, which will reduce the time for the hash functions generating and, accordingly, the load on the system as a whole.

The chart (Fig. 4) based on the table 1 shows that at an average during 1 second 3.55 frames are created. Hence we can determine the processing time of the file.

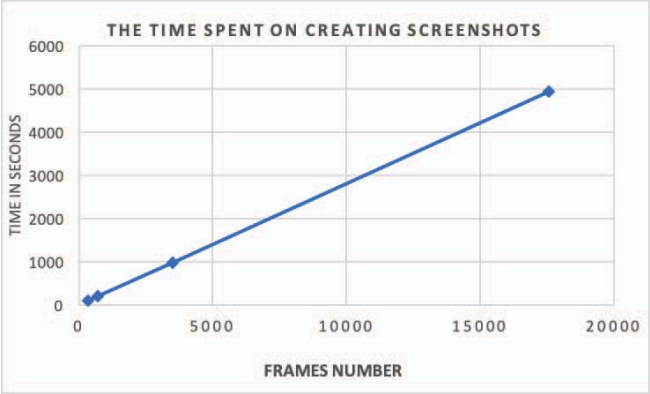


Fig. 4. The chart for time spent on creating screenshots (in seconds).

For example, processing of video file at 25 frames per second will take 87875 / 3.5 = 25107 seconds or 6.97 hours for a 58minutes and 15 seconds video file in quality 720p.

The spent time on frame generation at rate of one frame per second for video files with extensions: 720p, 1080p, and UHD are presented in the Table II.

TABLE II. THE SPENT TIME ON GENERATING HASH FOR A VIDEO FILE WITH A DIFFERENT EXTENSION

Quality	**Time (seconds)**
720p	975,39
1080p	1710,433
UHD	3216,774

We can conclude that the hash generating for the video with higher resolution is more time-consuming. Therefore, it is best to select the optimal resolution according the task. A comparison of the number of generated screenshots for different extensions per time unit is demonstrated in Fig. 5.

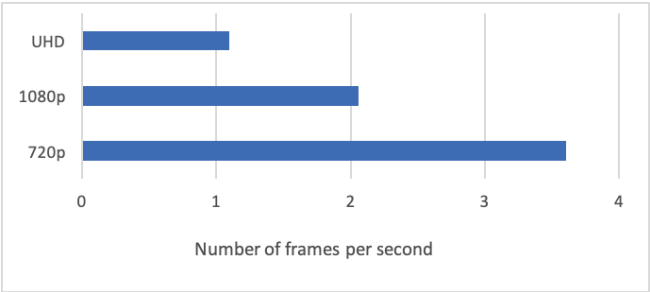


Fig. 5. The comparison chart of screenshots generation per unit time

As shown in Fig. 5 only one frame of video with UHD resolution can be processed during one second, at the same time for this video file with 1080 p resolution this characteristic have doubled.

IV. CONCLUSIONS

The approach to protect the integrity of video files based on the use of blockchain and cloud services is proposed in the paper. The proposed framework allows to store blocks of hash functions in private or in public blockchain. In our research we employed FFmpeg for processing of video files but it can be used for handling video stream, so this approach is suitable for video surveillance systems. In our paper we described and evaluated the time spent on generating hash and showed that it is directly proportional to the resolution of video file and number of selected frames.

REFERENCES

[1] A.M. Antonopoulos. “Mastering Bitcoin: Unlocking Digital Crypto-Currencies”. California, Sebastopol: O’Reilly Media, Inc., 2014.

[2] Satoshi Nakamoto. “Bitcoin: A Peer-to-Peer Electronic Cash System”. Internet: https://bitcoin.org/bitcoin.pdf

[3] Li Shancang, Li Da Xu, and Shanshan Zhao. “The Internet of Things: a survey”. Information Systems Frontiers,, Volume 17, Issue 2, pp 243–259, 2015.

[4] Whitmore Andrew, Anurag Agarwal, and Li Da Xu. “The Internet of Things - A survey of topics and trends”. Information Systems Frontiers 17.2, pp. 261-274, 2015.

[5] P. Brody, Veena Pureswaran. “Device democracy: Saving the future of the Internet of Things”. IBM, September, 2014.

[6] B. S. Panikkar, S. Nair, P.Brody. Pureswaran, V. “ADEPT: An IoT Practitioner Perspective”, 2014.

[7] P.Veena, S.Panikkar, S. Nair, P. Brody. Empowering the Edge - Practical Insights on a Decentralized Internet of Things. Empowering the Edge -Practical Insights on a Decentralized Internet of Things. IBM Institute for Business, Value 17, Apr. 2015. http://www-01.ibm.com/common/ssi/cgi-bin/ssialias?infotype=PM&subtype=XB&htmlfid=GBE03662USEN#loade

[8] P.Gallo, S.Pongnumkul, U. Q. Nguyen. BlockSee: Blockchain for IoT Video Surveillancein SmartCities. In 2018 IEEE International Conferenceon Environmentand Electrical Engineeringand 2018 IEEE Industrialand Commercial Power Systems Europe (EEEIC/I&CPS Europe) (pp. 1-6). IEEE. (2018, June).

[9] B. Gipp, K. Jagrut, and C. Breitinger, “Securing Video Integrity Using Decentralized Trusted Time stamping on the Blockchain”, in Proceeding sof the 10th Mediterrane an Conference on Information Systems (MCIS), Paphos, Cyprus, 2016.

[10] A.Knirsch, K.Unterweger, D.Karlsson, S.Engel, B. Wicker. “Evaluation of a Blockchain-Based Proof-of-Possession Implementation”,Center for Secure Energy Informatics, Salzburg University of Applied Sciences andSchool of Electrical and Computer Engineering, Cornell University, Technical Report 2018-01, 2018

[11] B. Deepayan, F. Tian. “The multimedia blockchain: a distributed and tamper-proof media transaction framework”. In: Digital Signal Processing (DSP), 2017, 22nd IEEE International Conference on, London, 2017-August, 3 November 2017.

[12] A. Fortuna “Naivechain: a blockchain implementation in 200 lines of code” Internet: https://www.andreafortuna.org/technology/blockchain/naivechain-a-blockchain-implementation-in-200-lines-of-code/, October 21, 2016 [Feb. 14, 2019]

2018 IEEE International Conference on Cloud Computing Technology and Science (CloudCom)

Validating data integrity with blockchain

Rosco Kalis
University of Amsterdam
Amsterdam, The Netherlands
roscokalis@gmail.com

Adam Belloum
University of Amsterdam
Amsterdam, The Netherlands
A.S.Z.Belloum@uva.nl

***Abstract*—Data manipulation is often named as a serious threat to data integrity. Data can be tampered with, and malicious actors could use this to their advantage. Data users in various application domains want to be ensured that the data they are consuming are accurate and have not been tampered with. To validate the integrity of these data, we describe a blockchain-based hash validation method. The method assumes that the actual data is stored separately from the blockchain, and then allows a data identifier and a hash of these data to be submitted to the blockchain. The actual data can be validated against the hash on the blockchain at any time. Several use cases are described for blockchain-based hash validation, and to validate the method it is implemented inside an application audit trail to validate the audit trail data. This implementation shows that blockchain-based hash validation is able to detect malicious and accidental changes that were made to the data.**

***Index Terms*—Blockchain, Ethereum, Data integrity, Data validation, Audit trail**

I. Introduction

US Director of National Intelligence James Clapper stated that the next big cyber threat is data manipulation [1] and technology magazine Wired listed it as one of the biggest security threats in 2016 [2].

Data integrity is paramount in many scientific and societal applications. Many data can be tampered with, and malicious actors could use this to their advantage by making others act on these compromised data, by distributing these data to spread misinformation, or by taking credit of work that isn't theirs.

Consumers of scientific, business, and other data want to be ensured that they can use data without having to worry about the integrity of these data. Likewise, producers of these data want to guarantee data integrity for their consumers, and they want to be ensured themselves that there is no one who can tamper with their data. In this paper, we are interested in the use case where companies want to guarantee data integrity within their organisation, and they want to be sure that all data that are handled through applications can not be changed outside the functionality of these applications.

We use the employment of audit trails to illustrate this use case. Companies employ audit trails to log all state changes made within their applications. This is already a way to enjoy more certainty about the data inside applications, and it offers a way to validate the current state of the application against all past state changes. However, if this audit trail is stored in the same way as the application data, it is equally vulnerable to tampering.

To make hard assertions about data integrity, a method needs to be developed to validate the integrity of any arbitrary data. For this, we look at blockchain [3], which is a promising technology that can improve data integrity. The last few years have really accelerated the progress in blockchain development, and especially smart contracts [4] have opened up new potential use cases for blockchain technology.

A. Blockchain

Blockchain is a data structure that distributes all its data over a network of nodes, so that there is no single point of failure, and no central control that might be compromised [3]. It uses a consensus algorithm that allows these independent nodes to approve correct transactions and reject malicious ones [4].

On the blockchain, data are stored in a chain of so-called *blocks* [4]. Every block header includes the root of a *Merkle tree*, which contains the actual data in the block [4]. Besides this, every block also includes a timestamp and a hash of the previous block in order to make it further resistant to manipulation [4]. This structure can be seen in Fig. 1.

If an attacker would change the data inside a past block, the hash of this block would change with it, and since the changed hash is never referenced by another block, it would not be accepted by the rest of the network [4], and it would effectively create a fork of the blockchain. The rule with forks is that the longest chain is always the leading one, so in order to have the modified block accepted by the network, the attacker would need to grow their chain faster than the rest of the network combined to pass the longest chain [4]. Because the resources of the entire network are extensive in major blockchains, this sufficiently guarantees the integrity of the data in the blockchain [4].

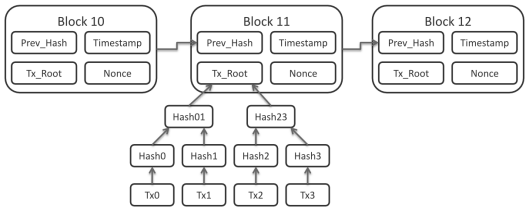


Fig. 1. This image shows the contents of blocks and how they are linked together. Image by Matthäus Wander (Wikimedia), downloaded from https://en.wikipedia.org/wiki/Blockchain in August 2018.

978-1-5386-7899-2/18/$31.00 ©2018 IEEE
DOI 10.1109/CloudCom2018.2018.00060

B. Smart contracts

Starting with Ethereum, blockchain technology has been extended with smart contract functionality. Smart contracts are a way to digitally enforce a contract or an agreement between parties through code. This concept predates blockchain more than a decade [5], but only when blockchain was implemented did it finally become possible to implement these smart contracts without the need for a trusted third party.

Ethereum offers the ability to publish smart contracts on its blockchain, which can be executed by the Ethereum Virtual Machine (EVM) [4]. By publishing these contracts to the Ethereum blockchain, all involved parties can easily inspect the contract and they will be assured that the contract will execute exactly as specified.

II. Related work

The need to ensure the data integrity of electronic audit trails and other data already existed decades ago. The different ways an audit trail could be corrupted were described by Weber in 1982, along with methods to overcome some of these corruptions [6]. Then, at the end of the century, Schneier & Kelsey described a method of securing audit log data on untrusted machines [7], which has since been cited in many other works on the subject of data integrity and auditing. US Patent 6968456 uses a similar approach to make the storage of audit trail data inside regular databases more secure [8].

Since the introduction of blockchain, multiple works have been published on the use of blockchain in the validation of data integrity and in other auditing processes. Deloitte has published their research into the use of blockchain in the field of accounting and auditing, in which they envision a hashing based approach similar to our own that allows third party auditors to easily verify the integrity of all data records [9].

Sutton & Samavi also describe a blockchain-based way of audit trail validation. They specifically focus on the non-repudiation of privacy audit logs in the light of privacy policy compliance [10]. US Patent application 20180025181 [11] and Zikratov et al. [12] also present hashing-based methods for data integrity validation. In contrast to our approach, these works specifically use their methods to validate the integrity of data files, while we focus on validating the integrity of any data.

III. Method

Smart Contracts on the Ethereum blockchain allow data to be stored directly inside these contracts as variables. Because of the nature of the blockchain, it is guaranteed that these data can only be changed using the smart contract's functionality, and every interaction with this contract gets recorded as a transaction on the blockchain.

Looking at these properties of blockchain and smart contracts, the easiest way to achieve data integrity for any data is to store all data directly on the blockchain inside these smart contracts. That way the data are easily verifiable, and both producers and consumers of the data are ensured that the data can be trusted. However, there are several issues with directly storing arbitrary data on the Ethereum blockchain. Most importantly these issues present themselves in transaction size limits, high transaction costs, and the potential need for (partial) data confidentiality. Taking these issues into account, a method is described for validating any arbitrary data using the Ethereum blockchain.

A. Transaction limits

Every operation that is executed on the Ethereum blockchain costs an amount of so-called *gas*, and the amount of gas that can be used within a single transaction is limited by the block gas limit. This gas limit is currently around 8 million gas for the Ethereum Main Net, but it can scale depending on demand [13].

The Ethereum Yellow paper specifies that every byte of passed data in a transaction costs 68 units of gas, and every transaction costs 21 000 gas to start with [13]. From this follows that the maximum amount of data passed in a single transaction is approximately 115 kB, as can be seen in (1).

$$(8000000 - 21000)/68 \approx 117300B \approx 115kB \qquad (1)$$

However, this only includes transaction data that are not stored. Data storage is one of the most expensive operations in terms of gas cost, so this can easily become a bottleneck when storing larger amounts of data. The Ethereum Yellow paper specifies that every byte of stored data costs 625 units of gas [13]. From this follows that the maximum amount of stored data in a single transaction is approximately 11 kB, as can be seen in (2).

$$(8000000 - 21000)/(625 + 68) \approx 11500B \approx 11kB \qquad (2)$$

For some use cases this limit will not be relevant, like when storing integer values. But when storing serialised application data or arbitrary-size blobs, this limit can be reached rather quickly. Realistically, this means that larger pieces of data would need to be split up over multiple transactions.

B. Transaction costs

Every unit of gas on the Ethereum blockchain needs to be paid for using the Ether cryptocurrency [13], so the monetary costs of storing data on the blockchain could increase rapidly. At the time of writing the price of one unit of gas is around 10 GWei, or 0.000 000 010 Ether, and the price of one Ether is around €300. This means the price per kB of data is around €2.1, as can be seen in (3). These increased costs make storing larger amounts of data impractical for realistic usage.

$$(625 + 68) \cdot 1024 \cdot 0.000000010 \cdot €300 \approx €2.1 \qquad (3)$$

C. Data confidentiality

All data published on the Ethereum blockchain are publicly accessible. For certain data, this is acceptable or even required, but there are many use cases in which data should only be shared with certain parties. Data that are meant for internal

use in corporations need additional confidentiality measures in order to leverage the strength of blockchain for data integrity.

Two different methods can be used to achieve data confidentiality on the blockchain: encryption and hashing [14]. It is visible from the earlier two points about transaction limits and costs that encrypted data would still realistically reach transaction limits quite often, and the costs of storage would skyrocket [14]. Therefore, only a method using data hashing is viable, as it easily stays under the transaction limits, has constant and relatively low transaction costs, and is able to shield confidential data from the public. In this paper we propose a method using data hashing to validate data integrity.

D. Data validation using data hashing

US Patent application 20180025181 [11] presents a method to reliably store data files and verify their integrity with blockchain technology. The method describes storing data files on a data storage module, and transmitting hashes of these data files to a public blockchain [11]. With this hash they store metadata, such as a timestamp and file size. The method suggests to monitors these files, and whenever a change occurs, this process is re-initiated, storing the new hash on the blockchain, with a link to the previous one [11].

While this method is specifically created for the validation of data files, a modified version of this method can be used for any type of data. The methods for hashing are the same for different types of data, while the metadata and the identifying data can be changed to fit the specific use case.

The method we propose hashes any arbitrary data and stores their hash on the Ethereum blockchain. This hash is identified by a unique identifier representing the specific data. The method for creating this unique identifier is left to the specific implementation of the method. The data can then be verified at any moment by validating that the hash stored for the data identifier matches the actual hash of the data. If the hashes don't match, we know that the data have been changed since they were stored on the Ethereum blockchain. We call the method blockchain-based hash validation.

In contrast to the method described in US Patent application 20180025181 [11], blockchain-based hash validation does not use monitoring to automatically update the hash of specific data, since this method is focused on validating the integrity of a specific version of the data. New versions of data can be added to the smart contract under a new unique identifier. This allows multiple versions of data to be validated simultaneously.

Since the data themselves are not stored on the blockchain, blockchain-based hash validation needs the data to be available in some way to validate them. This means that blockchain-based hash validation can not retrieve the original data when they have been lost or their integrity has been compromised. Use of this method should therefore always be coupled with a way to retrieve lost data, such as a sufficiently strong backup protocol and regular validations of the data. This allows changes in this data to be detected early, and the correct data to be restored.

IV. Proof of concept

Blockchain-based hash validation can be used in several different use cases, including the validation of published scientific results, the validation of audit trail data, and the validation of other shared data between different parties.

To show the possibilities of blockchain-based hash validation we implemented a proof-of-concept audit trail [1] that is validated against an Ethereum smart contract. This audit trail automatically logs all interactions that take place inside the application, and write identifier-hash mappings to the Ethereum blockchain for each of these log entries.

A. Smart contract functionality

The smart contract we created contains a mapping of the unique data identifiers to their corresponding data hash, as well as an array of all identifiers, so it can be iterated over.

Next, the smart contract includes an audit function that allows new data to be added to this mapping. This works by passing an identifier and a hash to this function; this identifier-hash pair is then stored in the mapping and the identifier is stored in the identifier array.

Finally, the smart contract contains a validation function that allows stored data to be validated by passing an identifier and a hash to this function and comparing them with the stored values.

```
contract AuditTrail {
  bytes32[] public identifiers;
  mapping(bytes32 => bytes32) public hashes;

  function audit(bytes32 identifier, bytes32 hash)
  external ownerOnly {
    require(hashes[identifier] == 0,
            "Identifier can only be audited once");
    hashes[identifier] = hash;
    identifiers.push(identifier);
  }

  function validate(bytes32 identifier, bytes32 hash)
  external view returns(uint8) {
    return hashes[identifier] == hash ? 0 : 1;
  }
}
```

B. Apache Isis

For the automatic audit trail we used Apache Isis [2], which is a Java software development framework based on Domain Driven Development and the Naked Objects pattern. It can be used to rapidly develop complex business applications because a UI and REST API are dynamically generated from the domain model of the application at runtime. This leads to a faster development cycle and higher agility, since the focus can remain on the domain model, and no extra time needs to be spent on the presentation layer of the application. More information on Apache Isis' design can be found in [14].

Apache Isis offers several services in the form of Application Programmer Interfaces (APIs) and Service Provider Interfaces (SPIs). The APIs are implemented by the framework

[1]Code can be found at https://github.com/rkalis/blockchain-audit-trail

[2]https://isis.apache.org/ (accessed 2018-04-10)

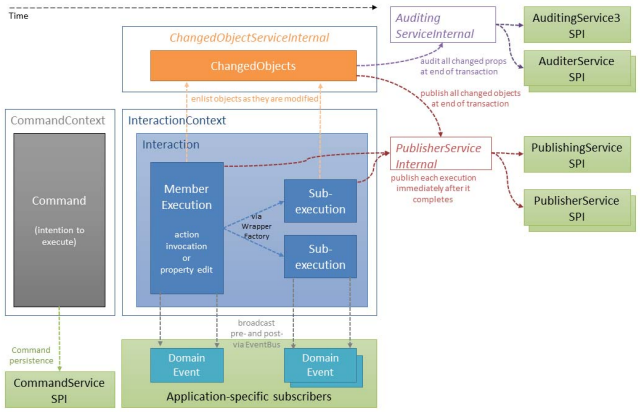


Fig. 2. This image shows the different domain services used for auditing within Apache Isis. Image downloaded from https://isis.apache.org/guides/rgsvc/rgsvc.html in April 2018.

and can be called by the application, while the SPIs are implemented by the developer, and will automatically be picked up and called by the framework [15]. Among these services are several SPIs for auditing or similar purposes, as illustrated in Fig. 2.

Most important of these services are the AuditerService and the PublisherService. The AuditerService captures the actual changes caused by an interaction [15], and the PublisherService captures a memento of the interaction, as well as a summary of the changed properties [15]. Important to note is that the AuditerService gets called every time a change in the database occurs, while the PublisherService gets called at the end of every interaction, which can contain multiple changes.

C. Audit trail implementation

For the automatic audit trail we wanted to log all changes, but at the same time keep the number of blockchain transactions to a minimum. This is why we implemented a combination of the AuditerService and PublisherService.

Every time the AuditerService' audit method gets called, its parameters get stored inside a single ThreadLocal *audit entry* object. At the end of the interaction, the PublisherService' publish method gets called, which is used to submit the audit entry's identifier and hash to the Ethereum smart contract, after which the ThreadLocal audit entry object is reset. The blockchain transaction is sent asynchronously, so that the rest of the application can continue running while the transaction is executed on the blockchain.

D. Audit trail validation

The audit entries inside the audit trail can be validated against the smart contract on the blockchain. Individual audit entries can be validated with a specific validation action, which calls the validation function on the smart contract to verify that the identifier-hash pair of the audit entry still matches the pair that is stored on the blockchain.

There is also the option to validate the entire audit trail. This is done by calling the smart contract's validation function for every single audit entry in the audit trail. In order to fully validate the audit trail, it is also checked for missing entries by traversing the array of identifiers inside the smart contract, and verifying that the audit entry with the corresponding identifier can be found inside the audit trail. The results of this validation are presented in three lists – of validated, invalidated, and missing audit entries.

V. Evaluation

To evaluate the implementation, three different scenarios have been created, in which the data inside the audit trail would be invalidated [14]. After executing these scenarios the audit trail is validated with the audit trail validation method.

For these scenarios, the blockchain audit trail implementation is added to two demo applications, which are based on actual applications that are being used. The first is based on Incode's Contact App [3], and is used for internal contact management within companies. The second is a larger scale application based on Estatio [4], which is a full-fledged estate management system.

Due to space limitation, we briefly describe the three scenarios used for the evaluations, and show the results of the first validation. More information on the scenarios, as well as more result images can be found in [14].

A. Scenario 1 - Inexperienced admin

A new system administrator in training gets a request from their coworker to restore some files from a backup. Lacking enough knowledge, they perform a full server backup, accidentally overwriting the application database. At the end of the day, the audit trail is validated, and the administrator's mistakes are discovered. Luckily, the company can reconstruct most of the correct data with the correct backups, but the changes that had been made the same day could not be recovered. This highlights a part of the weakness of the implementation, but it also displays the way these kinds of mistakes can be detected quite early on.

To simulate this scenario we take a backup of the application database, and restore it after making some changes in the database, effectively overwriting these changes. After following these steps, we run the audit trail validation. Figure 3 shows that the last two audit entries are reported as missing, as these are the ones that were deleted while restoring the database backup. A demo video of this scenario is available on YouTube [5].

B. Scenario 2 - Cover your tracks

An employee of Acme Corporation has decided they want to quit their job and retire. To fill the gap in their finances, they changed the recipient on one of the company's larger invoices to a private bank account, and changed it back to the original after the invoice had been paid. Because the company employs an audit trail, they wish to cover their tracks. They

[3]https://github.com/incodehq/contactapp (accessed 2018-05-29)
[4]https://github.com/estatio/estatio (accessed 2018-05-30
[5]https://youtu.be/nfekoK6pUqU

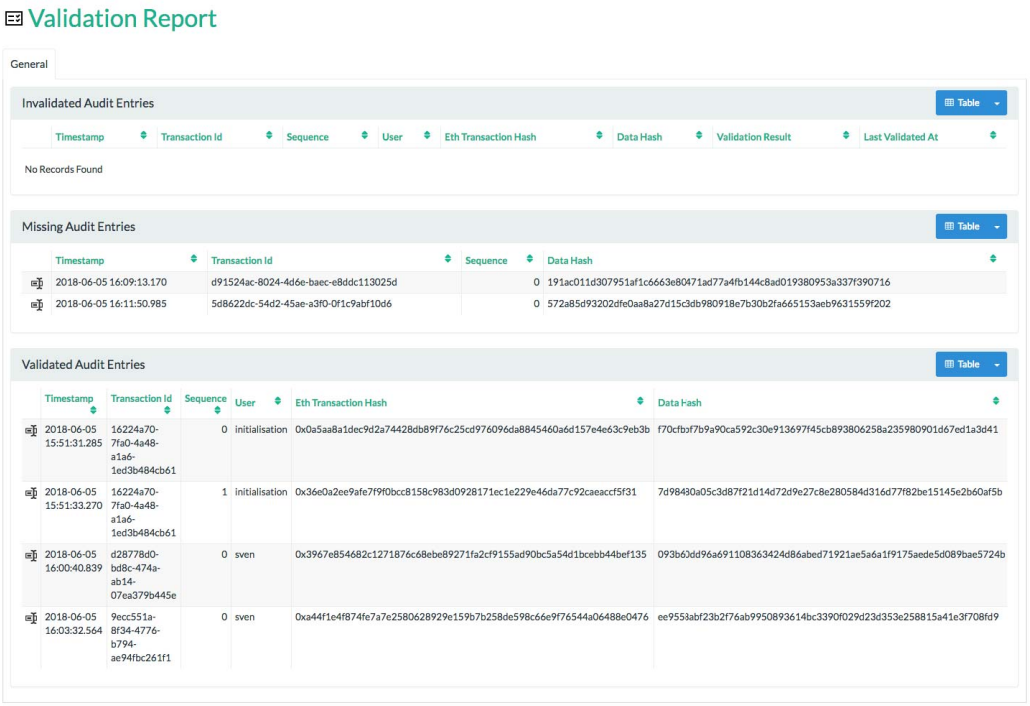


Fig. 3. The final two audit entries are missing as they have been removed in the restoring of the backup.

still have their old company credentials, so they log into the database and remove all audit entries that log their changes.

Because Acme has the policy to routinely validate their audit trail against the Ethereum blockchain they notice the missing audit entries at the end of the day. Since these audit entries had already been included in the company's backups, the correct data can easily be restored, and the employee's malicious actions come to light. Our implementation shows this by reporting the missing audit entries in the validation result.

C. Scenario 3 - Shift the blame

An employee of Acme Corporation maliciously changes the email address of a contact in the application. They then edit the corresponding audit entry directly in the database. In this audit entry he changes the logged user to his coworker to shift the blame. Afterwards, he reports his coworker to their superior. Because Acme corporation uses an audit trail that is validated against the Ethereum blockchain, it is quickly visible that the corresponding audit entry has been tampered with. The audit entry had not yet been included in the company's backups, so there is no hard proof of the employee's intentions. This is because our implementation shows the changed audit entries as invalidated in the validation report, but it does not show what the actual data inside the audit entry should have been.

VI. Discussion

In this paper we proposed a blockchain-based hash validation method. This method is used to validate the integrity of any data by storing a data identifier and a data hash in a smart contract on the Ethereum blockchain. The data can then be validated against the data stored inside the smart contract. We described several potential use cases for this method and we described one of these use cases in further detail in order to test the method. For this use case, we created an automatic audit trail for Apache Isis applications that uses blockchain-based hash validation to validate its data. This audit trail was evaluated using several scenarios.

A. Limits of the proof of concept

In the proof of concept audit trail, multiple changes are aggregated before sending the corresponding blockchain transaction. If the application would experience a crash or an outage during the auditing process, an incomplete audit entry could be written to the database, while nothing gets sent to the blockchain. This invalidates the audit entry

The blockchain transactions are sent asynchronously because it can take some time before a transaction is executed and accepted on the Ethereum blockchain. We observed that new transactions fail when five or more transactions are already being processed at the same time. This is usually not a problem for smaller applications, but can definitely impact larger applications. When using this implementation in a production environment, a method needs to be developed to handle these failures.

B. Conclusions

The evaluation of our proof of concept implementation showcased the strengths and weaknesses of blockchain-based hash validation. This evaluation showed that blockchain-based hash validation is able to detect tampering or loss of data. However, this evaluation also shows that it is not able to prevent this tampering or loss of data from happening, as it relies on the data being stored externally.

Therefore, in order to optimally use blockchain-based hash validation, it should be coupled with a way of restoring lost or corrupted data, such as a regular backup protocol or distributed data. Without these measures, the method will only function as validation, which is still valuable on itself.

C. Recommendations for further research

Because blockchain-based hash validation relies on the data being stored separately from the blockchain, it is only able to detect tampering, but it can not prevent tampering. It could be interesting to research we can fully prevent data tampering by storing the actual data on the blockchain.

The Ethereum gas limits put a limit of around 11 kB of data stored on the blockchain per transaction. The price of gas makes storing data on the blockchain very expensive. These issues greatly discourage the storage of larger amounts of data on the blockchain, but they do not make it impossible. While it is probably not practical for real-world usage, it is interesting to see if a proof of concept could be developed for storing any data on the blockchain as a way to guarantee their integrity.

There could also be potential in InterPlanetary File System (IPFS) for data storage. IPFS is a distributed file system that offers deduplication and version history for all stored data. IPFS could provide data storage that is resistant to tampering or corruption because of its distributed nature and checksum verification [16].

The IPFS website [6] also showcases the possibility to integrate IPFS with blockchain technology by storing large amounts of data with IPFS, and placing links to these IPFS data on the blockchain linking to certain specific versions of data in IPFS.

Finally, the current method stores an identifier-hash mapping in a smart contract to validate the integrity of the corresponding data. It is difficult to infer anything about the data by just looking at these data inside the smart contract. In the future, we might want to store additional metadata along with the hash inside the smart contract. This could be achieved by adding a MetaData struct to the smart contract, which could contain additional information. This MetaData struct could then be added to the mapping instead of just a data hash. An example MetaData struct could look like this:

```
struct MetaData {
    string user;
    uint256 timestamp;
    bytes32 hash;
}
```

[6] https://ipfs.io/ (accessed 2018-05-31)

ACKNOWLEDGEMENT

We would like to thank the EU PROCESS project (grant no 777533) for supporting this work. We would also like to thank Dan Haywood for his input throughout the implementation of our proof of concept.

REFERENCES

[1] S. Ackerman, "Newest cyber threat will be data manipulation, us intelligence chief says," *The Guardian*, 2015. [Online]. Available: https://www.theguardian.com/technology/2015/sep/10/cyber-threat-data-manipulation-us-intelligence-chief

[2] K. Zetter, "The biggest security threats we'll face in 2016," *Wired*, 2016. [Online]. Available: https://www.wired.com/2016/01/the-biggest-security-threats-well-face-in-2016/

[3] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," 2008. [Online]. Available: https://bitcoin.org/bitcoin.pdf

[4] V. Buterin, "Ethereum: A next-generation smart contract and decentralized application platform," 2013. [Online]. Available: https://github.com/ethereum/wiki/wiki/White-Paper

[5] N. Szabo. (1996) Smart contracts: Building blocks for digital markets. [Online]. Available: http://www.fon.hum.uva.nl/rob/Courses/InformationInSpeech/CDROM/Literature/LOTwinterschool2006/szabo.best.vwh.net/smart_contracts_2.html

[6] R. Weber, "Audit trail system support in advanced computer-based accounting systems," *The Accounting Review*, vol. 57, no. 2, pp. 311–325, Apr. 1982. [Online]. Available: https://search.proquest.com/docview/1301314968

[7] B. Schneier and J. Kelsey, "Secure audit logs to support computer forensics," *ACM Transactions on Information and System Security*, vol. 2, no. 2, pp. 159–176, May 1999. [Online]. Available: https://www.schneier.com/academic/paperfiles/paper-auditlogs.pdf

[8] A. Tripathi and M. Murthy, "Method and system for providing a tamper-proof storage of an audit trail in a database," Patent 6 968 456, 2005. [Online]. Available: http://patft.uspto.gov/netacgi/nph-Parser?patentnumber=6968456

[9] N. Andersen, "Blockchain technology: A game-changer in accounting?" Mar. 2016. [Online]. Available: https://www2.deloitte.com/content/dam/Deloitte/de/Documents/Innovation/Blockchain_A%20game-changer%20in%20accounting.pdf

[10] A. Sutton and R. Samavi, "Blockchain enabled privacy audit logs," in *The Semantic Web – ISWC 2017*. Cham: Springer International Publishing, 2017, pp. 645–660. [Online]. Available: https://doi.org/10.1007/978-3-319-68288-4_38

[11] I. Barinov, V. Lysenko, S. Beloussov, M. Shmulevich, and S. Protasov, "System and method for verifying data integrity using a blockchain network," Patent 2 018 025 181, 2018. [Online]. Available: http://www.freepatentsonline.com/y2018/0025181.html

[12] I. Zikratov, A. Kuzmin, V. Akimenko, V. Niculichev, and L. Yalansky, "Ensuring data integrity using blockchain technology," in *2017 20th Conference of Open Innovations Association (FRUCT)*, Apr. 2017, pp. 534–539. [Online]. Available: https://ieeexplore.ieee.org/document/8071359/

[13] G. Wood, "Ethereum: A secure decentralised generalised transaction ledger — byzantium version f72032b – 2018-05-04," 2018. [Online]. Available: https://ethereum.github.io/yellowpaper/paper.pdf

[14] R. Kalis, "Using blockchain to validate audit trail data in private business applications," Jun. 2018. [Online]. Available: https://esc.fnwi.uva.nl/thesis/centraal/files/f1051832702.pdf

[15] D. Haywood. (2018) Apache isis domain services. [Online]. Available: https://isis.apache.org/guides/rgsvc/rgsvc.html

[16] J. Benet, "Ipfs - content addressed, versioned, p2p file system (draft 3)," Unknown. [Online]. Available: https://github.com/ipfs/papers/raw/master/ipfs-cap2pfs/ipfs-p2p-file-system.pdf

2020 5th IEEE International Conference on Big Data Analytics

A Secure IoT Data Integrity Auditing Scheme Based on Consortium Blockchain

Guofang Dong
School of Electrical and Information Technology
Yunnan Minzu University
Kunming, 650031, China
e-mail: dongguofang1@163.com

Xia Wang
School of Electrical and Information Technology
Yunnan Minzu University
Kunming, 650031, China
e-mail: wangxiacsu@163.com

***Abstract*—With the development of IoT, a large amount of valuable data is stored in remote cloud services by enterprises or individuals, which raises concerns about the security of data. In order to ensure the integrity of the data in the cloud, scholars have proposed various auditing scheme, among which Third Party Auditor (TPA) is a distinctive feature of a class of scheme. But the security of a centralized TPA is difficult to ensure, and blockchain technology makes up for this shortcoming. In order to solve this, we propose a secure auditing scheme based on the consortium blockchain. Our scheme owns following feature: anonymity, fairness, accountability, security. The experiment results show that our scheme is efficient.**

Keywords Internet of Things; blockchain ; data integrity auditing; consortium blockchain; secure

I. INTRODUCTION

With the rapid development of IoT, more and more data run into the hard disk. For most small and medium-sized IoT companies, implementing local storage requires huge expenses, which is a huge economic burden for these companies. As a result, they often rely on relatively inexpensive cloud storage to store data. This demand has also promoted the development of cloud storage services. However, storing data in the cloud means losing direct control of the data. Due to natural disasters, network attacks, disk damage, and even human error, cloud storage services cannot fully guarantee data integrity. Once users use incomplete data for analysis, it will bring immeasurable losses. Therefore, how to ensure the integrity of the data has become a hot topic.

To ensure data integrity, scholars have proposed a series of data integrity auditing schemes. Among them, third-party-based auditing schemes can liberate users from heavy auditing burdens and also generate corresponding audits service, which is welcomed by academia and business. However, in previous schemes, this auditing third party is often a centralized role. This makes the role likely to have the following two problems: (1) single point of failure, whether it is hacked or hardware failure, will make TPA unable to perform audit tasks normally; (2) performance bottleneck, when a large number of users request at the same time, it is difficult for TPA to satisfy all users' requests, resulting in a delay in the feedback of audit service results. These two problems will affect the user's next use of data for processing.

The advent of blockchain technology provides new ideas for auditing programs. The decentralized, immutable and traceable characteristics of the blockchain meet the needs of data integrity auditing. There are also some scholars studying how to use blockchain for auditing.

Suzuki et al.[1] take the blockchain as a communication bridge for User-TPA-CSP which weakened the credible requirements of TPA, realized public audit, and can be accountable, but disclosed the pseudonym of the user and the link relationship between the user and the CSP. The existing work [2][3] have proved that the pseudo-name has a certain probability to be traced to the real identity. Therefore, the use of the public chain has exposed privacy. Ethereum smart contract is used for auditing by Nguyen et al.[4], although the TPA is eliminated, the operation is placed on the blockchain, the consensus nodes are involved in the operation, which increases the computational overhead, and the economic cost of using the smart contract is extremely high.

The use of the public chain will expose the user's audit information to the public's view, and may also expose the connection between itself and the CSP, which is unacceptable to the user. The Consortium chain is a good choice, both to preserve the characteristics of public audits and to ensure that audit information is exposed to a limited group of identified individuals.

In the Consortium chain, the most popular one is the Hyperledger Fabric [5]. In addition to inheriting the decentralized, tamper-proof and traceable features of the blockchain, Fabric uses the Member Service Provider (MSP) to ensure that members join-ing the Alliance are certified. The Identity Mixer [6] in Fabric provides users with multiple anonymity protections that protect user identity and hide links between users and CSPs. The transaction load in Fabric can be up to 99MB, which is much higher than the transaction load of other public chains. The fabric itself can set the time of the block, which allows the alliance to flexibly set the corresponding block time and improve efficiency according to the demand.

Based on Hyperledger Fabric, we propose a secure IoT data integrity auditing scheme. Using Fabric's own anonymity to achieve interactive identity privacy protection. Utilizing the characteristics of open and non-tamperable information of the blockchain achieves traceability. Fairness

978-1-7281-4111-4/20/$31.00 ©2020 IEEE

is achieved by using the characteristics of smart contract automatic execution.

Organization. The remainder of this paper is organized as follows. Section II introduce the concept of blockchain and Hyperledger Fabric. Section III shows the related work of auditing scheme. Section IV presents the proposed scheme in detail including the system model and the auditing process. Section V states the security analysis of the proposed scheme. Section VI presents the performance analysis of the scheme. Finally, the conclusions are drawn in Section VII.

II. BACKGROUND

In this section, we introduce the concept blockchain and the consortium blockchain Hyperledger Fabric used in the scheme.

A. Blockchian

Blockchain is a kind of peer-to-peer (P2P) distributed ledger which is append-only so that it is resistant to modification of data. In 2008, Satoshi Nakamote[7] was the first one to propose the design concept of blockchain, Bitcoin, and its implementation come to the world later, which triggered the wave of blockchain. Indeed, blockchain is a string of data blocks generated and chained by cryptography chronologically, each block containing some of network transaction. Nodes in the network send the transaction and keep one consistent ledger with a kind of consensus algorithm (e.g. PoW[8], PoS[8], PBFT[9]). Ethereum [10] is a Turing-complete blockchain-based platform, supporting a key concept, smart contract. The smart contract is executed by the network whose consensus does not require a trusted third party, no one can violate the contract. These two permissionless chains allow people to enter and exit at will, and they provide pseudo name system intend to protect the user's assets. The advantages of the blockchain are shining, attracting the attention of countless companies, but they are more inclined to permission chain to implement commercial applications. In this chain, everyone volunteers to work together to build this system and contribute in a contractual manner. Its controllability and limited anonymity make the chain very business friendly.

B. Hyperledger Fabric

Hyperledger Fabric is a modular and extensible open-source system for deploying and operating permissioned blockchains. Hyperledger fabric 1.0 and last is based on the PKI (Public Key Infrastructure) system and introduces the MSP (Membership Service Provider) Module which generates digital certificates to identify and manage the identity of members. The Fabric network is composed of different peers in channels. Within one channel, formal peer sends a transaction to the network, endorsing peers execute the chaincode (smart contract in Fabric) in its docker container to validity the transaction. Then ordering peers package transactions to a block and order them to work as consensus mechanism. At last the block was broadcasted to the network to keep a consistent ledger. Hyperledger Fabric v1.4 is the first long term support release, it use Identity Mixer[6] to provide anonymity to the others, in other words, we know each other but have no idea who is doing what. Of course, the administrator has right to audit the ledger and find the true identity for friendly purpose.

III. RELATED WORK

Data integrity and security in the cloud has always been the focus of people's attention. By checking the integrity, people can discover the problem of data damage and deletion in time, and take corresponding measures to deal with it. The emergence of blockchain technology provides new ideas for data integrity schemes. This paper focuses on the adoption of blockchain technology to solve the problem of scheme based on TPA. Ateniese et al. [11] first proposed the concept of provable data possession (PDP), which enables people to verify the integrity of remote data. Later, Wang.Q [12] introduced the concept of TPA into the PDP scheme, which was widely concerned by scholars in this field. TPA can audit on behalf of users, greatly reducing the burden on users. Since it is difficult to guarantee that TPA is credible in practice, there is the problem of data leakage. Wang et al. [13] [14] protect the privacy of users by proxy re-signature. Wang et al. [15] adopted the homomorphic authenticable ring signature for protecting the user's privacy, but it is not useful for large-scale users due to its cost. Huang et al. [16] used multiple TPAs to address the problem of collusion and centralization, but they introduced another centralized role named the receive server, which can learn the identity of users. Liu et al [17] solved the problem of malicious TPA faking identity for auditing, but it is not suitable to large-scale users.

Blockchain has the characteristics of decentralization, openness and transparency, tamper-proof, traceability and so on, which is very consistent with the demand of audit scheme and can effectively solve the problem of TPA. Therefore, many blockchain-based audit schemes have emerged in recent years. Suzuki et al. [1] set blockchain as the information channel among parties, and their experiment carried on the Bitcoin test network, but it was attacked by Malleability Attack [18]. Nguyen et al. [4] programmed the smart contract for auditing, but it has a large economic cost in Ethereum. Hao et al. [19] adopted verification peers for composing blockchain, and all peers are involved into the auditing process, which increases the computing overhead. Yu et al. [20] set blockchain as an information channel. Although the problem of TPA is avoided, the user cost is high, which is not what we expect. Liu et al. [21] proposed the a private blockchain based IoT data auditing framework, using smart contracts to audit. Dai et al. [22] logged data with private chain and uploaded it to public blockchain for public auditing.

IV. PROPOSED SCHEME

In this section, we will describe the proposed scheme, including the system model and the auditing scheme.

A. System Model

The system consists of four roles.

Users, have limited storage capabilities and need to use CSP to store data. In order to ensure the integrity of the data,

it is desirable to audit the data, but due to its limited computing power, it relies on TPA for auditing.

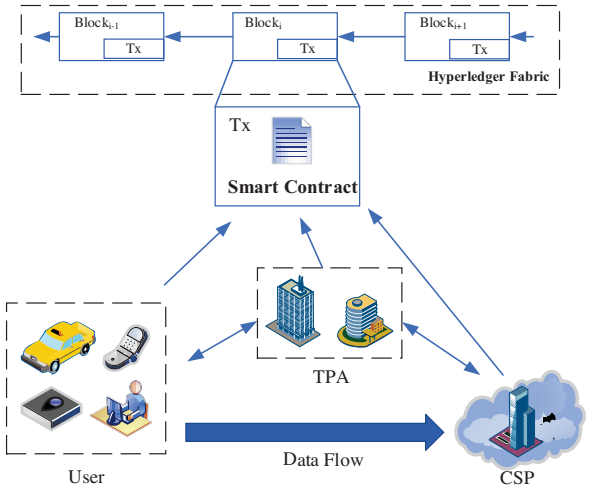


Figure 1. The system model

CSP, cloud storage service provider, is able to provide users with storage services.

TPA, third party auditor, with sufficient computing power, can provide users with audit services.

Hyperledger Fabric, an auditing information interaction platform, is credible because it is based on blockchain technology. Provide smart contract function to automate related business.

B. *Auditing Scheme*

The auditing scheme consists of three phases, several steps.

Initialization phase

- 1) All CSPs and TPAs join the Hyperledger Fabric. TPAs and CSPs maintain a distributed ledger.
- 2) Users join Hyperledger Fabric and purchase tokens. Users can join or leave if permission is granted. Users do not need to maintain the ledger, but they can verify the ledger. Tokens are used for auditing, users pay tokens to TPA, TPA can convert tokens to legal currency.
- All parties agree on auditing parameters, G_1, G_2 and G_T are multiplicative cyclic groups, p is the prime order of G_1 and G_2, g is the generator of G_2, $e: G_1 \times G_2 \to G_T$ is the bilinear pairing, $H: \{0,1\}^* \to G_1$ denotes the hash function that maps a string data to a point in G_1 and $h: G_1 \to Z_q^*$ indicates another hash function that maps a point in G_1 to a point in Z_q^*. The user generates a random secret key $x \in Z_q^*$ and computes the public key $y = g^x \in G_2$.

Storage phase

- 1) The user generates a random value $u \in G_1$ to compute the signature $\sigma_i = (H(m_i) \cdot u^{m_i})^x$ for each data block m_i and sends the signature and data to the CSP for storage.
- 2) The user deletes the local data but retains all signatures.

Auditing phase

- 1) The user issues an audit task using a smart contract. The task includes the size of the data to be audited, the deadline and the relevant tokens.
- 2) Every TPA can send transactions and accept the task. At this time, the remaining TPAs cannot accept the task. Only after the agreed time has passed, the remaining TPAs can accept the task.
- 3) The TPA accepting this task can be obtained from the user, the number of the data to be audited (such as 31,48,69,103), the corresponding signature (σ_i for each block m_i), a random array used to protect the data (v_i for each signature), and the target CSP.
- 4) The TPA challenges the target CSP, including the label of the data to be audited, the corresponding signature, and a random array.
- 5) The CSP first determines whether the user initiated an audit request through the blockchain. If so, CSP responds to the TPA $proof_{CSP} = \sum_{i \in I} m_i v_i$.
- 6) TPA calculates $proof_{TPA} = \prod_{i \in I} \sigma_i^{v_i}$ and performs verification using Equation(1), and calls the smart contract to send the audit result to the blockchain within the agreed time to obtain points. If TPA fails to send the audit results within the agreed time, it will be punished.

$$e(\prod_{i \in I} \sigma_i^{v_i}, g) = e(\prod_{i \in I} H(m_i)^{v_i} \cdot u^{\sum_{i \in I} m_i v_i}, y) \quad (1)$$

V. Security Analysis

In this section, we will talk about the new properties of our scheme from the Hyperledger Fabric. Based on it, our scheme realize anonymity, accountability and security. The fairness benefited from the smart contract.

Anonymity. 1) TPA cannot get the identity of user who sent the mission. 2) TPA cannot get the identity of CSP who return the proof.

proof: The identity of transaction sender is valid and anonymous, as Identity Mixer is based on Zero Knowledge Proof to maintain anonymous. Meantime, each member of the network will get a new ID when they send transaction, in this way, the member cannot trace the link between User and CSP, so does their identities.

Fairness: In the network, all TPAs can fairly gain any audit task if it was suitable.

proof: In our scheme, we use smart contracts to issue audit tasks instead of performing audit tasks, with corresponding point rewards. Every TPA has the same opportunity to accept this task. If the remuneration is not satisfied, TPA can choose not to accept it. Each task has a corresponding completion time, and the TPA that accepts the

task will be penalized after timeout, and then the task can be accepted by other TPA.

Accountability. In the network, any role that violates the rules will be exposed.

proof: Because the Ledger is maintained by many parties, the consensus mechanism is deterministic, once reached, irreversible. So the Ledger's data are extremely reliable. The main interactive information in the agreement is stored in the Ledger. When one party violates the agreement process, the record in the Ledger can be used as strong evidence to prove his breach of contract, thus achieving accountability.

Security. If someone try to interfere the scheme, the network can quickly handle it to make the network work properly.

proof: To join the alliance, the real identity of any party needs to be verified. In this way, it is difficult for adversary to be a member of the alliance. No matter which party of the alliance is compromised, administrator who can determine the real identity of the malicious party, has the flexibility to resolve it when malicious behavior occurs. Administrator can temporarily cancel malicious party's authority or expel him from the network. After the attack is eliminated, administrator can restore his authority or invite him to return to the network. If the malicious party is intentional, not be compromised, while expelling the bad guys from the network, the administrator can use the evidence saved in the ledger to safeguard the legitimate interests of the alliance through appropriate way.

VI. PERFORMANCE EVALUATION

The experiment was performed at PC laptop, which run CentOS7 on an Intel i5-4210M CPU at 2.6GHz and 4GB RAM. Our experiment consisted of three peers, a CSP, a User and a TPA. All of them are added in the same channel in Hyperledger Fabric v1.4. There are one chaincode consists of two functions, called Query and Invoke. The former is used to get the data from the task publisher. The latter is used to feed back the information for the next task. The user uses one Invoke to publish task and one Query to get the auditing result. The TPA uses one Query to get the information, one Invoke to own the task and one Invoke to response the task. The CSP uses one Query to confirm there a user asking TPA to issue an audit request. The time cost is shown in the Figure. 2.

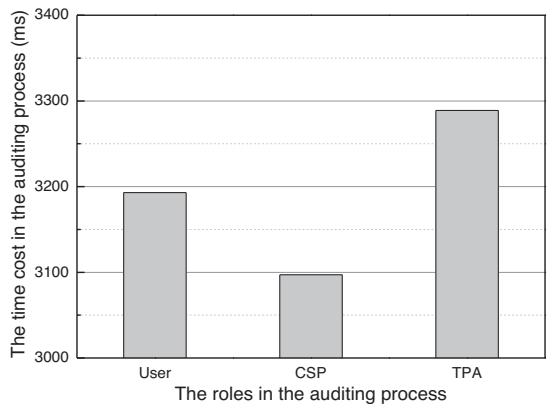


Figure 2 The time cost in the auditing process

VII. CONCLUSION

In order to solve the centralized problem of trusted TPA in auditing the integrity of data, we have proposed a secure auditing scheme based on Hyperledger Fabric. We use blockchain as a three-way communication tool, adopt the smart contract to publish task, and design a punishment mechanism. We analyzed the scheme with anonymity, fairness, accountability, security. Then we analyzed the feasibility of the scheme through experiments. With the development of Hyperledger Fabric, our system model will be fit audit better.

ACKNOWLEDGMENT

This work is supported by the National Nature Science Foundation of China (NO.61662089and NO.61761048).

REFERENCES

[1] S. Suzuki and J. Murai, "Blockchain as an audit-able communication channel," in 2017 IEEE 41st Annual Computer Software and Applications Conference (COMPSAC), vol. 2, pp. 516–522, IEEE, 2017.

[2] K. Liao, Z. Zhao, A. Doupé, and G.-J. Ahn, "Behind closed doors: measurement and analysis of cryptolocker ransoms in bitcoin," in 2016 APWG Symposium on Electronic Crime Research (eCrime), pp. 1–13, IEEE, 2016.

[3] F. Reid and M. Harrigan, "An analysis of anonymity in the bitcoin system," in Security and privacy in social networks, pp. 197–223, Springer, 2013.

[4] H.-L. Nguyen, C.-L. Ignat, and O. Perrin, "Trusternity: Auditing transparent log server with blockchain," in Companion of the The Web Conference 2018 on The Web Conference 2018, pp. 79–80, International World Wide Web Conferences Steering Committee, 2018.

[5] E. Androulaki, A. Barger, V. Bortnikov, C. Cachin, K. Christidis, A. De Caro, D. Enyeart, C. Ferris, G. Laventman, Y. Manevich, et al., " Hyperledger fabric: a distributed operating system for permissioned blockchains," in Proceedings of the Thirteenth EuroSys Conference, p. 30, ACM, 2018.

[6] J. Camenisch, S. Mödersheim, and D. Sommer, "A formal model of identity mixer," in International Workshop on Formal Methods for Industrial Critical Systems, pp. 198–214, Springer, 2010.

[7] S. Nakamoto et al., "Bitcoin: A peer-to-peer electronic cash system,", 2008.

[8] A. Baliga, "Understanding blockchain consensus models," in Persistent, 2017.

[9] M. Castro, B. Liskov, et al., "Practical byzantine fault tolerance," in OSDI, vol. 99, pp. 173–186, 1999.

[10] G. Wood et al., "Ethereum: A secure decentralised generalised transaction ledger," Ethereum project yellow paper, vol. 151, pp. 1–32, 2014.

[11] G. Ateniese, R. Burns, R. Curtmola, J. Herring, L. Kissner, Z. Peterson, and D. Song, "Provable data possession at untrusted stores," in Proceedings of the 14th ACM conference on Computer and communications security, pp. 598–609, Acm, 2007.

[12] Q. Wang, C. Wang, J. Li, K. Ren, and W. Lou, "Enabling public verifiability and data dynamics for storage security in cloud computing," in European symposium on research in computer security, pp. 355–370, Springer, 2009.

[13] B. Wang, H. Li, and M. Li, "Privacy-preserving public auditing for shared cloud data supporting group dynamics," in 2013 IEEE International Conference on Communications (ICC), pp. 1946–1950, IEEE, 2013.

[14] B. Wang, B. Li, and H. Li, "Panda: Public auditing for shared data with efficient user revocation in the cloud," IEEE Transactions on services computing, vol. 8, no. 1, pp. 92–106, 2015.

[15] B. Wang, B. Li, and H. Li, "Oruta: Privacy-preserving public auditing for shared data in the cloud," IEEE transactions on cloud computing, vol. 2, no. 1, pp. 43–56, 2014.

[16] K. Huang, M. Xian, S. Fu, and J. Liu, "Securing the cloud storage audit service: defending against frame and collude attacks of third party auditor," IET Communications, vol. 8, no. 12, pp. 2106–2113, 2014.

[17] C. Liu, J. Chen, L. T. Yang, X. Zhang, C. Yang, R. Ranjan, and R. Kotagiri, "Authorized public auditing of dynamic big data storage on cloud with efficient verifiable fine-grained updates," IEEE Transactions on Parallel and Distributed Systems, vol. 25, no. 9, pp. 2234–2244, 2014.

[18] C. Decker and R. Wattenhofer, "Bitcoin transaction malleability and mtgox," in European Symposium on Research in Computer Security, pp. 313–326, Springer, 2014.

[19] K. Hao, J. Xin, Z. Wang, Z. Jiang, and G. Wang, "Decentralized data integrity verification model in untrusted environment," in Asia-Pacific Web (APWeb) and Web-Age Information Management (WAIM) Joint International Conference on Web and Big Data, pp. 410–424, Springer, 2018.

[20] H. Yu, Z. Yang, and R. O. Sinnott, "Decentralized big data auditing for smart city environments leveraging blockchain technology," IEEE Access, vol. 7, pp. 6288–6296, 2019.

[21] B. Liu, X. L. Yu, S. Chen, X. Xu, and L. Zhu, "Blockchain based data integrity service framework for iot data," in 2017 IEEE International Conference on Web Services (ICWS), pp. 468–475, IEEE, 2017.

[22] H. Dai, H. P. Young, T. J. Durant, G. Gong, M. Kang, H. M. Krumholz,

[23] W. L. Schulz, and L. Jiang, "Trialchain: A blockchain-based platform to validate data integrity in large, biomedical research studies," arXiv preprint arXiv:1807.03662, 2018.

Proposing a Blockchain-based Solution to Verify the Integrity of Hardcopy Documents

Sthembile Mthethwa, Nelisiwe Dlamini, Dr. Graham Barbour
Council for Scientific and Industrial Research (CSIR)
Modelling and Digital Science Unit, Information Security
Pretoria, South Africa
{smthethwa, ndlamini2, gbarbour}@csir.co.za

Abstract—**Even with the ability to produce documents digitally, the paperless environment has yet to become a reality in South Africa. Hardcopy documents are still printed daily which makes them susceptible to document fraud. In South Africa, a case was reported recently, where someone who was creating fake documents was exposed. This introduces the challenge when using hardcopy documents which is loss of integrity. Thus, it is vital to have systems in place to verify document integrity and be able to determine when a document has been tampered with. Various techniques have been used to secure documents, yet the challenge persists. The combination of 2D barcodes, digital signatures, Optical Character Recognition (OCR), cryptographic hashing has proved the potential to achieve good results when combined. Recently, blockchain has been added as one of the techniques to be employed for document verification. This paper presents a proposed solution that incorporates the combination of 2D barcodes, OCR, cryptographic hashing and blockchain. As this is still on-going work, experiments are still required to demonstrate the viability of the solution.**

Keywords—blockchain, 2D barcodes, cryptographic hashing, digital signatures, document generation and validation, document verification, integrity, optical character recognition (OCR), secure hash algorithm (SHA-256), tesseract

I. INTRODUCTION

Nowadays digital documents have become a large part of every sector whether public or private resulting from the transformation of modern technology. This not only allows the dissemination of information but also preserves these documents in digital form and promotes paperless environments. However, a variety of these documents are printed every day such as; academic certificates, wills, case files, birth and marriage certificates, national identity documents, insurance documents, passports and drivers' licenses, etc. and issued to different people because printed documents are the most prevalent form of trusted communication. A great challenge is therefore experienced along with the advancement in technology; which now make it possible to easily reproduce falsified documents as they are now susceptible to unauthorized alterations [1, 2].

A series of forgery cases have been reported in the past. In India, a woman who was applying for a passport presented a falsified birth certificate, and it took some time for the concerned agency to detect the forgery. The police reported this as a common occurrence when it comes to applicants [5]. Incidents of falsified academic documents have also been reported, for instance, in Singapore, where foreign nationals were convicted after being charged with forgery of academic documents [4]. Additionally, in South Africa, two people presented fraudulent asylum documents at their workplace which they claimed were received from a state's agency official [3]. Recently a man, who produced unauthorized pay slips and bank statement and sold them to people, was exposed [6]. The counterfeiting of these documents enabled people to get credit and the credit companies are unable to trace these people after that, which has cost them a lot of money [6]. Such cases indicate the seriousness of hardcopy document fraud and demand a need to augment the solutions that solve this problem by introducing numerous methods to secure the original document from being forged in any way.

Luckily, this downside in the security of hardcopy documents has attracted attention from many researchers, and research in this area is advancing, with the aim to alleviate unauthorized altering and compromised integrity of these documents and the use of falsified documents. This entails proving that a scanned/physical copy of a document is the same as its original document [1, 2]. In doing this, various methods are implemented to record information pertaining to the original document, called the original template; and encode this information in a barcode then insert the barcode into the document and print it. Upon presentation of a copy of the original document, recognition techniques attempt to extract this information from the copy, producing a copy template. The two templates are then compared. Two key issues arise; first is the problem of storing the original template content, and secondly, the problem of extracting the copy template.

While the original template can be stored in a database, an ideal solution would be to store the original template as visible information on the original document itself. To supplement this, there are methods that are used to add information to the original document to ensure that a copy is not tampered with in any way. Amongst these methods are watermarks, document signatures, barcodes, hashing, etc. However, information stored using these methods is limited, for instance all document content cannot be included in a barcode because of the space limitations. Rather than storing the original template on the document, a hash is stored instead [2], and the problem of extracting the stored template from the copy emerges as a hash maps the document content to a string that represents the content, it does not contain the actual content in the original document.

Despite this, the use of hash values in blockchain-based methods, is undoubtedly a reliable solution that is now applied in document verification systems [7]. The blockchain which is

978-1-5386-6477-3/18/$31.00

a distributed, replicated and synchronized public ledger has made it possible to implement solutions that validate the integrity of the documents issued. However; more work still remains, the transition from eliminating the use of hardcopy documents to using digital documents hasn't been successfully navigated as yet considering that hardcopy documents are still widely used. In this paper we use the standard Optical Character Recognition (OCR) technique and incorporate the use of blockchain technology to present an agnostic solution that focuses on the ability to verify the integrity of both digital and hardcopy documents.

This paper is organized as follows, in Section 2, we present the literature review. Then a discussion of the proposed solution is presented in Section 3 and Section 4, concludes the study.

II. Literature Review

Over the years, technology has made it very easy to produce digital documents which are easy to retrieve, access and store, and encourages the change to more paperless environments. However; printed documents are still predominant and used to serve the purpose of communicating relevant information to people even though these documents have been perceived as cumbersome and inefficient [8, 9]. With the advancement of technology, the demand to verify important hardcopy documents has escalated, as the issue of fraudulent documents continues to aggravate. Falsifying a hardcopy document requires less effort these days, because these documents are inherently insecure and most have no passwords or digital signatures unlike digital documents. A long list of techniques have been proposed to mitigate the problem of forged documents mostly digital documents. But research in the area of securing hardcopy documents is starting to gain a lot of traction, since the use of these documents has become undeniable [1].

Some of the prevalent techniques include the use of watermarking, which aims to preserve the integrity of a document. Watermarking can either be in a digital or printed format [1]. This technique is still vulnerable to attacks, which may not necessarily remove the watermark imprinted, but rather disable its readability, the success of watermarks also relies on high quality printers, which incurs cost [8]. Nevertheless, it remains an active research area and continues to be improved [10]. The use of OCR, to recognise text from an image file is also prevalent. OCR is the best tool with regards to character recognition, whereby it takes in an image and returns the recognised text. Tesseract is quiet popular as an open source OCR tool, and is identified as having better accuracy and precision than other OCR tools, e.g. Transym OCR and GOCR [11]. The main issue with using OCR independently, is that it is not sufficiently reliable to determine the accuracy and is not generally 100% especially when a document has text that is not solidly black and a noisy white background, nonetheless it can be trained to achieve the expected accuracy [11, 12, 13].

Cryptographic techniques such as cryptographic hashing, Public Key Infrastructure (PKI) and digital signatures, are also very common in document verification, matter of fact it has been used in many studies to secure documents. [10] presented a solution to prove the authenticity of a document and verify it, using digital signatures. They also considered incorporating blockchain technology but decided using PKI digital signatures was sufficient for their system. Blockchain technology is flourishing in this area, its properties such as immutability, transparency and authenticity of digital records; has attracted it to a number of private and public sectors which have welcomed its use to counter document fraud [14]. Civic is one of the companies that have successfully implemented secure identity verification using blockchain technology, for this system to work, cryptographic hashing which plays a major role in Blockchain-based solutions, is employed [14]. Academic institutions have also adopted Blockchain use, e.g. Massachusetts Institute of Technology (MIT) is one of many institutions that now uses the blockchain to register digital educational certificates and allows people to authenticate these certificates, also applying vast use of cryptographic hashing for verification [6, 15]. Another system, Stampery uses a combination of blockchains, to ensure the integrity, existence and the ascription of any file or document, even communication. Once these files have been anchored to the blockchain anyone, anywhere in the world can verify their integrity [16].

Several research studies have also explored the use of two – dimensional (2D) barcodes, whereby information about the document is stored in a barcode and used later for the process of verification [1]. 2D barcodes are commonly used for document verification as they can store more data than 1D barcodes. To strengthen the security of barcodes, various cryptographic techniques are used i.e. PKI, data compression, hash functions, digital signatures [1]. [1] proposed a system whereby, barcodes are used with the help of these cryptographic techniques. Thus, showing the importance of integrating different components to design a suitable solution for the problem of document forgery. A limitation that comes with the use of barcodes is size (the amount of data that could be stored in a barcode) and once a document has numerous barcodes, it starts using a lot of space that can be used for content. Most of the proposed solutions that utilize barcodes store the entire document content in the barcode [17, 18]. In [19] we eliminated this by only storing the information that should be validated in a document, but still encountered the challenge of having numerous barcodes. Hence, in continuing with this ongoing work, the solution proposed in this study aims to decrease the number of barcodes by transferring the information used to verify the hardcopy documents to a blockchain.

From all these studies, and present implementations it can be concluded that efficient techniques must not only be effective but affordable, and implemented well to ensure the security of a document in making sure that unauthorized alterations can be detected. Thus, this study aims to provide an effective, simple and fast method of document integrity verification through the usage of 2D barcodes, OCR, cryptographic techniques and blockchain.

III. Proposed Solution

This section describes the proposed solution for verifying a hardcopy document and a digital document which is an extension of the solution that was presented initially in [19, 20]. In [19] the solution consisted of 4 components, namely; cryptographic hashing, digital signatures, OCR and 2D barcodes. Experiments were conducted using 3 different fonts i.e. Times New Roman, OCRB and AnyOCR and the highest accuracy obtained for AnyOCR was 100% which presented an opportunity to improve the solution so that it can work with different fonts. The documents generated in [20] consisted of 7 barcodes positioned at the bottom of the document, which presents a challenge to those who might want to adopt the system. The number of barcodes is dependent on the information that needs to be stored as each barcode has storage space limitation.

This led to some research on finding ways we can make the solution easily adoptable without the issue of having more barcodes to deal with when documents are generated as this might not necessarily fit in with the company's objective. All of the components used in the previous solution will still be used for this solution except only one barcode that contains information used to verify will be placed on the document. The solution is designed in a way that, if an attacker tries to tamper with the document, the system can detect those changes. In this paper we won't discuss the other components used for this solution as this was done in [20]. Only the added component and the modifications introduced are discussed.

A. Components of the proposed solution

1) Barcodes: This component was used in our previous solution, however; the limitations of storage space led to the use of more barcodes in order to accommodate all the information required to validate a document. The maximum capacity of a version 40 barcode in byte mode is between 1273 – 2953 bytes. In [20] , we observed that this barcode capacity presented flaws as it possessed high resolution which yielded negative results after printing and scanning the document. The quality of the data stored in the barcode was poor. To improve this the metadata was divided into 7 portions and stored in 7 smaller barcodes making sure that the capacity used in the barcodes is distributed equally and doesn't reach a growing rate that presents poor quality when the barcodes are decoded and read. The possibility of the capacity growing in the barcode therefore presents a downfall. This challenge might prevent the adoption of this system as a company would not agree to change their structure in-order to accommodate for more barcodes. For the reason that existing companies already have an acceptable format and layout they use to create documents. We realised when we demonstrated the first developed prototype a lot of questions were centred on the multiple barcodes, and some companies were concerned about the space used by the barcodes and the aesthetics of the document design. This challenge led us to trying other means in order to eliminate the inclusion of numerous barcodes. As a solution, we decided to transfer all the information previously stored in the barcodes to a blockchain and only have one small barcode in a document that would contain less information which will be used to verify the integrity of the document. Having one barcode would make it easier for the solution to blend perfectly with the existing structure of documents without changing it massively.

2) Blockchain: The introduction of blockchain has sparked a lot of interest in the research field. Researchers are constantly looking for means where this technology can be applied. The inception of this technology presented a lot of opportunities in the field and not only is it sparking interest in the field of cryptocurrencies, but its being studied for other purposes as well. Blockchain introduced a decentralized method of storing information, whereby all the participants have a copy of the blockchain. Thus eliminating one single point of failure. There is a plethora of blockchains e.g. Bitcoin, Ethereum, Hyperledger, Ripple, etc., one can choose from depending on what they are trying to achieve. Just like any other technology, blockchain posesses limitations i.e. the size of information one can store in the blockchain. This solution focuses more on the components selection rather than the cost of using each of these components particularly the use of barcodes and blockchain by the companies. The cost of the selected components and affordability of the company will be established accurately during the implementation of the solution, certain measures will be used to determine which blockchain will be used and deployment of this system in different company environments.

B. Proposed Solution Design

The solution consists of 2 main processes; generation and validation process. These processes utilize all the components discussed in the previous solution [20] as well as the ones discussed in the previous sub-section.

1) The Generation Process: This process includes the definition of two types of text; normal and validation text, validation text is hashed to produce a single hash value that is then encrypted with a secret key to obtain a digital signature. This process is illustrated in fig. 1 and 2.

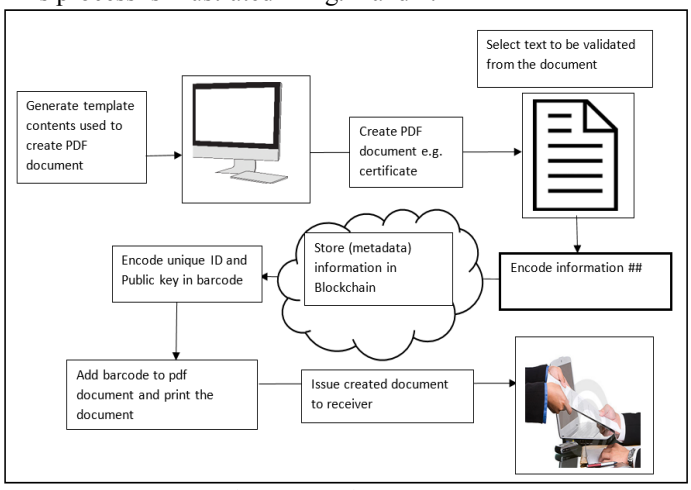


Fig. 1. Generation Process.

The digital signature and metadata are stored on the blockchain and a unique key is generated. This unique key and public key associated with the secret key used to create a digital signature are encoded to the barcode which is placed on a document. The metadata consists of; position, length, width and checksum values that are derived for each validation text, hash produced for all the validation text labels and timestamp of when the document was created. Finally, a digital copy is sent to the recipient and the pdf document is generated, printed and presented.

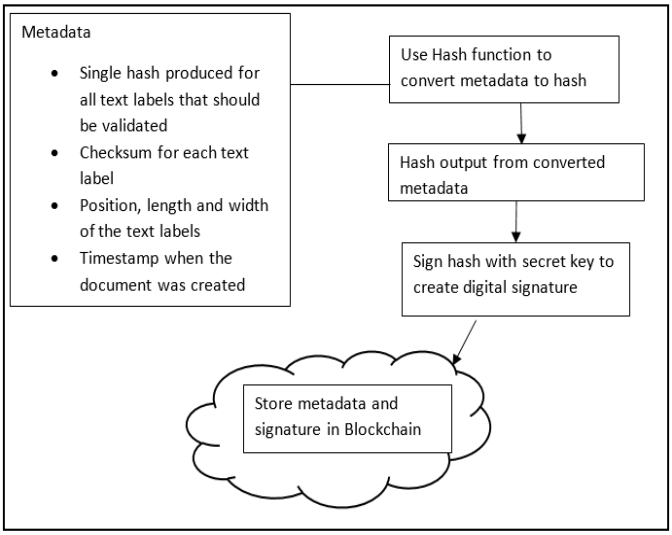


Fig. 2. Encode Information.

2) The Validation Process: Once the documents have been generated, printed and issued, the recipient can then use the documents in various cases, i.e. applying for jobs, applying for bank acounts etc. When these documents are submitted, they must be verified in order to determine whether their integrity has been maintained. Users can either submit a digital copy or a hardcopy document. If a hardcopy document is presented, the document must be scanned first in order to obtain a digital copy. To start the process of validation, the system reads in a scanned image and the barcode is identified and decoded. The barcode consists of a public key and unique key (which is used to fetch metadata related to the document from the blockchain). With the use of the public key extracted from the barcode, the digital signature is validated, if valid the metadata is extracted and used to locate the validated labels in the document. Thereafter, Tesseract OCR is used to validate the text labels. The hash (of all the validated text labels) is calculated and compared with the one from the original document (retrieved from the blockchain). If the comparison fails, it means the document has been altered. In addition to the hash that is included, a checksum for each text label is also calculated, this aids to point the exact text label that is not matching. Fig. 3, illustrates the process of document validation.

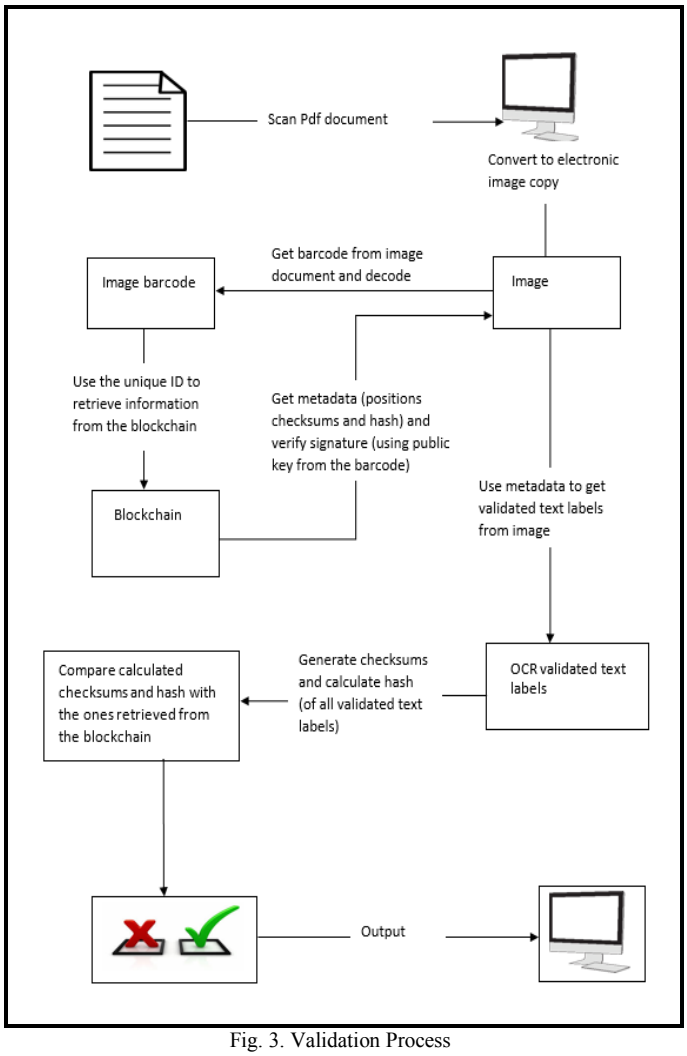


Fig. 3. Validation Process

The evaluation of the proposed solution will use a dataset generated using the generation process. The dataset would consist of 100 generated documents, using different fonts, i.e. Courier, Arial, Times New Roman etc., in order to determine the accuracy of OCR on these documents. The dataset will also be separated into different groups, on the size of text, which are; small, medium and large text. The validation process will be used to verify the integrity of the documents. To evaluate this, the information retrieved from the blockchain must be the same as the one presented when hashed.

Based on the studies conducted, the use of blockchain to verify documents is not something new and the implementations discussed in this paper have been a great success. Be that as it may most proposed and implemented solutions focus more on utilizing a single hash for verification purposes, whereby the document content is hashed and the calculated hash is saved on the blockchain [15]. To augment the existing solutions, our solution uses the blockchain to save more information about the document which is used during the process of validation, and also aims to exclude the irrelevant information in the document. Not only are we intending to identify a document that has been tampered with, but we also want to be able to show where the document has been changed.

This is made possible by including the exact position of the text that should be validated, (x and y coordinates) and the width of the text in the metadata. Before using OCR when the text is extracted from the document all white spaces are removed to minimize any additional white spaces introduced in the process. In most case a single hash value is used to represent the document's content, which cannot be obtained as a hash is designed to be a one way function, whereby we don't know the contents of the document, nor are we aware of the location of the altered text.

IV. Conclusion

This paper presented a proposed solution for the problem of document forgery, which is an extension to our previous proposed solution. The solution employed 4 techniques; OCR, cryptographic hashing, digital signatures and 2D barcodes. OCR was the first technique to be implemented, whereby documents were generated using a font known as AnyOCR (which is designed for OCR tools) and Tesseract was used to validate the documents. The experimental results yielded an accuracy of 100%, which is good. The second part of the experiment was to combine all the techniques, whereby new documents were generated and validation text was specified which was then added to the barcodes that are positioned at the bottom of the documents. Using our validation process, the system was able to detect when documents have been tampered with. This paper extends the previous solution by limiting the number of barcodes used to a single barcode and using blockchain to store the information that was previously stored in barcodes. This proposed solution will be implemented and tested for its practicability to detect forgery and ensure that the integrity and authenticity of a hardcopy document is maintained.

Acknowledgment

The authors would like to express their gratitude to the CSIR Modelling and Digital Science Unit for sponsoring this research.

References

[1] A. Husain, M. Bakhtiari, and A. Zainal, "Printed Document Integrity Verification Using Barcode," *Journal Teknologi (Sciences and Eng*, pp.99-106, 2014.

[2] M.H. Eldefrawy, K. Alghathbar, and M.K. Khan, "Hardcopy document authentication based on public key encryption and 2D barcodes," In *Biometrics and Security Technologies (ISBAST), 2012 International Symposium,* pp. 77-81, IEEE, March 2012.

[3] R. Jain, and D. Doermann, "Visualdiff: Document image verification and change detection," In *Document Analysis and Recognition (ICDAR), 2013 12th International Conference on* pp. 40-44, IEEE, August 2013.

[4] N. Ganesan, "*Three foreigners jailed in Singapore for submitting fake academic certificates _ Human Resources Online*, *Human Resources*," Available at: http://www.humanresourcesonline.net/three-foreigners-jailed-for-submitting-fake-academic-certificates/ (Accessed: 07 February 2018), 2017.

[5] S. Kalipa, "*Home Affairs official sold us fake papers _ IOL News*, *Crime and Courts IOL News*," Available at: https://www.iol.co.za/news/crime-courts/home-affairs-official-sold-us-fake-papers-1929048 (Accessed: 08 February 2018), 2015.

[6] C. Lewis, "SABC News exposes fake payslips, bank statements," 8 May 2018, 2018. [Online]. Available: http://www.sabcnews.com/sabcnews/sabc-news-exposes-fake-payslips-bank-statements/. [Accessed: 13-Sep-2018].

[7] B. Cresitello-Dittmar, "Application of the Blockchain For Authentication and Verification of Identity," Independent Paper, 2016.

[8] Y. S. Joshi, "The Future of Enterprise Printing: Securing Hardcopy Documents in the Digital Age: white paper," CIO Insight, no. July, pp. 1–12, 2014.

[9] C. Lakmal, S. Dangalla, C. Herath, C. Wickramarathna, G. Dias, and S. Fernando, "IDStack - The common protocol for document verification built on digital signatures," 2017 Natl. Inf. Technol. Conf. NITC 2017, vol. 2017–Septe, no. September, pp. 96–99, 2018.

[10] S. R. M. Oliveira, M. A. Nascimento, and O. R. Zaiane, "Digital Watermarking: Status, Limitations and Prospects," Technical Report TR 02-01, Department of Computing Science, Alberta University, Edmonton, Alberta, Canada, 2002.

[11] S. Dhiman, and A. Singh, 2013. "Tesseract vs gocr a comparative study," *International Journal of Recent Technology and Engineering*, *2*(4), pp.80, 2013.

[12] C. Patel, A. Patel, and D. Patel, 2012. Optical character recognition by open source OCR tool tesseract: A case study. *International Journal of Computer Applications*, *55*(10), 2012.

[13] V. S. Chandel, "Deep Learning based Text Recognition (OCR) using Tesseract and OpenCV," 06 June 2018, 2018. [Online]. Available: https://www.learnopencv.com/deep-learning-based-text-recognition-ocr-using-tesseract-and-opencv/. [Accessed: 13-Sep-2018].

[14] B. Karanjia, A. G. Karanth, S. Veerapaneni, S. Goswami, A. Sharma, and M. Boda, "Blockchain in the Public Sector – Transforming Government Services through Exponential Technologies," 2017.

[15] Universa, "Blockchain in Education," 23 May, 2018. [Online]. Available: https://medium.com/universablockchain/blockchain-in-education-49ad413b9e12. [Accessed: 04-Sep-2018].

[16] A. S. de P. Crespo and L. I. C. García, "Stampery Blockchain Timestamping Architecture (BTA) - Version 6," 2017, pp. 1–21.

[17] M. Salleh, and T.C. Yew, "Application of 2D Barcode in Hardcopy Document Verification System," In *ISA*, pp. 644-651, June 2009.

[18] C.M. Li, P. Hu, and W.C. Lau, "Authpaper: Protecting paper-based documents and credentials using authenticated 2D barcodes," In *Communications (ICC), 2015 IEEE International Conference,* pp. 7400-7406, IEEE, June 2015.

[19] S. Mthethwa and N. P. Dlamini, "Verifying the Integrity of Hardcopy Document Using OCR," in 2nd International Women in Science Without Borders (WiSWB)-Indaba, 2018.

[20] N. Dlamini, S. Mthethwa, and G. Barbour, "Mitigating the Challenge of Hardcopy Document Forgery," in International Conference on Advances in Big Data, Computing and Data Communication Systems (icABCD), 2018, pp. 1–6.

2018 IEEE 24th International Conference on Parallel and Distributed Systems (ICPADS)

Blockchain Based Data Integrity Verification in P2P Cloud Storage

Dongdong Yue[1], Ruixuan Li[1] ✉ , Yan Zhang[2], Wenlong Tian[1], and Chengyi Peng[1]
[1]Intelligent and Distributed Computing Laboratory, School of Computer Science and Technology, Huazhong University of Science and Technology, Wuhan, China
[2]School of Computing, Engineering and Mathematics, Western Sydney University, Sydney, Australia
Email: rxli@hust.edu.cn

***Abstract*—With the popularity of cloud storage, how to verify the integrity of data on the cloud has become a challenging problem. Traditional verification framework involves the Third Party Auditors (TPAs) which are not entirely credible. In this paper, we present a framework for blockchain-based data integrity verification in P2P cloud storage, making verification more open, transparent, and auditable. In this framework, we present Merkle trees for data integrity verification, and analyze the system performance under different Merkle trees structures. Furthermore, we develop rational sampling strategies to make sampling verification more effective. Moreover, we discuss the optimal sample size to tradeoff the conflict between verification overhead and verification precision, and suggest two efficient algorithms of order of verification. Finally, we conduct a series of experiments to evaluate the schemes of our framework. The experimental results show that our schemes can effectively improve the performance of data integrity verification.**

***Keywords*-P2P Cloud Storage, Blockchain, Data Integrity Verification, Sampling, Merkle Trees**

I. Introduction

Due to the rapid growth of information sharing and exchange, more and more companies and individual users choose to store their data on the cloud. Traditional data privacy and integrity is ensured through data encryption, multiple signatures, anonymous mechanisms and so on. However, users lose control of these data when these data are stored on the cloud. Therefore, how to verify the integrity of the data stored on the cloud becomes an important problem. The data storage and integrity verification workflow framework under traditional cloud storage is shown as Fig. 1. In this framework, there are three objects: Clients, Cloud Storage Servers (CSS), and Third Party Auditor (TPA) [17]. The client stores his own data on the CSS, and sends relevant information to the TPA to verify the integrity of the data. When data integrity verification is performed, the CSS will submit the proofs to the TPA. Finally, the TPA verifies the integrity of the cloud-stored data based on these proofs and the user's previously transmitted useful information.

The rise of Peer-to-Peer (P2P) cloud storage, which exploiting a large amount of idle disk space, makes it possible to rent cheap storage space. In a P2P cloud storage system, each user can be either a client that rents storage space or a lender that lends his own idle storage space. Users can obtain cheap storage space, and lenders can benefit from lending their idle space. Sia [15] and Storj [18] are two mature P2P cloud storage platforms. Each user in these platforms can share its own storage space and gain revenue from the sharing. The redundant backup mechanism in these platforms makes the data storage more reliable.

Starting from an article by Satoshi Nakamoto in 2008 [11], Bitcoin has entered our horizon and triggered an upsurge in blockchain technology. In a simple term, blockchain is a distributed database that includes transactions, blocks, consensus mechanisms, smart contracts, and so on. Each work in the blockchain is recorded in the form of a transaction. Multiple transactions form a block, and multiple blocks are linked together to form a blockchain. The header field of each block contains the hash of the previous block, thus forming an ordered chain. The advantage of blockchain technology is that it provides a decentralized, open, transparent, auditable, and tamper-proof record. All blockchain participating nodes can verify the transactions recorded on the chain. These transactions are permanently recorded on the chain and cannot be maliciously modified. The consensus mechanism in the blockchain ensures that the state of the entire blockchain is consistent without the participation of any third parties. Smart contracts are contracts stored in the blockchain. When the system meets the contract execution conditions, the contract automatically executes the corresponding content. The emergence of smart contracts makes the blockchain more intelligent.

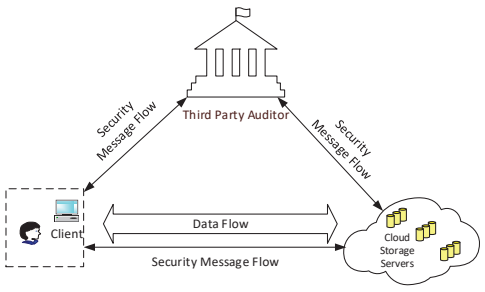


Fig. 1. Data storage and integrity verification on the cloud under a traditional architecture. There is only data flow between client and cloud storage servers. Third party auditor assists clients in verifing data integrity.

978-1-5386-7308-9/18/$31.00 ©2018 IEEE
DOI 10.1109/ICPADS.2018.00079

In the traditional scenario where the TPA is introduced for data integrity verification, the TPA may not be completely trusted. However, there are also trust issues in verification scenarios that do not involve TPA. From the perspective of CSS, malicious Clients may intentionally claim that their data is not completely preserved by CSS, thus extorting damages from CSS. From the Clients perspective, the dishonest CSS may still claim to guarantee the integrity of the data without saving the complete data, thus harming the Clients interests. Therefore, we can introduce blockchain for data integrity verification in the case of mutual distrust between CSS and Clients. Sia [15] is a P2P cloud storage platform that combines blockchain for data integrity verification. On this platform, blockchain is used to record relevant information for data integrity verification. Such information will be stored permanently on the blockchain and cannot be modified, which makes the verification results more reliable. However, Sia [15] neither provided a complete verification mechanism, nor considered how to select partial data shards for verification. While in the case of limited resources or high real-time requirements, it is necessary to verify the integrity of the whole data by verifying partial data shards. Therefore, how to select partial data shards and guarantee the performance of data integrity verification is a problem worth studying.

In this paper, we firstly propose a general data integrity verification framework to solve this problem for blockchain based P2P cloud storage. Then we analyze the performance of Merkle trees with different structures, and propose a sampling strategy to select shards for validation. The main contributions of this paper are summarized as follows.

- We propose a general data integrity verification framework for blockchain based P2P cloud storage. This framework solves the problem of untrustworthy in traditional verification mechanism. In this framework, clients and cloud servers that do not trust each other can interact.
- Based on the proposed framework, we use Merkle trees for data integrity verification based on blockchain and analyze the performance of Merkle trees under different structures. Then, we present a sampling strategy and discover an optimum sample size to verify data integrity. These make it more effective to verify the integrity of data when only a part of data can be verified because of limited computation resources.
- We conduct extensive simulations to evaluate the performance of the proposed framework by implementing a prototype system. Simulation results demonstrate the feasibility of the proposed framework and validates our theoretical analysis.

The rest of this paper is organized as follows. In Section II, we discuss the related work from three aspects. Then, Section III describes the detailed design of our framework. Experimental studies are presented in Section IV. Finally, Section V concludes the paper with some remarks.

II. Related Work

This section firstly reviews the P2P cloud storage. Then, we introduce how data integrity verification works in traditional cloud storage. Finally, the research work on applying blockchain to verify data integrity is elaborated.

A. P2P Cloud Storage

The mainstream cloud storage systems, such as Google's GFS (Google File System) [3], Amazon's elastic cloud, and open source HDFS (Hadoop Distributed File System), have adopted a similar centralized architecture, in which there is a huge risk of single point failure. The central server is easy to become the bottleneck of the system. Once the central server collapses, it may cause the whole cloud storage service to be unavailable. Many features of P2P systems, such as non centralization, scalability, robustness, high performance and load balancing, can solve this kind of problem. Ji-Yi *et al.* proposed a general model of P2P based cloud storage system, which can provide higher quality of cloud storage services [8].

Some other researchers work on data management in P2P cloud storage. As cloud computing is generally regarded as the technology enabler for Internet of Things, Teing *et al.* tried to ensure the most effective collection of evidence from P2P cloud enabled IoT infrastructure [14]. Since cloud severs and users usually locate outside the trusted domain of data owners, P2P storage cloud brings new challenges for data integrity and access control when data owners store data on it. To address this issue, He *et al.* designed a ciphertext-policy attribute-based encryption scheme and a proxy re-encryption scheme [7]. Based on these schemes, they further proposed a secure, efficient and fine-grained data access control mechanism for P2P storage cloud. However, there is little work on data integrity verification in P2P cloud storage.

B. Data Integrity Verification in the Traditional Cloud Storage

There are mainly two types of traditional data integrity verification mechanisms. One is Provable Data Possession (PDP); the other is Proofs of Retrievability (POR). PDP can quickly verify whether the data stored on the cloud is intact, while POR can restore the damaged data when the data integrity is compromised. The basic PDP authentication method is proposed by Deswarte *et al* [6]. Before the user uploads his own data, he uses the Hash-based Message Authentication Code (HMAC) to calculate the Message Authentication Code (MAC) value of the data and saves it at local. When verifying these data, the user first downloads the data stored on the cloud, then calculates the MAC value of the downloaded file, and compares it with the MAC value previously saved to determine whether the data integrity is guaranteed.

Although this mechanism is simple, directly downloading complete data requires a lot of resources and may lead to leakage of data privacy. Then Seb *et al.* proposed a block-based scheme to reduce the computational overhead [12]. Due to the deterministic verification method, the verification result may not be completely correct. Then Ateniese *et al.*

proposed using probabilistic strategies to complete the integrity verification. They used the homomorphic properties of RSA signature mechanism, gathered evidence in a very small value, which greatly reduced the communication overhead [1]. Subsequently, Curtmola *et al*. implemented the data integrity verification mechanism in the case of multiple copies, but it didn't support the dynamic operation of data [5]. Ateniese *et al.* first considered the dynamic operation of data. They presented simple modified mechanism of the PDP based on their previous work [1], making it support dynamic data manipulation [2]. However, this mechanism does not support insert data. In response to this problem, Wang *et al.* implemented a PDP mechanism that supports full dynamic operation. This mechanism uses the Merkle tree to guarantee the correctness of the data block, and uses the Boneh-Lynn-Shacham (BLS) signature to guarantee the correctness of the data block value [17]. Later, they also proposed a privacy protection verification scheme that uses random masking techniques to make TPA unable to know the data information provided by cloud service providers.

Although the PDP authentication mechanism can efficiently verify the integrity of data, it cannot recover invalid data. Juels *et al.* proposed a sentinel-based POR mechanism [9]. Nevertheless, it can only conduct a limited number of verifications. Subsequently, Shacham *et al.* used the BLS short message signature mechanism to construct homomorphic verification tags, which can reduce the communication overhead for verification [13]. However, it is difficult to be implemented. Wang *et al.* proposed using the linear features of the error correction code to support partial dynamic operations, while it could not support the dynamic insertion of data [16]. Chen *et al.* optimized Wang's mechanism and used the Reed-Solomon erasure code technique to recover the failed data, which can improve the recovery efficiency, but increase the computational cost [4].

C. Blockchain based Data Integrity Verification

The problems of incomplete trust caused by traditional data integrity verification make it an inevitable trend to integrate blockchain technology into data integrity verification of cloud storage. Liu *et al.* applied the blockchain technology to the Internet of Things (IoT) and proposed a blockchain-based data integrity service framework. Without relying on TPA in this framework, data owners and data consumers can be provided with more reliable data integrity verification [10]. However, their work is to target at IoT and is not applicable to scenarios with P2P cloud storage. How to design a universal data integrity verification framework in P2P cloud storage is a worthy research issue. It is because that each node can be either a storage provider or a storage renter under the combination of P2P cloud storage and blockchain. Therefore, this paper focuses on proposing a general and practical data integrity verification framework combined with the blockchain under the P2P cloud storage scenario.

III. The Proposed Method

In this section, we firstly introduce our framework for data integrity verification. Then, we describe the structure of the Merkle trees. The performance of different structures of Merkle trees are analyzed in terms of computation overhead and communication overhead. Finally, we detailedly illustrate some strategies of sampling verification and propose the method about calculating the best sample size.

A. Data Integrity Verification Framework

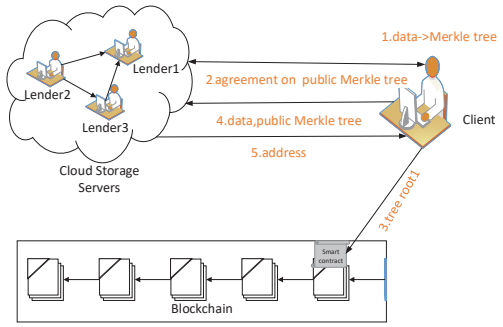
The framework of data storage and integrity verification under blockchain based P2P cloud storage is shown as Fig. 2. In this framework, there are three entites: Clients, Cloud Storage Servers (CSS), and Blockchain (BC). Clients upload their own data to the CSS and use BC to verify data integrity. The overall workflow is divided into two stages. As shown in Fig. 2(a), there are five steps in the preparation stage. In the first step, the client will slice his data into several shards, then uses these shards to construct a hash Merkle trees. In the second step, the client and CSS will agree on the hash Merkle trees. In the third step, the client will store the root of this hash tree denoted as $root1$ on the blockchain. In the fourth step, the client uploads his data and public Merkle trees to CSS. In the fifth step, CSS return the address that stores the client's data to client. As shown in Fig. 2(b), there are also five steps in the verification phase. In the first step, the client will send an challenge number si to CSS, which selects shard i to verify. In the second step, CSS use hash function to calculate a hash Digest i', according to si and shard i. In the third step, CSS send Digest i' and the corresponding auxiliary information to BC. In the fourth step, the smart contract on the blockchain will calculate a new hash root denoted as $root2$, and compare $root1$ with $root2$. If they are equal, the data integrity has been guaranteed; otherwise, the data integrity has been corrupted. In the last step, the BC will return the verify result to the client.

In this framework, clients will place the root of the Merkle trees on the blockchain before uploading the data. Due to the non tamperability property in blockchain, any client or CSS cannot modify the root stored on the blockchain, which makes integrity verification more credible. At the same time, due to the distributed nature of the blockchain, there is little possibility that the data on the blockchain will be damaged. Hence, the data integrity verification is more reliable.

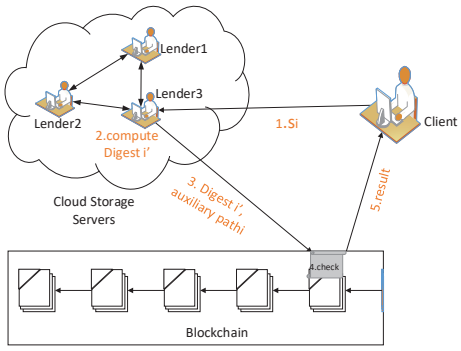
B. Structure of the Merkle Tree

The advantage of using Merkle trees to verify the data integrity is that the entire data file can be verified by a small segment of the entire data shards, which is relatively small regardless of the size of the original file. The structure of Merkle trees is shown as Fig. 3. The public part of this tree needs to be uploaded to the P2P CSS to assist in validating each data shard of the private part. The private part of this tree is composed of data shards $shard_i$ and random challenges r_i. The random challenges can only be sent to the P2P CSS when the client needs to verify the corresponding data shards.

Therefore, the private part is locally saved by the clients. The data uploaded by clients to P2P CSS are the data shards after data slicing. Since it is a tree structure, we can study the Merkle trees with different branches. This paper analyzes the different structures of Merkle trees, then discusses the communication overhead and computational overhead of the system under these structures.



(a) Preparation stage



(b) Verification stage

Fig. 2. Illustration of blockchain based data integrity verification framework. (a) shows the workflow of user uploading data to P2P cloud storage servers. (b) shows the workflow of verifying data integrity in P2P cloud storage servers combined with blockchain.

1) Communicational Cost: Since the public tree needs to be passed from the user to the cloud servers, the size of the public tree is proportional to the communication cost. Assuming that the output degree of each node of the tree is m, and the total number of leaf nodes, namely the total number of shards, is n, then the total number of nodes of the public tree is:

$$\begin{aligned} sum(m) &= m^0 + m^1 + m^2 + \cdots + m^{log_m n} \\ &= m^0 + m^1 + m^2 + \cdots + n. \end{aligned} \tag{1}$$

To explore the relationship between m and $sum(m)$, we assume that the number of branches of the two types of Merkle trees are m_1, m_2 respectively, which is satisfied with $m_1 = (m_2)^2$.

When $m_1 = (m_2)^2$ and n is fixed, the statement that $sum(m_1) < sum(m_2)$ is true. So we can get the conclusion that when m (the branching of the Merkle tree) increases, the size of public tree decreases, the communication cost decreases.

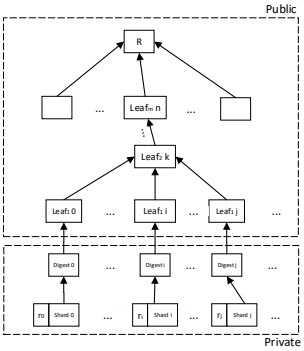


Fig. 3. The structure of Multi-Branch Tree. There are two parts of this Merkle Trees, public and private. The arrows in this figure represent performing a hash function. The bottom layer of the private part consists of shards $shard_i$ and random challenging numbers r_i. The second layer of the private part is the hash result $Digest_i$ of the bottom layer. $Leaf_m n$ in the public part of the Merkle tree is the nth leaf node of the m layer. The top of the tree is the hash root of this Merkle Trees, denoted as R. The public part is uploaded to cloud storage servers, and the private part is stored by the client.

2) Computational Cost: We measure the computational cost by calculating the delay in completing the computation, because the computational cost is proportional to the computational latency.

(i) Verify Shards - The latency of verifying shards = calculation times $\times$ time cost of each calculation, and the calculation times of each shard is $F1(m) = log_m n$. When n is fixed, $F1(m)$ decreases as m increases. So the latency of verifying shards decreases as m increases.

(ii) Generate Merkle Trees - The latency of generating Merkle trees is proportional to the size of public tree. From previous we know that the total number of nodes of the public tree decreases as m increases. So the latency of generating Merkle trees decreases as m increases.

(iii) Generate Auxiliary Path - The latency of generating the auxiliary path is proportional to the size of the auxiliary path. The size of the auxiliary path is $F2(m)$.

$$\begin{aligned} F2(m) &= (m-1)log_m n = (m-1)\frac{lnn}{lnm} \\ &= lnn(\frac{m}{lnm} - \frac{1}{lnm})(m \geq 2). \end{aligned} \tag{2}$$

$F2(m)$ increases as m increases. But each element in the auxiliary path is a hash string, so that it takes up little memory overhead. It is also very quick to get the auxiliary path through the Merkle trees (as shown in the experiment), the cost of calculation is small. Therefore, the additional communication and computational costs of the auxiliary path caused by the multi-branch Merkle trees structure are negligible.

C. Sampling Verification

Due to the limitation of real-time requirement, there is no need to verify all data shards to confirm the data integrity. In these cases, we need to choose a part of shards to verify.

Selecting a portion of the shards from the overall data shards for validation is regarded as a sampling problem. In our scenario, we choose random sampling strategy because the difference between each data shard is very small. Then, we adopt repeated sampling (sampling with replacement) to ensure that the probability of each shard to be chosen is the same, which guarantees the fairness.

1) Sampling Strategy: We adopt two sampling strategies: simple random sampling and stratified sampling. Firstly, simple random sampling is to generate sampling data through random functions. Secondly, stratified sampling is to stratify the overall data according to certain characteristics and then perform simple random sampling in each layer.

(i) Simple Random Sampling - The random function is used to randomly select shards for verification. Simple random sampling was first adopted at the beginning of the operation of the system, because we knew little about the service providers at this time.

(ii) Stratified Sampling - After a period of simple random sampling, we would get the service providers' ability to guarantee data integrity. Higher ability means a higher probability of preserving complete data. According to the grade of service providers' ability, we can divide the providers into several layers. Then perform random sampling over each layer. Assuming that providers are divided into three layers, denoted as $R1, R2, R3$ respectively, the sample sizes for each layer are $N1, N2, N3$ respectively, and the sample size of sampling is N. We need to ensure $N = N1 + N2 + N3$. The sample size for each layer decreases proportionately to the grade of the service providers' ability to guarantee data integrity.

These two sampling strategies are combined to perform sampling. At the beginning of the system, Simple Random Sampling will be performed to get these service providers' credit rating. Higher credit rating means the ability to ensure higher data integrity. Then, according to these credit rating, we can divide the providers into several layers, and perform Stratified Sampling. After a while, we will rerun the Simple Random Sampling to update the providers at each layer, then perform Stratified Sampling continue.

2) Sample Size: The total number of validated shards is called sample size. Sample size of sampling will affect the cost and precision of verification. For the verification cost, the larger the sample size, the more pieces of shards need to be verified, and the higher the verification cost. That is the verification cost is positively correlated with the sample size. For the verification precision, the larger the sample size, the more representative it is of the overall data, and the higher the verification precision. It is means that the verification precision is also positively correlated with the sample size.

(i) Verification Cost - A simple linear function can be used to express the relationship between sample size N and verification cost C:

$$C = c_0 + c_1 N, \tag{3}$$

where $c_0 > 0, c_1 > 0$. c_0 represents the basic cost, and c_1 represents the influence degree of sample size. The values of c_0 and c_1 are not of direct interest, but used to establish a linear relationship between C and N.

(ii) Verification Precision - Suppose the total number of data shards is n, where there are f invalid (lost or tampered) shards, and the sample size of sampling is N. The variable V is used to represent the number of invalid shards detected in the sampled data, then the probability P_V represents at least one invalid shard has been detected, which is:

$$\begin{aligned} P_V &= P\{V >= 1\} = 1 - P\{V = 0\} \\ &= 1 - (\frac{n-f}{n})^N. \end{aligned} \tag{4}$$

Since we used repeated sampling, the probability that a shard selected randomly was valid is $\frac{n-f}{n}$. So when the sample size is N, the probability that no invalid shard is detected is $P\{V = 0\} = \underbrace{\frac{n-f}{n} * \frac{n-f}{n} * \cdots * \frac{n-f}{n}}_{\mathrm{N}}$.

(iii) Overall Consideration - The ideal situation is to spend as little verification cost as possible and obtain as high verification precision as possible. However, the verification cost and the verification precision are in contradictory relationship. What we need to do is finding an optimal sample size to tradeoff the contradiction between the verification cost and the verification precision. Therefore, we propose a Loss Function $L(N)$ to comprehensively consider the impact of sample size on verification cost and verification precision, which is:

$$\begin{aligned} L(N) &= C + \lambda \frac{1}{P_V} \\ &= c_0 + c_1 N + \lambda(\frac{1}{1 - (\frac{n-f}{n})^N}), \end{aligned} \tag{5}$$

where $N \in (0, n], c_1 > 0, c_0 > 0$. The relationship among loss, cost and precision reflected by the loss function should conform to the actual situation. That is, the higher the cost, the greater the loss, i.e. the loss is proportional to the cost. The higher the precision, the smaller the loss, i.e. loss and precision are inversely proportional. Therefore, we adopted Equation. 5 to express the relationship among loss, cost and precision in a simplified way. λ balances the importance between verification cost and verification precision. In practice, if we have a different emphasis on validation precision and validation overhead, we can change the value of λ. In order to simplify the analysis, we set $\lambda = 1$ in our paper. As c_0, c_1, n, f can be obtained as constants, $L(N)$ can be regarded as a function of variable N. Our goal is to find an optimal N to make $L(N)$ minimum, we could get the following theorem to calculate the optimal N.

Theorem 1: When $N \in (0, n]$, there exists the optimal $N = N2$ to make $L(N)$ minimum, where $N2 = log_a \frac{(2c_1 - lna) - \sqrt{lna^2 - 4c_1 lna}}{2c_1}$.

3) Order of Verification: After getting the samples, appropriate strategies can be used to determine the order, in which the samples are verified. We can abstract this issue as follows. Given the sample size N, assuming there exists an invalid shard, denoted as i, to discover invalid shard i, which kind of validation strategies should be adopted so we can verify the least amount of shards, namely the verification cost is minimal.

Here we apply several basic algorithms, which are sequential verification, block verification, exponential verification, binary verification and fibonacci verification, to our new scenario. Due to the space limitation, we do not elaborate on the implementation steps of each algorithms in this paper.

IV. Experiments

In this section, we firstly describe the implementation of a prototype system of the proposed framework. Then, we conduct some experiments about the structure of Merkle trees and sampling verification, and analyze their performance through the experimental results.To simplify the description, symbols used in this paper are shown in Table I.

TABLE I
NOTATIONS IN THIS PAPER

Notation	Description
CSS	Cloud Storage Servers
TPA	Third Party Auditor
BC	Blockchain
n	the total number of shards
m	the branch number of the Merkle trees
BBT	Binary-Branching Merkle trees
FBT	Four-Branching Merkle trees
EBT	Eight-Branching Merkle trees

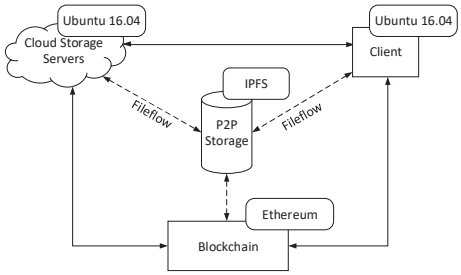


Fig. 4. The implementation of data integrity verification framework. In this fig, solid lines represent logical relationships and dotted lines represent actual interactions.

A. Framework Implementation

Fig. 4 shows the structure of the data integrity verification framework. The blockchain system is implemented by Ethereum as it is the most mature blockchain platform that supports smart contract. Clients and CSS are emulated through Ubuntu 16.04. The P2P cloud storage servers is implemented by IPFS, standing for Inter-Planetary File system, which is an attempt to share files in an HTTP manner. This paper focuses on how to select a part of data shards for data integrity verification and guarantee high performance. As for the impact of blockchain on the throughput of the system, this is the other issue deserves further study, which is not discussed in the experimental part of this paper. The implementation steps of this framework is illustrated in Table II.

TABLE II
IMPLEMENTATION STEPS OF DATA INTEGRITY VERIFICATION FRAMEWORK

Step	Entities	Operation
	Preparation Stage	
1	Client	data $\rightarrow$ Merkle trees
2	Client $\longleftrightarrow$ CSS	agreement on Merkle trees
3	Client $\rightarrow$ IPFS	upload root1
4	IPFS $\rightarrow$ Client	return ipfs-address-root1
5	Client $\rightarrow$ BC	upload ipfs-address-root1
6	Client $\rightarrow$ IPFS	upload data, Merkle trees
7	IPFS $\rightarrow$ Client	return ipfs address of each shard
	Verification Stage	
1	Client $\rightarrow$ IPFS	send ipfs address of shard i
2	IPFS	compute new root2
3	IPFS $\rightarrow$ BC	send ipfs-address-root2
4	BC	compare(ipfs-address-root1, ipfs-address-root2)
5	BC $\rightarrow$ Client	send verification result

B. The Structure of Merkle Trees

we conduct the experiments under three different Merkle trees structures, which are Binary Branching tree (BBT), Four-Branching tree (FBT), Eight-Branching tree (EBT). Then assuming the total number of shards is from 16 to 16384 (16, 64, 128, 256, 512, 1024, 2048, 4096, 8192, 16384). The performance of the three Merkle trees structures was compared in terms of the time cost of verifying shards, the time cost of building Merkle trees, and the time cost of generating auxiliary path.

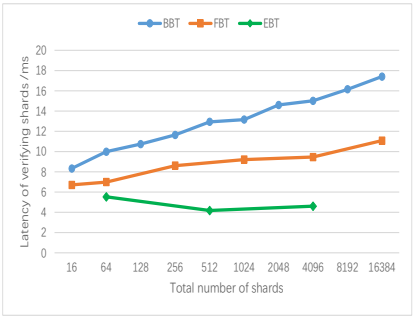


Fig. 5. The relationship between the verification latency and the total number of shards.

Since n and m need to satisfy the relationship $n = m^k (k = 0, 1, 2, ...,)$ to form a full tree, the n that different m can take is not completely the same. In the figure, the total number of shards that BBT, FBT, EBT can obtain is not exactly the same. Fig. 5 shows that EBT performs best and BBT performs worst in terms of the time cost of verifying shards. From Fig. 6, we can see that FBT and EBT are significantly better than BBT. This is mainly reflected in the following two aspects : (1) the latency of generating the Merkle Tree of FBT and EBT is always smaller than BBT and (2) as shards grow, the FBT's and EBT's latency growth rate are significantly smaller

than BBT's. As the computation cost is positively related to the computation delay, FBT and EBT are better than BBT in computation overhead. In terms of the auxiliary path, the previous theoretical analysis has concluded that the auxiliary path size will increase with the number of branches. However, each additional element of the auxiliary path is a hashed string, the increased storage space is small. At the same time, it can be seen through Fig. 7 that the time delay for generating auxiliary paths does not significantly increase with branches. Therefore, the additional communication and computation costs of the auxiliary path caused by the multiple branching tree structure are negligible. In summary, the performance of FBT and EBT are better than BBT.

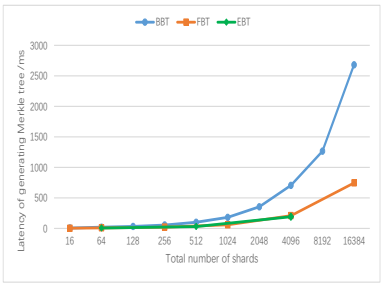


Fig. 6. The relationship between the latency of generating Merkle Trees and the total number of shards.

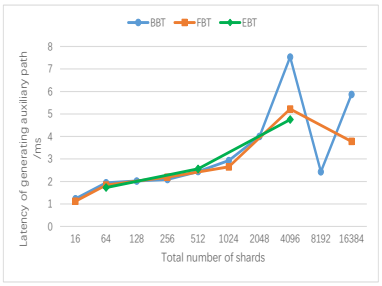


Fig. 7. The relationship between the latency of generating auxiliary path latency and the total number of shards.

C. Sample Size

In Section III, we have introduced the Loss Function $L(N)$, which involves c_0, c_1, n, f, N, to decide the suitable sample size. To simplify the calculation, we set $c_0 = 0$, then conduct two sets of experiments. In the first set of experiments, we assume the total number of shards $n = 10000$, $c1 = 0.053$ and take four different value of f, that is $f/n = 0.001, f/n = 0.002, f/n = 0.01, f/n = 0.05$. In the second set of experiments, we assume the total number of shards $n = 10000$, $f/n = 0.002$ and take four different value of $c1$, that is $c1 = 0.7, c1 = 0.4, c1 = 0.1, c1 = 0.01$. Then describe the general direction of $L(N)$ as N changes.

Fig. 8 and Fig. 9 show that as the sample size N increases, the value of the Loss Function $L(N)$ decreases from a value firstly, after reaching a minimum value, it starts to increase continuously. Thus, there exits a most approprite value of N leading to the minimum value of $L(N)$. From Fig. 8, we can see that when f/n is larger, the minimum value of $L(N)$ is closer to the Y-axis and X-axis. That means the more shards fail, the smaller the optimal sample size. As f/n increases, the minimum value of the Loss Function $L(N)$ decreases. This means that the overall validation performance of this system increases as the number of failed shards increases. From Fig. 9, we can see that when $c1$ increases, the optimal sample size decreases while the minimum value of the Loss Function increases. It means that when the weight of verification overhead increases, the optimal sample size decreases, while the overall validation performance of this system decreases.

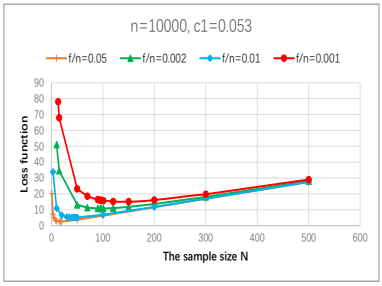


Fig. 8. The relationship between Loss Function $L(N)$ and the sample size N when f/n changes.

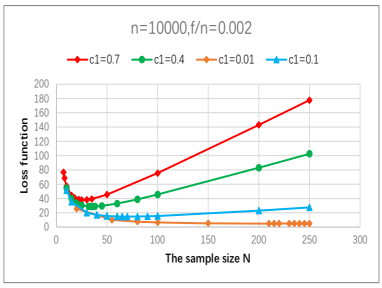
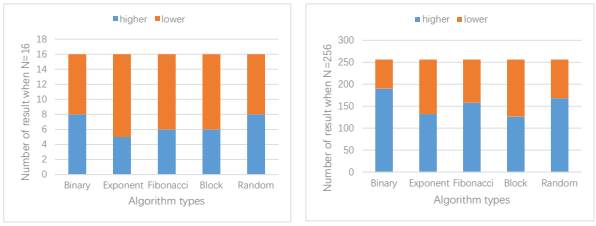


Fig. 9. The relationship between Loss Function $L(N)$ and the sample size N when $c1$ changes.

D. Order of Verification

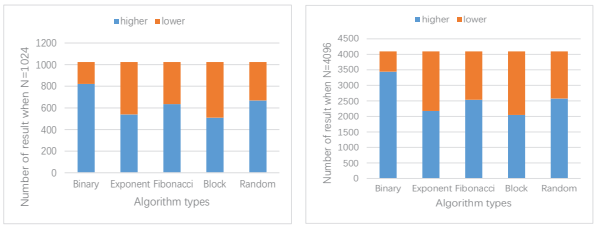
In order to compare the verification performance of different algorithms, we set the sample size from 16 to 4096 (16, 256, 1024, 4096), and use sequential validations' cost as a benchmark. We make the sample data shard one invalid at a time, then count the number of verification costs higher or lower than the baseline at the failure location using different algorithms.

Fig. 10-11 show that the proportion of verification cost higher than the benchmark increases with the increase of sample size N. From Fig. 10a we know that when N is small, almost each algorithm works better than the benchmark. Comparing Fig. 10b, Fig. 11a and Fig. 11b, we can see that with the increases of sample size N, the performance of Binary verification is getting worse and worse, while the performance



(a) N=16 (b) N=256

Fig. 10. Statistics on validation overhead when N=16 and N=256.



(a) N=1024 (b) N=4096

Fig. 11. Statistics on validation overhead when N=1024 and N=4096.

of other algorithms remain stable. Fibonacci verification and Random verification are work better than the baseline when $N = 16$. Exponent verification and Block verification always work better than the baseline no matter how N increase. Thus, we can get the conclusion that although the verification costs of these algorithms are increasing, Exponential verification and Block verification work better than others.

V. Conclusion and Future Work

In this paper, we have proposed a general verification framework in P2P cloud storage. It solves the problem of untrustworthy in traditional verification mechanism by utilizing blockchain. As the data integrity verification method in blockchain, we also analyze the verification performance under various Merkle trees structures. To improve the verification performance while also to keep a high verification precision, we further design rational sampling strategies and calculating the optimal sample size. Finally, we demonstrate the feasibility of the proposed framework by implementing a prototype system and validating our analysis of Merkle trees and sampling strategy through extensive experiments. In the future work, we will further improve the presentation of equations, elaborate our experiments and evaluate the performance of the system with real Blockchain systems.

VI. Acknowledgement

This work is supported by the National Key Research and Development Program of China under grants 2016QY01W0202 and 2016YFB0800402, National Natural Science Foundation of China under grants 61572221, U1401258, 61433006 and 61502185, Major Projects of the National Social Science Foundation under grant 16ZDA092, Science and Technology Support Program of Hubei Province under grant 2015AAA013, Science and Technology Program of Guangdong Province under grant 2014B010111007 and Guangxi High level innovation Team in Higher Education InstitutionsÍInnovation Team of ASEAN Digital Cloud Big Data Security and Mining Technology.

References

[1] G. Ateniese, R. Burns, R. Curtmola, J. Herring, L. Kissner, Z. Peterson, and D. Song. Provable data possession at untrusted stores. In *ACM Conference on Computer and Communications Security*, pages 598–609, 2007.

[2] G. Ateniese, R. D. Pietro, L. V. Mancini, and G. Tsudik. Scalable and efficient provable data possession. In *Proceedings of the 4th International Conference on Security and Privacy in Communication Netowrks*, pages 1–10, 2008.

[3] G. Chairscott, L. Michael, P. Chairpeterson, and Larry. Proceedings of the nineteenth acm symposium on operating systems principles. 2003.

[4] B. Chen and R. Curtmola. Robust dynamic remote data checking for public clouds. In *Proceedings of the 19th ACM conference on Computer and Communications Security (CCS 2012)*, pages 1043–1045. ACM, 2012.

[5] R. Curtmola, O. Khan, R. Burns, and G. Ateniese. Mr-pdp: Multiple-replica provable data possession. In *The 28th International Conference on Distributed Computing Systems (ICDCS2008)*, pages 411–420. IEEE, 2008.

[6] Y. Deswarte, J.-J. Quisquater, and A. Saïdane. Remote integrity checking. In *Integrity and Internal Control in Information Systems VI*, pages 1–11. Springer, 2004.

[7] H. He, R. Li, X. Dong, and Z. Zhang. Secure, efficient and fine-grained data access control mechanism for p2p storage cloud. *IEEE Transactions on Cloud Computing*, 2(4):471–484, 2014.

[8] W. U. Ji-Yi, F. U. Jian-Qing, L. D. Ping, and Q. Xie. Study on the p2p cloud storage system. *Acta Electronica Sinica*, 35(5):1100–1107, 2011.

[9] A. Juels and B. S. Kaliski Jr. Pors: Proofs of retrievability for large files. In *Proceedings of the 14th ACM Conference on Computer and Communications Security*, pages 584–597. ACM, 2007.

[10] B. Liu, X. L. Yu, S. Chen, X. Xu, and L. Zhu. Blockchain based data integrity service framework for iot data. In *Proceedings of the 24th IEEE International Conference on Web Services (ICWS 2017)*, pages 468–475. IEEE, 2017.

[11] S. Nakamoto. Bitcoin: A peer-to-peer electronic cash system. https://bitcoin.org/bitcoin.pdf, 2008.

[12] F. Sebé, J. Domingo-Ferrer, A. Martinez-Balleste, Y. Deswarte, and J.-J. Quisquater. Efficient remote data possession checking in critical information infrastructures. *IEEE Transactions on Knowledge and Data Engineering*, 20(8):1034–1038, 2008.

[13] H. Shacham and B. Waters. Compact proofs of retrievability. *Journal of Cryptology*, 26(3):442–483, 2013.

[14] Y. Teing, A. Dehghantanha, K. R. Choo, and L. T. Yang. Forensic investigation of P2P cloud storage services and backbone for iot networks: Bittorrent sync as a case study. *Computers & Electrical Engineering*, 58:350–363, 2017.

[15] D. Vorick and L. Champine. Sia: simple decentralized storage. http://www.sia.tech/, 2014.

[16] C. Wang, Q. Wang, K. Ren, and W. Lou. Ensuring data storage security in cloud computing. In *17th International Workshop on Quality of Service (IWQoS 2009), Charleston, South Carolina, USA*, pages 1–9, 2009.

[17] Q. Wang, C. Wang, J. Li, K. Ren, and W. Lou. Enabling public verifiability and data dynamics for storage security in cloud computing. In *Proceedings of the 14th European Symposium on Research in Computer Security (ESORICS 2009), Saint-Malo, France. Proceedings*, pages 355–370, 2009.

[18] S. Wilkinson, T. Boshevski, J. Brandoff, and V. Buterin. Storj a peer-to-peer cloud storage network. https://storj.io/, 2014.

2019 IEEE/ACM 1st International Workshop on Software Engineering Research & Practices for the Internet of Things (SERP4IoT)

An IoT Integrity-First Communication Protocol via an Ethereum Blockchain Light Client

Elizabeth Reilly *, Matthew Maloney *, Michael Siegel * and Gregory Falco *
* Massachusetts Institute of Technology, Cambridge, Massachusetts 02139
{reillye, maloneym, msiegel, gfalco}@mit.edu

***Abstract*—Smart cities and advanced energy delivery systems are examples of IoT rich environments. These systems are responsible for communicating critical data about urban infrastructure that keeps our modern cities functioning. Today, IoT devices lack communication protocols with data integrity as a priority. Without data integrity, these systems are at risk of actuating urban environments on compromised data. Attackers can use this IoT communication flaw to wage cyber-physical attacks. We designed and developed an integrity-first communication protocol for IoT that is distributed and scalable based on the Ethereum blockchain. Our light client ensures data communication integrity for systems that require it most.**

***Index Terms*—IoT Security, Blockchain, Smart Cities, Energy Delivery Systems, Communication Integrity, Ethereum, IoT Communication Protocol, Cybersecurity**

I. Introduction

Despite the prevalence and growing popularity of smart city IoT devices such as smart meters or CCTV security cameras, many of these devices lack security [9]. There are hundreds of videos dedicated to demonstrating how to hack into IoT devices in minutes [5]. While some communication protocols exist, the variation in manufacturers makes standardizing communication for these memory and processing power-constrained devices difficult [8]. Our research focuses on a specific security gap for urban critical infrastructure IoT - improving communication integrity.

Prioritizing integrity, what we call integrity-first communication, is important for smart city IoT. Smart cities involve digitizing critical infrastructure and are often comprised of many IoT devices. While the protocol we describe is applicable to any IoT device, including energy delivery systems, it is most important in smart cities because these systems are often cyber-physical in nature. Compromising the integrity of the communication for these specific systems can be disastrous.

Because many IoT devices have different operating systems and configurations, it is hard to establish one single communication protocol that can be universally applied [6]. IoT devices with limited memory and computational resources pose an even greater concern as they often lack the resources to host a communication protocol [12]. Several IoT communication protocols have been proposed that attempt to prioritize communication integrity, yet they are not scalable [13].

Given the distributed nature of IoT devices and their scale, a suitable integral communication protocol is needed. We propose using the blockchain for this purpose. Blockchain is a distributed ledger that allows participating devices to post transactions and verify the transactions of other participants. Consensus around verified transactions form the blockchain, which is immutable and secure. Therefore, blockchain provides a scalable, distributed record that enforces consensus across all participating nodes [14]. Once a transaction has been approved by the majority of nodes, it cannot be tampered with or erased unless an attacker controls over 50% of the nodes. Possessing over 50% of the nodes, an attacker could control the direction the chain grows. The risk of the ”51% attack” is extremely low in large, public blockchains because it would be expensive and difficult to gain control of 51% of the nodes in a large network. Therefore, large blockchains could be effective at guaranteeing the integrity of IoT communications.

Historically, hosting a blockchain node on IoT has been problematic because each node must store the chain of transactions. Many IoT devices have memory and computational limits which keep them from being able to store entire chain data. While current solutions have bypassed this issue by using lighter clients or hardware solutions, each have limitations [2][1][8][15]. We designed and built an Ethereum blockchain light client that overcomes these gaps. Our light client provides integrity-first communication for IoT devices.

II. Related Works

As of this publication, there is no industry standard for how IoT devices should communicate securely. Many current communication protocols for IoT devices either poorly address security, or are not scalable. Researchers have only recently begun investigating blockchain as an IoT communication protocol. Existing IoT blockchain implementations are either too large, too centralized, too expensive or use hardware solutions.

A. Legacy Communication Protocols

Modbus was one of the original proposed communication mechanisms for IoT devices [10]. Modbus was originally designed for isolated systems so the integrity of messages was not taken into consideration. DNP3 is another early communication protocol for IoT devices built upon TCP/IP. It has a master-slave structure and likewise does not provide integrity of messages and no authentication mechanism between master and slave. Both of these systems do not ensure integrity-first communication and were designed for early industrial IoT.

B. Modern IoT Communication Protocols

IoT devices with limited resources are often referred to as constrained nodes. Currently, DTLS is the default security

978-1-7281-2253-3/19/$31.00 ©2019 IEEE
DOI 10.1109/SERP4IoT.2019.00016

protocol used for application messages between constrained nodes [2]. DTLS is built upon the UDP protocol [12]. However, DTLS lacks scalability. Similarly, the IETF has attempted to create a standard for communication protocols for constrained nodes. One such protocol is CoAP, which is also built upon UDP and includes a subset of HTTP functions that are optimized for resource scarce environments [16]. However, similar to DTLS, this protocol does not perform well under high load or congestion [16]. Also, UDP is known for being unreliable and messages are often lost [1]. Therefore, none of the security protocols are able to achieve a distributed, scalable form of integral communication.

MQTT is another communication protocol popular with constrained nodes. It is built on TCP and IP [17]. MQTT has 3 different reliability standards called Quality of Service Levels. The first level is 0, in which a message is delivered at most once and no acknowledgement of reception is required. The second is level 1, in which every messaged is delivered at least once and acknowledgement of reception is required. The third level is 2, in which a four-way handshake mechanism is used to deliver a message exactly once. Despite these different levels of reliability, MQTT lacks scalability as nodes connect to a centralized broker performs handshake procedures [13].

C. IoT Blockchains

1) Tangle: Blockchain has been proven to be highly scalable but has not been largely applied to IoT devices nor explored as a source of integrity-first communication. The most well known variation of blockchain specifically designed for IoT is the tangle, which is accompanied by the IOTA coin [15]. The tangle stores the chain in a DAG structure to reduce space and also has a light variation specifically for small devices. In order for nodes to get their transactions approved, they must approve 2 other transactions first. This makes the transactions fee-less but also means that there are no miners in the system. Without miners, the system has less incentive to grow. Furthermore, nodes can decide for themselves which transactions to approve. This can prevent certain transactions from being approved. Therefore, a coordinator that decides the order in which transactions will be approved is currently used. This is problematic because it creates a level of centralization and single point of failure within the network [3]. Also, the tangle is a relatively small blockchain variation making it vulnerable to a 51% takeover attack.

2) IoT Chain: IoT chain uses a DAG structure similar to the tangle and also uses Simplified Payment Verification (SPV) to facilitate operations on smaller devices [1]. SPV allows devices to conduct payment verification without maintaining complete blockchain information as long as block headers are preserved. IoT chain also uses practical byzantine fault tolerance (PBFT) to ensure fast consensus. PBFT is a state machine replication algorithm that is based on the consistency of message passing. The combination of the DAG structure and PBFT allow transaction times for IoT chain to be milliseconds. Despite these benefits, the primary limitation of IoT chain is that it requires a specific operating system and linking module. This makes it a hardware solution which is more difficult to integrate with older devices. Also, the relatively small size of IoT chain makes it vulnerable to a 51% attack.

3) IoTex: IoTex similarly uses PBFT and SPV to ensure fast transaction times and limited storage space [2]. The key concept for IoTex is the idea of blockchains within blockchains. Essentially, different blockchains are used for different kinds of IoT devices. This is done because different devices have different features and by separating these features into different blockchains, no one device has to store large chains. For example, one chain might record transactions and another one might have turing-complete contracts on it. There is a root blockchain that manages all the blockchains. However, this root blockchain is problematic because it introduces a layer of centralization. Like the tangle and IoT chain, the small size of IoTex makes it vulnerable to a 51% attack.

4) NeuroMesh: One blockchain that successfully provides secure communication for IoT devices, which is most similar to our design, is NeuroMesh [8]. NeuroMesh functions as a "friendly" botnet to fight against other botnets and communicates security commands to IoT devices using the Bitcoin blockchain as the communication protocol. While this technology does provide integral communication because of the size of the Bitcoin network (minimizing the 51% attack risk), it has several operational constraints. First, NeuroMesh is only able to send 80 bytes of characters at a time. This might be fine for sending security commands to IoT devices, but is insufficient to handle substantial data transfers. Second, the Bitcoin network is slow and expensive compared with Ethereum [18]. Finally, Neuromesh uses SPV which must store block headers [8][14]. This limits how small it can become.

III. Design Parameters

For our light client, we focused on allowing constrained nodes to participate in global blockchain networks without needing to host an entire node. There were many parameters we had to consider in this light design. First, we needed to select an implementation of blockchain to use. There are many different implementations of blockchain, including both Bitcoin and Ethereum [18][14]. We initially considered creating our own blockchain, however, small blockchain networks are often vulnerable to 51% attacks. Also, we wanted to choose a blockchain implementation that was fast, inexpensive and reliable. Therefore, we decided on Ethereum.

Next, we had to find a way to scale down the size of Ethereum, both in terms of code size and chain data. We chose not to store the chain data at all and to reduce the code size using compiler tricks as well as manual stripping of the code.

Finally, we had to figure out how to avoid storing the chain data while also maintaining the correct parameters of the network, such as nonce and gas price. We achieved this by first storing these parameters locally, and occasionally querying the network to ensure that locally stored data matched global data. With all of these considerations in mind, we have implemented a light client version of Ethereum that is able to send and receive data from other devices.

IV. Light Client Implementation

A. Using Ethereum as a Base

As previously mentioned, the integrity of blockchain communication comes from the size of the network. For example, when the number of participants in the blockchain is small, it is easy for a malicious actor to execute a 51% attack. While Bitcoin is the largest blockchain network, we sought a large blockchain that does not limit data transfer to 80 bytes nor has a high cost of transaction. For these reasons we chose the Ethereum blockchain [18]. Like Bitcoin, Ethereum uses proof of work as a decentralized consensus algorithm to hash blocks to the blockchain [18]. The coins exchanged in the Ethereum network are ethers rather than bitcoins [18].

B. Avoiding Storing Chain Data

A considerable challenge was avoiding the need to store the Ethereum chain data (including headers) on constrained nodes so that they still could participate in the Ethereum network. Nodes rely on chain data to determine their nonce [18]. A node's nonce is a count of its outgoing transactions [18]. Every time a node wants to make a new transaction it must include its current nonce, otherwise the transaction will be rejected. Keeping the nonce consistent is crucial for the light client. In the light client implementation, nodes keep track of their nonce locally and occasionally query other nodes in the network to gain a consensus on their correct nonce.

In addition to the nonce, nodes need to have correct gas limit and gas price values in order to ensure that their transactions are mined and stored in the chain. Gas pays for the computation of the transaction regardless of whether it is accepted or not. This transaction fee is equal to gas limit * gas price. These values have been hard coded at a gas price of 30 Gwei (equal to 0.00000003 ether) and a gas limit of 210,000 units of gas because these values are likely to get transactions added to the chain. Therefore, light nodes do not need to query for this data. With the gas and nonce set correctly, the light node can exchange transactions without storing chain data.

C. Reducing Code size

Eliminating the need to store chain data greatly reduces the size of the code that needs to be stored on the node. However, the current open source code for Ethereum is still 30 MB. Therefore, we have stripped as much code as possible from the node in order to reduce its size. Since the light nodes are not acting as miners in the system, there are many aspects of the code base that can be removed for the light client, such as any code related to mining. To date, we have stripped the code base to 5 MB while preserving the ability to send transactions and query other nodes. Although 5 MB is sufficient for many smart city devices (CCTVs), we aim to reduce this size more.

D. Architecture and Communication

As seen in Figure 1, the light client architecture interacts with full nodes and other light nodes in a distributed manner. The light nodes do not store any chain data but can both send and receive data as indicated by the two-way transaction broadcast arrows. The light nodes send data by adding that data to a transaction and then sending the transaction to another node. When the light node wishes to send a transaction, it pings all Ethereum nodes, just like any full node would. Only a full node can process the light node's data, which is indicated by the red "x" shown between light node communications. When a full node successfully mines the light node's block (indicated by a checkmark), other full nodes will no longer be able to mine the same block (indicated by an "x"). After the light node's block is successfully mined it is posted to the Ethereum network as an immutable value on the ledger. The light client must pay a small gas fee should it wish to send data, usually on the order of a few cents [11].

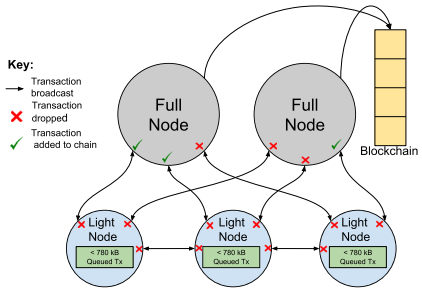


Fig. 1. Distributed Node Architecture

Light clients are able to receive data from the full nodes by pulling data from the blockchain. Although these light nodes will not store the blockchain themselves, they can query other nodes for information about the chain. Therefore, these nodes can query to see whether any recent transactions have been sent to them and then download these specific, small transactions. Therefore, the light nodes and full nodes are able to send and receive messages effectively, and the small node does not need to store the entire chain data.

If the light client has a large amount of information it wants to send all at once, it can queue transactions and aggregate the transfer. The gas limit of an Ethereum transaction is about 3,000,000 gas which allows up to 780 kB of data per transaction [4]. Therefore, at any given time we queue up to this amount.

V. Discussion

Our Ethereum light client communication protocol for smart city IoT devices provides assurance that data was not compromised in transit. The light client authenticates the origin of a data source based on the unique Ethereum address that initiated the transaction. This can be trusted because of the proof of work consensus algorithm and the size of the Ethereum chain. Many urban critical infrastructure sectors rely on integral communications. For example, electric grid infrastructure requires device state information to be transferred over networks at regular intervals. If an attacker compromised the integrity of state data, there could be cyber-physical damages.

Despite the benefits of using the light client for communications, there are limitations of this approach which restrict

its practical implementation to certain IoT use cases. First, the light client will require an operating system to run. This considerably limits the number of devices it is applicable for. Also, the block time of our light client is on average 15 seconds meaning that there is a transaction delay for data [11]. This is superior to Bitcoin's block time which is closer to 10 minutes [11]. Therefore, devices requiring real-time data transfer should not use this protocol. Another consideration is that the average cost to send an Ethereum transaction using our light client implementation is 10 cents for gas [11]. This is considerably less expensive compared to Bitcoin-based NeuroMesh which costs an average of 75 cents per transaction [7]. However, for IoT use cases that require constant transactions, this could become very costly over time. While our light client is both faster and less expensive than NeuroMesh, we acknowledge the use case limitations.

Another advantage of our IoT blockchain compared with IoTex, IoT chain and the tangle is that the light client resides on a major blockchain. Ethereum is widely used - over 10,000 transactions are sent every hour and within a 24 hour time period, around 200,000 addresses will be active [11]. The high volume of users makes it difficult for an attacker to gain control of 51% of the network, unlike other IoT blockchains.

To date, we have tested the light client on approximately 100 devices in a laboratory setting. The transactions have reliably been sent and received within the expected constraints of the Ethereum network. By releasing the light client to the research community, we hope to test the light client at scale and under failure conditions, such as when a large portion of the network crashes or a node's transaction is continually rejected.

VI. Conclusion and Future Work

Our light client aims to address the problems with communication integrity for IoT devices. Public blockchains such as Ethereum have given us the ability to disseminate data in a scalable and distributed fashion. Future work on the light client will include further reducing its size and establishing optimal conditions for its function on smart city IoT devices. We also aim to develop an agent that will interpret data from the light client and implement commands on the IoT endpoint. Building an agent for the light client will be the basis for an integrity-driven approach to performing updates for IoT devices at scale.

Acknowledgements

This material is based, in part, on work supported by the Department of Energy under Award Number DE-OE0000780. The views and opinions of the authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

References

[1] Iotchain: A blockchain security architecture for the internet of things. *In Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC 2018)*, pages 1–6, 2018.

[2] Iotex: A decentralized network for internet of things. 2018. https://iotex.io/white-paper.

[3] Our response to 'a cryptocurrency without a blockchain has been built to outperform bitcoin. 2018.

[4] *Ethereum Blockchain App Platform*, (Accessed: 2019-02-13). https://www.ethereum.org.

[5] *Shodan*, (accessed April, 2018). www.shodan.io/.

[6] Riccardo Bonetto, Nicola Bui, Vishwas Lakkundi, Alexis Olivereau, Alexandru Serbanati, and Michele Rossi. Secure communication for smart iot objects: Protocol stacks, use cases and practical examples. *2012 IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM)*, 2012.

[7] David Easley, Maureen O'Hara, and Soumya Basu. From mining to markets: The evolution of bitcoin transaction fees. (May 1, 2018). https://ssrn.com/abstract=3055380.

[8] Gregory Falco, Caleb Li, Pavel Fedorov, Carlos Caldera, Rahul Arora, and Kelly Jackson. Neuromesh: Iot security enabled by a blockchain powered botnet vaccine. *ACM Proceedings: International Conference on Omni-Layer Intelligent Systems (COINS)*, 2019.

[9] Gregory Falco, Arun Viswanathan, Carlos Caldera, and Howard Shrobe. A master attack methodology for an ai-based automated attack planner for smart cities. *IEEE Access*, pages 48360–48373, August 28, 2018.

[10] Igor Fovino, Andrea Carcano, Thibault Murel, and Alberto Trombetta. Modbus/dnp3 state-based intrusion detection system. *2010 24th IEEE International Conference on Advanced Information Networking and Applications*, 2010.

[11] Adam Gencer, Soumya Basul, Ittay Eyall, Robert van Renessel, and Emin Sirer. Decentralization in bitcoin and ethereum networks. *arXiv preprint arXiv:1801.03998*, 2018.

[12] Jiyong Han, Minkeun Ha, and Daeyoung Kim. Practical security analysis for the constrained node networks: Focusing on the dtls protocol. *2015 5th International Conference on the Internet of Things (IOT)*, 2015.

[13] Andrew Minteer. *Analytics for the Internet of Things*.

[14] Satoshi Nakamotoi. Bitcoin: A peer-to-peer electronic cash system. 2008. https://bitcoin.org/bitcoin.pdf.

[15] Popov Sergui. The tangle. 2015. https://iota.org/IOTAWhitepaper.pdf.

[16] Zhengguo Sheng, Shusen Yang, Yifan Yu, Athanasios Vasilakos, Julie McCann, and Kin Leung. A survey on the ietf protocol suite for the internet of things: standards, challenges, and opportunities. *IEEE Wireless Communications*, pages 91–98, 2013.

[17] Dinesh Thangavel, Xiaoping Ma, Alvin Valera, Hwee-Xian Tan, and Colin Tan. Performance evaluation of mqtt and coap via a common middleware. *2014 IEEE Ninth International Conference on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP)*, 2014.

[18] Gavin Wood. Ethereum (eth) – whitepaper. 2015. http://gavwood.com/paper.pdf.

Creating student's profile using blockchain technology

V. Juričić, M. Radošević and E. Fuzul
Faculty of Humanities and Social Sciences, University of Zagreb, Croatia
vedran.juricic@ffzg.hr, matea.radosevic@ffzg.hr, efuzul@gmail.com

Blockchain technology offers advantages such as integrity, anonymity, credibility and independence of the institution and time because of its decentralized nature. One of the current areas for tackling challenges and potential application of blockchain technology is education. Higher education is a complex system with various challenges that could be potentially improved by introducing this technology. This paper shows a brief overview of current systems, their current state and a direction of their development, but also proposes specific guidelines and upgrades for current systems, that would enhance them and, consequently, the area of education. It also proposes blockchain integration levels in higher education that, combined with data extraction, can expand and deepen the area of education for the benefit of institutions, employers and students.

Keywords - blockchain; data extraction; education; student's profile

I. INTRODUCTION

Blockchain is an emerging technology that is based on decentralized and anonymous architecture which confronts current technological systems and the usual network topologies. It is based on a distributed ledger, i.e. a database on a peer-to-peer protocol [11]. Blockchain technology records a series of transactions between nodes in the network, grouping data into blocks and using special cryptographic methods to create an infinite array of blocks. The technology is anonymous and protects its user's identity, unchangeable and trustworthy due to its decentralized architecture.

Network anonymity is achieved due to lack of exposure of private data in the network and by using encryption to identify and authenticate users. The network of trust marks blockchain technology as the unmistakable true source of information. Cryptographic methods involved in block creation as well as in linking blocks make each blockchain transaction stable and immutable. The consensus algorithms presented in the blockchain maintain functionality and truthfulness of the network by rewarding their users who contribute and validate transactions. Decentralization is a great advantage of blockchain technology compared to current systems, because it removes a central unit or central node in a network that maintains, monitors, and most frequently charges network usage services. Centralized systems generally control all streams of information within the network and user data, and the stability and functionality of the entire system depends on the main central node. Blockchain works on a series of equally important nodes that maintain the network and in a case of a failed node, the network is not compromised. Copies of the database, i.e. ledger, are distributed to all nodes of the network and can be accessed at any time to see the last state of the database.

Many states, institutions and organizations have launched blockchain initiatives and are developing their own technology that will potentially improve the current one [1]. Although blockchain technology is mostly connected of cryptocurrencies, it can be applied to a lot of different areas.

II. BLOCKCHAIN SYSTEMS IN EDUCATION

As a very promising technology, blockchain has found high-quality applications in the field of education. Today's systems and networks in the field of education have largely been centralized, have full control over their students' knowledge and are largely bureaucratized and nonautomated. Blockchain systems applied in the field of education offer a turn of ownership. Due to its characteristic of equality in the network nodes and the exclusion off a central node, students would have full independence of their personal data. It will allow total independence of the students towards the institution and with complete control over their data. Data stored on the blockchain is permanently recorded and encrypted using cryptographic methods to ensure integrity, immutability through hash functions, authenticity and anonymity.

The blockchain's potential in education goes much further, opening the student's opportunity of anonymity over their personal data, independence of institution, immutability of records of official documents and certificates with a complete confidence in the truthfulness and infallibility due to networks architecture. With the emphasis that the student manages all his data. Blockchain opens up a new approach to education. Using this technology, it is possible to reduce the administrative costs and the cost of the study since most of administrative work would be automatized. It also opens an opportunity for a different approach in paying tuition and gives opportunities for more customized and online studies. [22][25]

Although it is currently in its experimental beginnings, the potential of the distributed education system does not remain unnoticed. Many universities, organizations and companies are launching their own blockchain initiatives and exploring the benefits and applications in the field of education [14][18]. There are numerous publications addressing the potential applications and a large number of projects and co-operations in the field of education.

The most up to date university project is Blockcert, an open-source software developed by Media Lab at MIT university and the Learning Machine company [2][21]. The project seeks to expand existing systems and create a universal software that is applicable to all educational institutions which issue certificates. The Blockcert system allows educational institutions to issue academic certificates and give users full control of their own official documents and personal data. Such system offers the sovereignty and independence of a third party after the issuance of the certificate since all data is written on a blockchain. User data as well as all other documents are the exclusive property of the user and the user can make decisions about sharing his personal data. Without questioning absolute truthfulness of the data provided by the systems such as Blockcert, the user does not need to provide insight into a range of other private data to support the integrity of the certificate. Since every transaction on a blockchain has two anonymous parties who are participating in transaction, personal identity does not need to be revealed since it is connected with the participants public key.

Blockert is currently based on Bitcoin blockchain, using unique identifiers to store the hash of a certificate, or to be more precisely bash of certificates. This allows systems to store any kind of certificate or any kind of data to the blockchain because it is not storing document, but only its hash. Previously Blockcert delivered certificates to all attendees of a workshop by hashing document content and then hashing the group of documents which is known as the Merkle root. The institution issues a certificate through its own system which, upon issuance, records the evidence on the blockchain about the specific certificate, data about the person to whom the certificate was issued and the time it was issued. After the issuance, the user and all the third parties with whom the user wishes to share the obtained certificate can verify the certificate through the Blockcert verification system. Blockcert allows users to control and retrieve certificates through a mobile application. Blockcert creates an opportunity for institutions to deliver and safely store all official documents in an easy way without the risk of forgery or document loss, all while not exposing private user data.

Currently, only the Massachusetts Institute of Technology, the University of Nicosia and the University of Birmingham research center develop their own systems based on the Blockcert specifications. The University of Nicosia is the first university based on blockchain architecture [9]. Besides offering a degree program on cryptocurrencies and providing students with tuition fees in cryptocurrencies, they offer a free online course and issue academic certificates on the blockchain. For the students who successfully complete the course, the faculty issues a digital certificate that is recorded on the blockchain. Every certificate is separately hashed and then included in the index document. Afterwards, the index document is hashed and its hash is placed on the blockchain.

Sony Corporation and Sony Global Education have developed a certification education system using blockchain technology. The system is used to authenticate, share and store documents generated through education [10]. Woolf University is the first university completely based on blockchain technology [19].

The technical capabilities of each system differ depending on various factors such as selecting platforms, tokens, etc.

III. IMPROVEMENT PROPOSAL

The current development of blockchain technology in education is mostly focused towards issuing academic certificates to ensure integrity, trust and availability regardless of institution or time. Furthermore, in combination with a variety of available services, issuing certification can be further improved.

Systems that offer biometric identification, protect copyrights, archive documents [15][16][17], etc. surely open question just how broad and applicable area in which technology together with education comes in, and to the which extent we can apply blockchain technology and the all benefits it can bring.

In the case of issuing official records on a blockchain, current systems can expand their functionality and in combination with other technologies could enable much more features. There are three main processes involved in issuing a certificate: institution issuing a document by recording it on a blockchain, verification of a document, and sharing the document with all interested parties.

A diagram of components and their interconnections in an educational environment in Blockcert system is shown in figure 1.

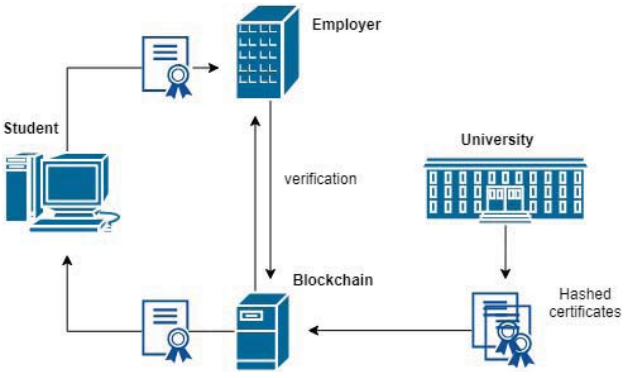


Figure 1. Components of a blockchain in an educational environment

Figure 1 shows a blockchain that is used by certain university, its students and employers. The university issues students' diplomas, uses hashed certificates and then adds them to the blockchain. Employers, who also

have access to this blockchain, can use it to search and verify students' diplomas.

But blockchain technology offers much more than just easier verification of diplomas. Technology is regularly new, but its benefits are proved in many fields, although it requires certain adjustments, not just in technical aspects but also in management and organizational aspects. Our proposal for improvements in educational institutions is based on multiple various dimensions and each is discussed in a separate section.

A. Division to courses

In most educational systems the only proof of fulfillment is the final diploma. However, during the years of studying, students acquire different key performance indicators which in sum result in a degree and the aforementioned document representing it, the diploma. In order to allow students a more modular approach to education, the diploma may be deconstructed down to a smaller collection of key performance indicators, i.e. grades and ECTS points. If the diploma is deconstructed and students have the ability to retrieve only aspects of it, it would allow a more flexible educational system.

One of the problems in the mentioned approach is the increase of issued documents which the faculty must handle. However, by introducing modern blockchain systems the problem would be solved through automation, all while keeping the integrity of certificates.

To reiterate, the certificate is an evidence of individual's learning. The first, almost logical, upgrade to current systems could be the dissection of a degree down to individual courses. The courses are constituent units of the curriculum whose successful graduation results in a diploma. By using blockchain technology or systems such as Blockcert, the process can be improved by recording and grouping certificates of all successfully passed courses instead of delivering a single diploma. Systems that function on the principle of saving hash to blockchain can generate any form of certificate because the blockchain does not record the contents of the document, but only their hash and thus is primarily used for verification.

For each course, the system would generate a new certificate that would be recorded on the blockchain and sent to the student. The user interface would consist of a series of passed certifications, i.e. courses. Certificates would be generated similar to the current system, bashed or in the index document, but with additional data that could be grouped.

B. Employer benefits

By recording the courses, it is possible to gain a deeper insight into the knowledge gained during the course of the study and create a student profile by listing all the courses and grouping them into categories. The course categories can refer to semesters, years or levels of study, and issuing a certificate for each course would open a possibility for the validation of a specific subject. Another possibility would be grouping the course as a defined set of knowledge that the student has acquired through the study and as a factor in assessing the knowledge of the particular area. With the extension of the system, employers would receive, apart from the diploma itself, information on the passed courses.

In this way, the employment process could be more improved. Adding a course to a student profile would allow the search of required courses, i.e. the required courses or category of courses for the required position, and thus the better filtering of potential employees. In addition, students who are in the process of acquiring a diploma or have not completed a degree, but have a number of passed courses, this would allow qualification on positions based on acquired knowledge, despite the lack of a diploma. It could also be a potential motivation for more active students. Such an approach would yield a more just system.

Furthermore, splitting diplomas into smaller objects can go much further and the student profile can be elaborated in more detail. Courses are comprised of conditions and activities that the student must successfully pass in order to pass the course. Courses can thus be divided by the formal obligations that the student performs through the semester. A number of smaller units such as presence, assignments, performance ratings, home works can bring a much wider picture of the quality of the acquired knowledge. All of the above-mentioned conditions in the course represent the total of knowledge gained through the study. Although these course units are already defined as a set of obligations and requirements in the course, they can be always be reevaluated, reused on other courses and further dissected.

C. Data ownership

In a system without a central authority, students are becoming rightful owners of their data. Applying blockchain technology in a field of education is a step to a more privacy-oriented system. Private data is no longer collected and contained and there is no central point where data can be exposed. All the data in the blockchain is associated with user's public key and only way it can be accessed is through a linked private key. Private key is in a exclusive ownership/property of a student and without its private data remains encrypted to all third parties. Current systems [2][9] are not recording student's personal data on a blockchain but only hash values of a certificate. Due to this one-way function it is impossible to retrieve original information. In blockchain terminology public key stands for public address which is linked to a student's identity. Once a certificate has been issued and recorded on a blockchain, the student holds all means of controlling the certificate. The university no longer has to hold any information about the student. Once a hash value is stored on a blockchain and student has received his certificate, university could crease to exist but the student will still be able to verify his acquired knowledge. The right to be forgotten is the most difficult part regarding blockchain technology since once the data has been hashed and stored on the blockchain, it cannot be removed. However, the student is the sole owner of the private key needed to unlock and access the data, therefore all private and personal data covered via the GDPR directive is truly and solely owned by the individual.

D. *Introducing smart contracts*

One of the most important features of blockchain architecture are smart contracts, which enable an extension of its functionality. Smart contract is an executable program that is triggered when system's state satisfies contract's conditions or requirements [12]. Although they are not so present in the most of current education systems, their implementation greatly shortens administrative tasks and increases level of trust in the entire education system. Along with the current proposals, by introducing smart contracts, it is possible to deliver all the previous items by fulfilling the conditions. Specifically, the diploma could be delivered to the student after the passing of all courses or by obtaining the required ETCS score. The smart contracts in education could be triggered after the fulfillment of all conditions required from the course, without any need for manual revision. As an example, students would automatically receive a certificate upon acquiring a positive grade in the final exam, attending 80% of lectures and completing 25 exercises defined by the course.

Employers would also have many benefits. Such an approach would allow them to search and verify candidates on a lower level, with precise criteria, including students who have only passed a certain course, regardless of their other knowledge or their diploma.

The question is not to what extent it is necessary to analyze studies and courses and divide them to smaller units, because technically this is entirely possible. The question is to determine an ideal level of granularity, in order to achieve the maximum effectiveness for students and potential employers.

E. *Extraction of skills from courses*

The next proposal relates to the extraction of features or skills that students acquire through the course of studying. By using machine learning algorithms and clustering techniques, it would be possible to extract knowledge and achievements [3] from a particular course and improve the student profile. Profiles would be much more individualized by combining the list of formally acquired skills predefined from the course syllabus, and the extracted skills from the course syllabus using machine learning. By combining these factors, it would be possible to gain insight into the entire student's study and the competences it has acquired. The process and result could be further extended by including additional information dissected from the final grade, such as percentage of attendance and other performance indicators defined by the course.

From a perspective of an employer, it would be possible not only to filter potential employees by courses, but to search by defining specific sets of knowledge. This would ease the process of hiring by allowing the search queries to be more flexible due to the extracted informal features generated by machine learning algorithms.

The next step in development of technology could lead in the direction of cognitive skills especially in the direction of modern cognitive abilities. The current educational system is rigid and non-adaptive to the everyday needs defined by potential employers. The current system within its course syllabus define formal skills such as mathematical competency, logical reasoning, etc., while potential employers search for a combination of skills which also include collaboration and teamwork efforts, creative thinking, public speaking, adaptiveness to new situations, etc. [4][5]

The required skills could be extracted by the aforementioned machine learning algorithms by extending the data observed in them, specifically by inserting job descriptions and intersecting them with previous results. Skills could provide an insight into a different perspective on access to education, creating a focus not only on adopted knowledge but on ways of acquiring knowledge. They could show student skills beyond the courses, and profiles would gain breadth and individuality in describing the student.

Blockchain student cognitive profiling, along with all the features of blockchains such as anonymity, integrity, independence, offers a different approach to employment and employee selection, and offer the fullest and widest approach to education.

F. *Extending student's profiles with different sources*

The traditional education system presumes that all needed competencies and skills may be acquired through the defined curriculum. However, in practice this proves wrong in many cases due to many factors, from bureaucratic problems of changing the curriculum to the inability to support all the new skills presented. As a result, students nowadays refer to multiple locations in order to achieve the most out of their professional education. Additional sources include extracurricular studies, private schools and courses and online free courses which include certification, e.g. the popular online learning platforms.

By allowing the students to be full owners of their own data, in this case certificates and diplomas, if all of the certificates are registered on the blockchain and implement the same process as mentioned in previous sections, the final product would be an extensive profile of students' skills, courses and education located and owned by the student and verifiable from the same source - the blockchain.

Benefits for the employers in such scenarios are obvious and extend the previously mentioned. A single source of truth presenting all of the competencies which a potential employee has presented, a combination of traditional course outcomes and modern soft skills, would allow the student to be fully showcased and would greatly benefit the end-employer in its searching and selection process.

IV. CONCLUSION

The blockchain technology opens a new to approach to education widening the possibilities of user protection data ensuring authenticity and security to various academic records, transcripts and certificates. With decentralized access to that kind of valuable information it becomes truly independent as its issuers and allows open secured access to it.

The number of projects and ideas is growing as well as the number of members and organizations involved and their mutual co-operation. It is difficult to predict in what direction and how far technology will be faced with everyday new implementation and adaptation proposals, but also on many challenges in terms of scalability, security, application of different consensus algorithms and different platforms. Although for the most part the system of crediting students through blockchain is in the test phase, the potential for applying this technology in the field of higher education is limitless considering value of secured data. Blockchain technology allows students more flexible education and creates a different approach to employment and employee selection.

Possibilities of blockchain technology has not yet reached it's full potential but it's decentralized nature, security and independence are becoming more valued in various industries and institutions that stress most value on the authenticity and credibility of certified data.

V. REFERENCES

[1] Grech, A. and Camilleri, A. F., "Blockchain in education." Publications Office of the European Union, Joint, Research Centre, 2017.

[2] Blockcerts: The Open Standard for Blockchain Credentials. Retrieved January, 2019 from https://www.blockcerts.org/

[3] Scheuer, O., & McLaren, B."Educational data mining". In N. Seel (Ed.), Encyclopedia of the sciences of learning. Boston, MA: Springer, 2012.

[4] Kantar, Rebecca et al. "Constructing cognitive profiles for simulation-based hiring assessments." in proceedings of the 11th International Conference on Educational Data Mining, EDM 2018.

[5] P. Williams, "Does competency-based education with blockchain signal a new mission for universities", Journal of Higher Education Policy and Management, vol. 41. pp. 104–117, January 2019.

[6] J. David Judd, "Cryptocollege: how blockchain can reimagine higher education", International Journal on Innovations in Online Education 2(2) 2018.

[7] Sharples M., Domingue J., "The blockchain and kudos: a distributed system for educational record, reputation and reward". In: Verbert K., Sharples M., Klobučar T. (eds) Adaptive and Adaptable Learning. EC-TEL 2016. Lecture Notes in Computer Science, vol 9891. Springer, Cham, 2016.

[8] Chen, G., Xu, B., Lu, M., and Chen, N.-S. "Exploring blockchain technology and its potential applications for education", Smart Learning Environments 2018 5:1.

[9] Academic certificates on the blockchain – UNIC blockchain initiative. Retrieved January, 2019 from https://digitalcurrency.unic.ac.cy/

[10] Sony Global - Sony develops system for authentication, sharing, and rights management using blockchain technology. Retrieved January, 2019 from https://www.sonyged.com/2017/08/10/news/press-blockchain/

[11] S. Nakamoto S. "Bitcoin: A peer-to-peer electronic cash system.", 2009.

[12] A. Morrison, How smart contracts automate digital business, Retrieved January, 2019 from https://usblogs.pwc.com/emerging-technology/how-smart-contracts-automate-digital-business/

[13] E. Durant, A. Trachy, "Digital diploma debuts at MIT | MIT News." Office of undergraduate education. Retrieved January, 2019 from http://news.mit.edu/2017/mit-debuts-secure-digital-diploma-using-bitcoin-blockchain-technology-1017

[14] EUBlockchain. An initiative of the European Commission, Retrieved January, 2019 from http://ec.europa.eu/citizens-initiative

[15] Civic Secure Identity Ecosystem - decentralized identity & reusable KYC." Retrieved January, 2019 from https://www.civic.com/

[16] Open identity system for the decentralized web. Retrieved January, 2019 from http://uPort.me

[17] Bernstein - Blockchain for intellectual property. Retrieved January, 2019 from https://www.bernstein.io/

[18] Malta, the first nation state to deploy blockchain in education pilots - connected learning., Retrieved January, 2019 from https://connectedlearning.edu.mt/malta-first-nation-state-to-deploy-blockchain-in-education/

[19] Woolf - Building the First Blockchain University, Retrieved January, 2019 from https://woolf.university

[20] J. Rooksby, K. Dimitrov, "Trustless education? A blockchain system for university grades", School of Computing Science University of Glasgow Scotland, presented at New Value Transactions: Understanding and Designing for Distributed Autonomous Organisations. Workshop at DIS2017, June 2017.

[21] Learning machine, Learning machine company. Retrieved January, 2019 from https://www.learningmachine.com

[22] G. Chen, B. Xu, M. Lu and N. Chen, "Exploring blockchain technology and its potential applications for education", Smart Learning Environments, 5(1), 2018.

[23] Certificates, Reputation, and the Blockchain – MIT MEDIA LAB. Retrieved January, 2019 from http://certificates.media.mit.edu/

[24] Meet TrueRec by SAP: Trusted Digital Credentials Powered by Blockchain. Retrieved January, 2019 from https://news.sap.com/meet-truerec-by-sap-trusteddigital-credentials-powered-by-blockchain/

[25] M. Turkanovic, M. Hölbl, K. Kosic, M. Hericko, A. Kamisalic, "EduCTX: A Blockchain-Based Higher Education Credit Platform.", Faculty of Electrical Engineering and Computer Science, University of Maribor, IEEE Access 6: 5112-5127, 2018.

[26] O. Patrick; Flanagan, Brendan; O. Hiroaki, "Connecting decentralized learning records: A Blockchain Based Learning Analytics Platform", in proceedings of the 8th International Conference on Learning Analytics and Knowledge (2018): 265-269, 2018.

[27] S. Kolvenbach, R. Ruland, W. Gräther, W. Prinz, "Blockchain 4 Education." In Proceedings of 16th European Conference on Computer-Supported Cooperative Work - Demos and Posters, Reports of the European Society for Socially Embedded Technologies, 2018.

Blockchain Security Architecture: A Review Technology Platform, Security Strength and Weakness

Latifa.Al-Abbasi *, Wael El-Medany †

*† College of Information Technology, University of Bahrain
*Information and eGovernment Authority, Bahrain
latifamsa@gmail.com, welmedany@uob.edu.bh

Keywords: Blockchain Security, Blockchain Architecture, Blockchain structure, Blockchain types, Blockchain attacks.

Abstract

Blockchain, is the new trending revolution innovation that leverages business process and data sharing, by eliminating the centralization or intermediates. The technology has been attracting huge attention and many organizations are looking to adopt it. The decentralization, security immutability, data sharing and other blockchain advantages that have been explored reinforce that. Decision makers need to understand all potential risks to the blockchain technology to take the right decision for their business and users. The aim of this research is to study the blockchain technology Platform architecture, the security strength and weakness of the technology, how the technology addresses the information security key areas, integrity, availability and confidentiality, to help all stakeholders and decision makers in designing and architecting their blockchain applications.

1 Introduction

The Public started knowing Blockchain when they were introduced to Bitcoins. They always thought they are the same. Bitcoin is only a one-use case or application of Blockchain technology, as Blockchain potential use cases are much more than Cryptocurrency. Recently, the technology has become one of the most trending subjects in the conferences and technical events, with a lot of new applications and use cases of discussion. In Bahrain eGovernment forum, some of these use cases were discussed among different areas. In fintech, Blockchain has more application than Bitcoin, in Sweden for Example, they use "Green Assets Wallet" to trade green investments. And because of the blockchain specifications, of transparency and anti-corruption, it has been used for Voting in West Virginia and Russia, where the voting in South Korea are still piloting. In Dubai a more comprehensive approach to Blockchain technology is used, 20+ use cases of Smart Dubai project [1].

More applications and use cases of Blockchain are flows, smart contract to manage agreements automatically with no interaction. In Internet of Things word (IoT) [2], [3], a common issue of data privacy appears, and the usage of the smart contract facilities to solve access control and data privacy in IoT services as proposed in [4], [5]. Experts are exploring the integration of Blockchain and IoT to consolidate the modern technology and fill the privacy gap of IoT.

The first and main service that needs to be applied, is Identity Management. This use case can be implemented as a foundation and enabler for any other applications that needs identity verification. If Bahrain's Government, for example, started building a trusted digital identity management using Blockchain, that will help the government to accelerate more integrated digital transactions. For example, voting can be easily integrated with digital identity. This applies also to any digital payment including cryptocurrency or other digital currency using identity authentication. As proposed in [6], it can go further by designing that in a way, that the users are a self-governing on their profile, they will be the custodian of their own data by giving permission to a third party like an insurance company to access the data or not.

This Paper contributes to provide comprehensive reference security architecture of the Blockchain to decision maker, developer or researcher to understand the Blockchain security architecture, and contribute to their guidance in designing their Blockchain architecture according to the business needs for confidentiality, integrity or availability. The paper will include a detailed explanation of the Blockchain security structure and security layers and the importance of each layer to increase the Blockchain security. It also discusses the Blockchain smart contract application contribution to enhance the security architecture of the Blockchain and IoT devices as an advanced layer.

Section 2 discusses the related work, section 3 is the problem and objective of the technology as risks and opportunities, 3.1 provides a Blockchain overview, 3.2 explains Blockchain features, section 3.3 illustrates Blockchain model were illustrated and 3.4 to explain an end to end blockchain transaction. In section 4 is the Blockchain Security Architecture, section 5 discusses the security risk to CIA trade. Finally, section 6 is the conclusion.

2 Related work

Overall potentials and innovation for Blockchain uses, especially in government sector are discussed in [1] during the Bahrain International E-Government forum 2018. [4] and

[5] proposed the Blockchain smart contract application to solve the data privacy issue associated with the IoT **devices,** the smart contracts are used to control the access on the IoT devices and thus leveraging the power of IoT by eliminating the data privacy issue. In a technical Gartner report [9] security risks have been explored among all the layers in the blockchain security model as in Figure3. By exploring all possible risks to the Blockchain starting from the business processes, the blockchain features that will support these processes and the risk associated to any business starting an innovation with the key questions that must be clarified before engagement to this technology. It then goes more in depth to the risk of other layers of the trust model of the blockchain whether public or private. The exploration then started moving from business to matching the technical aspects, the identity access management and public-key infrastructure with the associated risks to them. It also discussed the network risks associated to p2p blockchain network into the physical layer and processes. In [10], the author proposed a solution to solve the privacy issue in the blockchain by mixing the transaction and confusing any attacker trying to monitor and identify the sender or receiver. In [6], the author introduced the Decentralized Identity that increases data privacy and security, where users are the controller of their data and can share it with any organization when needed. The proposed solution can be integrated to any system for identity verification and data sharing, it can be used for bank transactions, government services, health care institutions or any other service that requires identity verification. In 2018 [8] and [7], In 2018 [8] and [7], the authors clarified the technology concepts to eliminate any confusion related to media and assumption, the report was directed to decision makers to understand Blockchain concept and its impact to their business. Blockchain platform was assessment done in [11] to select the appropriate one for the organization business model. [12] explained the Blockchain structure and its features. While the author in [13] introduced some Blockchain security powers. Blockchain risks and threaten attacks were highly demonstrated in [14].

3 Blockchain Opportunities and Risks

Although much attention is directed towards Blockchain technology by the decision makers and CEO's want to be part of this technology, there are a lot of ambiguities and unclarity in their mind about the technology. This is because Blockchain is not an enhancement of the current technology, it is externally different than what we have, and we need a different perspective to understand its capabilities. Also, many blockchain applications are not fully implemented and tested [7]. According to Gartner survey in 2018 [8] 66.6% of the CIO are attentive to Blockchain technology, while only 1% are implementing or investing into it and 22% planning adopt soon. By 2022, 10% of the organizations will leverage their business using the Blockchain and on the worst estimate, one of them will achieve a wealth of ten billion dollars. So, this is not a simple decision, it should be on a clear ground with no assumptions. A 100 of senior-level managers were asked a question about their opinion of the Blockchain risk and opportunities to their organizations as illustrated in Figure1 [7].

The decision maker wants to know when they can adopt this technology? How to start? And in which area? What are the resources needed? What is the cost? Is it worth the investment and when is the appropriate time for such investment? Will waiting leave us behind the orchestration?
This paper is to provide a comprehensive view of the Blockchain technology overview and the security powers and risks, what is the associated risks and threat to the technology? To help the decision maker, developer, and researcher to know more about this technology, what are the strengths and the weaknesses? This can help them to take the right decision at the right time to achieve the right objective according to the Blockchain technology.

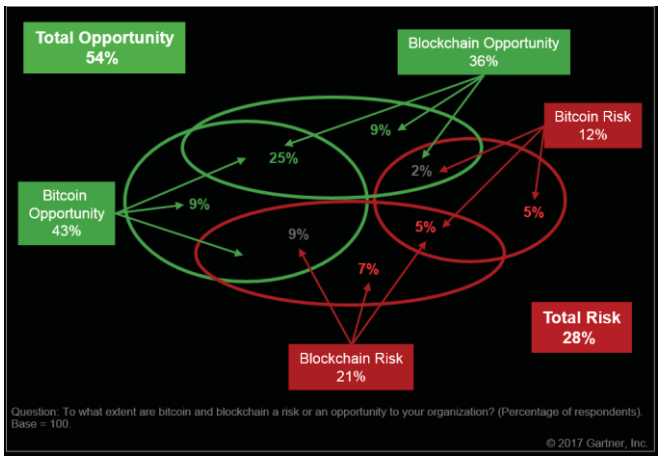


Figure 1: Board opinion of the Blockchain risk and opportunities to their organizations.

3.1 Blockchain Overview

The blocks in the blockchain technology are digital information that are linked to each other using cryptography, in a Peer-to Peer network. It is another representative of database. The blocks structures and links has been clarified in figure2, each block has a header which consists of the previous block except the first Block, a block time stamp, a 4-byte nonce that validates the next new block to be added and the block transaction hash which will be linked to the next block by adding it to its header as well. And this will form the chain that gives it the immutability power of changing or tampering. Any change will break the chain and will be detected. New blocks can be added to the chain or retrieve the exit block information with access rights privileges, but once the block is created, it cannot be changed at all. The block body contains the transaction and its counter, all transaction hashed to the Merkle Root Tree figure3. This ensures the transaction and Blockchain integrity, and the transaction is authenticated using asymmetric digital signature (Public Key).

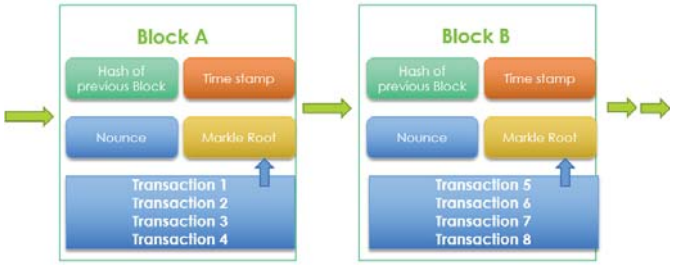


Figure 2: Blockchain Structure.

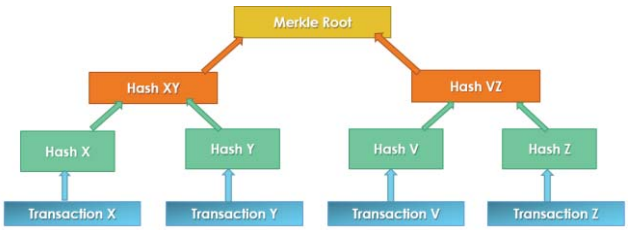


Figure 3:Merkle Root Tree

3.2 Blockchain Features

Security: One of the most fundamental features of blockchain that is no central or mediator controller of the Blockchain, and that will reduce the cost of and increase the security of the blocks. To tamper or change the blockchain, the attacker has to attack most of the nodes connected to it, otherwise it will be detected and recovered immediately.

Integrity: Each block is connected to the previous block by holding its hash value; this produces a connected chain of blocks that are traceable and immutable for any change. Any change is broadcasted to all the nodes in the block for verification. This will ensure the accuracy of the blockchain.

Disintermediation: The Blockchain eliminate the mediator, this will reduce the cost and timing for mediation and increase the data pooling and sharing.

Automated: Any Blockchain process like in smart contract for example is automated depending on predefined protocol of rules to be executed without human interaction or central controller.

Availability: The Blockchain uses Peer to Peer network, where all the nodes share their resources with each other's. This also will increase the Blockchain availability, if one of the nodes goes down, others are available. See figure 4

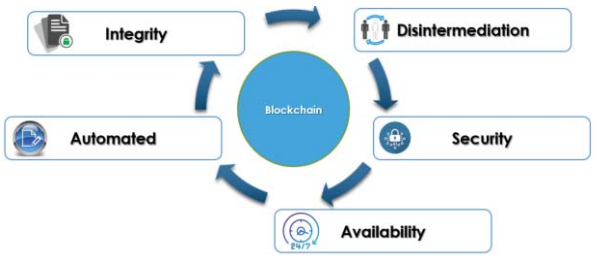


Figure 4:Blockchain Features

3.3 Blockchain Models

Gartner [11], explained the three Blockchain models and illustrated them in figure 5.

Public Model:
The Bitcoin is an example of Public Blockchain, anyone can participate, read and write to the chain, and can connect with no permission control and it totally depends on the trust control algorithm.

Consortium Model:
In this model, the membership is more closed to specific authenticated members, with permission mode. The validation of the transaction can be limited to one or few members of the Consortium blockchain, the members can grow, and new members can be authenticated. This model is used for specific industrial that blend traditional and algorithm trust controls.

Private Model:
A very strict permission and access controls to the Private Blockchain, one or multiple entity can evaluate the transaction, the evaluation is not algorithm trust control, it is more depending on business process controls.

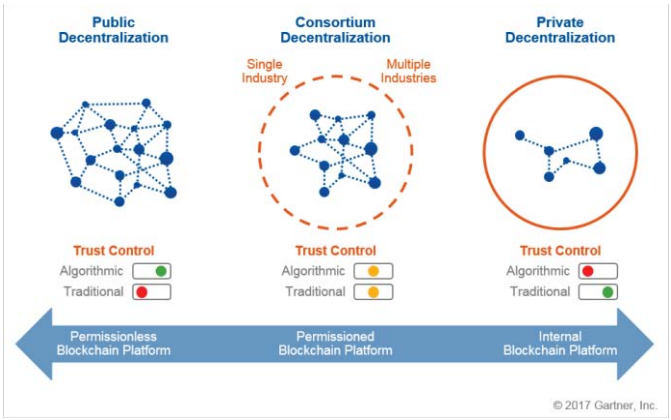


Figure 5: Blockchain Model, Decentralization and Trust Control.

3.4 Blockchain Transaction Operation

The block chain complete transaction is explained also in figure6 [11], the transaction starts by initiating the new transaction from one of the members, then the verification starts before broadcasting to all nodes, and the execution with the validation of future statues, after that the transaction will be strobed in all the nodes, the block will be added to the chain and then synchronized to all and finally confirming the new transaction.

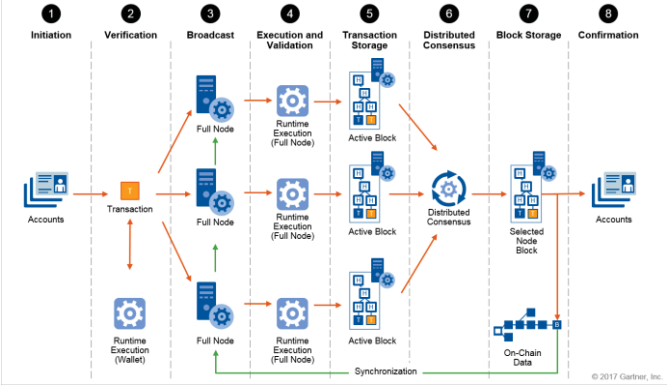


Figure 6: Blockchain Platform Simple Flow of Transaction Operations.

4 Blockchain Security Architecture

There are no formal or standard Blockchain architecture, but the simplest way of presenting it is in figure7 [12], with 5 layers. Data layer for data type, size, cryptography hashing or public key algorithm, as well as the protocols of adding new transaction. The second is the network layer for P2P sharing resources network and the node validation that validates any node before connecting it to be part of the Blockchain network. Where the third layer is Consensus layer for permissions read and write to Blockchain, consensus algorithm that verify any transaction before storing in the Blockchain. The fourth layer is optional and is used in the public Blockchain, in private blockchain the hyper ledger fabric is used. Finally, is the application layer is where the end user interacts with the technology. For more security we can add more layers, like adding a smart contact application in after the consensus layer for more access control.

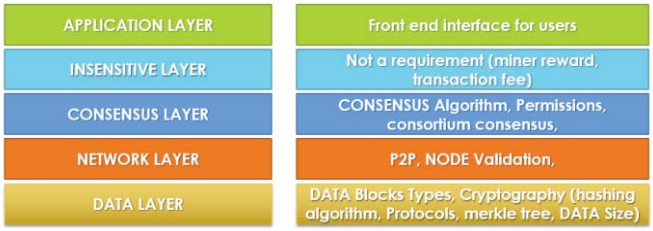


Figure 7: Blockchain security Architecture

5 Blockchain Security Risks

5.1 Blockchain Integrity

The author explained the integrity of Blockchain, the blockchain is known for its immutability and integrity of data. But there is a concern connected to the validation protocol of the data, the 51% attacks as it is called. It states that if 51% of the connected nodes agreed on data validation, it will not require the complete nodes to agree. By this protocol a majority number of malicious nodes can tamper the data. This can be enhanced by improving the protocols that control the blockchain validation according to business process.

5.2 Blockchain Availability

The Availability of Blockchain is very strong; only few incidents were captured since the bitcoin on 2009. The distribution ledger on all nodes with the updated and live copy of the whole blockchain. The Blockchain maintaining independently by each node. If any left the network, the other can operate excellently without being affected. This is totally true for private blockchain. While in public blockchain, the blockchain operators are unknown and the objective of the network is not clear, and by the validation control they can exclude and compromise some nodes that affect their network availability.

5.3 Blockchain Confidentiality

In Public Blockchain, the confidentiality is considered as a major risk, any activity is visible to public. In Bitcoin for example, anyone can see all transaction details, the date and the amount, the confidentiality is only about the user identity which can be observed by analysing other transactions details. Two solutions are proposed by the author, rotating public keys or unseeing permissioned blockchain. Another author [10] proposed mixing the transaction by participating all users of adding devises transaction to confuse the sniffer, a cycle of transaction that will end up with one real transaction but cannot be traced by the sniffer. Private Blockchain is confidentiality secured because it is a permissioned Blockchain and not opened publicly.

6 Conclusion

Blockchain is a very fresh technology, the most matured application is the Bitcoin and still it has been evaluated as highly vulnerable. Public Blockchain like the bitcoin are very risky, their integrity can be compromised if the 51% of the node decide to. With no confidentiality it is vulnerable to identity theft attack and encryption key compromise. The solution of Service and transaction mixing [10] is risky because it depends on trusting the other users to return the fake transaction, if it was payment transaction, there is a risk of stealing that money, another disadvantage is the time consuming of mixing transaction to finish in order to get to the final and real transaction, and how many numbers of transaction is sufficient. This solution is not practical and consumes time with no trust guaranteed. Where [14] proposed key rotation in order to confuse the sniffer and avoid identity theft. This solution needs to be examined more to clarify its effectiveness to solve the issue. Another solution proposed by the same author, to use permissioned Blockchain, this can be very promising solution for enterprises and private blockchain, where it cannot be inherited for public blockchain yet due to it openness to public.
Decentralized digital Identity can solve this issue for any blockchain platform [6], for private blockchain by using a

permissioned blockchain network, where for public blockchain, permission less blockchain network. But with both, it will increase the trust due to authentication and key management system; it can be used as a single point of trust for any integrated application or other blockchain platform. This solution can Increase confidentiality and decrease the potential of identity theft, check figure8. Future research about the decentralized Identity management adoption in Bahrain Smart card system (CRS) can be very useful as a foundation for other blockchain application integration, this integration can be also with non-government entities like banks or insurance company that need to retrieve this information from current CRS system or user authentication is required. The future research will examine the decentralized digital identity with the proper blockchain platform that satisfies the project requirement.

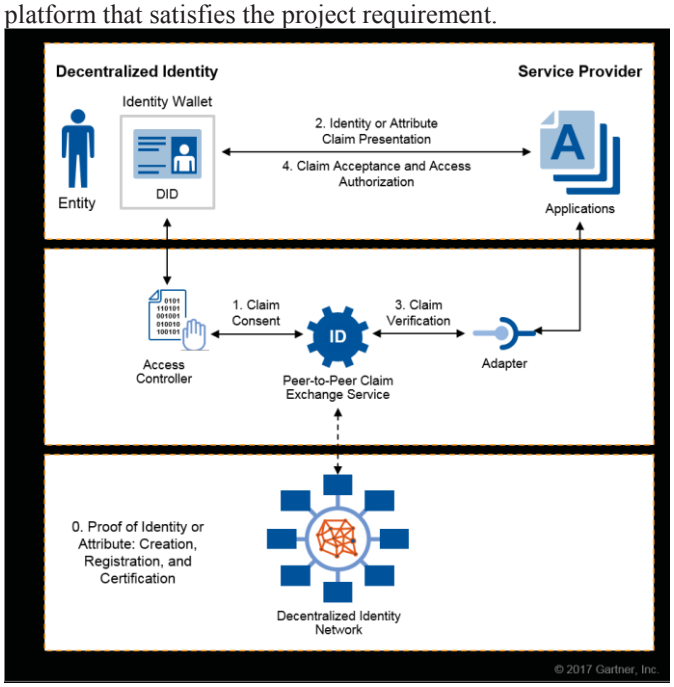


Figure 8: Decentralized Identity Service.

References

[1] Gregory G. Curtin. Blockchain for eGovernment and the fourth industrial revolution(4ir). volume 2018 of 3, pages 4{6, Ritz Carlton Bahrain, October 2018. Bahrain: Information and eGovernment Authority, Bahrain International E-Government forum.

[2] A. Ghasempour, "Optimum number of aggregators based on power consumption, cost, and network lifetime in advanced metering infrastructure architecture for Smart Grid Internet of Things," 2016 13th IEEE Annual Consumer Communications & Networking Conference (CCNC), Las Vegas, NV, 2016, pp. 295-296.

[2] A. Ghasempour and T. K. Moon, "Optimizing the Number of Collectors in Machine-to-Machine Advanced Metering Infrastructure Architecture for Internet of Things-Based Smart Grid," 2016 IEEE Green Technologies Conference (GreenTech), Kansas City, MO, 2016, pp. 51-55.

[4] Truc D. T. Nguyen, Hoang-Anh Pham, and My T. Thai. Leveraging blockchain to enhance data privacy in iot-based applications. Computational Data and Social Networks Lecture Notes in Computer Science, page 211{221, 2018.

[5] Muhammad Salek Ali, Koustabh Dolui, and Fabio Antonelli. Iot data privacy via blockchains and ipfs. Proceedings of the Seventh International Conference on the Internet of Things - IoT 17, 2017.

[6] Homan. Farahmand. Blockchain: Evolving decentralized identity design. Technical report, Gartner, 2018.

[7] David. Furlonger and Ray. Valdes. Practical blockchain: A Gartner trend insight report. Technical report, Gartner, 2017.

[8] Rajesh. Kandaswamy and David. Furlonger. Blockchain based transformation: A gartner trend insight report. Technical report, Gartner, 2018.

[9] Mark. Horvath, Jonathan. Care, and David. Mahdi. Evaluating the security risks to blockchain ecosystems. Technical report, Gartner, 2018.

[10] Daniel Genkin, Dimitrios Papadopoulos, and Charalampos Papamanthou. Privacy in decentralized cryptocurrencies. Communications of the ACM, 61(6):78{88, 2018.

[11] Homan. Farahmand. A technical primer for assessing a blockchain platform. Technical report, Gartner, 2017.

[12] Complete Guide on Blockchain Technology. (2019, March 28). Retrieved from https://www.udemy.com/certified-blockchain-expert/.

[13] Rajesh. Kandaswamy and David. Furlonger. Pay attention to these 4 types of blockchain business initiatives. Technical report, Gartner, 2018.

[14] Joerg Fritsch. Blockchain technology: How security relates to use cases. Technical report, Gartner, 2018.

Hybrid Blockchain-based Unification ID in Smart Environment

Nakhoon Choi*, Heeyoul Kim*

* Department of computer Science and engineering, Kyonggi University, Korea

skrgns0411@naver.com, heeyoul.kim@kyonggi.ac.kr

***Abstract*— Recently, with the increase of smart factories, smart cities, and the 4th industrial revolution, internal user authentication is emerging as an important issue. The existing user authentication and Access Control architecture can use the centralized system to forge access history by the service manager, which can cause problems such as evasion of responsibility and internal corruption. In addition, the user must independently manage the ID or physical authentication medium for authentication of each service, it is difficult to manage the subscribed services. This paper proposes a Hybrid blockchain-based integrated ID model to solve the above problems. The user creates authentication information based on the electronic signature of the Ethereum Account, a public blockchain, and provides authentication to a service provider composed of a Hyperledger Fabric, a private blockchain. The service provider ensures the integrity of the information by recording the Access History and authentication information in the Internal-Ledger. Through the proposed architecture, we can integrate the physical pass or application for user authentication and authorization into one Unification ID. Service providers can prevent non-Repudiation of responsibility by recording their authority and access history in ledger.**

***Keywords*— Blockchain, Identity, Ethereum, Hyperledger**

I. INTRODUCTION

According to the report of the 'smart factory industry and market trend' [1] Korea Technology Transfer Centre for National R&D Programs in 2018, the domestic smart factory demand market is expected to grow 11.2% per year by 2020 and 8% annual growth of the global smart factory demand market.

Smart Factory have the advantage that automated processes reduce product defect rates and allow for on-site operation by a small number of highly authorized personnel. However, there are problems such as control of employees' access to Smart Factory and production facilities, and responsibility for accidents caused by high authority. Currently, most organizations and enterprises use a server-based centralized architecture for Access Control and user authentication. Such a centralized architecture enables denial of access history through log manipulation and deletion by the administrator, thus avoiding responsibility for accidents. Therefore, the existing authentication and Access Control model is not suitable as a means of authentication and Access Control of Smart Factory.

In order to solve the above problems, this paper records the access history using the integrity of the blockchain, and integrates the authentication means used for each Smart Factory, or facilities, through the hybrid blockchain structure. This allows you to create a Unification ID to replace a physical pass or an application for access, record the history, and use it for future proof.

II. BACKGROUND

A. Blockchain

In this paper, we use the blockchain for the integration of user ID and decentralization of authority record, and the public blockchain using consensus algorithm represented by PoW (Proof of Work) [2]. The blockchain ensures the integrity of the information contained in the block through consensus among untrusted participants on the network and selects the block creator to maintain a decentralized network without administrators. We use Ethereum [3], a public blockchain platform, for the proposed model. Unlike Bitcoin, which supports Turing incomplete scripts that can only perform simple operations, Ethereum supports the creation and distribution of Smart Contracts [4] for developing ĐApp (Decentralized Application). Written in distributed Ledger written in Solidity, a Turing complete language, Smart Contract ensures automated execution and integrity of results.

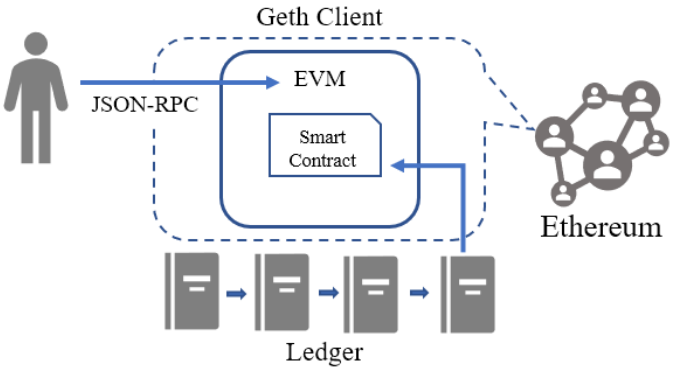


Figure 1. Operation of Ethereum Smart Contract

Smart Contract written in Solidity is compiled and stored in Ledger in the form of EVM (Ethereum Virtual Machine) Byte Code to ensure the integrity of the contract. Geth (Go-Ethereum), an Ethereum Client, calls the Byte Code of the Smart Contract and executes it in the EVM environment, ensuring automated execution and the integrity of the results according to the contract. Users operate Smart Contracts without Ethereum Clients on websites that can be accessed

`

through MetaMask [5], replacing authentication and information storage in a decentralized way.

B. Hyperledger Fabric

Hyperledger Fabric [6] is a blockchain technology development project for enterprise among the Hyperledger projects led by the Linux Foundation. It is the most popular open source-based private blockchain platform. Unlike the structure of public blockchain that anyone can participate in, it uses a certificate and PKI to verify the participant with a licensed network and is managed through the Membership Service Provider (MSP).

The Hyperledger Fabric refers to the companies and organizations that make up the network as "Organization". Organizations have their own set of peers and use peers to form a network. Blocks created in the blockchain network are delivered to each peer and stored. Peers of each Organization ensure data integrity and enable distributed storage of data through verification of each other's blocks. Also, unlike PoW of Public Blockchain(e.g. Bitcoin.), Hyperledger Fabric does not require much resources for consensus for block creation, and it collects transactions of peers through ordering service and creates them in blocks and distributes them to the network.

III. PROPOSAL MODEL

A. Proposal Model Overall

The proposed model is composed of Ethereum (Public) - Hyperledger Fabric (Private) Hybrid-Blockchain. Private layer means internal system such as each Smart Factory and Smart City.

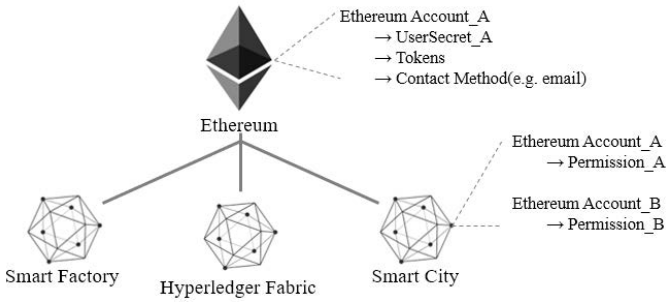


Figure 2. Proposal Model Overall

The user creates a Unification ID through Ethereum Smart Contract and sets authorization based on the Unification ID by user authentication of the organization composed of Hyperledger. Set authority information is saved in Ledger of Hyperledger, and user can prove authority to several organizations through one Ethereum Account. In addition, Token, information about authority, is stored in Ethereum ID and used as proof of history.

B. Proposal Model Layer

*1) **Unification Management Layer - Ethereum:*** Integrated management layer, Ethereum stores the information in Table 1 is based on the User Account through the Smart Contract.

TABLE 1. USER IDENTIFICATION

Identification	Meaning
UserSecret	Identifying information for initial authentication of users
Token	Identifying information of the organization issued for secondary authentication of user
Contact Method	How users receive tokens

UserSecret = Keccak-256(Name, Birth, PhoneNum) (1)

UserSecret is created through (1) and does not expose personal information through the irreversibility of Keccak256 Hash Function. In addition, Smart Contract performs access control by comparing the transaction sender account for the above information with the ID creator's account.

TABLE 2. SMART CONTRACT FUNCTION

Function	Meaning
registration	Create a User ID and save the UserSecret to Ethereum
editSecret	Modify the saved UserSecret, check the revision-transaction generator, and perform access control
addToken	Store Token for secondary authentication in Ethereum

Table 2 shows the functions that operate on the surface of Ethereum Smart Contract.

*2) **Permission Layer – Hyperledger Fabric:*** Each department of the Group, such as Smart Factory and Smart City, forms the Organization of Hyperledger, and creates a Channel consisting of the Consortium for access control and peers of each department. The node of the Ordering service is managed by the Group manager. Privileges and access history are stored in Ledger, and all participants in the network are given read permission to the Ledger. Each Group meets the requirements of Table 3.

TABLE 3. REQUIRED COMPONENTS

Components	Meaning
Internal Information System	Creates a Secret by processing the information provided by the user's group subscription according to (1)
Ethereum Sync Node	User authentication is performed by reading a Secret and a Token stored in the Ethereum smart contract by Ethereum Node.
Admin	Hold a key that can represent an organization to sign a Token for secondary authentication. It also establishes appropriate access rights for the position for authorized users.
QR Reader	Access Control to objects is performed and includes ecRecover function.

`

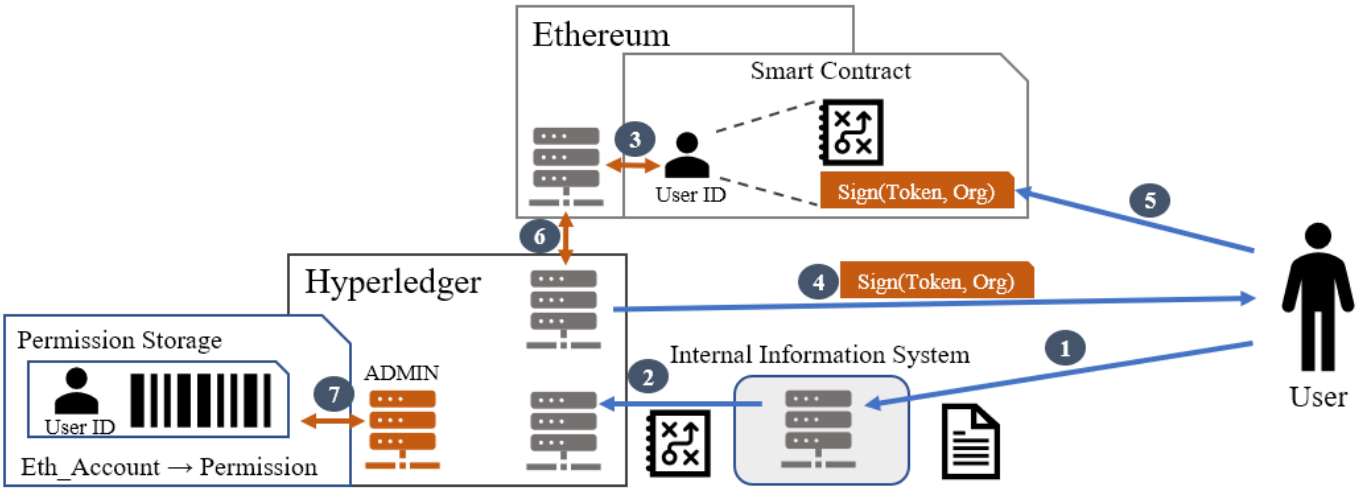


Figure 3. Permission Setting Flow

In addition, Group of the Legacy System can be included in the permission layer, and the compatibility conditions are shown in Table 4.

TABLE 4. REQUIRED COMPONENTS IN LEGACY SYSTEM

Components	Meaning
Ethereum Sync Node	User authentication is performed by reading a Secret and a Token stored in the Ethereum smart contract by Ethereum Node.
PKI	Public Key Infrastructure (PKI) that can represent organizations signing tokens for secondary authentication
QR Reader	Access Control to objects is performed and includes ecRecover function. Performs an equivalent role as the Client App of Hyperledger Fabric

C. Unification ID Create

Before using the proposed model, the user creates Unification ID based on his Ethereum Account in the Smart Contract through Web DApp and registers Secret and Contact Method(How to get your token shipped). This paper registers email in the Contact Method and receives the token of the organization.

D. Permission Setting

Figure 3 shows how a user who completes the ID Create process is granted access based on their Ethereum ID in a new organization.

- User Alice joins an organization that uses Hyperledger (e.g. Smart Factory) and provides personal information including Secret generation standards (Name, Birth, PhoneNum).
- Node linked with Smart Factory's Internal Information System generates Secret according to generation standard based on the above information and delivers it to Ethereum Sync Node of Hyperledger Fabric.
- Ethereum Sync Node checks whether the created Secret is registered in Smart Contract.
- If Alice is a registered user in Ethereum, she creates a token for each user who is digitally signed as shown in (2) as the administrator's key of Hyperledger or the representative key of the Group for the second authentication of the ID.

$$\text{Token} = \text{Sign}(\text{Keccak-256}(\text{Seed\_Alice}), \text{Org}) \quad (2)$$

⑤ Alice registers the issued token in her ID.

⑥ The Ethereum Sync Node accesses Alice's ID on the Smart Contract and checks if the issued token is registered.

⑦ Hyperledger's Admin stores permission information in Permission Storage (ACL: Access Control List) based on Alice's Ethereum Account.

E. Access Control

Figure 4 is the process in which Alice, whose Permission Setting has been completed, performs access control to the Smart Factory's Access-System.

① User Alice creates a QR through the QR Generator App registered with her private key and performs authentication for access or operation of the Smart Factory. QR includes Timestamp and Sign (Timestamp, Alice) to prevent Replay Attack.

② QR Reader reads Alice-generated QR and delivers it to Hyperledger's Access Control Node with QR_Serial_Number.

③ Access Control Node verifies Signature Data (r, s, v). It also restores Alice's public key and account through ECDSA-based public key recovery algorithm ecRecover [7].

④ Retrieve the restored account from Permission Storage and check the access right for QR_Serial_Number.

⑤ When the access right is confirmed, record the access history in the history storage and grant access.

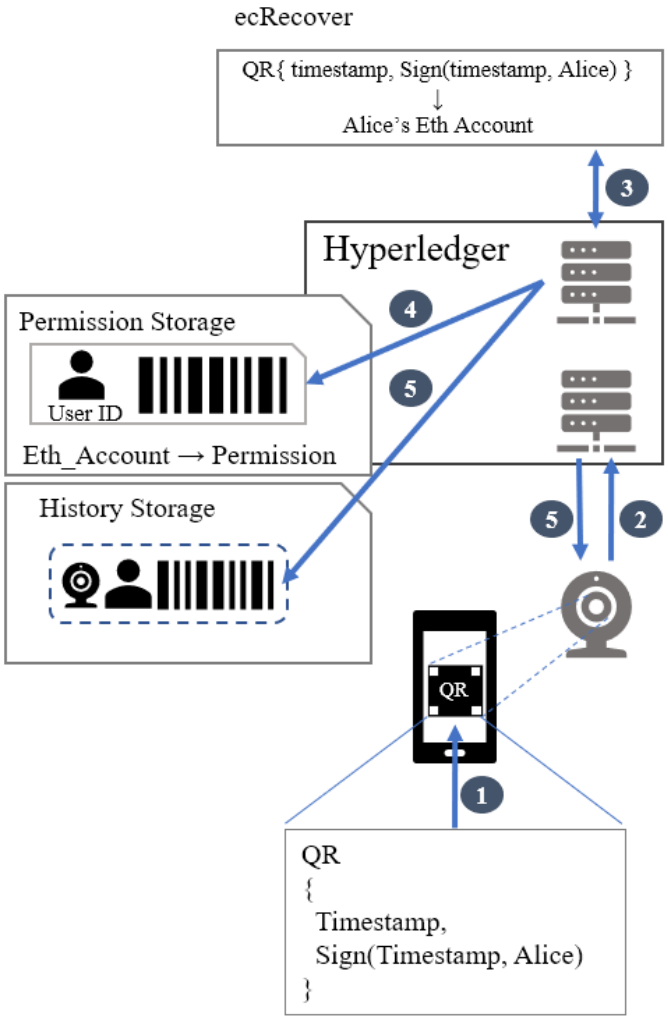


Figure 4. Access Control Flow

IV. IMPLEMENTATION

In this paper, a prototype was implemented to verify the behaviour of the proposed model.

A. Development environment

TABLE 5. DEVELOPMENT ENVIRONMENT

	Environment	Language
WebApp	Chrome, MetaMask	JavaScript
Smart Contract	Ethereum Ropsten Testnet	Solidity ^0.5.0
Chaincode	Hyperledger Fabric 1.4	Node.js
QR App	Android Emulator	Java
QR Reader	Raspberry Pi	Node.js

B. Implementation

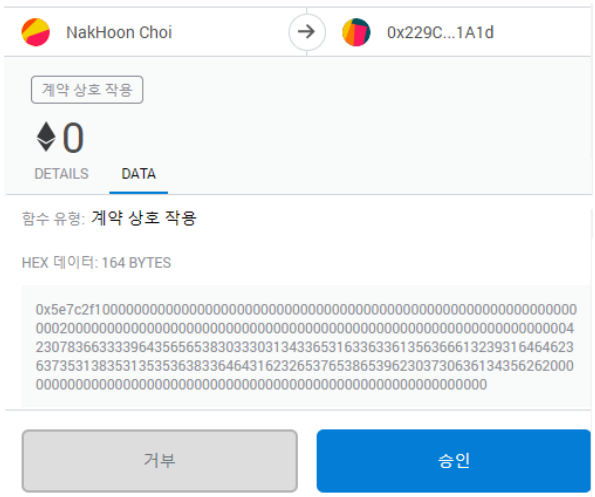


Figure 5. Create Unification ID in MetaMask

Figure 5 shows the user creating a transaction through MetaMask to register Secret and Contact Method in Web DApp.

```
=== newUser Detected ===
Name : Alice
Address : Suwon, Kyonggi
Phone : 010-0000-0000
User MetaInfo : 0x6c39d5ee8030143e1c63a5cfa291ddb675185155683dd1b2e7e8
Serving to Hyperledger...
Success.
```

Figure 6. Log of User Join an Organization

Figure 6 shows the log that Internal Information System sends user's Secret to Hyperledger Fabric when user joins group.

```
=== newUser Detected ===
User MetaInfo : 0x6c39d5ee8030143e1c63a5cfa291ddb675185155683dd1b2e7e8e9
Searching in Ethereum...
=== User Detected ===
User Address : 0xF2563715Ca207a40efb2008b35D590766d7D01e3
Contact Method : skg4463@gmail.com
Serving Token : 0xcF9F37229CcF9F374542174547af815835817af81583d581C9bE7C
```

Figure 7. Log of User Registration Comfirmation

When Secret is delivered to Hyperledger through Figure 6, Ethereum Sync Node checks if the above Secret is a registered user in Smart Contract, and sends the Token containing the organization's signature to the user's Contact Method (Email). Token is registered through addToken function of.

```
=== Token Registration Detected ===
User Address : 0xF2563715Ca207a40efb2008b35D590766d7D01e3
User Name : Alice
Token Confirm : 0xcF9F37229CcF9F374542174547af815835817af81583d581
Permission Set.
```

Figure 8. Log of Permission Setting

As shown in Figure 8, when the user registers the received token and completes the second authentication, the Ethereum Sync Node checks this and records the Ethereum Account and access rights in the Hyperledger's Permission Storage.

`

```
QR{
q0 : 0x9c304714ae532199b8cb0dd67eaf3248fe4d77b15672633855c
8c579329eedff5145c2d9f502e5d80c1a4f8422dc426f7174161c701f1
ea6b711adddb5f5ff621c,
q1 : 1570294439,
q2 : 0001
}
q0 : signature data,
q1 : UNIX timestamp,
q2 : QR Reader Serial Number
```

Figure 9. QR Generator Information

Figure 9 shows the information contained in the QR Code generated by the user through the QR Generator App. q0 is user's Signature Data (r, s, v), q1 is Timestamp used for Signature, q2 is Serial Number of QR Reader which read QR Code.

```
=== User Access Detected ===
QR Signature : 0x9c304714ae532199b8cb0dd67eaf3248fe4d77b15672633855c8c5793
QR Timestamp : 1570294439
QR Serial : 0001

ecRecovering...
Recovered User Address : 0xf2563715ca207a40efb2008b35d590766d7d01e3
0xf2563715ca207a40efb2008b35d590766d7d01e3 is 'Alice', An authorized User.
History Save.
```

Figure 10. Log of ecRecover

QR Reader sends q0-2 in Figure 9 decrypted from QR to the Access Control Node. The Access Control Node recovers the user's public key and account through ecRecover. After that, access control is performed based on Serial of QR Reader. In Figure 10, you can see that your account has been successfully recovered from QR.

V. Scalability

The proposed model of this paper is aimed at Unification ID in smart environments. From a broader perspective, it integrates the authentication methods required for various smart factories, companies and organizations in the smart city, and manages authentication and access history. Currently, the details are stored only in the Private layer, but by additionally storing them in the Public layer, all access details in the smart city can be integrated and managed.

VI. Conclusions

As smart environments such as Smart Factory and Smart City have increased, the possibility of manipulating the access history of the centralized model for access and access control has become a problem. We proposed a model using the blockchain, a decentralized platform, to ensure the integrity and transparency of information.

The proposed model integrates authentication means for access control in Smart Environment through Hybrid Blockchain. This allows users to prove their authority through a single Ethereum Account without creating a physical pass or application for authorization of multiple organizations. In addition, the access history and access to the facilities of the Smart Factory are stored in the blockchain, thus preventing non-repudiation of responsible materials in the event of an accident.

`

Acknowledgment

This research was supported by Ministry of Science ICT Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Education(2018R1C1B6002903)

References

[1] Hyeon-gyu Lee, Smart Factory Industry and Market Trends, Korea Technology Transfer Centre for National R&D Programs .2018
[2] NAKAMOTO, Satoshi, et al. Bitcoin: A peer-to-peer electronic cash system. 2008.
[3] The Ethereum Foundation. "White Paper: A Next-Generation Smart Contract and Decentralized Application Platform" website.[Online].Availalbe:https://github.com/ethereum/wiki/wiki/White-Paper
[4] SZABO, Nick. Smart contracts: building blocks for digital markets. EXTROPY: The Journal of Transhumanist Thought,(16), 1996, 18: 2.
[5] METAMASK. website.[Online].Availalbe: https://metamask.io
[6] (2019)Hyperledger-fabric read the docs website.[Online].Availalbe: https://hyperledger-fabric.readthedocs.io/en/release-1.4/
[7] Danial R. L. Brown. "SEC1: Elliptic Curve Cryptography," Standard for Efficient Cryptography, pp. 47-48, May. 2009

Nakhoon Choi is in the undergraduate course of Computer Science Department, Kyonggi University, His major research interests include security and blockchain.

Heeyoul Kim received the B.E. degree in Computer Science from KAIST, Korea, in 2000, the M.S. degree in Computer Science from KAIST in 2002, and the Ph.D. degree in computer science from KAIST in 2007. From 2007 to 2008, with the Samsung Electronics as a senior engineer. Since 2009 he has been a faculty member of Department of Computer Science at Kyonggi University. His major research interests include cryptography, security and blockchain.

2020 8th Iranian Joint Congress on Fuzzy and intelligent Systems (CFIS), September 2-4, 2020, Mashhad, Iran

A Distributed Blockchain-based Video Sharing System with Copyright, Integrity, and Immutability

Molud Esmaili[1] Reza Javidan[2]

[1] MSc. Student
Computer Engineering and Information Technology Department
Shiraz University of Technology, Shiraz, Iran
Mo.esmaili@sutech.ac.ir

[2] Associate Professor
Computer Engineering and Information Technology Department
Shiraz University of Technology, Shiraz, Iran
Javidan@sutech.ac.ir

***Abstract*— Today's video-sharing systems are platforms that have been developed to be a hub for the introduction and sharing of videos created by video creators from all over the world. These systems generate significant revenue through advertising or copyright transferring they receive from video viewers, which share only certain percentage with video creators. However, To make more profit for video creators, in this paper we proposed a distributed video sharing architecture with the possibility of performing financial transactions that works on the Ethereum. In this new architecture, the Blockchain network separated from storage; because video storage in Blockchain results to be costly for mining in the nodes in the network. Ethereum's smart contract is responsible for monitoring the validity, transparency, immutability, integrity, and security of transactions as well as controlling video access. On the other side of the architecture, there is an Ethereum-compatible distributed file storage called Swarm for video storage. Due to the lack of video storage in the Blockchain network, the integrity of the videos is questionable. To solve this problem in the proposed scheme, a Merkel tree of video hashes in the swarm is created and stored in the Blockchain to ensure the immutability and integrity of the videos and, to be transparent and valid to everyone. Finally, the performance of the proposed architecture in terms of security, structure distribution, downtime probability, fault tolerance, transparency, content integrity, immutability, and sustainability has been compared with similar systems and its performance has been proven.**

Keywords— Distributed Video Sharing; Blockchain; Distributed Storage; Smart Contract; Copyright Management; Content Integrity

I. INTRODUCTION

With the advent of the Internet, the production, release, and use of media are increasing day by day. Video is especially important as one of the most popular and, of course, the most challenging media. Video sharing systems such as YouTube and Netflix have also gained popularity[1]. However, these online streaming platforms have several important disadvantages: 1) Low profit for content video creators, 2) centrality and low privacy for consumers and 3) Low advertising effects for advertisers.

Blockchain as the third generation of the Internet is the brainchild of a person or a nickname Satoshi Nakamoto. Blockchain is a peer to peer network consists of a distributed database of blocks that are interconnected using cryptography [2]. A Blockchain network consists of many technologies including peer to peer network, distributed ledger, asymmetric cryptography, consensus mechanism and smart contract [2]. Blockchain features include data decentralization, data immutability, fault tolerance, attack resistance, transparency, and security. These features have made it possible to implement this platform in many other applications in addition to the use of these networks in the field of digital currencies[3]. Blockchain networks can be used for content sharing because of their distribution, trust, transparency and secure financial transactions. validity, transparency, immutability, integrity, and security of transactions as well as controlling video access is monitored by Blockchain.

Ethereum is a Blockchain network with a Turing-complete programming language that allows people to write smart contracts and distributed applications with rules, transaction formats, and arbitrary state transition functions[4]. In Ethereum many features and functions, especially consensus mechanisms, have been improved over bitcoin, and also transition state functions were added. A smart contract is the most important application that Ethereum developed for using in distributed applications. This is a computer protocol that is digitally created to validate, facilitate, or re-execute the contract and negotiation process[5].

Swarm is a distributed storage platform developed by Ethereum. This storage has features such as zero downtime, DDOS attack resistance, censorship resistance, error tolerance and also guarantees the sustainability of uploaded data[6].

This paper presents a distributed video sharing architecture on the Blockchain platform to solve the problem of centrality and copyright infringement. In this new scheme, the Blockchain network separated from the storage. Because video storage in Blockchain results costly mining in the nodes in the network. Ethereum's smart contract is responsible for keeping the validity, transparency, immutability, integrity, and security of transactions as well as controlling video access. On the other side of the architecture, there is an Ethereum-compatible distributed file storage called Swarm for video storage. The performance of the proposed system in terms of security, structure distribution, downtime probability,

978-1-7281-7301-6/20/$31.00 ©2020 IEEE

fault tolerance, transparency, content integrity, immutability, and sustainability has been compared with similar systems and its performance has been proven.

The key contributions of this paper are summarized as follows:

- This paper proposed a distributed video sharing system based on the Blockchain that works in a fully distributed way, unlike the current centralized architectures.
- The proposed system is implemented based on the Transparent Blockchain Network and used Ethereum Smart Contract to perform video-sharing transactions in a secure, transparent way without Interference of any third party[7].
- In the proposed architecture, video storage is separated from Blockchain for decreasing mining costs and increasing system efficiency also videos are stored in a distributed storage called Swarm.
- In the proposed scheme, is created a Merkel tree of video hashes and is stored in the Blockchain to guarantee the integrity and immutability of the videos and also be transparent and valid for everyone.
- Ethereum is used as a Blockchain network and the distributed storage is implemented by Swarm. A distributed application using Python programming language is also developed for the connection of parts of the system and the user interface.

The rest of this paper is organized as follows. Related works are discussed in Section II. The proposed new architecture is introduced in Section III. Section IV presents the implementation, evaluation, and analysis of the architecture compared to the previous approaches. Finally, the paper concludes in section V.

II. Related Works

Zhaofeng et al.[8] introduced a Blockchain-based approach to digital rights management called DRM, which is a service for the detecting right content. Their proposed architecture utilizes two Blockchain interfaces, which are responsible for maintaining basic information as well as encrypted information. In DRM architecture, separate flexible storage is used to store the original and encrypted content and then connects to the Blockchain using hashes. In the implementation part, Ethereum is used as a Blockchain network and an interplanetary file system for distributed storage. It also used encryption and watermarking algorithms to manage copyright and prevent free release.

In 2018, Zheng et al.[2] introduced a Blockchain-based privacy-scalable sharing system. This paper uses Paillier encryption to prevent the manipulation of shared data by Blockchain technology and to validate them. The proposed system enables secure transactions and its information is protected by Paillier encryption. The proposed system is tested in cloud storage platforms and laboratory environments and is showed good performance. But despite the use of Blockchain networks as decentralized networks, in this system, the shared data is stored in centralized storage. This increases the probability of DDOS attacks and the presence of a central breakpoint in the system.

Fotiou et al.[9] proposed a name-based distributed scheme to provide security in these networks due to the importance of user name, content and device security, especially in data-name networks that work with Blockchain and hierarchical-identity authentication encryption algorithm. In this system, each user has a private key generator that is used to generate the original private key and system parameters required by the HIBE algorithm. Storing private keys is one of the disadvantages and challenges of the proposed algorithm.

Liu et al.[10] introduced a new Blockchain-based network architecture using edge computing in mobile networks for video streaming services. They first designed an incentive mechanism to improve connection of video creators, coders, and viewers. They then propose a method for block size adaptation in Blockchain-based video streaming systems. Also, for their new architecture, two loading modes were designed to avoid overloading the edge nodes; the loading mode from the nearest edge nodes in the mobile networks and the loading mode from the user to the device. The researchers proved the performance of architecture and algorithms by simulation.

In 2017, Xu et al.[11] Cited problems such as lack of media quality in networks, lack of copyright, and lack of a complete and proper model for media networks, proposed a method of media copyright management. In this way, Media production, copyright, transactions, and user behavior were managed using Blockchain features. The disadvantage of this scheme is storing large media in Blockchain and having high costs for mining and storing.

In 2017, Bhowmik et al.[12] introduced transaction-based, distributed Blockchain-based multimedia network, using Blockchain Technology. In this research, with the use of watermarking algorithms on media such as image, media ownership is protected and any claim of ownership or distortion of valuable media is prevented. Watermarked information contains the transactions history and an image of hashes that preserves retrievable original media content. In this study, Ethereum was used to implement the Blockchain network, and no details of the media storage were discussed.

The common disadvantages of the articles mentioned is the use of centralized storage or the use of a Blockchain network for storing content. This will slow down the network speed and increase the cost of Mining. Also, in article [8], which isolated and distributed storage is used, the integrity and immutability are not preserved.

In this proposed architecture, in addition to the distribution and use of the Blockchain network, video storage is not stored in the Blockchain. Swarm is used for distributed storage, which is highly compatible with the Ethereum platform. Also, the integrity and immutability of the videos are achieved by the Merkel tree added to the smart contract.

III. Proposed Architecture

In this article, a video sharing system is proposed that works in a fully distributed way, unlike current systems. Distribution in this architecture is obtained using Blockchain networks, which are networks that are trustworthy, transparent, and immutable. In these networks, blocks are created by consensus mechanisms, and block validation requires a majority vote. In this architecture, nodes have the right to publish the videos as well as using others published

videos; as a result, there is no centralized client-server architecture.

In Blockchain network transactions validate by miners and store in blocks. The cost of mining is paid by the transaction owner and depends on the volume of computing and the amount of storage. Since video is a massive media, it will cost a lot to store it in the Blockchain. So, in this architecture, video storage is separated from the Blockchain and only the system transactions are stored in the Blockchain network. This will result in very high performance and a much lower cost of mining for network nodes. In other words, the Blockchain network as a system intelligence is responsible for controlling, monitoring and validating system transactions, and the Swarm acts only as a storage for storing the contents beside it.

The architecture of the proposed distributed video sharing system is shown in Figure 1.

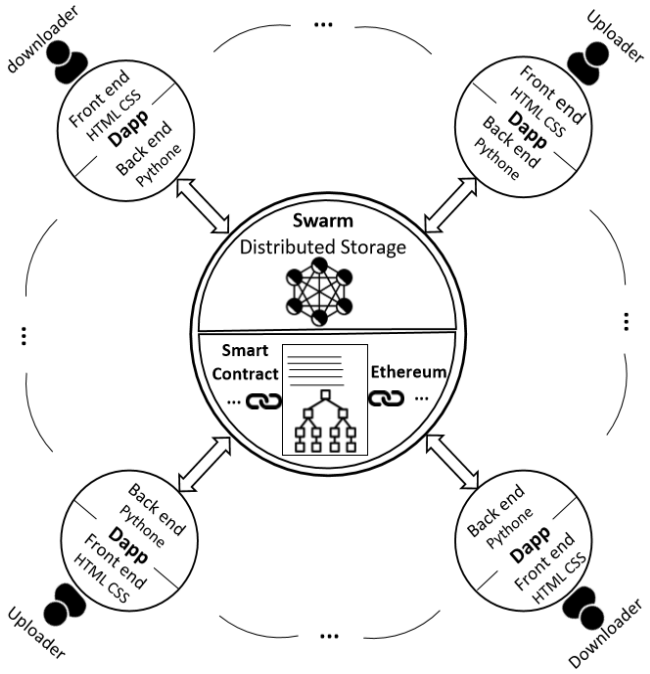


Fig. 1. Proposed architecture for distributed video sharing system

A. Blockchain in Proposed Architecture

In the new scheme, the Ethereum Blockchain network is used. Ethereum offers a very flexible platform for developing distributed applications, smart contracts, a digital currency called Ether, and improved algorithms; So It is very reasonable choice to work with a Blockchain network

In this architecture, Ethereum's smart contract is used as a system intelligence and as a supervisor for the validity of system transactions. It also monitors the proper performance of the content publisher and consumer.

B. Distributed Storage in Proposed Architecture

Since the system is introduced to eliminate centralization, a distributed storage called Swarm is also used to store the videos, developed by Ethereum to adapt and work with this network. This storage is a peer to peer network in which the content is addressed with its hash and is very similar to the IPFS file system [13]. Also, Swarm can define as a torrent-like system that guarantees data sustainability[6]. Its Features include distribution, censorship-resistant, zero downtime, DDOS resistant, Fault-tolerant, and self-sustaining. It should be noted that Swarm is still in test mode and has very limited features and functions.

In addition to joining the Blockchain network, people on this system must register and have an account in Swarm. Each account in Swarm has an address and password that are required to access this network and use its contents. On the other hand, every video is added to the system by an uploader and distributed throughout the network; then, every downloader anywhere in the world can access the content with its hash address.

C. Decentralized Application in Proposed Architecture

Distributed applications are scalable and useful applications that are more developed and used with the emergence of peer to peer architecture and Blockchain networks[14]. Features that a distributed application must have included: open-source, internal currency support, a consensus algorithm, and no central breakpoint[15].

In the proposed video sharing system, a distributed application is used to user interaction with the smart contract, the user with the storage as well as the storage and the smart contract. If the user submits a request to download or upload the video to the system, Dapp downloads the file or uploads it to the system. It also transmits system events from the smart contract to the user. Dapp also as an intermediary, transmits control messages from Swarm to smart contracts.

D. Access Control in Proposed Architecture

Access control is implemented in this architecture by using Swarm distributed storage and Ethereum smart contract monitoring. Each video owner has a list of grants that they must upload to the Swarm at the same time as uploading the video. The grant list contains a list of 64-character public keys, made using the Swarm address of each person. Only people whose public keys are on the list, are allowed to access the video.

On the other hand, People who want to download a video will send their request to the smart contract through the distributed application. The smart contract, after reviewing and validating both parties of the transaction, reduces the value(Ether) of the video from Downloader balance and adds it to the video owner's balance. After the transaction has been verified in Blockchain, the Dapp with the smart contract command updates the grant list and then the downloader user can access the video. This architecture, while protecting the copyright of the content owner, prevents illegal and unauthorized publishing in the network. It also raises the validity level of content on the Internet and prevents the publication of untrustworthy, untrue, and unworthy content. As people spend time and money on content, they use and select it more carefully and sensitively.

E. Integrity and Immutability in Proposed Architecture

Blockchain networks are immutable and integrate. This means that since all blocks are chained together, it will not be possible to change blocks once a block has been created. On the other hand, the Merkle tree is used to integrate transactions and data within a block. This way data integrity is supported across this network.

In the proposed architecture, as mentioned, due to the inability to store videos in blocks (incurring very high mining costs), they were stored in separate distributed storage. The

problem is that data integrity and data immutability can not be guaranteed in this architecture; because the videos are no longer stored in the Blockchain. To solve this problem, we added another Merkle tree, where the hash of videos in Swarm are stored. Because the tree is stored in Ethereum Blockchain, it will be transparent and unalterable for everyone. In this way, the integrity and immutability is preserved in addition to the financial transactions for the videos. The structure of the Merkle tree is shown in Figure 2.

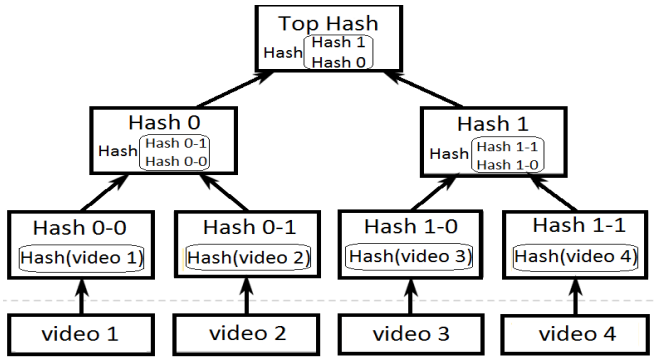


Fig. 2. Merkel tree structure to store the hash of videos

The sequence diagram for video downloading and uploading in the video sharing system presented in this paper is shown in Figure 3.

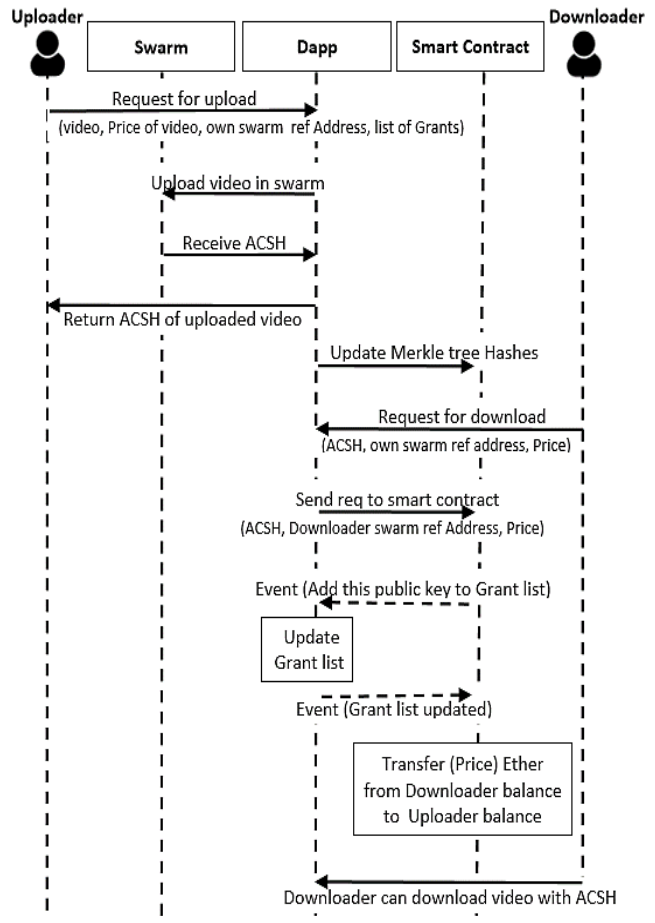


Fig. 3. Sequence diagram of video downloading and uploading

IV. Implementation and Evaluation

As shown in Figure 1, the architecture presented in this paper consists of three main parts: smart contract, storage, and distributed application. To implement the Ethereum smart contract, Solidity 0.5.1[16] is used and developed in the Remix.ide online development environment. Distributed storage is implemented using the Swarm platform 0.5.4 unstable-7deac569[17]. A distributed application is also implemented with Python 3.5.1 programming language and the web3 library was used to communicate with the smart contract[18, 19].

Implementation of the proposed system is performed using 3 nodes. uploader node, downloader node with access control, and downloader node without access control. Table 1 details the environments and tools used to implement the system.

TABLE I. Development Environments of Proposed Architecture

OS	windows X64	Hardware	GBdisk
RAM	8 GB	CPU	intel i5
Blockchain	ethereum	Development tool	python, solidity, HTML CSS, Ganache, Remix ide
External DB	swarm0.4.3 unstable-7deac569	Number of Nodes	4
Dapp (Backend)	python 3.5.1	Privacy Protection	Grant list of Public key
Dapp (Frontend)	CSS HTML	File Extension	py, sol

Algorithm 1 shows the pseudo code for the decentralized application of upload video.

Algorithm 1 upload the video

```
uploaderID = sys.argv[1] #get uploaderID (bzzaccount) of
uploader
file_name = sys.argv[2] #file to upload
grantPath = sys.argv[3] #grant_list_path
value = int(sys.argv[4]) #value of the video
connect_to_smart_contract() #connect to smart
contract(ganache) with specefic address
swarm_hash = upload_file_swarm(file_name, uploaderID,
grantPath) #upload file to swarm
add_video_smart_contract(swarm_hash,value) #add
swarm_hash of video and value of video to smart_contract
#wait for buyer to buy the video
while True:
if video_buy_event():
add_secp256_to_grantList()
update_grant_list()
```

The decentralized application output for upload a video is shown in the Figure 4.

```
grantList_version python3 upload.py 9641828f90563031af1433a393d6d22a746fc0de
file.txt ./grantList.txt 10000
swarm hash is:
ae3ee83b30174a52f7d9b0600fd23541b1706eb278c4971dbffe6a92749babaf47c6150
e9016539d1fe245ac3489f5ecdd1bd484f03faec2c33d63ac6542eb3b
Submitted with this hash:
ab7259cfc8b84831436033262e750ffc180894bbc200578e21d132e5797cdc75
```

Fig. 4. Output of decentralized application for upload a video in swarm

Algorithm 2 shows the pseudo code for the decentralized application of download video.

Algorithm 2 download the video

```
swarm_file = sys.argv[1] #swarm hash of a file to download
secp256 = sys.argv[2] #secp256 of downloader account
value = int(sys.argv[3]) #value of the video (wei unit)
connect_to_smart_contract() #connect to smart
contract(ganache) with specefic address
buyVideo smart contranct(swarm_file,secp256,value) #smart
contract public function
time.sleep(5) #wait for uploader to add secp256 of downloader
to grant_list
get_the_video_swarm() #get the video with swarm commands
```

The decentralized application output for illegal download is shown in the Figure 5.

```
~ swarm down bzz:/ab7259cfc8b84831436033262e750ffc180894bbc200578e21d132
e5797cdc75 out.txt
Fatal: download: file from address ab7259cfc8b84831436033262e750ffc180894bbc2
00578e21d132e5797cdc75: unexpected HTTP status: 500 Internal Server Error
```

Fig. 5. Output of decentralized application for illegal download a video from swarm

A. Performance Analysis

- *Distributed Structure:* Distribution and centrality removal are the main targets of the third-generation of the Internet and many new technologies, like Blockchain. The proposed architecture also eliminates centralization in content storage and replaces central servers with peer to peer, Fault-tolerant, censorship-resistant, Ethereum-compliant, and DDOS resistant storage. Whereas in articles [2,9,10,11,12] distribution is only seen on the Blockchain network and centralized memory is used to store the content.
- *Zero Downtime:* Due to not using the centralized server to store content, system Downtime probability close to zero. On the other hand, Swarm ensures that due to the redundancy and distribution of content in this peer to peer network; the content will be accessible even if the owner node is not online[19]. Whereas in articles [2,9,10,11,12] the system will fail if content servers crash.
- *Fault tolerance:* As noted above, duplication and dissemination of content between nodes and elimination of coding to store content increase fault tolerance and thus access to the system. As a result, the proposed system will present greater Fault resistance and availability than the systems proposed in[2,9,10,11,12].
- *Content integrity and immutability:* With the Merkel tree of video hashes added to the smart contract, altering the videos will not be possible. Because in the Merkel tree, altering each node will affect the root of the tree. This makes the content generated by each node in the network no longer modified or republished by any node. The content owners can put their signature or logo on the videos and use the content immutability to protect their ownership. Also, since all the hashes in the Merkel tree are linked together and displayed with a unique hash (the root of the Merkel tree), the whole content stored in the system is connected and integrated.
- *Content sustainability:* The incentives created to disseminate content to others within a peer to peer network will ensure consistency in content. That is, each node will host the content of other nodes and, in turn, they distribute its content across the network[19]. In [8], IPFS storage is used to store distributed content. Swarm is more motivated than IPFS in distributed storage for nodes, because of its deep integration with Ethereum, which enhances data sustainability and availability[20].
- *Transparency and trust:* The proposed system establishes transparency in the financial transactions in the system due to the use of the Ethereum smart contract. Video trading deals are done in the Ethereum smart contract, and its code and transactions are stored in the Ethereum. These resources are transparent and visible to all nodes of the network and, despite the absence of a central intermediary, establishes trust in the network.

B. Security Analysis

The proposed new architecture rejects today's centralized content sharing structure and offers a fully distributed structure in the monitoring system (Ethereum Blockchain network) as well as in the storage (Swarm storage). This change eliminates the problem of a single point of failure. Because there is no longer any centralized point in the system that attacking or modifying that threat the performance and security of the system.

On the other hand, the lack of a central breakpoint and server makes it impossible for DDOS attacks in the system. This is because there is no centralization in the system that traffic increasing on which will compromise the performance of the system.

The lack of a central server in this architecture also makes content censorship and distortion impossible in the system and users can freely publish their content and opinions on the system. Swarm storage is more resistant to censorship than the IPFS file system[8,20].

The video sharing system presented in this paper is compared with models in papers [2,8,9,10,11,12] and is summarized in Table II.

TABLE II. COMPARISON OF PERFORMANCE BETWEEN THE PROPOSED SYSTEM AND SIMILAR SYSTEMS

	Proposed scheme	[8]	[2]	[9]	[10]	[11]	[12]
distributed Structure	✓	✓	✓	✓	✓	✓	✓
Separated Storage	✓	✓	✓	✗	✓	✗	✗
Distributed Storage	✓	✓	✗	✓	✗	✓	✓
Content Integrity	✓	✗	✗	✓	✓	✓	✓
Transparency	✓	✓	✓	✓	✓	✓	✓
Security	✓	✓	✓	✓	✓	✓	✓
Low Downtime	✓	✗	✗	✗	✗	✗	✗
Content Sustainability	✓	✗	✗	✗	✗	✗	✗

C. Scalability Analysis

Research on the IPFS file system has shown that download delay increases with increasing nodes in this network. This is due to the increasing number of duplicate blocks with increasing nodes in this network. This problem also exists in Bittorrent and swarm systems. But it improved in IPFS compared to Bittorrent. This problem has also been improved in swarm over IPFS[21].

D. Time complexity Analysis

The time complexity of the system depends on the algorithm of adding and validating video in the Merkel tree. The time complexity of adding and validating a video hash both are equal to O(log(n)).

Experiments on the proposed system showed that as the video size increased, the download time increased. This increase is seen after 128 KB, Because the videos are stored in 128 KB chunks. As the number of chunks of 128 KB increases, the download time will increase by approximately the same. While the second time download is fast and in a matter of seconds and this time remains constant with increasing video size. The results are shown in Figure 6.

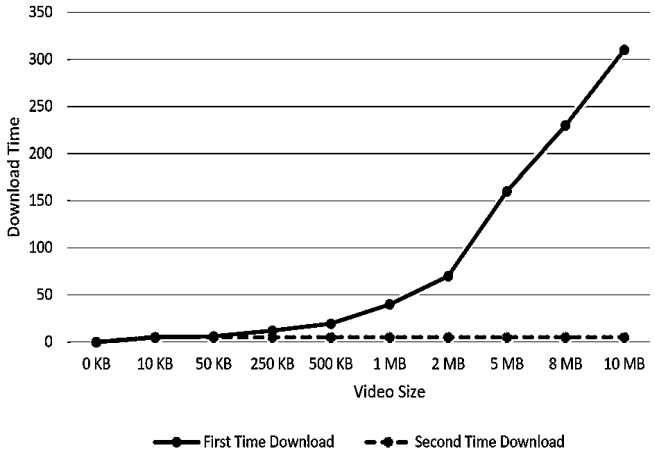


Fig. 6. The chart of download time vs video size

V. CONCLUSION

In this paper, a distributed video sharing system is introduced and implemented that works on a Blockchain network. In this architecture, to reduce mining costs and save time, video storage was transferred from the Blockchain to a distributed storage compatible with the Ethereum network called Swarm. Also to maintain the integrity and immutability of the videos in the new storage, the hash of videos was stored in a Merkel tree and kept in the smart contract.

Evaluation and comparison of the proposed system with similar systems prooved that new architecture in terms of structure distribution, zero downtime, fault tolerance, content integrity and immutability, content sustainability, transparency, trust, and also Security is at a high level.

REFERENCES

[1] Sh. Alam Chowdhury, D. Makaroff, "Characterizing Videos and Users in YouTube: A Survey", in *2012 Seventh International Conference on Broadband, Wireless Computing, Communication and Applications,* Victoria, BC, Canada, 2012.

[2] B. Zheng, L. Zhu, M. Shen, F. Gao, Ch. Zhang, Y. Li, J. Yang, "Scalable and Privacy-Preserving Data Sharing Based on Blockchain", *Journal of Computer Science and Technologies,* vol. 33, no. 3, pp. 557-567, 2018.

[3] I. Lin, T. Liao, "A Survey of Blockchain Security Issues and", *International Journal of Network Security, ,* vol. 19, no. 5, pp. 653-659, 2017.

[4] V. Buterin. "A next-generation smart contract and decentralized application platform," GitHub, 2014. [Online]. Available: https://github.com/ethereum/wiki/wiki/White-Paper (Accessed on 22 June 2019)

[5] S. Rouhani, R. Deters , "Security, Performance, and Applications of Smart Contracts: A Systematic Survey", *IEEE Access*, vol. 7, pp. 50759 – 50779, 2019.

[6] K. Ozyilmaz, A. Yurdakul, "Designing a Blockchain-Based IOT With ethereum, Swarm and LoRa", *IEEE Consumer Electronics Magazine,* vol. 8, no. 2, pp. 28-34, 2019.

[7] B. Kumar Mohanta ; S. S Panda ; D. Jena, "An Overview of Smart Contract and Use Cases in Blockchain Technology", in *2018 9th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*, Bangalore, India, 2018.

[8] Zh. Ma, M. Jiang, H. Gao, Zh. Wang, "Blockchain for Digital Right Management," *Future Generation of Computer Systems,* vol. 89, pp. 746-764, 2018.

[9] N. Fotiou, G. C. Polyzos, "Decentralized Name-based Security for Content Distribution Using Blockchains", in *2016 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, San Francisco, CA, USA, 2016.

[10] M. Liu, F. Yu, Y. Teng, V. Leung, M. Song , "Distributed Resource Allocation in Blockchain-based Video Streaming Systems with Mobile Edge Computing", *Transactions on Wireless Communications,* vol. 18, no. 1, pp. 695 - 708, 2018.

[11] R. Xu ,L. Zhang ,H. Zhao ,Y. Peng , "Design of Network Media's Digital Rights Management Scheme Based on Blockchain Technology", in *2017 IEEE 13th International Symposium on Autonomous Decentralized Systems (ISADS)*, Bangkok, Thailand, 2017.

[12] D. Bhowmik ,T. Feng , "The Multimedia Blockchain: A Distributed and Tamper-Proof Media Transaction Framework," in *2017 22nd International Conference on Digital Signal Processing (DSP)*, London, UK, 2017.

[13] K. Ozyilmaz, M. Do ˘gan, Yurdakul, "IDMoB: IoT Data Marketplace on Blockchain", *2018 Crypto Valley Conference on Blockchain Technology*, Zug, Switzerland, 2018.

[14] S. Ravel, "Decentralized applications", O'Reilly, CA, 2016.

[15] W. Cai, Z. W, J. B. Ernst, Z. hong, C. Feng, V. Leung, "Decentralized Applications: The Blockchain-Empowered Software System", *IEEE Access*, vol.6, pp. 53019–53033, 2018.

[16] C. Dannen, "Introducing Ethereum and Solidity: Foundations of Cryptocurrency and Blockchain Programming for Beginners", Apress, Berkeley, CA, 2017.

[17] “Swarm for DApp-Developers”, 2019. [Online]. Available: https://swarmguide.readthedocs.io/en/latest/dapp_developer/index.html (accessed on 30 July 2019).

[18] K .Iyer, Ch. Dannen, “Building Games with Ethereum Smart Contracts”, Apress, Berkeley, CA, 2018.

[19] “Web.py”, 2018. [Online]. Available: https://web3py.readthedocs.io/en/stable/(Accessed on 5 August 2019)

[20] “IPFS & Swarm”, 2019. [Online]. Available: https://github.com/ethersphere/swarm/wiki/IPFS-&-SWARM(Accessed on 2 october 2019)

[21] “IPFS performance”, 2018. [Online]. Available: https://github.com/ipfs/go-ipfs/issues/5226. (Accessed on 1 november 2019).

2019 International Conference on Vision Towards Emerging Trends in Communication and Networking (ViTECoN)

Immediate Detection of Data Corruption by Integrating Blockchain in Cloud Computing

E. Angelin Kanimozhi
Department of Computer Science and Engineering
Thiagarajar College of Engineering (TCE)
Madurai, India
angelinkanimozhi@gmail.com

M. Suguna
Department of Computer Science and Engineering
Thiagarajar College of Engineering (TCE)
Madurai, India
mscse@tce.edu

Dr. S. Mercy Shalini
Department of Computer Science and Engineering
Thiagarajar College of Engineering (TCE)
Madurai, India
hodcse@tce.edu

***Abstract* – Data has become an invaluable asset in recent times. Although, so many technologies for storing and processing these data, are in existence, cloud computing is known as the best in most of the parameters. Cloud computing allows its users to store and process a huge amount of data that are in the remote servers, thus reducing the burden on user side. Since, the users have to hand over the control over their data to an unknown authority, this method gives birth to a new challenge of data protection in terms of confidentiality, integrity and availability. There have been numerous researches on data integrity protection, so far, using cryptographic tools and data replication strategies. Yet, there is always a need to trust on a third party auditor for performing data integrity verifications. This has led to the threat of being cheated on, if the cloud authority colludes with the third party verifier. Also, the verification strategies that are already proposed and executed failed to identify the data corruption at the time of occurrence itself. To overcome this problem, we propose a data integrity verification scheme that uses blockchain technology for storing the data in the cloud. The blockchain properties like immutability and decentralization are integrated to identify the tampering of data immediately. Hence, the third party auditor is removed by enabling the minors to take control over the blocks in the blockchain. This paper shows the design and working model of the integration of blockchain in cloud storage.**

***Keywords* - Blockchain, data integrity, immutability, data corruption, decentralization, hash value, server, clients**

I. INTRODUCTION

The fundamental properties of cloud computing in data protection are confidentiality, integrity and availability which are together called as CIA properties [1]. These properties can be compromised in the act of cyber-attacks, by the attackers.

Confidentiality is the property of the data in which the particular data must not be disclosed to an unauthorized user. This is a passive attack where the data are not affected, but the trust on the storage management authority.Hence, this attack often remains undetected and will cause no damage to the data. Data availability is the state of the data which ensures that the data is available to the users whoever is demanding, at anytime and anywhere. This property may be compromised temporarily like in the case of denial of service attack. But by using replication methods, this problem can be solved to some extent. Data integrity is a property which should be handled more carefully than all other challenges because this can damage the data and also remain undetected. If a certain set of data is deleted or modified from the entire set, the results inferred from that data will be largely affected and creates a big impact on the business decisions taken by the module.

Focusing on data integrity verification, we find many protocols like provable data possession (PDP) and proof of retrievability (POR). By provable data possession protocol, the user can verify whether his/her data is safe in the hands of the cloud server [2,3]. Usually, the user uses a third party auditor (TPA) for challenging some blocks of data in the server by performing some hash algorithm on them. The generated tags are finally compared and verified by the TPA and the report is given to the requested user. If the data is found to be corrupted, the users can claim for the incentive [4] to the server. But there is a chance that, the server can compromise the algorithm or can collude with the TPA to cheat the user. The POR protocol [5] deals with the ways to recover the data from a loss in the server. This ensures the user that even after a disaster or any other error happens with the server, which could cause a data loss, the cloud has a solution or an algorithm to recover the users' data. Like in previous protocol, here also we have a chance that, the server might collude with the TPA to cheat the user.

In the proposed method, we apply blockchain technology for storing the data in the cloud. Since we have two great properties like immutability and decentralization, it is helpful to maintain the data integrity of the data stored in the cloud. In blockchain, due to decentralization, there is no central authority. By the property of immutability, if one block is modified, it affects the whole chain. While using PDP protocols, the integrity verification is done if and only if requested by the user. But, by using blockchain data corruption is immediately reflected in the chain and the data user will be notified. Since, we have local copies of the blockchain in every node (the client's system) the data recovery can also be done. The proposed model deals with the detection of the data corruption at the time of occurrence itself and notifies the data user about the corruption.

Previously, the blockchain structure is used in Bitcoin transactions for avoiding the third party management to keep records of all transactions and to avoid conflicts. This paper first explains how blockchain is used in Bitcoin transactions and how can it be used in cloud data storage.

978-1-5386-9353-7/19/$31.00 ©2019 IEEE

This paper is organized as follows. Section II speaks about the researches done so far on data integrity verification strategies and the blockchain implementations. Section III explains the terms involved in the blockchain working model. Section IV deals with the ways in which the blockchain is implemented in Bitcoin transactions to record and maintain them without conflicts. Section V defines the exact design of the proposed model in which the blockchain is integrated into the cloud storage. Section VI compares and contrasts the experimental results of the proposed model with the existing model. Section VII concludes the proposed ideas and the future work that can be added to the model for improvement.

II. PRELIMINARIES

Blockchain allows the users to transfer assets and record their transaction details [6]. Blockchain technology runs around some common terms. Any node or the user can initiate a transaction. After initiating, the data are recorded and broadcasted to the network [7,8]. This data is received by other nodes in the network and they verify the data by running a proof of work algorithm or a proof of stake algorithm for validation, before storing into the block. All the nodes have to come to an agreement before adding any new block to the chain. This is achieved by using the consensus algorithm.

A. Byzantine fault tolerance problem and the consensus algorithm

Byzantine fault tolerance problem occurs where we apply decentralization and we have a number of nodes [9,10]. Consider an army that is waiting for the order from the commander. The messages can be "attack" or "retreat". If the messengers in between are traitors and passes conflicting messages to the lieutenants or the commander himself a traitor and passes the conflicting messages, then the army might lose the war. This problem can occur in blockchain transactions, since we are working with decentralized nodes. To solve this problem, an algorithm called a consensus algorithm is developed, where the nodes have to come to an agreement before committing any new block into the chain.

B. Proof of work

At the time of a block creation or a new transaction to be committed, then miners (the calculating nodes) calculate the hash values for the blocks simultaneously [11]. The miner completing the calculation first will be the winner and the block is set to be committed.

C. Proof of stake

Proof of stake is slightly different from proof of work. In proof of stake, the validator of a block will be selected on the basis of the capacity or the amount of assets the miner holds [12,13]. This validator then validates a block before adding into the chain.

III. BLOCKCHAIN IN BITCOIN TRANSACTIONS

Blockchain technology is initially used for bitcoin transactions[14]. Let us consider an example, where money transactions are recorded and managed by an third party auditor (TPA).

There are three users, namely, A, B and C. A has Rs.100 in hand. A requests the TPA sends Rs. 100 to B. This is transaction T_1. Third party auditor verifies if there is enough amount with A to perform the requested transaction. The TPA approves the transaction requested by A. The user A again requests a transfer of Rs. 100 to user C. The TPA just checks the available amount with user A regardless of the previous request. The user A performs the two granted transactions simultaneously. Rs. 100 will be debited from the user A and credited to user B and user C. This creates conflicts in transaction records and in this way A can cheat the TPA and the other participants in the transaction.

This problem can be efficiently removed in a blockchain based management of bitcoin transactions. Every transaction is recorded in the blockchain at the time of it's commitment. Let us take the same example. User A sends 100 bitcoins to user B. This transaction T_1 is recorded in the blockchain as following:

T_1 : A-100 ; T_1 : B+100 ;

When user A tries to send another 100 bitcoins to user B in transaction T_2, the system finds out that there is no enough money with user A and the transaction T_2 initiated by A, gets aborted. In this way, TPA is replaced by a blockchain structure which records and manages the bitcoin transactions among the nodes (users) in the network. The data about the transactions are recorded in the blocks in the blockchain, with their respective timestamps. For each block, a hash is generated by subjecting the data inside the block into hashing algorithm like MD5 or SHA. The hash generated by one block will be sent to the next block. For example, if user A has 300 bitcoins initially, and he sends 100 bitcoins to user B and 100 bitcoins to user C , then the block B_1 consists of the following information.

T_1: A-100 ; T1: B+100 ;

T_2:A-100; T2: C+100;

$HV_0 = 0$;

Here, HV_0 represents the hash value of the previous block. Since, the block B_1 is the genesis block, the previous block's hash value is initially assigned as 0. The data in the block including the HV_0 are subjected into an hash algorithm. The generated hash value HV_1 is written in the next block B_2. Let us assume that the user C wants to send 200 bitcoin s to user A, then the data in next block B2 are,

T_3: C-200; T_3: B+200;

HV_1 = E32;

For this block, a hash is generated and sent to next block. In this way, each and every block is connected in the chain [15]. If an unauthorized user tries to make any changes in the data about the transactions, written in the block, the chain will be affected completely, i.e. modification in any block will affect its previous block which in turn affects the whole chain.

This happens because of the immutability property of the blockchain. Decentralization property of the blockchain allows it to be replicated by all of it's nodes. The nodes are nothing but the users in the networks that can participate in the transactions. The

nodes have the local copies of the blockchain. Every updation in the blockchain will be recorded in the local copies of the nodes also. If a block is modified and the act is identified, then the data can be rolled back to the original form by using the data stored in the other copies of the nodes. In this way, data recovery can be achieved.

IV. BLOCKCHAIN IN CLOUD COMPUTING

In this paper, we propose a model, where the users can store their immutable data in the cloud and can preserve it without any data corruption..

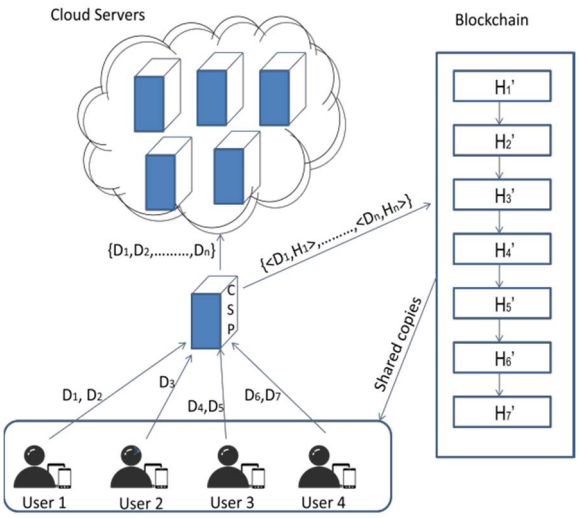


Fig. 1: Blockchain in cloud computing

Fig. 1 shows the architecture of the integration of blockchain into data storage in cloud . A cloud service provider (CSP) is responsible for the distribution of the users' data among the virtual servers. D_1, D_2, ..., Dn represents the data id of the data sent by the users and H_1, H_2, ..., Hn are their respective hash values generated by the server. H_1', H_2',....., H_n' are the new hash values that are calculated by the blockchain by linking with the previous blocks while adding every new block in the chain.

The users send their data (encrypted or non- encrypted) to the CSP. The CSP sends the data to any of it's virtual servers. The server generates hash value for the given data and handovers to the CSP. The client ID (user who has uploaded the data), the data ID and the hash of the particular data are sent to the blockchain.

This blockchain is shared with all the clients using that cloud. The blockchain on adding a new block, it creates a new hash H' in link with the previous block. After every updation, the blockchain is written in all the local copies of the clients. In the above example, D_1 and D_2 are uploaded by the user 1 ; D_3 is uploaded by the user 2; D_4 and D_5 are uploaded by the user 3; and D_6 and D_7 are uploaded by user 4.

V. DETECTION OF DATA CORRUPTION BY BLOCKCHAIN

As mentioned in the previous sections, blockchain offers immediate detection of the data corruption. When anyone tries to modify the data in the server, the blockchain function is invoked immediately. The blockchain gets the data ID that is suspected to be corrupted and it insists the server to generate a hash value for the particular data again.

This new hash value is compared against the old hash value that is stored in the block. If these values don't match with each other, then the notification is sent to the particular client about corruption. This process is explained in Fig. 2.

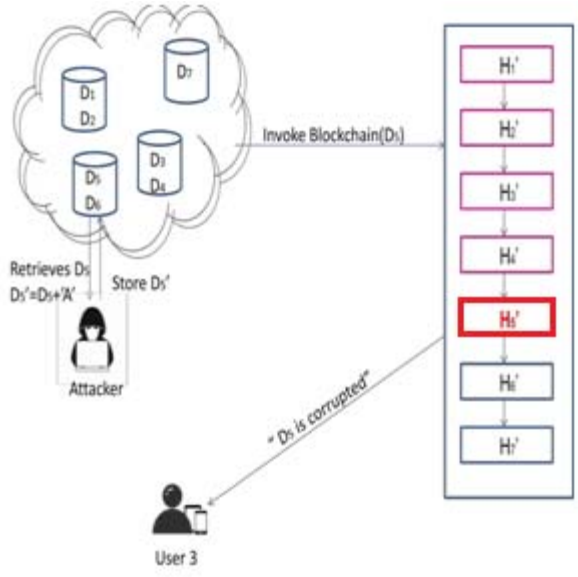


Fig. 2: Data corruption

In the given example the CSP tries to tamper the data D5, hence the blockchain function is invoked. The hash comparison happens and the message about the corruption is sent to the respective client.

VI. RESULTS AND ANALYSIS

The data integrity protocols so far proposed have detected the data corruption if and only if the user demands for verification.

TABLE 1: COMPARISON OF OVERHEADS

Verification Method	Computational Cost		Communication overhead
	Client	*Server*	
Atneise et. Al [2]	O(t)	O(t)	O(1)
Erway et. al [3]	O(t log n)	O(t log n)	O(t log n)
Blockchain based integrity verification	O(1)	O(log n)	O(1)

The cost and the overhead caused by the existing PDP (Provable Data Possession) algorithms and the proposed model are shown in Table 1, where n is the total number of data blocks of a file, t is the number of challenged data blocks in an auditing query, s is the number of sectors in each data block and ρ is the probability of block/sector corruption, where n denotes the size of the input.

The time taken for the verification in the existing model and the proposed blockchain based data integrity verification model

are compared and tabulated in Table 2 along with the other required parameters.

By the comparison made in the table below, the proposed scheme can be termed as more efficient than the existing provable data possession protocols.

TABLE 2: COMPARISON OF VERIFICATION TIME AND OTHER PARAMETERS

Verification Method	Probability of Detection	Usage of TPA	Verification Style	Time Taken ms/GB
Atneise et. al [2]	$1-(1-\rho)^t$	Yes	On demand	62.5
Erway et. al [3]	$1-(1-\rho)^t$	Yes	On demand	30
Block-chain based integrity verification	1	No	Auto-generated	8

VII. CONCLUSION

A design for integrating blockchain in the cloud storage is proposed and discussed. The working model of the original blockchain that is used in bitcoin transactions have been explained. In this way, we can implement blockchain in the cloud for storing data and detecting the tampering of data. The immutability and the decentralization properties of the blockchain technology can be incorporated into different fields to obtain different results. In the proposed model, we took advantage of these properties to achieve the data integrity. The proposed model has been compared against the existing protocols in terms of cost and time, and have been proved efficient than other protocols. In future, this can be extended to attain other properties of data protection too. Also, the data recovery can be addressed in the upcoming researches. The proposed design will be suitable for all immutable records like a government's data about its citizens. This design can be modified further to allow data modifications, so that it suits the mutable data storage in future.

REFERENCES

[1] E. Gaetani, L. Aniello, R. Baldoni, F. Lombardi , A. Margheri and V. Sassone, Blockchain-based database to ensure data integrity in cloud computing environments, in Proceedings of the First Italian Conference on Cybersecurity (ITASEC17), Venice, Italy, 2017.

[2] G. Ateniese, R. Burns, R. Curtmola, J. Herring, O. Khan, L. Kissner, Z. Peterson and D. Song, Remote data checking using provable data possession, ACM Transactions on Informatin and System Security (TISSEC), pp 12:1-12:34, 2011.

[3] C.C. Erway, A. Kupcu, C. Papamanthou and R. Tamassia, Dynamic Provable Data Possession, ACM Transactions on Information and System Security, Vol. 17, No. 4, Article 15, 2015.

[4] H. Wang, D. He, J. Yu and Z. Wang , Incentive and unconditionally anonymous Identity-based public Provable Data Possession, IEEE Transactions on Service Computing, 2016.

[5] S. Shen, H. Lin and W. Tzeng, An effective integrity check scheme for secure erasure code- based storage systems, IEEE transactions on Reliability,vol. 64, no. 3,pp. 840-851, 2015.

[6] D. K. Tosh, S. Shetty, X. Liang, C. A. Kamhoua, K. A. Kwiat and L. Njilla, Security implications of blockchain cloud with analysis of block withholding attack.,17th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGRID), 2017.

[7] A. P. Joshi, M. Han and Y. Wang, A survey on security and privacy issues of blockchain technology, Mathematical foundations of computing, pp. 121-147, 2018

[8] P. J. Ho and P. J. Hyuk, Blockchain security in cloud computing: Use cases, challenges and solutions, Symmetry, 1–13, 2017.

[9] L. Lamport, R. Shostak and M. Pease, The byzantine generals problem, ACM Transactions on Programming Languages and Systems, Vol. 4, No. 3, 1982

[10] .R. Kotla, L. Alvisi, M. Dahlin, A. Clement and E. Wong, Zyzzyva: Speculative byzantine fault tolerance, in ACM SIGOPS Operating Systems Review , ACM, pp. 45– 58, 2007.

[11] M. Vukolic, The quest for scalable blockchain fabric: Proof-of-work vs. BFT replication, International Workshop on Open Problems in Network Security, Springer,112–125, 2015.

[12] D. Larimer, Transactions as proof-of-stake, 2013.

[13] D. Larimer, Delegated proof-of-stake white paper, 2014.

[14] J. A. Garay, A. Kiayias and N. Leonardos, The bitcoin backbone protocol: Analysis and applications., in EUROCRYPT, Lecture notes in Computer Science, pp. 281–310, 2015.

[15] G. Wood, Ethereum: A secure decentralised generalised transaction ledger, Ethereum Project Yellow Paper, pp. 1–32, 2014.

Smart FIR: Securing e-FIR Data through Blockchain within Smart Cities

Nasir D. Khan, Chrysostomos Chrysostomou
Department of Electrical and Computer Engineering and Informatics, Frederick University.
Nicosia, Cyprus.
st017021@stud.frederick.ac.cy, ch.chrysostomou@frederick.ac.cy

Babar Nazir
Department of Computer Science, COMSATS University Islamabad.
Abbottabad, Pakistan.
babarnazir@cuiatd.edu.pk

***Abstract*—Electronic First Information Report (e-FIR) is a basic document filed to the police stations by a victim or someone on his/her behalf when a cognizable offense such as murder, kidnapping, rape, theft, etc. is committed. In the e-FIR database, the offense's record can be compromised due to its centralized nature, and further the intentional registration of false e-FIR can occur. Thus, data integrity and transparency are key concerns in e-FIR database. In this paper, e-FIR data integrity and false registration appended with police stations in a centralized database are addressed via a consensus-based distributed blockchain solution, as an integral part of a smart city environment. Specifically, a smart contract based intelligent framework has been utilized to explore the potential of Ethereum blockchain in providing integrity to e-FIR data stored in a police station's database. Local database is interfaced with Ethereum blockchain using Web3 Remote Procedure Call (RPC) protocol. Multiple simulations have been performed to evaluate the performance of the proposed framework. Our results show a trade-off between different hashing algorithm security level for the offenses data and number of transactions stored in a single block on blockchain ledger.**

***Index Terms*—e-FIR, Smart cities, Smart contract, Blockchain, Data integrity.**

I. Introduction

Smart cities strongly rely on the concept of Information and Communication Technologies (ICT), which invest in human social life to improve their citizens' quality of life, by stimulating economic growth, sustainable good governance, wise resources management, and efficient mobility, whilst they guarantee the security and privacy of their citizens [1]. Giant companies like Intel, IBM and Siemens are hugely investing in futuristic smart cities [2], as the latest statistics show that urbanization is progressing at an unprecedented pace. According to the UN report of 2018, currently, more than 50 percent of the World's population lives in metropolitan cities and is expected to grow up to 66 percent by 2050 [3]. Moreover, smart city infrastructure needs efficiency in many aspects, from resource allocation to energy consumption, social security to health management [4], and safe city [5] to the criminal record management system.

In a smart city of smart vehicles, smart schools, smart hospitals, smart infrastructure etc., where everything is connected to the Internet (IoE) [6] to share tremendous data volume daily, this city should also provide a smart and secure system for Electronic First Information Report (e-FIR) data management in a police station as shown in Fig. 1. e-FIR is a simple document that has been written out and filed to the police by the victim or someone on his/her behalf when a cognizable offense such as murder, kidnapping, rape, theft, etc. is committed. Reporting a crime and filing a cognizable offense manually in a police station consumes a lot of time because the police to people ratio in some of the commonwealth countries are tremendously high as shown in Table I. Instead, the e-FIR mechanism is used in some of the commonwealth countries i.e. Pakistan, India, Bangladesh, Malaysia, Japan and Singapore, while the mechanism for filing an offense in Europe and USA is, apparently, different than the aforementioned countries [7].

Non-registration, false registration and integrity of e-FIR data are the main concerned problems connected with it. These problems are due to police corruption, inefficiency and lack of accountability. Initially, e-FIR data is stored in a central database of police station locally, which is then shared with the headquarter (HQ) of police stations. Here the e-FIR data could easily be manipulated as the control of e-FIR database is local within the police station. Therefore, to address this problem, applying blockchain technology can help us to better respond to the security challenges and can endeavour data integrity, as blockchain is a fraud-resilient, distributed ledger, which can record all the transactions in a Peer-to-Peer (P2P) network. Blockchain has a decentralized architecture, and its popularity in the cryptocurrency world in securing the distributed network communication has been remarkable [8].

In this paper, the major contributions are twofold: Firstly, a blockchain-enabled framework providing efficient integrity to e-FIR data is proposed, which is applicable in, and been an integrated part of, a smart city environment. Secondly, false registration of e-FIR is minimized by resolving it through the concept of blockchain. To the best of our knowledge, this is a first attempt restraining false registration and providing integrity to e-FIR data using blockchain.

The rest of paper is organized as follows. Section II discusses e-FIR and relevant approaches therein. Section III presents the proposed system architecture. The proposed framework implementation and evaluation results are shown in Section IV. Finally, the concluding remarks with future work is given in Section V.

978-1-7281-5207-3/20/$31.00 ©2020 IEEE

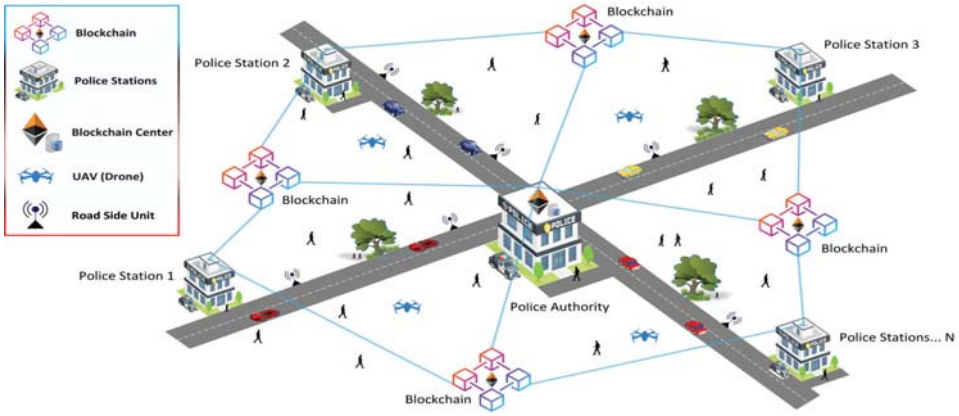


Fig. 1: System model of smart police stations in a smart city environment.

Table I: POLICE TO PEOPLE RATIO IN SOME COUNTRIES [19]

No.	Country	Police-People Ratio
1	Bangladesh	1:1138
2	India	1:728
3	Pakistan	1:625
4	Singapore	1:614
5	Malaysia	1:450

II. E-FIR Background And Related Work

In various systems, criminal records and different offenses data are usually stored in centralized storage. However, there can be multiple deficiencies in centralized systems, such as single point of failure. On the other hand, different offenses data stored in local database in a police station are highly vulnerable to the following issues:

- **Data Tampering:** Storing data in a local database of an institution can allow the superior authority to manipulate the crucial data without taking any other authority into consideration. The only way to solve this issue is to mark every single data with digital signature and distribute it among different entities to keep the data transparent.
- **False Registration:** Police officials having access to data stored in local database can register false case on anyone without disclosing the personal identification number (ID) and credentials of the officer in-charge (admin) with the case. To identify the right person being involved in the false case is a challenge, and it can only be handled by sharing the admin credentials with different entities, so it could be used for auditing purpose.

In order to improve system security and provide integrity to the offenses data, a decentralized consensus-based approach is required, where the user can trust the system to interact and share information without being concerned about data tampering.

Blockchain has recently gained prominent popularity, mostly due to its distributive nature, where the blockchain decouples the centralized hold from single entity and gives control to multiple participating entities, who validate the authenticity of the records and make the ledger completely transparent. There are two main types of networks in blockchain, that is, public and private network blockchain. Bitcoin [9] and Ethereum refers to public blockchain using Proof-of-Work (PoW) concept, and Hyperledger-fabric refers to private blockchain using Proof-of-Authority (PoA) concept, where all operate in a trustless environment for online P2P transactions. The most hyped alternative created for the cryptocurrency application is the smart contract paradigm, where Ethereum and Bitcoin were deployed and served as cryptocurrencies [10], [11]. A smart contract is a software-defined protocol that can digitally verify, facilitate or even enforce the negotiations of a contract. Smart contracts execute intelligent transactions without any third party's intervention and those transactions are traceable and irreversible [12]. Ethereum is one of the blockchain platforms, which allows us to interact with object-oriented solidity programming for writing smart contracts.

Researchers have opted Blockchain for many diverse problems. Antra *et al.* [13] have discussed an idea of how to secure online FIR with blockchain by registering the complainer, suspect and witness to the system interface. In this work, the pre-registration of the process is conducted by the officer in-charge and the user credentials are stored in a local database, which can result in non-registration of FIR by making changes to the user authentication data. The authors also lacked in not addressing the issue of false FIR handling. Maisha *et al.* [14] have proposed a blockchain-based system for securing merely the criminal data into the blockchain distributed ledger and restraining the data from any unlawful changes by unauthorized personnel. A technique of pre-registering users to the system has been used and the criminal data is uploaded to the cloud repository. The authors lacked in addressing the integrity of user's data stored on cloud database, which eventually does not consider the case of false FIR registration. Kirti *et al.* [15] have proposed a portal based e-FIR system, in which an administrator ensures the authenticity and integrity of the FIR data by only filing the pre-registered FIR in the local database, which provides transparency using e-governance. However, the

authors lacked in addressing the data integrity even if they use the pre-registering technique. Muhammad Baqer Mollah *et al.* [16] have introduced a system in which, the home ministry would be connected with all the police stations in a city in Bangladesh, called the 'Third Eye', and its sole purpose would be to keep track on police stations activities and records. Here, home ministry officials have access to the data and could be tampered easily due to the existence of a central database, which is solely managed by the home ministry officials.

According to the literature review, no previous work has a focus on providing intelligent integrity to e-FIR data and handling false registration of e-FIR stored in central database in a police station. For this issue, we propose a consensus-based blockchain framework, where multiple participating entities are involved to maintain the transparency of e-FIR data.

III. Proposed Blockchain-Based Framework

The proposed intelligent framework utilizes benefits of the blockchain technology by addressing an important challenge, namely, how intelligent integrity could be provided to e-FIR data stored in the centralized database of a police station in a fully connected digital city (smart city) interoperability scenario. The vision is to decentralize the authorities hold on e-FIR data in a central database of police station among different entities to provide transparency. In this paper, the proposed novel framework is specifically twofold:

- An intelligent system is proposed to provide e-FIR data integrity through distributed blockchain ledger, using smart contract, which is tamper-proof and fraud resilient.
- False e-FIR registration is also dealt by collecting the credentials of both the user and the admin and storing them on the blockchain for auditing purpose.

A. System Architecture

We assume that the ID of citizens stored in a national database of a country are safe and secure, and the system interface (SI) from which e-FIR can be registered by the user, is connected with the national database for user authentication. The workflow of the proposed system architecture, as shown in Fig. 2, is briefly elaborated as follows:

1) **Registration of Police Station:** The superintendent of police (SP) in the HQ generates a unique account address for every single police station, called hash of the police station and it is stored/registered on the blockchain ledger using smart contract. In PoW, all the addresses initiate the mining process in a consensus and whichever participating address achieves in solving the complex puzzle, then its block is mined. On the other hand, in PoA, the authority address is only responsible for mining the blocks. In the hash of an individual police station, the following details are integrated with it, which are used for auditing purpose.

- City and location of the police station.
- In-charge (admin) of the police station.
- Names of all the investigating officers.

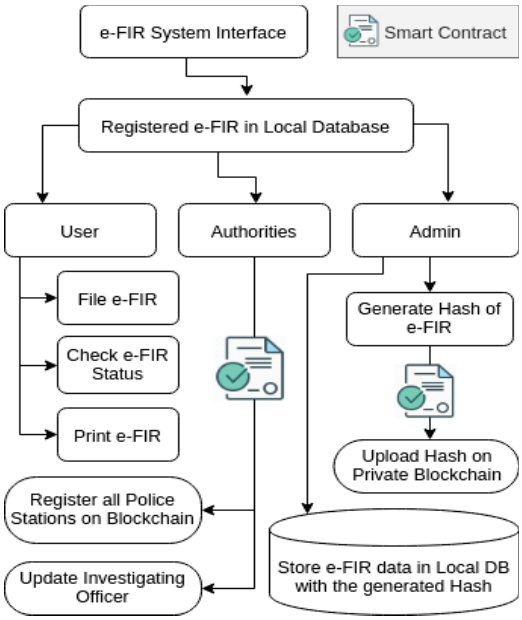


Fig. 2: Flow diagram of the proposed blockchain-based architecture.

In case of a new investigating officer selected in the police station, the admin of that particular police station would be liable to inform the HQ, so that the officials of HQ would update the credentials of that police station and make a new transaction on the blockchain. All the participating addresses (police stations) will also know about the new appointment. Likewise, the same procedure would also be followed for the admins of the police stations.

2) **User Filing e-FIR:** The user interacts with SI by entering ID for validation, which is done on run-time basis from the citizen's national database as it is connected with the SI, and allow him/her for filing an e-FIR in case of cognizable offenses only. If the user had filed e-FIR and it is still pending, then he/she will not be allowed to modify it, as it will result in change in the original hash value, which indicates changes made to the original e-FIR data that would help in identifying fraud. The user will have to provide all the following details when filing an e-FIR (with any additional information if any).

- Time, date and place of the offense and the reporting.
- All personal details of the complainant and accused.
- Complete detailed description of the offense.
- Any additional evidences for proof (if any).
- Description of the property stolen (if any).
- Police station where the offense is registered.

3) **Admin Approving e-FIR Transaction:** The admin will be responsible for authorizing all the transactions on blockchain. When user files an e-FIR, the admin of police station assigns one of the investigating officers to verify the information provided by the user, check the originality of evidences and solve the case. If the data is found to be valid, the admin generates a hash of that e-FIR data and uploads

Fig. 3: Transaction of a single block in Ethereum blockchain (Ganache).

it on the distributed private blockchain using smart contract. Also, the e-FIR details provided by the user are uploaded to the police station central database digitally signed with the exact hash generated for that e-FIR data. The hash will account as an ID for the e-FIR. If e-FIR data is found as fraud, the admin will not approve the transaction and hence, the case and the transaction will be dropped.

4) **Handling False e-FIR:** If the user or admin of police station intentionally tries to register false e-FIR against someone, then the accused user will have the privilege to request to the SP for an auditing of false e-FIR. The SP has access to the hash data and all other details of the case such as the city and police station, admin of the police station, investigating officer, and e-FIR data, which is allegedly filed against the accused user. As the credentials of all involved persons who have filed alleged e-FIR, are saved on blockchain in the form of hash, they are unable to withdraw their identities from blockchain ledger to vanish the evidence of not being involved in the case. Blockchain also stores the time-stamp of every block transaction, which can further aid in identifying the involvement of a person in fraudulence.

IV. Implementation and Results

We have created a local database of e-FIR data in Matlab and that local database is interfaced with Ethereum blockchain using the intelligence of smart contract. The details of our proposed model implementation is explained below.

A. Platform Interfacing

In a P2P interfacing, we have connected Matlab with Python IDE, and eventually, the Python IDE is connected with Ganache [17], which is a tool used for Ethereum blockchain. We have deployed the smart contract on Ethereum online Remix IDE [18]. The smart contract is deployed on a Web3 Remote Procedure Call (RPC) environment using a specific port number. Python IDE receives data from Matlab (database) on that specific port number, which then forwards the data to Ganache (Ethereum Blockchain). The port number of Ganache IDE should be set to the same port number used by Matlab, Python and Remix IDE [20]. Complete details of e-FIR data stored in single transaction on blockchain is shown in Fig. 3.

B. Ethereum Blockchain

We have used Ganache software for Ethereum blockchain, which is a development tool provided by Ethereum developers. The benefit of using Ganache is that it provides 10 different accounts with each account having 100 ethers, and the purpose of those ethers are merely for development purpose. For the scalability of the system, we need to build and implement personal blockchain, where we can generate multiple unique addresses for every single node. The benefit of personal blockchain is that, if we define an authority for mining the block, then the mining process becomes very fast, because the authority is only responsible for utilizing computational power for mining the block. The mining process of Ethereum uses PoW concept; however, defining functions in a smart contract and allowing specific addresses to specific operations can provide the benefits of PoA. Specifically, in our model, we have made some addresses as an authority that can do specific operations, which other addresses can not do, i.e. registering all the police stations from SP node address, and approving e-FIR transaction from admin node address.

The more number of registered e-FIRs in police station database, the more number of transactions occur on blockchain ledger using smart contract, as shown in Fig. 4. The graph is plotted against SHA-256 hashing algorithm (more details can be found in Section IV.C).

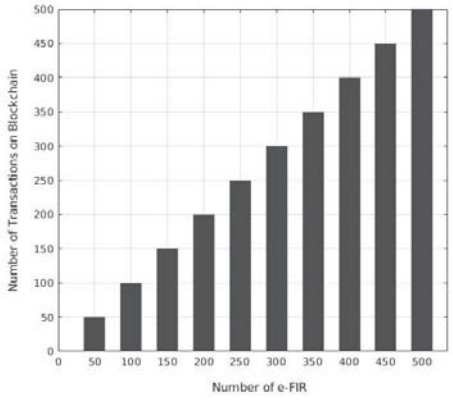


Fig. 4: Number of transactions vs. e-FIRs on blockchain.

C. Smart Contract

In our model, we have developed smart contract in a solidity programming language in Remix IDE used for Ethereum

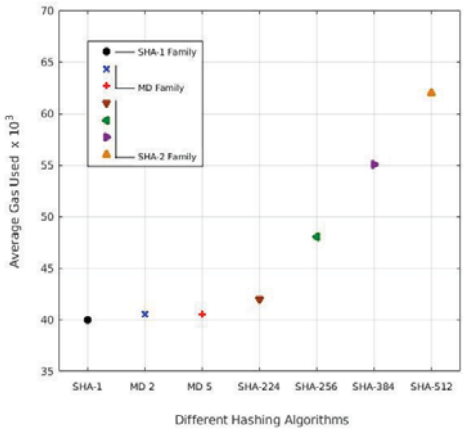


Fig. 5: Different hashing algorithms.

blockchain, which gets data in the form of hash and stores it on the blockchain. The functions in smart contract contains the following functionalities as discussed in Section III.

- Registering all Police Stations.
- Uploading Hash on Blockchain.
- Updating Investigating Officer.

We have used a number of different hashing algorithms in the implementation of the proposed blockchain-based framework to test the impact of these hashing functions on the performance of our proposed framework. We used the Secure Hash Algorithm (SHA) family and the Message Digest (MD) family of hashing algorithms. Specifically, we used SHA-1, SHA-224, SHA-256, SHA-384, SHA-512, MD-2, and MD-5. Gas in Ethereum is defined as a special unit that measures the amount of computational effort that it will take to execute certain operations. From the results obtained, as shown in Fig. 5, the minimum average Gas used by SHA-hashing family is 40,000 and maximum Gas being used is around 62,000. Likewise, the average Gas used by MD-hashing family is 40,500. SHA-512 (512 bits) is considered to be the most advanced and secured hashing algorithm, but it uses more Gas, as shown in Fig. 5, resulting in less transactions per block in blockchain. On the other hand, using SHA-1 (160 bits) can benefit us with more transactions in a single block, but with less hashing security, as SHA-1 is not as much secure as SHA-512 is. Using SHA-256 (256 bits) could endeavor data integrity by having sufficient hashing security level while using moderate Gas value, as observed in Fig. 5.

D. System Specifications

We have tested the proposed e-FIR model on the following system specifications, as shown in Table II.

Table II: SYSTEM SPECIFICATIONS

System RAM	8 GB DDR3
Hard Drive	128 SSD/640 HDD
System Core	Intel Core i5
Operating System	Ubuntu 18.04

V. Conclusion And Future Work

This paper examines the relatively under-developed area of record management in police stations for the prevention of data tampering and false report filing, using the concept of blockchain technology. Research conducted in this paper has presented a consensus based solution for providing integrity to the offenses data stored in police station database using blockchain. In the proposed framework, Matlab is interfaced with Ethereum blockchain using Python and Web3 RPC to intelligently secure the e-FIR data transaction through smart contract. Multiple simulations have been performed to demonstrate the trade between number of transactions occur in a single block and different hashing security level for e-FIR data.

The proposed system will further be investigated in future for dynamically selecting different hashing algorithms based on classification and criticality of the offenses data. The system will also efficiently utilize the Gas value in Ethereum blockchain by identifying the offense's data type and its importance, in order to maximize the number of transactions stored in a single block.

References

[1] P. A. Perez-Martinez et al. "Privacy in Smart Cities- A Case Study of Smart Public Parking," Proc. 3rd Int'l Conf. Pervasive Embedded Computing and Commun. Sys., pp.55–59, 2013.
[2] M. Dohler et al., Eds., "Feature Topic on Smart Cities", IEEE Commun. Mag., vol, 51, no. 6, 2013.
[3] [Online: January, 2019] Urban Population Growth statistics by UN; https://population.un.org/wup/Publications/Files/ WUP2018-Report.pdf
[4] Agusti Solanas et al., "Smart Health: A Context-Aware Health Paradigm within Smart Cities", IEEE Commun. Mag., vol, 52, no. 8, 2014.
[5] Jaime Ballesteros et al. "Safe Cities. A Participatory Sensing Approach", IEEE LCN, 2012.
[6] Paola G. V. et al., "FOCAN: A Fog-supported Smart City Network Architecture for Management of Applications in the Internet of Everything Environments", J. Parallel Distrib. Comput., 2018.
[7] [Online: March, 2019] Website of US department of justice for reporting a crime; https://www.justice.gov/actioncenter/report-crime.
[8] R. M. Parizi et al., "Empirical vulnerability analysis of automated smart contracts security testing on blockchains" CASCON, IBM Corp., 2018.
[9] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," white paper, 2008.
[10] T. T. A. Dinh et al., "Un-tangling Blockchain: A Data Processing View of Blockchain Systems,"in IEEE Transactions on Knowledge and Data Engineering, vol. 30, no.7, pp. 1366-1385, 1 July, 2018.
[11] Jean Bacon et al., "Blockchain Demystified: A Technical and Legal Introduction to Distributed and Centralized Ledgers", 25 RICH. J.L. and TECH., no. 1, 2018.
[12] Reyna et al., "On blockchain and its integration with IoT. Challenges and opportunities." Future Generation Computer Systems, vol 88, 2018.
[13] Antra Gupta et al., "A Method to Secure FIR System using Blockchain", IJRTE, Vol. 8, Issue-1, 2019.
[14] Maisha A. Tasnim et al., "CRAB: Blockchain Based Criminal Record Management System", SpaCCS, LNCS 11342, pp. 294–303, 2018.
[15] Kirti Marmat et al., "E-FIR using E-Governance", IJIRST, vol. 3, 2016.
[16] Muhammad Baqer Mollah et al., "Proposed E-Police System for Enhancement of E-Govemment Services of Bangladesh", IEEE/OSA/IAPR, 2012.
[17] [Online: January, 2017] Personal blockchain for Ethereum development; https://www.trufflesuite.com/docs/ganache/overview.
[18] [Online: November, 2019] Josh Cassidy, Article for Online Remix IDE- writing smart contract; https://kauri.io/remix-ide-your-first-smart-contract/124b7db1d0cf4f47b414f8b13c9d66e2/a.
[19] [Online: October, 2011] Bangladesh Police's Website, Police to People Ratio; http://www.police.gov.bd/index5.php?category=48.
[20] A. B. Masood et al., "Realizing an Implementation Platform for Closed Loop Cyber-Physical System using Blockchain", IEEE 89th VTC, 2019.

Preserving Location Data Integrity in Location Based Servers using Blockchain Technology

Mahesh Kumar K.M* and Sunitha N.R†
Department of CSE,
Siddaganga Institute of Technology
Tumkur, Karnataka, India
Email: *maheshkumarkm87@gmail.com,
†nrsunitha@sit.ac.in

***Abstract*—**

Data integrity or data quality plays a vital role in the storage reliability and security. With the advances in technological trends in storage, new challenges of data integrity surfaces. In this paper we discuss some key challenges in data integrity and suggest a blockchain based solution to solve data integrity problem in Location Based Servers. We designed and implemented a smart contract and query/update application to suit the needs of Location Based Servers. Implemented the Geo server on Hyperledger Fabric and tested the working of smart contract and query/update application on the testbed created. Our work show that proposed blockchain based solution is efficient and helps preserve privacy and confidentiality, highlighting the rich features and benefits.

***Index Terms*—Blockchain, Data Integrity, Geo Location, Hyperledger Fabric, Location Based Services**

I. Introduction

One of the fundamental concepts of information security is "Data Integrity", in its broadest sense Data Integrity refers to accurate and consistent data that is stored in any of the construct like data mart, data warehouse, database. The term data integrity is synonymous with the term data quality, which describes a state, a function or a process. Data that preserves the integrity/quality is termed complete, characteristics associated with data correctness are business rules, definitions, dates, relations and lineage. Error checking and validation rules helps to preserve data integrity in database, as an example numeric columns restrict users from entering alphabets in to the cells.

Data integrity as a process guarantees that data in transit is unaltered from creation to consumption, Data integrity as a condition or state quantifies the data object for its validity and fidelity, as a function related to security it is responsible for maintaining the information as it is when it was in-putted, audit-able to confirm its reliability. In order to support decision-making data undergoes several operations like capture, store, retrieve, update and transfer. Data integrity serves as a performance measure i.e., based on the error rate that occur during the operations. It is critical to protect data from corruption/modification and prevent unauthorized disclosure to take control of critical business processes. Accidental errors/inaccuracies are expected to occur for e.g. programming errors, or through malicious means e.g. hacking, hence database security professionals resort to several practices that assures data integrity, including:

- Data encryption: to apply lock on data using encryption techniques.
- Data backup: to back up data at alternate location, used in case of disaster recovery
- Access controls: gaining control over read/write privileges.
- Input validation: preventing wrong entry of data.
- Data validation: certifies data transmitted is correct.

Data integrity plays a crucial in Location Based Services (LBS), due to the fact that the success of the LBS application relies on the accuracy of the location or location information being provided as a service to the application user. Geo Location data is stored on Geo servers in the form of Points of Interest (PoI) database, user access this data via wireless media, PoI can be any location of user interest like nearest ATM, shopping mall, hospital, fuel station e.t.c. Accuracy of the location data determines the accuracy of the location information, leading to better LBS and good revenue.

In the following sections we discuss blockchain technology, starting from merkle tree a data structure which is used by blockchain, followed by blockchain along with its constituents and blockchain platforms available. Rest of the paper is organized as follows: section II highlights the challenges, solutions and attacks related to data integrity in general and from cloud perspective, also suggests how blockchain can be a better solution to address data integrity in LBS, section III discusses details of the proposed scheme, section IV lists the features and benefits of the proposed scheme, followed by conclusion and future work in section V and finally we list references.

A. Merkel Tree

Merkle tree was introduced by Merkel etal [1] in the year 1979, it has its wide spread use in the field of cryptography and computer science. Typically, a merkle tree (see Fig. 1) or a hash tree, is a tree structure in where data/transaction is represented by the leaf node and cryptographic hash of the concatenated hash value of its child nodes non-leaf nodes.

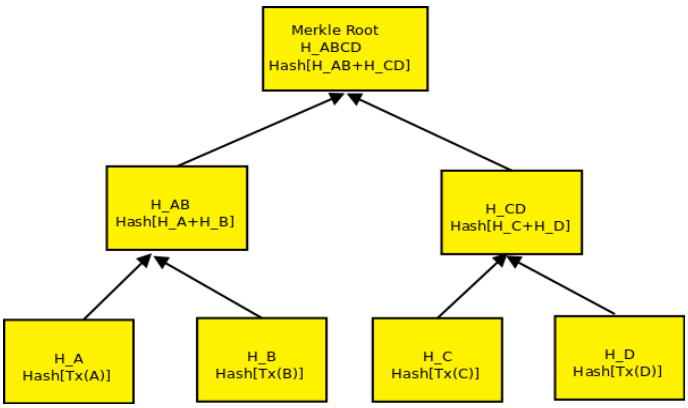


Fig. 1. Merkel Tree

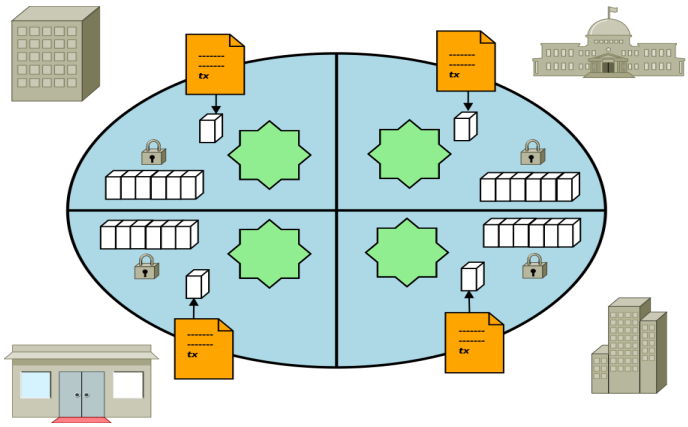


Fig. 2. Blockchain

Hash trees are an efficient and secure way to verify the contents stored in a large data stores.

B. Blockchain

Satoshi Nakamoto was the first researcher to conceptualize the first blockchain in the year 2008 [2] and to implement it as a core component of bitcoin (digital currency), where blockchain serves as the public ledger for all transactions. Satoshi Nakamoto solved the double spending problem by introducing blockchains, this was made possible without the need for central server or trusted authority. The bitcoin design has inspired many researchers to apply this technique to other applications.

A blockchain [2] is a list of records called blocks (see Fig. 2) which grows continuously, cryptography is used to maintain links and secure the blocks. Each block consists of hash pointer (serves as pointer to previous block), a timestamp and data related to transaction details. Blockchain is designed to resist modification of data (tamper evident), it is "an open, distributed ledger that can record transactions between two parties efficiently and in a verifiable and permanent way". Blockchains are managed by peer-to-peer network, which abides to the protocol in order to validate the new block generated. It is impossible to alter the data once recorded, without altering all the subsequent blocks, which requires coordination of the majority of the network.

Security, fault tolerance and decentralized consensus property are achieved in blockchains by design and is a good example of distributed computing system, Hence blockchains serves as the preferred choice for the record management activities like medical, transportation, e.t.c., and also it can efficiently handle event such as documenting provenance, food traceability, identity management or transaction processing.

1) A Distributed Ledger: Distributed ledger forms the heart of the blockchain network, it is responsible for recording all the transactions taking place in the network. A blockchain ledger is decentralized in nature, a replica of the ledger is stored in all the network participants, who take part in the maintenance of the ledger. Collaboration and decentralization are the powerful and desirable attributes that mirror the way the businesses function in the real world scenario.

Additionally, blockchain works in append-only manner, with the help of cryptographic techniques it is guaranteed that transactions once recorded cannot be altered. Immutability property ensures provenance of information and increases the confidence among the network participants, that they can be sure about the information, hence blockchain is often termed as systems of proof.

2) Smart Contracts: Smart contracts are the mechanisms that provide controlled ledger access, to support consistency in information update and to enable ledger functions like transacting, updating, querying e.t.c., also they encapsulate information and try to keep it simple over the entire network. Smart contracts has the capability to automate few aspects of the transaction. A smart contract can, for e.g., regulate the cost of the shipment of an item based on the arrival time (on-time or delayed) depending on the agreement made by both parties on the ledger, funds transfer takes place automatically after the shipment of item.

3) Consensus: Consensus is a critical mechanism that deals with ledger transaction synchronization, it is responsible to keep the transaction ledger synchronized across the network. It guarantees that ledger updates happen only upon the approval of the appropriate participants, preserving the transactions and order in which the transactions took place.

C. Blockchain Platforms

There are many blockchain platforms available like BigChainDB, Chain Core, Corda, Credits, Eris:db, Etherium, HydraChain, Hyperledger Fabric, Hyperledger Sawtooth Lake [3] e.t.c.

Hyperledger Fabric: Hyperledger was founded by the Linux Foundation in the year 2015 in order to enhance cross-industry blockchain technologies. There is no single standard in blockchain, instead a collaborative approach via a community process is taken to develop blockchain technologies in order to encourage open standards and adopt the key standards in course of time.

Hyperledger Fabric is sub blockchain project under the Hyperledger. It is equipped with a ledger, makes use of smart contracts and consensus by which network participants handle their transactions. Unique property of Hyperledger Fabric is it uses private and per-missioned blockchain. Unlike other blockchain technologies which use open permission-less system which allow unknown identities to take part in the network (based on proof of work), Membership Service Provider (MSP) provides the membership service to participants of a Hyperledger Fabric network, through which they can enroll and take part in network.

Hyperledger Fabric is highly modular in nature and supports several pluggable options. Different ledger data formats can be used for storage, allows switching in and out of consensus mechanism and also several variants of MSPs are supported. Channel creation ability is offered and Hyperledger Fabric enables a group of participants to create their own ledger of transactions, This is important in a scenario where competitors are part of the same network and do not want every transaction they make to be known to every other participants for e.g., a special price offered to some participants should not be revealed to other participants.

II. Literature Survey

The normal practice in the past was that, users would download the necessary application software on to their machine and run the software on to their physical computer/server located in their building. However, with the introduction of the cloud application users can now use/access the same application through Internet.

Location Based Services (LBS) are now being hosted on the cloud for the following benefits provided by the cloud [4]:

- Flexibility.
- Disaster recovery.
- Automatic software updates.
- Capital-expenditure Free.
- Security.

Data integrity schemes, discussed in our work are of probabilistic nature, tamper evident (not tamper resistant) and, can be applied only on secondary storage (non-volatile data). Integrity schemes for volatile data are not discussed in this paper. There are three entities in a data integrity scheme: (i) Data owner who outsources his data, (ii) a cloud storage provider (CSP) to whom data are outsourced and (iii) an auditor who verifies the integrity of the data. An auditor can be the data owner himself or some Third Party Auditor (TPA) who is chosen by the data owner.

With the help of data integrity schemes, any data corruption or deletion can be timely identified and thus necessary measures can be taken to recover the data. Several data integrity scheme are discussed based on different attributes.

Approach: The approach refers to the nature of guarantee provided by a data integrity scheme, which can be deterministic or probabilistic. Deterministic schemes [5], [6] need to access the complete file to determine the integrity and give 100% possession guarantee. Whereas, probabilistic schemes [7] use randomly chosen blocks of data to verify the integrity and give less than 100% guarantee.

Nature of data: This attribute describes the nature of data, on which the scheme is applicable. Data can be of static nature [8] (archival data, backups, or data that are never modified but are appended only) or of dynamic nature [9], [10] (that frequently change due to operations like create, read, update and delete).

Setup: The setup refers to the nature of environment in which data integrity scheme will be deployed. Since data are placed over the cloud, which can be simple (self-sufficient, providing all types of capabilities itself) or hybrid/cloud of cloud/multi-cloud (managing some resources internally and other are provided externally; [11])

Tendency: The tendency of a data integrity scheme can be verification only (identification of corruption/deletion of data) or verification with data recovery (if any corruption is identified). From the perspective of tendency, data integrity schemes are categorized into proof-of-retrievability (POR) [12] or provable data possession (PDP).

Metadata: All data integrity schemes use some additional metadata along with the original data, which are utilized in integrity verification process. This metadata can be "tags" [8], [10], [13], having homomorphic verifiable property, which help in generating an aggregated value in the verification process. The metadata may be "signatures" [14], [15], which are used as an alternative to tags, e.g. algebraic signatures to improve techniques like Reed-Solomon codes.

Encryption: Over the year's data, integrity schemes have utilized both symmetric [16], [17] and asymmetric encryption [18], [19] for security.

Data integrity schemes are vulnerable to a variety of attacks. Therefore, care must be taken in designing such a scheme. Following are the possible attacks that may be launched against a data integrity scheme along with their possible solutions:

Tag forgery attack: Forgery of tags is possible by malicious CSP in an attempt to hide the user's data damage and bypass the auditing challenge. A CSP may try to forge the tags to deceive the verifier successfully [20]. Cryptographic homomorphic tags can provide unforgeability [7], [11].

For more detailed survey on cloud computing data integrity schemes interested readers can go through this survey paper [21].

Today's transaction networks are just a slightly updated version of the traditional transaction networks that have existed ever since business records have been kept. The members of the business transaction network lack proper coordination, because each member has his own way storing the transaction records and proving the provenance of the transaction is hard.

The latest technological advancement enabled the transaction networks to take the transaction records paper folders to cloud platforms, but it has retained the age-old methods. There is no unified system to identify the network participants, proving provenances is hard and to clear transactions it still takes several days to weeks. Most of the contracts are still

signed and executed manually, database is centralized leading to single point of failure. With the today's fractured information sharing approach it is not possible to build a system of record which can span the complete transaction network, even though visibility and trust are critically needed in business network.

Business network like LBS, demand standards for managing the identity of the participant on the network, storing data and executing transactions. Provenance of asset should be easy to establish just by looking through transactions list, also transactions once recorded cannot be changed and can therefore be trusted. In this paper we apply blockchain technology (Hyperledger Fabric platform) to location based servers to preserve location data integrity in LBS, our choice of applying blockchain is justifiable by the rich features provided the Hyperledger Fabric platform.

III. Proposed scheme

LBS application has three stakeholders, location provider, location service provider and the mobile user. Location provider as the name suggests is the one responsible for assisting the mobile user to know his location, which will be used to access the location service in the form of making queries to the PoI database (see Table III-3 for sample records) residing on the Geo server. In our proposed system we use Hyperledger Fabric to implement PoI database hosted on Geo Server. Writing an application to run on Hyperledger Fabric are discussed in detail with the example of Geo Server, in its simplest form, application running on blockchain network should enable its users to run a query against the ledger, queries of type select all/specific record or update existing or create a record.

We compose our Geo Server application using JavaScript, which uses the Node.js SDK in order to interact with the network on which the ledger resides. Three standard steps involved in writing the simple Geo Server application are as listed:

1) Test Network (Testbed)
2) Designing Smart Contracts
3) Developing Application

1) Test Network (Testbed): First step is to build a test network or testbed, in our paper we make use of Hyperledger Fabric to start our blockchain network (see section IV for benefits of using Hyperledger Fabric). Our test network consists of elementary components like Certificate Authority (CA) which is the backbone of test network, an ordering node, CLI container and a peer node. These components are essential in order to query or update the ledger. See Fig. 3 for simple blockchain network, a single script can launch this network on Hyperledger Fabric.

2) Designing Smart Contracts: In order to know what the smart contract does, we should look in to its functionality, our smart contract for the simple Geo Server application contains the functions to read the complete(holistic) and single(granular) record, to create a new Geo record and to update "type" field of an existing record. See Fig. 4 for smart contract information and Table III-3 for sample PoI records.

TABLE I
Sample PoI database records

Sl. No.	Type	Xcoord	YCoord	Address
1	ATM	13.32	77.12	SIT-EXT, Tumkur.
2	College	13.35	77.05	SIT, BH Road, Tumkur
3	Restaurant	13.22	77.10	OPP SIT, BH Road Tumkur

3) Developing Application: We develop two simple applications which are capable of interacting with the blockchain network. First application is designed to query the ledger residing on blockchain network and the second application is designed to update the ledger. Our applications make use of the SDK APIs available on Hyperledger Fabric in order to interact with the test network and finally call the functions provided by smart contracts see Fig. 5.

4) Implementation Details: We implemented our Geo server application on Hyperledger Fabric v1.0 installed on the machine running on Ubuntu 16.04 LTS 64 bit operating system (interested readers can go through the installation steps given in [22]). Test network was setup using the basic-network provided by Hyperledger Fabric v1.0, which in turn does the following activities:

- Launches CA, an ordering node, a peer node and CLI container.
- Creates a channel named "Mychannel" and joins all the peer to Mychannel.
- Installs the smart contract onto the peer's system, instantiates smart contract on Mychannel.
- Calls the initLedger function, which in turn populate the ledger with the unique Geo location records.

Once the test network and smart contract are in place we can proceed to query the ledger. We can query for single record or multiple records or all records based on the functionality provided by smart contract, it is also possible to perform complex searches like looking for all assets that contain certain keywords but the ledger has to be written to support JSON Objects which is a rich data storage format (interested readers can read [23]).

5) Evaluation Metrics: The following performance metrics can be used to test and validate the blockchain based Geo server application:

- Latency: it is the delay experienced per transaction, general strategy is to block new transaction until the current transaction commits.
- Throughput: it is the number of successful transactions per second.
- Scalability: increasing the throughput gracefully with optimized latency with the increased workload.
- Fault tolerance: it is the ability to handle failures gracefully for e.g., byzantine failures.

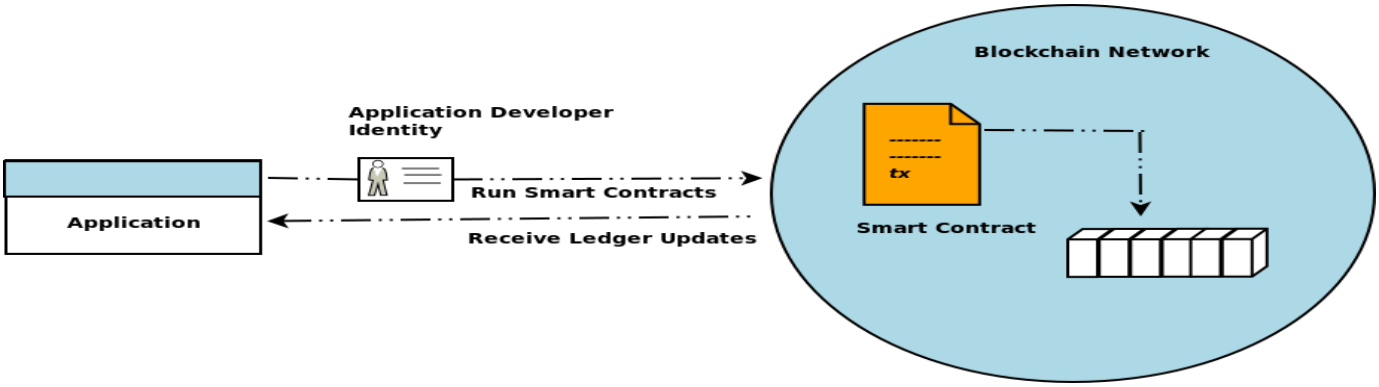


Fig. 3. Network

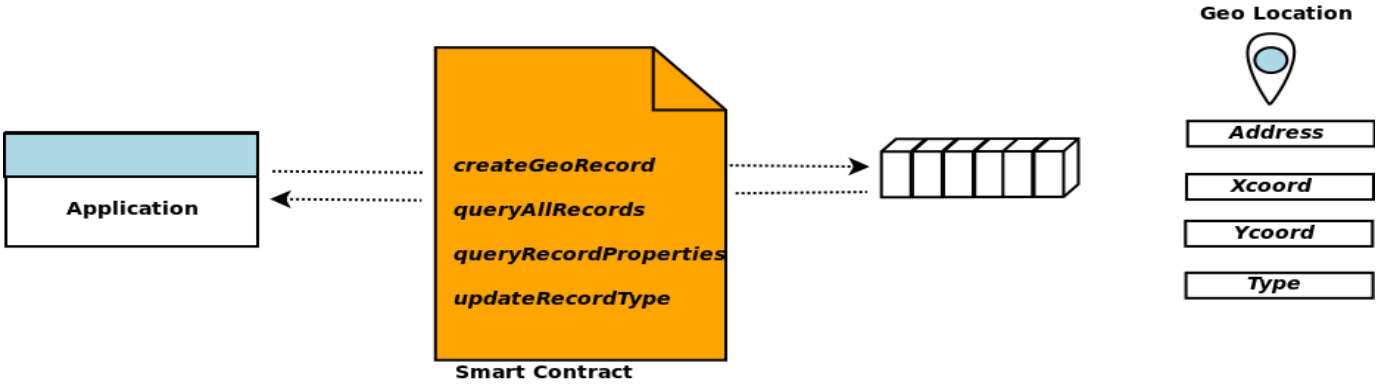


Fig. 4. Smart Contract

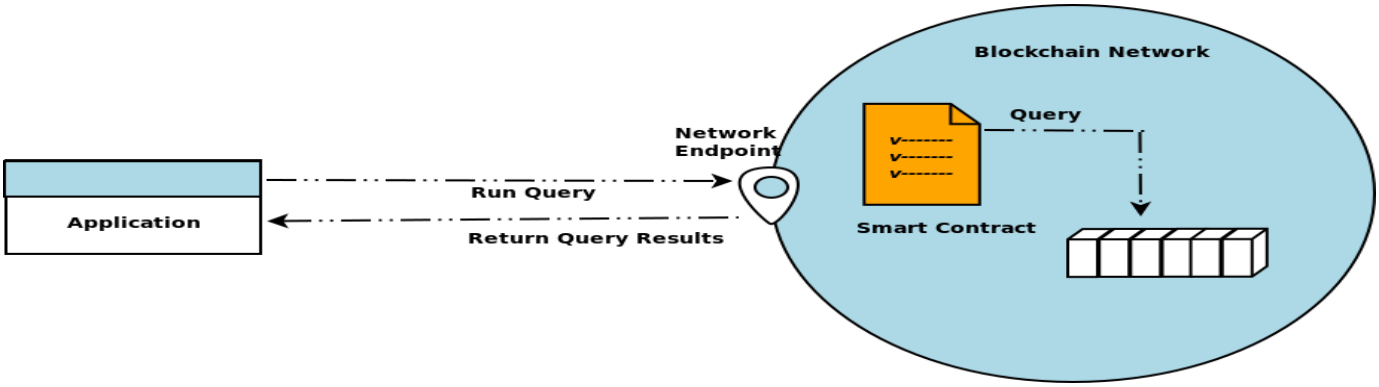


Fig. 5. Query

6) Adversary: Handling Byzantine failures is important to blockchain, Byzantine failure are a result of misbehavior caused by an attacker. Any rogue or compromised node which is part of the system constitutes the adversary. Under this threat model, safety property of the underlying consensus protocol defines the security of a blockchain system.

IV. Features and Benefits

Some key features which helped us to choose Hyperledger Fabrics include:

- Identity management: membership identity is provided as service, which manages all user identity and authenticate all the participating entities in the network.
- Privacy and confidentiality: provides private channels, restricting the messaging paths to provide transaction confidentiality and privacy to network entities.
- Efficient processing: provides concurrency and parallelism.
- Chaincode functionality: encodes logic that can be invoked based on the type of the transaction taking place on the network.
- Modular design: provides functional choice i.e. specific algorithms can be plugged in by network designers for identity or encryption.

Some key benefits of using Hyperledger Fabric are listed below:

- Consensus: All participants agree on transaction being made.
- Provenance: There is a single point of origin for all transactions.
- Immutability: Records cannot be changed or removed.

V. Conclusion and Future work

The main objective of the paper is to promote the idea of using blockchain network to solve the data integrity issues of Geo location database used in LBS applications, in our paper we have addressed the data integrity issue with the help of Hyperledger Fabric platform. We designed smart contract and query/update application to cater the needs of Geo server, we implemented and tested the same on testbed (blockchain network).

We recommend the use of Hyperledger Fabric platform to address data integrity and security issues of LBS application. In our future work we plan to design and build an application to support the complex searches using rich data storage formats, evaluate the work against a strong adversary model and prove the security and performance of the scheme.

References

[1] R. C. Merkle, R. Charles *et al.*, "Secrecy, authentication, and public key systems," 1979.

[2] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system", http://bitcoin.org/bitcoin.pdf," 2008.

[3] R. Nagpal, "Blockchain platforms-a brief introduction," https://medium.com/blockchain-blog/17-blockchain-platforms-a-brief-introduction-e07273185a0b, 2017, accessed: 2017-10-17.

[4] L. Liu and M. Parashar, Eds., *IEEE International Conference on Cloud Computing, CLOUD 2011, Washington, DC, USA, 4-9 July, 2011*. IEEE Computer Society, 2011.

[5] S. Jajodia and L. Strous, Eds., *Integrity and Internal Control in Information Systems VI - IFIP TC11/WG11.5 Sixth Working Conference on Integrity and Internal Control in Information Systems (IICIS) 13-14 November 2003, Lausanne, Switzerland*, ser. IFIP, vol. 140. Springer, 2004.

[6] X. Huang and J. Zhou, Eds., *Information Security Practice and Experience - 10th International Conference, ISPEC 2014, Fuzhou, China, May 5-8, 2014. Proceedings*, ser. Lecture Notes in Computer Science, vol. 8434. Springer, 2014.

[7] G. Ateniese, R. Burns, R. Curtmola, J. Herring, L. Kissner, Z. Peterson, and D. Song, "Provable data possession at untrusted stores," in *Proceedings of the 14th ACM conference on Computer and communications security*. Acm, 2007, pp. 598–609.

[8] J. Yuan and S. Yu, "Proofs of retrievability with public verifiability and constant communication cost in cloud," in *Proceedings of the 2013 international workshop on Security in cloud computing*. ACM, 2013, pp. 19–26.

[9] R. Curtmola, O. Khan, and R. Burns, "Robust remote data checking," in *Proceedings of the 4th ACM international workshop on Storage security and survivability*. ACM, 2008, pp. 63–68.

[10] Y. Zhang and M. Blanton, "Efficient dynamic provable possession of remote data via balanced update trees," in *Proceedings of the 8th ACM SIGSAC symposium on Information, computer and communications security*. ACM, 2013, pp. 183–194.

[11] Y. Zhu, H. Hu, G.-J. Ahn, Y. Han, and S. Chen, "Collaborative integrity verification in hybrid clouds," in *Collaborative Computing: Networking, Applications and Worksharing (CollaborateCom), 2011 7th International Conference on*. IEEE, 2011, pp. 191–200.

[12] J. Xu and E.-C. Chang, "Towards efficient proofs of retrievability," in *Proceedings of the 7th ACM symposium on information, computer and communications security*. ACM, 2012, pp. 79–80.

[13] Y. Zhu, H. Hu, G.-J. Ahn, and M. Yu, "Cooperative provable data possession for integrity verification in multicloud storage," *IEEE Transactions on Parallel and Distributed Systems*, vol. 23, no. 12, pp. 2231–2244, dec 2012.

[14] W. Luo and G. Bai, "Ensuring the data integrity in cloud data storage," in *Cloud Computing and Intelligence Systems (CCIS), 2011 IEEE International Conference on*. IEEE, 2011, pp. 240–243.

[15] S. R. Tate, R. Vishwanathan, and L. Everhart, "Multi-user dynamic proofs of data possession using trusted hardware," in *Proceedings of the third ACM conference on Data and application security and privacy*. ACM, 2013, pp. 353–364.

[16] G. Ateniese, R. Di Pietro, L. V. Mancini, and G. Tsudik, "Scalable and efficient provable data possession," in *Proceedings of the 4th international conference on Security and privacy in communication netowrks*. ACM, 2008, p. 9.

[17] N.-Y. Lee and Y.-K. Chang, "Hybrid provable data possession at untrusted stores in cloud computing," in *Parallel and Distributed Systems (ICPADS), 2011 IEEE 17th International Conference on*. IEEE, 2011, pp. 638–645.

[18] A. F. Barsoum and M. A. Hasan, "Integrity verification of multiple data copies over untrusted cloud servers," in *Proceedings of the 2012 12th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (ccgrid 2012)*. IEEE Computer Society, 2012, pp. 829–834.

[19] B. Chen and R. Curtmola, "Robust dynamic remote data checking for public clouds," in *Proceedings of the 2012 ACM conference on Computer and communications security*. ACM, 2012, pp. 1043–1045.

[20] K. Yang and X. Jia, "Data storage auditing service in cloud computing: challenges, methods and opportunities," *World Wide Web*, vol. 15, no. 4, pp. 409–428, Jul 2012.

[21] F. Zafar, A. Khan, S. Malik, M. Ahmed, A. Anjum, m. i. Khan, N. Javed, M. Alam, and F. Jamil, "A survey of cloud computing data integrity schemes: Design challenges, taxonomy and future trends," vol. 65, 11 2016.

[22] Cendhu, "Installation of hyperledger fabric v1.0 from source on ubuntu 16.04 lts," https://blockchain-fabric.blogspot.in/2017/09/installation-of-hyperledger-fabric-v10.html, 2017, accessed: 2017-10-18.

[23] Ananymous, "Writing your first application," http://hyperledger-fabric.readthedocs.io/en/latest/write_first_app.html, 2017, accessed: 2017-10-18.

Blok Zincir Teknolojilerine Genel bir Bakış: Çalışma Prensibi, Fırsatları ve Zorlukları
An Overview of Blockchain Technologies: Principles, Opportunities and Challenges

Gültekin Berahan Mermer
ING Bank
İstanbul, Türkiye

Engin Zeydan
Türk Telekom Labs
İstanbul, Türkiye

Şuayb Ş. Arslan
MEF Üniversitesi
İstanbul, Türkiye

***Özetçe* —Blokzincir, toplumumuzun birbiriyle iletişim kurma ve ticaret yapma biçiminde devrim yapma potansiyeline sahip, yakın zamanda ortaya çıkmış olan bir teknolojidir. Bu teknolojinin sağladığı en önemli avantaj aracı gerektiren bir oluşumda güvenilir bir merkezi kuruma ihtiyaç duymadan değer taşıyan işlemleri değiş tokuş edebilmesidir. Ayrıca, veri bütünlüğü, dahili orijinallik ve kullanıcı şeffaflığı sağlayabilir. Blokzincir, birçok yenilikçi uygulamanın temel alınacağı yeni internet olarak görülebilir. Bu çalışmada, genel çalışma prensibi, oluşan fırsatlar ve ileride karşılaşılabilecek zorlukları içerecek şekilde güncel blokzincir teknolojilerinin genel bir görünümünü sunmaktayız.**

Anahtar Kelimeler—Blokzincir, kriptoloji, mutabakat.

***Abstract*—Blockchain is a recently emerging technology that has the potential to revolutionize the way society interacts and trades between each other. The main advantage this technology provides is its ability to exchange transactions without relying on a trusted third party entities of any means. It can also provide data integrity, in-built authenticity and user transparency. Blockchain is envisioned to be the new internet on which many revolutionary applications will be based. In this work, we provide an overview of blockchain technologies including their principles, opportunities and challenges ahead.**

Keywords—Blockchain, cryptogtaphy, consensus.

I. Giriş

İnternet, sosyal ağlar, arama motorları veya bulut kullanımı gibi son derece merkezi platformların oluşturulmasını sağladı. Ancak bu merkezileşme, kişilerin kişisel verilerini, bu platformların sahipleri tarafından potansiyel ticari veya yanlış kullanıma maruz bırakabilmektedir. Blok zincir teknolojileri, İnternet gibi dağınık bir ağ üzerinde, aracılar olmaksızın ticari işlemleri ve diğer verileri güvenli bir şekilde paylaşmak, depolamak ve güvence altına almak için merkezi olmayan yöntemleri desteklemektedir. Ayrıca blok zincirlerine dayanan merkezsizleştirme mekanizmaları, kullanıcıların kendi kişisel verileri üzerinde tam kontrol sahibi olmalarını da sağlayabilir.

Bir Block Chain - Blok Zincir (BC), birbirine zincirlenerek giderek artan blokların kayıtlarını tutan dağıtılmış bir veritabanıdır. BC Bitcoin'in altında yatan teknoloji olarak olarak öne çıkmakta olup ve Satoshi Nakamoto tarafından bitcoin özelinde kullanılmıştır [1]. Ayrıca, Bitcoin'in işlem, depolama ve para basılması için bütün referans kodunu yayınladı. Yazılımı da açık olarak bütün dünyanın kullanımına sundu. Bitcoin tarafından kullanılan teknikler üzerine Zerocash [2] veya Ethereum [3] gibi diğer blok zincirler inşa edilmiş ve bitcoin'in sunduğu avantajlar daha iyi mahremiyeti sağlama veya daha anlamlı akıllı sözleşmeler (smart contracts) üretme olarak genişletilmiştir. Telekomünikasyon alanında Orange, Verizon gibi telekom operatörleri son birkaç yıldır blok zinciriyle ilgili projelere, yeni teşebbüslere, prototiplere ve çerçevelere yatırım yapmaktadır [4]. British Telecom ve AT&T, telekomda blok zincirle ilgili birçok patente başvurusu ve IBM, Microsoft gibi şirketler telekomünikasyonu blok zincirle birleştiren ve telekomünikasyon için blok zincir tabanlı platform ve servislerde çalıştıklarını gösteren önemli sanayi olaylarını gösterdiler [4]. Hyperledger projesi açık kaynaklı dağıtık defter çerçevesi oluşturmayı hedefleyebn başka bir BC teknolojisidir [5]. Türkiye'de Bankalararası Kart Merkezi (BKM) tarafından Blok zincir teknolojilerinin detaylarını anlatan "Blockchain 101" kitabı [6]'de yayınlanmıştır.

Bu bildiride, BC teknolojilerine genel bakış, çalışma prensipleri, faydaları ve eksik kısımları tartışılmıştır. Bildirinin geri kalanı şu şekilde organize edilmiştir. Bölüm II'de blok zincir teknolojilerine genel bakış, mimari yapı, çalışma prensibi ve ağ türleri özetlenmiştir. Bölüm III'de BC teknolojilerinin genel gereksinimleri, faydaları ve mevcut eksiklikleri özetlenmiştir. Son olarak Bölüm IV'da ise bildirinin sonuçları verilmiştir.

II. Blok Zincir Teknolojilerine Genel Bakış

A. Genel Yapıtaşları ve Çalışma Prensibi

BC'in temel yapıtaşları şu şekildedir: **(i) Paylaşılan defter**- Yalnızca eklenmiş paylaşılan kayıt sistemi ilgili tüm katılımcılara işlem görünürlüğü sağlayan iş ağı. Blok zincir mimarisi, katılımcıların, bir işlem gerçekleştiğinde eşler arası çoğaltma ile güncellenen bir defteri paylaşmasına olanak tanır. **(ii) Akıllı Sözleşme** - İş terimleri, işlem veritabanında tutulmaktadır ve bir işlem gerçekleştiğinde uygun sözleşmelerle yürütülmesi şeklinde çalışır. Ağ katılımcıları, işlemlerin doğrulanması için mutabakatta bulunur veya benzeri mekanizmalar yoluyla ağın doğru kalmasını sağlar. **(iii) Gizlilik** - İşlemler güvenilir, doğrulanmış ve doğrulanabilir şekilde depolanır. **(iv) Güven**

978-1-5386-1501-0/18/$31.00

- İşlemler ilgili katılımcılar tarafından onaylanır. **(v) Şeffaflık** - Ağdaki tüm katılımcılar, onları etkileyen tüm işlemlerden haberdardır. **(vi) Kriptografi-** Ağ katılımcılarının yalnızca ilgili defterin bölümlerini görmesi ve işlemlerin güvenilir, doğrulanmış ve doğrulanabilir olmasını sağlamaya yardımcı olmak için kullanılır.

BC'nin genel çalışma prensibi Şekil 1'de numaralandırılarak verilmiştir. Öncelikle A ve B kurumları veya bireyleri birbirleri arasında işlem yapmak istediklerinde (**adım 1**), bu yapılacak işlemin tüm detayları bloklar halinde tutulur (**adım 2**). BC'deki tüm işlemleri bloklar halinde ele alır ve her bir blok öncelikle belirli kurallara göre oluşturulur. Birden çok işlem, doğrulanmış işlem blokları halinde birleştirir. Bu blok yapısında nonce (tek seferlik anahtar), bir önceki BC başlığının hash bilgisi, blok işlemlerinin Merkle ağacının kök hash'i, zaman bilgisi ve zorluk derecesi (çalışma belgesi-proof of work) bulunmaktadır [7]. Oluşturulan bu blok tüm dağıtık kayıt defterinin içerisindeki tüm uç noktalara yayılmaktadır (**adım 3**). Bir başka deyişle, BC içerisinde yer alan her uç nokta başlangıçtan itibaren tüm kayıtların kopyasını tutar. A ile B arasındaki işlemin gerçekleştiğini diğer uç noktalar doğrular (**adım 4**). BC içerisinde bulunan tüm uç noktalar birbirleri ile iletişim halinde bulunarak sistemde herhangi bir bozukluk olup olmadığını teyit ederler. Eğer BC yapısında herhangi bir halkada bir değişiklik olursa zincir kırılır ve sistem geneli mutabık olmaktan çıkar. Kırık halka dağıtık kayıt defteri ağından çıkarılana kadar ve kalan halkalar üzerinde mutabık kalınmasına kadar sistemin çalışması bozulur. Yeni bir blok yapısı oluştuğunda ise bir önceki bloğun kriptografik özeti alınarak (i.e. hash fonksiyonundan geçirip) ikinci bir blok yapısı oluşturulur ve bir zincir yapısı oluşturulur (**adım 5**). Bu şekilde her bir blok yapısı geldiğinde bir önceki yapının özeti ile ilişkili olacak şekilde tüm bir zincir yapısına sahip olur. Daha sonra A ve B arasında işlem gerçekleşmiş olur (**adım 6**). Örneğin, BC teknolojilerinin en önemli örneği olan Bitcoin'deki BC yapısında Blok üreticisi (Madenciler) bloğundaki tüm işlemlerin geçerli olmasını sağlar. Madencilerin önemli bilgi işleme gücü vardır. En yüksek bilgisayar gücüne sahip olan madencinin kazanma ihtimali yükselir ve blok zincirin sonuna eklenir. Daha fazla yeni para basmasına ve bunları tutmasına izin verilerek madenci de ödüllendirilmiş olur. Bilgiyi işleme gücü kanıtı ise verilen bir bulmacayı çözme kapasitesiyle orantılıdır. En yüksek birikimli zorluk derecesine sahip zincir, ana zincir olarak seçilir.

III. Blok Zincir Genel Gereksinimler, Fırsatlar ve Mevcut Zorluklar

A. Genel Gereksinimler ve Fırsatlar:

Blok Zincir tabanlı teknolojilerin yaygın olarak kullanılması için bazı gereksinimleri şunlardır: (i) toplum, çevre ve / veya ekonomi üzerindeki olumlu etki yaratması ve büyük bir vatandaş topluluğu tarafından kabulü gerçeklenmesi (ii) şeffaflık, hesap verebilirlik, gizlilik (iii) Tümsel bir çözüm olarak kullanılabilirlik ve kapsayıcılık (iv) teknik kısıtlamaların çözülmesi konusunda büyük ölçekte yaşama kabiliyeti. (v) geleneksel yapıdan daha etkili ve verimli çözümler üretmesi.

Bitcoin ile insanların hayatına giren blok zinciri teknolojisi ödeme işlemleri dışında insanların kafasında net bir uygulama alanı olarak yer etmemektedir. Bitcoin'in finansal bir çözüm olması ve blok zincirinin ilk, en uzun ömürlü ve anlaşılabilir uygulaması olması insanların blok zincirini bitcoin ile sınırlı olarak düşünmesine yol açmaktadır. Ancak blok zinciri teknolojisi çok daha fazla kullanım alanı, imkanı olan ve sektöre hitap eden bir teknolojidir [8] [9].

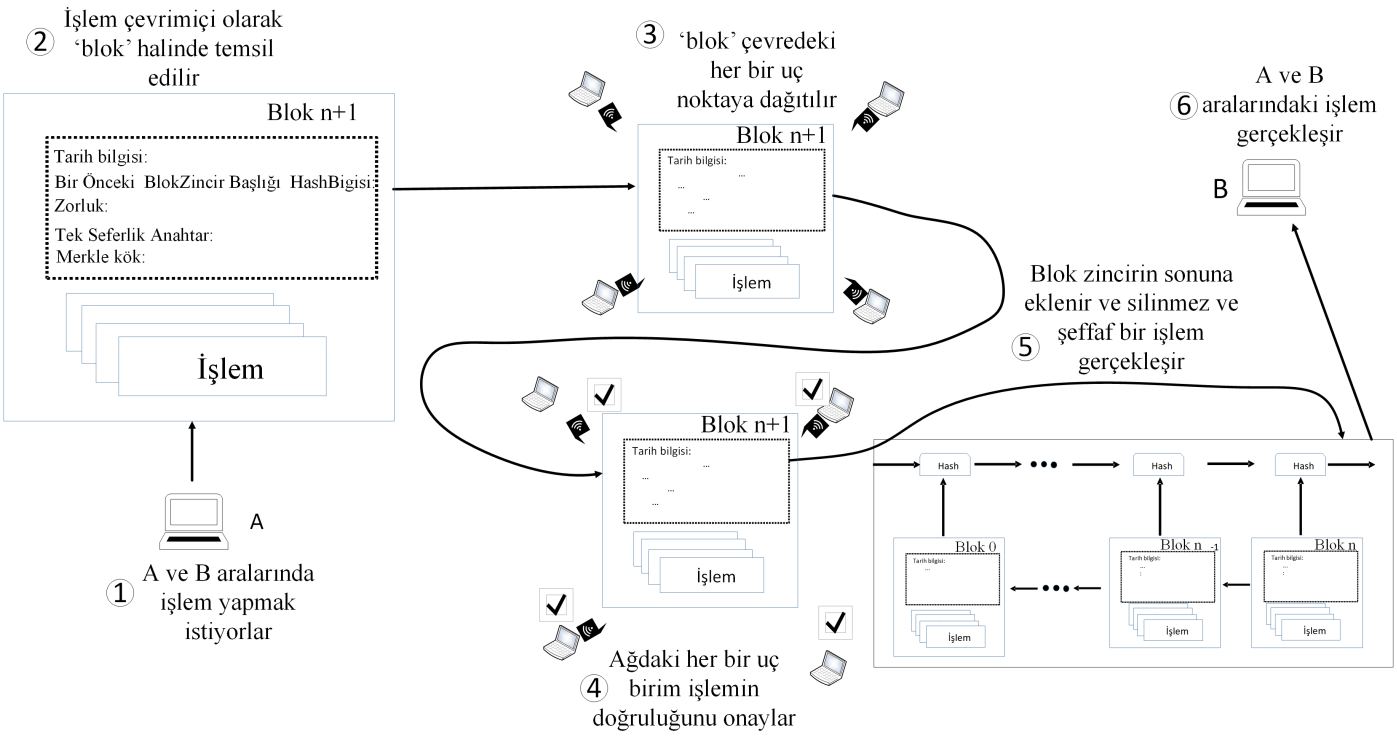
Blok zincirinden yararlanma imkanı olan bankacılık, telekom, enerji, veri depolama, hukuk ve devlet yönetimleri gibi birçok sektör bulunmaktadır. Blok zincirinin o alana değer sağlaması için Şekil 2'deki faktörlerden bir ya da birkaçını içermesi gerekmektedir. Eğer bir sektör aşağdaki şartlardan birindeyse blok zinciri teknolojisinden faydalanması yüksek ihtimaldir:

Sahtekârlık: Çeşitli işlemlerde yer alan dolandırıcılık aktivitelerine müsait bir ortam söz konusuysa, blok zinciri teknolojisi dolandırıcılık olasılığını azaltmaya yardımcı olabilir. İçerdiği kriptografik algoritamalar, dağıtık şekilde tutulan bilgiler ve onaylama adımları sayesinde yapılan sahtekârlıklar sisteme dâhil olmaz. Böylece yapılan işlemlerde sahtekarlık ihtimali ortadan kaldırılmış ve güven, blok zinciri sayesinde sağlanmış olacaktır. Bu yüzden blok zinciri uluslararası finansal işlemlerde kullanmak için bir çok önemli banka tarafından tercih edilmektedir [10].

Aracılık: Aracıların bulunduğu bir ortam söz konusuysa ve aracıların değer zincirine katkısı yoksa, blok zinciri teknolojisi sayesinde aracıları devre dışı bırakmak mümkündür. Özellikle paylaşım ekonomisi diye adlandırılan Uber ve Airbnb gibi şirketlerin öncü olduğu sektörlerde bu aracıları aradan çıkararak hem hız hem de tasarruf sağlamak mümkün olacaktır [11].

İşlem Sayısı: Blok zinciri teknolojisi saniyede yüksek işlem hızı gerektiren yerlerde de kullanılmaya uygun altyapı sunar. Özellikle nesnelerin interneti (IoT) ya da akıllı cihazların haberleşmesi konusunda büyük katkı yapması öngörülmektedir. Gelecekte yapay zeka ile çalışan araçlar ve eşyalar birbirleri arasında iletişim kurarken ve yerine göre ticaret yaparken çok sayıda işlem gerçekleşecektir. Bu da günümüz sistemleri ile özellikle işlem hızı açısından karşılanması zor gereksinimler oluşmasına neden olacaktır. Hem doğrulama hem de işlemin kendisinin hızlı yapılmasını sağlayacak altyapı blok zinciri teknolojisinde mevcuttur [12]. Blok zinciri teknolojisinin ilk uygulamalarından olan Bitcoin hız olarak şu anda mevcutta kullanılan ödeme teknolojilerine göre yavaş kalmaktadır. MasterCard ve Visa saniyede 80.000 işlem işleme kapasitesine sahipken, Bitcoin yalnızca saniyede 7 işlem yapabilir [13]. Ancak bu teknoloji üzerine yapılan çalışmalar sayesinde gelecekte saniyede işlenen işlem sayısı 400.000 kata kadar çıkabilecektir ve bu da beraberinde çözülmesi gereken güvenik problemleri oluşturmaktadır [14]. Doğrulama protokolleri değiştirilerek de güvenlik açısından problem yaratmadan yüksek hızda işlemler gerçekleştirmek mümkündür. Örneğin IOTA adı verilen Tangle teknolojisini kullanan bir çözüm mevuttur. Bu çözüm işlemleri doğrulamak için blok zincirinde ilk işleme kadar gitmek yerine doğrulanması gereken işlemden önce gerçekleşmiş 2 işlem üzerinden doğrulama yaparak saniyede işlenebilen işlem sayısının çok fazla artmasını sağlamaktadır [15].

Stabil Veri: Bir blok zinciri uygulaması için anlık değişen uçucu veriler kullanılması bu teknolojinin katacağı değeri sınırlı kılmaktadır. Bu yüzden verilerin stabil olduğu, kısa zaman dilimleri içinde değişmediği sektörlerde blok zinciri



Şekil 1: Blok Zincir Yapısı ve Çalışma Prensibi

teknolojisi sayesinde yok edilemez, değiştirilemez, şeffaf ve dağıtık depolanabilen bilgileri tutmak mümkün olacaktır. Tapu sahipliği, kişisel veriler, sağlık kayıtları, dijital haklar ve IP hakları gibi güvenli tutulması gereken ve kısa zamanda değişmeyen veriler için çok uygun kullanım alanları sunmaktadır [16].

Bununla beraber, gelecek nesil Internet of Things - Nesnelerin Interneti (IoT), mobil ve akıllı sistemler için otomatik ve güvenli bir işlem altyapısı sağlayabilirken, kripto para birimleri madenciliğine gönüllü olarak katılım, yeni dağıtılmış hizmetlerin geliştirilmesine izin verebilir (ör. şu anda mevcut "ücretsiz hizmetler" in çoğunun kullandığı reklam tabanlı gelir sistemi).

Şekil 2: Blok Zinciri Teknolojisinin Değer Sağlaması için Gerekli Faktörler

B. BC içerisindeki Mevcut Zorluklar:

Güvenlik ve Gizlilik Geleneksel yöntemler ile sağlanan güvenlik ve mahremiyet koruma yöntemleri, merkeziyetçi olarak çalışmakta olup, bütün sisteme giriş yapan tüm cihazların tanımlanması, doğrulanması, yetkilendirilmesi ve merkezi bulut sunucuları aracılığıyla bağlandığı merkezi aracılıklı iletişim modellerine dayanır. Bu model ise çok fazla sayıda bağlantı sağlandığı durumlarda ölçeklenebilir bir çözüme doğru ilerlemez. Bununlar beraber bulut sunucuları genellikle tek bir başarısızlık noktası (single-point-of-failure) olmayı sürdürebilir. Ayrıca şu anki güvenli iletişim mimarilerinde, genellikle kullanıcının izni olmaksızın tüm verilerin değiştirilmesi veya gürültülü veya özetlenen verilerin talep sahibine açılması gibi gizlilik için önemli açıklar yaratabilecek işlemler de düşünülmemiştir [16].

Veri büyüklüğü: Tarihsel verilerin arşivlenmesi ile birlikte zaman ilerledikçe bu saklanan veriler büyür ve sürdürülemez olurlar. Veri arşivlenmesi için farklı yöntemler blok zinciri ekosistemindeki çeşitli oyuncular tarafından araştırılmaktadır [17].

Regulatif boşluk: Herhangi bir yeni teknolojide sıkça olduğu gibi, blok zincir mevcut yönetmeliklerden ve çerçevelerinden daha hızlı bir oranda geliştirilmekte ve uygulanmaktadır. Bu nedenle, dijital ve akıllı sözleşmeler için hem güvenlik hem de gizlilik bakımından farklı yöntemler sunan blok zinciri teknolojsi kullanıcılarının anlaşmalarının uygulanması için net bir düzenleme veya şartname mevcut değildir. Bu amaçla düzenleyici otoriteler, bir çok alanda potensiyel kullanım alanı bulunan blok zincir gibi teknolojilerin daha hızlı uygulanması için veri koruma kanunları gibi esnek yasal ve şartname çerçevelerini de etkinleştirmelidirler. [18].

BC güvenlik, değiştirilemezlik ve mahremiyet gibi özellikleri sayesinde yukarıda bahsedilen zorlukları çözmek için yararlı bir teknoloji olabilir.

C. Mevcut Eksiklikler:

Takip Edilemeyen Para Akışı: BC teknolojisine dayalı Bitcoin gibi teknolojilerde yapılan islemleri takip etmek nitekim yapılabiir. Lakin "Monero" gibi kripto paralar "takip" meselesini neredeyse imkansıza yaklaştırmıştır [19]. Ayrıca, Bitcoin farklı yasadışı işlemler için de kullanılmaya açık bir teknolojidir. Özellikle Deepweb'de illegal işlemlerde ödeme aracı olarak kullanılmasından dolayı otoritelerin tepkisini çekmiş ve gerçekten vaat ettiği değerin anlaşılmasına uzun bir süre gölge düşürmüştür [20]. Şu anda teröre finansman sağlama ve illegal işlerde kullanılmasını engelleyecek bir çözüm malesef bulunmamaktadır. Bankacılığın en büyük problemlerinden olan kara para aklama kripto paralardan bağımsız olarak yıllardan beri süregelen bir problemdir [21]. Yine de vaat ettiği değer düşünüldüğünde ve şu andaki reel para birimlerinin de bu tür illegal işlerde kullanıldığını düşündüğümüzde, teknolojinin gelişmesi ile birlike devletlerin reel para birimi ve kripto para dönüşümü sırasında gerekli kontrollerle bu sorunların aşılması beklenmektedir [22]. Ayrıca, devletler kripto paranın inşanlara sağladığı kazançlardan da vergi almak istemektedirler. Şu anda altın, hisse senedi ve bono gibi finansal yatırım araçlarından, kişilerin bu araçlardan sağladıkları kazanca göre vergi alınmaktadır. Ancak kripto paralarda henüz bir regülasyon olmadığı için devletlerin vergi alması zorlaşmaktadır. Bu da hem insanların regüle olmamış bu teknolojinin sağladığı avantajlardan uzak durmasına hem de devletlerin sert yaptırımlarına maruz kalmasına neden olabilir [23].

Karmaşık teknoloji: Blok zinciri teknolojisi öncelikle merkezi yetkisi olmayan dağıtılmış bir sistemdir ve hali hazırda çok karmaşık bir sistem üzerine kurulmuştur. Kriptografi, dağıtık veri tutma ve madencilik gibi anlaşılması zor ve uzmanlık gerektiren konuları içermektedir. Bunun adaptasyonu da teknolojinin karşmaşıklığından dolayı zor olacaktır. Böyle bir altyapıyı telekomunikasyon veya banka sistemleri gibi karmaşık sistemleri olan yapılarla entegre etmek ölçeklenebilirlik, hizmet kalitesi, güvenlik ve hata toleransı bakımından ciddi sorunlar yaşatabilir.

IV. Sonuçlar

Blokzincir merkeziyetçi bir kuruma veya herhangi bir aracıya ihtiyaç duymadan bireyler veya kurumlar arasında işlemlerin güvenli bir şekilde sağlanmasını sağlayan bir teknolojidir. Blok zincirinden yararlanma imkanı olabilecek birçok farklı sektör de bulunmaktadır. Bu bildiride son yıllarda önem kazanan ve ileriye yönelik teknolojilerde yer alması olası blok zincir teknolojisinin çalışma prensiplerini, fırsatlarını, zorluklarını ve mevcut eksikliklerini sunduk.

Kaynakça

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

[2] E. B. Sasson, A. Chiesa, C. Garman, M. Green, I. Miers, E. Tromer, and M. Virza, "Zerocash: Decentralized anonymous payments from bitcoin," in *Security and Privacy (SP), 2014 IEEE Symposium on*, pp. 459–474, IEEE, 2014.

[3] G. Wood, "Ethereum: A secure decentralised generalised transaction ledger," *Ethereum Project Yellow Paper*, vol. 151, 2014.

[4] A. Lannquist, "Blockchain in enterprise: How companies are using blockchain today." https://blockchainatberkeley.blog/a-snapshot-of-blockchain-in-enterprise-d140a511e5fd. [Internet üzerinden; 31-Mart-2018'de erişildi].

[5] Hyperledger, "Hyperledger Project (HLP)- Blockchain Technologies for Business." https://www.hyperledger.org/, 2017. [Internet üzerinden; 31-Mart-2018'de erişildi].

[6] A. Usta and S. Dogantekin, "Block chain 101." http://www.bkm.com.tr/wp-content/uploads/2017/05/blockchain-101.pdf, 2017. [Internet üzerinden; 31-Mart-2018'de erişildi].

[7] A. M. Antonopoulos, *Mastering Bitcoin: unlocking digital cryptocurrencies*. " O'Reilly Media, Inc.", 2014.

[8] D. Tapscott and A. Tapscott, *Blockchain revolution: how the technology behind bitcoin is changing money, business, and the world*. Penguin, 2016.

[9] S. Underwood, "Blockchain beyond bitcoin," *Communications of the ACM*, vol. 59, no. 11, pp. 15–17, 2016.

[10] S. Williams, "5 big banks currently testing ripple's blockchain." https://www.fool.com/investing/2017/12/17/5-big-banks-currently-testing-ripples-blockchain-t.aspx, 2017. [Internet üzerinden; 31-Mart-2018'de erişildi].

[11] A. Pazaitis, P. De Filippi, and V. Kostakis, "Blockchain and value systems in the sharing economy: The illustrative case of backfeed," *Technological Forecasting and Social Change*, vol. 125, pp. 105–115, 2017.

[12] M. Conoscenti, A. Vetro, and J. C. De Martin, "Blockchain for the internet of things: A systematic literature review," in *Computer Systems and Applications (AICCSA), 2016 IEEE/ACS 13th International Conference of*, pp. 1–6, IEEE, 2016.

[13] R. Vlastelica, "Why bitcoin won't displace visa or mastercard soon." https://www.marketwatch.com/story/why-bitcoin-wont-displace-visa-or-mastercard-soon-2017-12-15, 2017. [Internet üzerinden; 31-Mart-2018'de erişildi].

[14] M. Vukolić, "The quest for scalable blockchain fabric: Proof-of-work vs. bft replication," in *International Workshop on Open Problems in Network Security*, pp. 112–125, Springer, 2015.

[15] S. Popov, "The tangle." https://iota.org/IOTA_Whitepaper.pdf, 2017. [Internet üzerinden; 31-Mart-2018'de erişildi].

[16] G. Zyskind, O. Nathan, *et al.*, "Decentralizing privacy: Using blockchain to protect personal data," in *Security and Privacy Workshops (SPW), 2015 IEEE*, pp. 180–184, IEEE, 2015.

[17] X. Liang, S. Shetty, D. Tosh, C. Kamhoua, K. Kwiat, and L. Njilla, "Provchain: A blockchain-based data provenance architecture in cloud environment with enhanced privacy and availability," in *Proceedings of the 17th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing*, pp. 468–477, IEEE Press, 2017.

[18] G. W. Peters, E. Panayi, and A. Chapelle, "Trends in crypto-currencies and blockchain technologies: A monetary theory and regulation perspective," *arXiv preprint arXiv:1508.04364*, 2015.

[19] A. Kumar, C. Fischer, S. Tople, and P. Saxena, "A traceability analysis of monero's blockchain," in *European Symposium on Research in Computer Security*, pp. 153–173, Springer, 2017.

[20] D. Stroukal *et al.*, "Bitcoin and other cryptocurrency as an instrument of crime in cyberspace," in *Proceedings of Business and Management Conferences*, no. 4407036, International Institute of Social and Economic Sciences, 2016.

[21] P. Reuter, *Chasing dirty money: The fight against money laundering*. Peterson Institute, 2005.

[22] P. Twomey, "Halting a shift in the paradigm: The need for bitcoin regulation," *Trinity CL Rev.*, vol. 16, p. 67, 2013.

[23] O. Y. Marian, "Are cryptocurrencies' super'tax havens?." http://scholarship.law.ufl.edu/facultypub/358, 2013. [Internet üzerinden; 31-Mart-2018'de erişildi].

Content Addressed P2P File System for the Web with Blockchain-Based Meta-Data Integrity

Chaitanya Rahalkar
Department of Computer Engineering
Pune University
Pune, India
chaitanyarahalkar4@gmail.com

Dhaval Gujar
Department of Computer Engineering
Pune University
Pune, India
dhvlgjr@gmail.com

***Abstract*—With the exponentially scaled World Wide Web, the standard HTTP protocol has started showing its limitations. With an increased amount of data duplication & accidental deletion of files on the Internet, the P2P file system called IPFS completely changes the way files are stored. IPFS is a file storage protocol allowing files to be stored on decentralized systems. In the HTTP client-server protocol, files are downloaded has from a single source. With files stored on a decentralized network, IPFS allows packet retrieval from multiple sources, simultaneously saving considerable bandwidth. IPFS uses a content-addressed block storage model with content-addressed hyperlinks. Large amounts of data can is addressable with IPFS with the immutable and permanent IPFS links with meta-data stored as Blockchain transactions. This timestamps and secures the data, instead of having to put it on the chain itself. Our paper proposes a model to use the decentralized file storage system of IPFS, and the integrity preservation properties of the Blockchain, to store and distribute data on the Web.**

***Index Terms*—IPFS, Blockchain, decentralized Systems, Peer-To-Peer Systems**

I. Introduction

With the ever-expanding World Wide Web, the data generated on the web has grown vastly. The amount of data generated daily is at a staggering 2.5 quintillion bytes. [1] This pace is gaining constant momentum due to the inclusion of new IoT devices every day. Sensory data produced by IoT devices get bulkier as modern devices are added to the Internet. To counter the problem of data handling, many distributed file systems were introduced. The popular ones being Napster, BitTorrent, KaZaA, supporting millions of distributed users. Among all of them, HTTP - one of the oldest protocols on the Internet is the biggest distributed file system, when coupled with browsers allowing users to share files globally. With the increase in the scalability of the World Wide Web, the reliability of HTTP began to degrade. Keeping track of terabytes of data and moving these files over the Web is a difficult task. Several other protocols were introduced to tackle the problem of scalability and decentralization with an intention of replacing the well-established reign of HTTP. The other problem with HTTP is security and data integrity. As a countermeasure, the inclusion of Blockchain technology was introduced. Blockchain technology cannot be used to store the entire data due to its distributed ledger protocol. [3] This protocol states that every node in the Blockchain must preserve a copy of the data, on the chain. Hence, storing petabytes of data on the Blockchain is infeasible. This model proposes to store only the file metadata, summarizing necessary information about data, on the Blockchain. This data, being in bytes for a single file, reduces the overall size of the ledger. [8]

II. Motivation

IPFS (InterPlanetary File System) is a proposed protocol that enhances HTTP. [2] We are entering the era of data distribution with new challenges like:

- Hosting petabytes of datasets
- Computing large data across organizations
- High volume, high definition, on-demand, real-time streaming of data
- Versioning and linking of massive datasets
- Preventing accidental disappearances of important files

To tackle all these problems, HTTP does not provide a scalable solution. Adding the middleware of Blockchain technology for preserving the file metadata helps maintain the integrity of the files that are stored. Blockchain technology induces its peculiar characteristics of data integrity, data security, and transparency to this file system. [7] Blockchain technology is a distributed ledger system that will preserve all the file metadata, including file size, author information, checksums, date of creation and modification, etc. To summarize, the distributed technology of IPFS and the data integrity feature of the Blockchain to preserve file-related information creates a full-fledged data serving model for the Internet. [6]

III. History

The origin of the IPFS protocol dates back to the time when the DHT (Distributed Hash Table) was created. [5] The backbone of IPFS relies on the DHT protocol. It is a key-value store that uses distributed technology to store data. Key distribution takes place among nodes using a deterministic algorithm. Each node is assigned a portion of the hash table and it stores only the assigned data in the hash table. It uses advanced routing algorithms for data retrieval. The main disadvantage of DHT is data integrity and privacy. Since every node does not have a copy of all the data stored on the network, downtime of specific nodes may lead to data loss

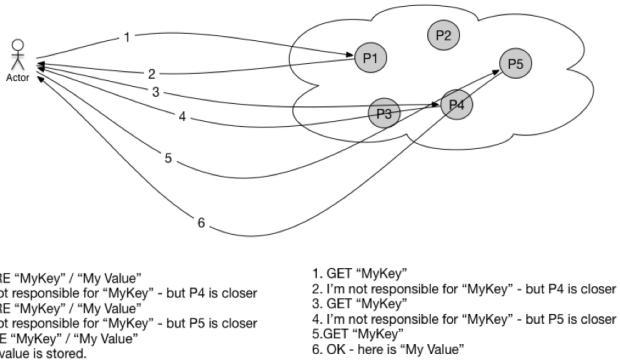


Fig. 1. Distributed Hash Tables

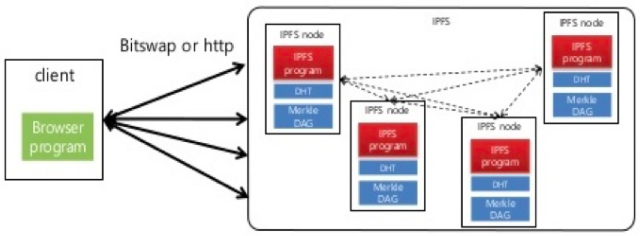


Fig. 2. Content Addressed Filesystem

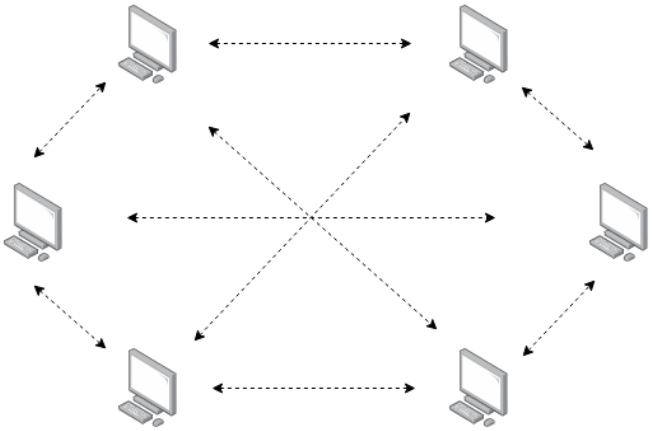


Fig. 3. Peer to Peer Network

or non-availability of data. Also, the security of the data is compromised since data in the DHT nodes is not protected. Hence, Blockchain technology serves as an additional layer above DHT. In the Blockchain, copies of all the metadata of the files stored on the IPFS will be with every node. IPFS is a proposed replacement for the existing HTTP protocol.

IV. Theoretical Concepts of Implementation

A. Brief implementation

The IPFS is a distributed system similar to the BitTorrent protocol. Data is broken down into pieces and distributed across nodes in the network. The data element is assigned a unique IPFS hash that acts as an identifier for the file. The file is accessible via the IPFS hash. Every node in the IPFS network has its own IPFS daemon that communicates with other nodes in the network. When a file is uploaded to the network, a unique IPFS hash of the file is created, and uploaded to the Blockchain network. Along with that, file-related metadata like author information, file size and the file type is also uploaded.

Retrieval of files from the network is done via the network DHT. The network DHT uses advanced routing algorithms, beyond the scope of this paper, that find the data in log(n) time complexity, where n is the number of nodes in the network. After retrieving the data, the hash of the file is generated. The hash is searched on the Blockchain network, and if found, the metadata is retrieved from the network. This metadata is compared with the metadata of the file. Any mismatch indicates manipulation or file corruption without authorized permission. [10] The entire model is composed of four essential terminologies:

1) Distributed Hash Tables: A distributed hash table (DHT) is a type of a decentralized distributed system that works on the lookup mechanism similar to the hash table data structure. Key-value pairs are stored in a DHT, and nodes which are a part of the distributed system can efficiently retrieve the value associated with a given key. Keys are identifiers which map to particular values which in turn can be addresses, documents or arbitrary data. This allows a DHT to scale horizontally and handle continuous node arrivals, departures, and failures.

2) Content Addressed File System: Everything on the World Wide Web is addressed with a URL that maps to the location of the file on the Internet. The IP address assigned to a website locates the file on the WWW. However, in a Content Addressed File System, the file is accessed based on its content, and not on its location. [9]

3) Blockchain Technology: Blockchain technology is a decentralized system similar to a ledger, a continuously-growing list of records without the possibility of tampering and revision. In a Blockchain, each node of the network stores the entire ledger data. So, the Blockchain mechanism differs completely from DHT, in which data is distributed among nodes. Every new transaction entering the Blockchain is validated using a process called mining.

4) Peer-to-Peer Protocols: Peer-to-Peer computing is a network in which each workstation has equivalent capabilities and responsibilities. They are typically situated physically near to each other and run similar networking protocols and software. Peers in a P2P network make a portion of their resources, available for other network participants, without the need for a central server. The P2P architecture is designed around the notion of equal peer nodes, i.e., peers are both suppliers and consumers of resources, simultaneously functioning as both ”clients” and ”servers” to the other nodes on the network.

Following are the components of the P2P Model:

1) Identities: Nodes are identified by a Node Id, the cryptographic hash of the node’s public-key, generated with

S/Kademlia's crypto puzzle. Nodes store their public and private keys (encrypted with a passphrase). The Node Ids can be regenerated per daemon initialization.

2) Network: IPFS nodes communicate regularly with hundreds of other nodes in the network, potentially across the Internet
3) Routing: It is the mechanism to maintain information about location of specific peers and objects. The routing mechanism responds to both local and remote queries.
4) Exchange: It is a novel block exchange protocol also called - BitSwap, that governs efficient block distribution.
5) Objects: Every file in the P2P network is considered as a blob. This blob is made addressable with a Merkle DAG (Directed Acyclic Graph) of content-addressed immutable objects with links.

B. Need for Implementation

The exponential growth rate of the World Wide Web has caused HTTP to start showing its limitations. There is a need to reinvent the protocol. Following are some of the limitations of HTTP:

1) HTTP is highly inefficient and costly: HTTP downloads a file from a single computer at a time, instead of getting segments from multiple computers simultaneously. With video delivery, a P2P approach has the ability to save 60% in bandwidth costs. IPFS allows distribution of high volumes of data, effectively.
2) The web's centralization limits opportunities: The Internet has been accelerating innovation and has levelled the playing field. However, the increasing unification of control is a threat to this model.
3) Humanity's history is deleted daily: IPFS stores a versioned history of files and makes it simple to set up robust networks for mirroring of data.
4) Preservation of data integrity with Blockchain: Data tampering chances are incredibly high in the case of sensitive data. With the Blockchain middleware, data becomes immutable. Any attempt to change the metadata results in an invalid block, detecting the problem immediately.

C. Application Scope

With the current demands of the industry, IPFS and Blockchain is a perfect pair to perform scalable and fault-tolerant tasks. Following are some of the use cases of this model:

1) Preservation of Massive Datasets: With the distributed technology, the model allows people to store large datasets, showing fast performance with decentralized archiving system. Along with that, the integrity of the datasets can be preserved.
2) Sensitive Data Storage: Sensitive government documents, bond papers, contracts, etc. can be securely and safely stored with this model, avoiding cases of fraud. [12]

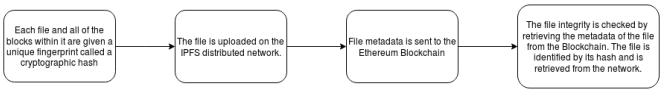


Fig. 4. Model Flow

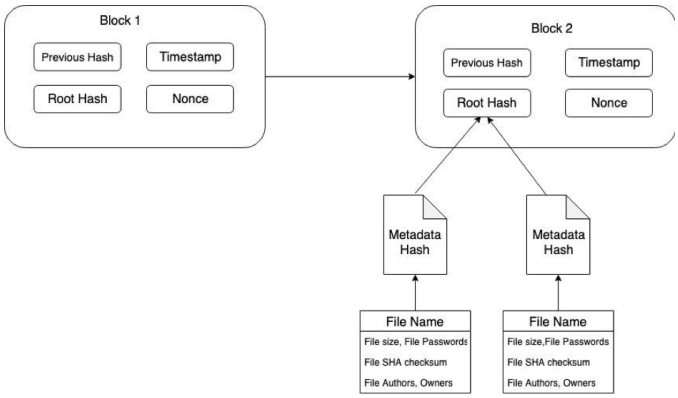


Fig. 5. Contents of Blockchain

3) Content Delivery: Secured P2P content delivery saves millions in bandwidth, also providing better performance.

V. Model Flow

Figure 4 explains the flow of the model. When a file/folder is uploaded to the P2P network, a specific hash of the form "Qm…" is generated for the file and all the files within a folder. The files are uploaded on the P2P network, and the DHT is updated. These content addressed files can now be accessed via their unique hash. A proposed naming system called InterPlanetary Naming System (IPNS) analogous to the DNS is used to map file names to their unique hashes. The metadata of the file is then uploaded to the Blockchain. The Proof-of-Stake system of Ethereum [4] [13] is used for incentivizing. The authenticity and integrity of the file is verified by retrieving the hash and its data from the Blockchain.

Figure 5 shows the contents of the block in the Blockchain. Every block has a Merkle DAG(Directed Acyclic Graph) structure. The block comprises of four elements - The root hash, created from all the metadata hashes, the root hash of the previous block, the timestamp at which the block was mined, and the nonce. (used in the Proof-of-Work system)

TABLE I
File Metadata Stored on the Blockchain

	File creation date	IPFS Hash Value	File Access Date	File Access Time	IPFS Hash of Modified File
1	12/06/18	$Qm....2$	14/06/18	$12:45PM$	$Qm....9$
2	16/06/18	$Qm....19$	19/06/18	$1:05PM$	$Qm....25$

Table I shows the example file metadata. It comprises of parameters like file creation date, file creation time, access date, access time and other related file parameters.

VI. Implementation Details

1) IPFS(Interplanetary File System) has a framework created for interfacing with the P2P network. This framework allows one to interact with the IPFS network. It connects with the IPFS daemon and translates and transfers requests via the TCP socket that it initiates.
2) An uploaded file is transferred to the IPFS network via the TCP socket. At the same time, file metadata is sent to the Ethereum Blockchain. A sample Blockchain entry is as shown in Table I
3) The IPFS gives a content-based address that can distinctively identify the file on the network.
4) When a file is to be retrieved from the network, the file integrity is verified by retrieving the metadata from the Blockchain. Each file is uniquely identified by a hash. The hash of the current file is verified with the one stored on the Blockchain. This validates the integrity of the file.

VII. challenges

Despite the consistency of the IPFS protocol, a few issues are yet to be fully resolved. Firstly, the content addresses generated on IPNS are not human-friendly. These links can be shortened to easy-to-remember names using a Domain Name System (DNS), but this has the possibility of introducing an external point-of-failure for distribution of content. Many reports have suggested that IPNS can be slow at domain name resolution, with delays of up to a few seconds. Nodes may choose to "clear cached data" to save space, since there is little to gain for the nodes by maintaining a backup for longer. Theoretically, this may lead to disappearance of data if no nodes have a copy of that data. This is not a significant issue as of now, but for IPFS to be a viable, long-term solution, long term backups need to be strongly incentivized.

VIII. conclusion

A secured and integrity compliant system was proposed using the P2P feature of IPFS and the tamper-proof principle of Blockchain technology. This model is a complete solution to the various problems faced by HTTP and data security. With minimal hardware requirements, any node in the decentralized network can serve data, improving bandwidth, latency, and availability. The four main components namely DHTs, Blockchain, P2P Networks and Content Addressed File System, together, make the model a secured, reliable, and fault-tolerant system.

References

[1] Marr, Bernard. "How Much Data Do We Create Every Day? The Mind-Blowing Stats Everyone Should Read." Forbes, https://www.forbes.com/sites/bernardmarr/2018/05/21/how-much-data-do-we-create-every-day-the-mind-blowing-stats-everyone-should-read/. Accessed 5 Oct. 2019.

[2] Benet, J. (2014). IPFS - Content Addressed, Versioned, P2P File System. ArXiv, abs/1407.3561.

[3] Nakamoto, Satoshi. (2009). Bitcoin: A Peer-to-Peer Electronic Cash System.

[4] The Ethereum Wiki, 2019. GitHub, https://github.com/ethereum/wiki

[5] Labs, Protocol. IPFS Is the Distributed Web. IPFS, https://ipfs.io/. Accessed 6 Oct. 2019.

[6] Kelly, Mat, et al. "InterPlanetary Wayback: Peer-To-Peer Permanence of Web Archives." Research and Advanced Technology for Digital Libraries, edited by Norbert Fuhr et al., Springer International Publishing, 2016, pp. 411–16.

[7] Zyskind, G., et al. "Decentralizing Privacy: Using Blockchain to Protect Personal Data." 2015 IEEE Security and Privacy Workshops, 2015, pp. 180–84. IEEE Xplore, doi:10.1109/SPW.2015.27.

[8] Wang, Huaimin & Zheng, Zibin & Xie, Shaoan & Dai, Hong-Ning & Chen, Xiangping. (2018). Blockchain challenges and opportunities: a survey. International Journal of Web and Grid Services. 14. 352 - 375. 10.1504/IJWGS.2018.10016848.

[9] Benet, Juan. "IPFS - Content Addressed, Versioned, P2P File System." ArXiv:1407.3561 [Cs], July 2014. arXiv.org, http://arxiv.org/abs/1407.3561

[10] Chen, Y., et al. "An Improved P2P File System Scheme Based on IPFS and Blockchain." 2017 IEEE International Conference on Big Data (Big Data), 2017, pp. 2652–57. IEEE Xplore, doi:10.1109/BigData.2017.8258226.

[11] Anjum, A., et al. "Blockchain Standards for Compliance and Trust." IEEE Cloud Computing, vol. 4, no. 4, July 2017, pp. 84–90. IEEE Xplore, doi:10.1109/MCC.2017.3791019.

[12] Sarıtekin, R. A., et al. "Blockchain Based Secure Communication Application Proposal: Cryptouch." 2018 6th International Symposium on Digital Forensic and Security (ISDFS), 2018, pp. 1–4. IEEE Xplore, doi:10.1109/ISDFS.2018.8355380.

[13] Wood, D.D. (2014). ETHEREUM: A SECURE decentralized GENERALISED TRANSACTION LEDGER.

THE APPLICATION OF CLOUD DATA INTEGRITY VERIFICATION SCHEME IN INTERNET OF THINGS SECURITY

WEI SAI, XINPENG ZHANG, CHENGYANG XIE, HONG LI, HUI ZHANG

After-service logistics information center in Chengdu Military Region, Chengdu, Sichuan 610015, China
E-MAIL: carriage1029@163.com

Abstract:

With the development of IoT applications, the scale of IoT data is rapidly increasing. Relying on cloud computing to process IoT data becomes an inevitable choice of IoT data management. However, IoT data will get more and more security challenges and its data integrity verification needs are increasingly high to ensure security. This article is based on a realistic IoT application data integrity verification model, analyzed the IoT cloud storage data integrity verification needs, and through the analysis of existing cloud storage data integrity verification schemes, some exploration paths in lightweight data integrity verification and key security update are proposed.

Keywords:

Internet of things; Cloud computing; Data Integrity Verification; Privacy Protection; Key Update

1. Introduction

At present, IoT devices have been widely used in consumer, commercial, automotive, industrial and medical markets. According to statistics, as of the end of 2015, more than 15 billion IoT devices have been connected to the Internet. It can be seen that, with the development of wireless technologies, Internet of Things devices (especially various mobile devices) have become a convenient and universal Internet service access terminal. However, IoT devices have limited storage space, weak computing power, and limited storage capacity, all of which make it incapable of supporting complex data mining and big data storage. As a rich resource with powerful computing power and storage capacity, with use of cloud computing can reverse this situation and greatly expand the capabilities of IoT devices. Coud computing can make IoT devices more and more intelligent due to massive storage data and potential data analysis capabilities. Being able to access intellectual support from the Internet, this will make a profound difference to its corporate value. For example: Microsoft Marketing Director claims the cloud will play a key role in expanding IoT device computing in the embedded operating system (Windows Embedded). Microsoft is currently working on software that aims to realize the important functions of integrating IoT devices with the cloud computing.

2. Security Challenges for IoT devices when using Cloud Computing data

As one of the main service areas of cloud computing, for IoT device users are to store data on the cloud servers, this resolves the problem for lack of storage capacity of IoT devices. This new storage service has many advantages, such as reducing IoT equipment data management and maintenance burden. However, this will produce many security challenges. Due to the loss of physical possession and control of out-sourced data, IoT device managers are concerned that their data may have been distorted or deleted for the following reasons. First, in spite of the dependable reliability measures cloud providers provide, irreparable data corruption can still result from infrastructure problems. Second, cloud storage service providers do not deserve full trust in cloud storage. Cloud service providers may discard infrequently accessed data for economic reasons, but they claim that data is still well stored or that data loss events are hidden to maintain their reputation. The Cloud Security Alliance (CSA) conducted a systematic survey of weaknesses in cloud computing such as power outages, downtime, data loss, etc. CSA revealed the top three threats "insecure interfaces and APIs," "data loss and leaks," and "hardware failures." These three threats accounted for 64% of all cloud disruptions, while "data loss and leakage" accounted for 25%. Therefore, for a secure IoT device cloud storage data model, it is essential to ensure the integrity of the data in the cloud computing or to verify the integrity of the data.

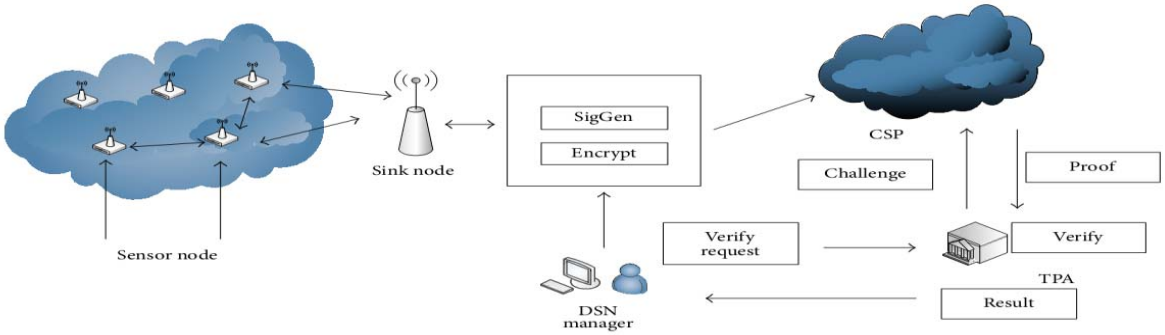


Fig.1 DSN based cloud storage data integrity verification framework

3. IOT cloud storage data integrity verification

3.1. Framework of IoT cloud storage data integrity verification

Combined with a common IoT device-sensor structure and distributed sensor networks (DSN), the following article presents a suitable cloud storage data integrity verification framework. In a distributed sensor network application scenario, for the sake of simplicity, assume that after sink nodes collect sensor nodes data, there is a DSN data manager to process and transmit DSN data to the cloud server. Because the DSN data manager does not have additional computing resources, he can only utilize the limited capabilities of the Sink node for secure storage of DSN data. In this scenario, DSN data managers are part of the DSN, but for cloud service providers, they can be viewed as a special cloud user. Specifically, assuming that Zhang is the owner of a DSN, his task is mainly to collect sensor network data for processing and thus to complete a variety of customer-facing application services. On one hand, Zhang hopes to focus his energy on application services rather than data management. On the other hand, he thinks that purchasing or renting specialized storage devices may cost him a lot of budget, so he tends to save his data to a cloud server provided by a cloud service provider (CSP). In the meantime, he is worried about the following questions:

(1) When DSN data is stored in a cloud server that provided by CSP, he will not be able to control his data using traditional control methods, he fears that CSP will not acknowledge or maliciously tamper with the data after the data is corrupted. Therefore, he needs to be able to verify that the data stored on the cloud server is fully validated at any time.

(2) Zhang's main job is to respond to various applications proposed by the sensor network, so he needed an effective integrity verification scheme that would allow third-party auditors to perform periodic data integrity verification tasks on his behalf.

(3) In practical applications, CSP and TPA can perform storage tasks and auditing tasks honestly, but they are curious too. He may wonder what data the user has stored on the cloud server. However, Zhang wants to ensure that the data he stored is confidential, in addition to himself and his designated authorized to get the real data, the other people cannot get the real storage. Therefore, he needs to ensure that this storage model protects the zero-knowledge privacy of his stored data.

(4) Zhang rely on cloud servers for data storage and management. For a variety of applications, he needs to be able to interact with the cloud server for dynamic data, to access and update his stored data.

(5) Zhang's DSN may be arranged in multiple regions. To simplify management, he needed the TPA to be able to handle audit tasks from multiple Sink nodes (DSN data managers) simultaneously.

(6) In the DSN data cloud storage model, DSN data managers utilize the Sink node's limited ability to compute data validation labels, and upload the data with the corresponding verification tag to the cloud server. DSN data managers therefore hope cloud data integrity verification schemes should be as lightweight as possible.

(7) In the DSN data cloud storage model, Sink nodes may be damaged due to various reasons and need to be replaced. The digitally signed certificate of the device may expire after a period of time. DSN data managers therefore hope cloud data integrity verification schemes should support the dynamic updating of keys.

As shown in Fig.1, The DSN data cloud storage model includes the following components: Sensor node、Sink node、Cloud service provider (CSP), DSN data manager、Third-party auditors (TPA). The Sensor node is responsible for collecting information from the target and sending it to the Sink node. Zhang is the data owner of a distributed

sensor network. He assigned a DSN data manager to sign and encrypt the data. Then, the DSN data manager transmits the generated outsourced storage data and the corresponding data tag to the cloud server provided by the CSP, and deletes the locally stored data. If the DSN data manager wants to verify the integrity of the data stored on the cloud server, he needs to send a query to the TPA. After receiving this query, the TPA verify that the query is valid or not. If it is valid, the TPA will generate a challenge message to ask the cloud service provider for a data verification ticket stored on the server. The TPA verifies this information and returns a valid authentication result to the DSN Data Manager. Finally, DSN data manager will tell Zhang audit results.

3.2. Goals of IoT cloud storage data integrity verification scheme

The cloud storage data integrity verification scheme for IoT devices models should meet the following security and efficiency goals.

(1) Public Verification: Allow TPA to verify the integrity of cloud storage data without retrieving the entire data or increasing user's online burden.

(2) Storage Correctness: Ensure that no dishonest cloud servers pass TPA audit validation if the storage blocks are corrupted.

(3) Privacy Protection: Ensure that the TPA cannot obtain the DSN data manager's data content information in any way during the audit.

(4) Batch Auditing: TPA can fulfill the audit validation task requests made by a large number of different DSN data managers at the same time during auditing, and perform data audit validation tasks safely and efficiently.

(5) Verify Lightweight: Make sure that the communication costs and computational costs of interacting with the DSN data manager are as small as possible for the TPA to perform audit validation.

(6) Security Key Update: Ensuring that a device replacement or device digital signature certificate update process is secure and does not create new security issues.

4. The technical route of the IOT data integrity verification scheme

4.1. Overview of the Cloud Storage Data Integrity Verification Scheme

Juels et al. conducted a prior research on data integrity verification in [1], and they proposed a protocol named Proofs of Retrievability (POR). Using this technique, the cloud storage data can be validated effectively and can provide some degree of data recovery function. In the meantime, Ateniese et al proposed a Provable Data Possession (PDP) protocol in [2]. A PDP protocol can be turned into a POR protocol by adding a forward error correction code. Using a combination of homomorphic authenticators and sampling strategies based on Randomized Digital Signature (RSA) in the PDP scheme, the verifier can use all the data without downloading it , Publicly verify the integrity of cloud data. Although the PDP first considered the public integrity test, which used a third party instead of the user to verify the data integrity, the PDP did not give a formal proof of security definition and security.

Later, Shacham and Waters proposed another POR technology in [3], which uses Boneh-Lynn-Shacham (BLS) to validate cloud storage data integrity and for the first time we defined public audit security and safety certificate. Follow-up work is followed by Shacham and Waters's work, have the same security model and threat model. Based on the literature [3], many schemes for data integrity verification of cloud storage are proposed one after another.

With the development of cloud computing, more and more attention has been paid to third-party public integrity verification schemes that support privacy protection, reusable proof schemes for dynamic data, and full verification schemes that support batch audit tasks. In order to reduce the user's computational burden, user's en-trust a credible third-party auditor (TPA) to periodically verify the data stored on the cloud server. At the same time, cloud users may want to keep their data private to the TPA. In 2009, Wang et al. proposed an integrity verification scheme [4] to ensure data security in support of TPA auditing and a publicly verifiable scheme [5]. Reference [4] considers the dynamic storage of data under distributed conditions and applies the challenge response protocol to verify the correctness and wrong location of data. The literature [5] based on the BLS signature technology, while introducing the RSA structure, but the program only supports some dynamic operation, and does not provide privacy protection. In 2010, Wang et al. proposed an improved public integrity verification scheme for cloud storage data with privacy protection. Data owners delete the data and verification labels locally after they are stored on the cloud server. When there is an audit request, a request is made to the TPA to perform an audit task by the TPA. In the TPA audit process, random mask technology is used to ensure that the TPA and the cloud server cannot obtain any useful information about the original data. In 2011, Wang et al. proposed a public integrity verification scheme for cloud storage data using the Merkle Hash Tree (MHT) for dynamic operations such as data modification, insertion and

deletion. However, Xu et al. Point out that this protocol cannot resist the attacks of malicious cloud servers and external attackers. Even if the cloud server passes the auditing of the TPA, it cannot guarantee the authenticity and integrity of the data stored by the cloud user. In 2013, Wang et al once again proposed an improvement plan [6], which is more resistant to forged attacks from outside. However, in some practical applications it is not sufficient for an auditor to obtain information from the entire file block. For example, an auditor can initiate an offline guessing attack to get files from several blocks of files stored on the cloud server. In [6], Zhu et al. proposed a collaborative and provable data holding scheme that supports batch auditing of cloudy servers and also supports dynamic auditing. However, multi-user bulk auditing is not supported because in the scenario, user-generated data validation labels use different parameters and TPAs cannot be audited using a linear combination of labels from different users. Wang and Zhu proposed a series of classical solutions for data integrity verification of cloud storage. However, their solutions have heavy computational overhead, which greatly affects audit performance [7], and then more support the integrity of data dynamic operations Verification schemes are proposed one by one.

4.2. IoT device using cloud storage data model focuses on data integrity verification scheme

Although many cloud data integrity verification schemes have been proposed so far, they are all designed in a traditional cloud storage environment and do not take into account the specialized applications of IoT device data cloud storage models. For an IoT device cloud storage data audit application, there is a higher demand for ensuring the integrity of data in the cloud computing or for verifying the integrity of the data:

(1) emphasize public verification

Until now, there are two types of data integrity verification schemes that are based on the public key infrastructure (PKI) mechanism and based on the pseudo-random function (PRF) mechanism. Protocols based on pseudo-random functions (PRFs) are themselves verifiable, meaning that only the data owner can check the integrity of the in-memory data. As a comparison, a publicly verified integrity verification scheme allows external third-party auditors to verify the integrity of the data when needed. IoT devices are typically resource constrained and do not support complexity calculations. Therefore, for IoT users, it is crucial to be able to verify the public integrity of IoT data cloud storage data. In addition, the number of IoT devices determines that the verification scheme must support bulk user requests and bulk data verification processing.

(2) Emphasize privacy protection

Another important issue in the IoT cloud data integrity verification scheme is the privacy protection of data. In other words, the content of the challenge file should not be disclosed to third-party auditors during the integrity verification process. Note that some embedded devices, such as smartphones, smart cameras, Apple watches and sports bracelets, may generate a large amount of personal data. If this sensitive information is exposed during the integrity verification process, IoT users' privacy, such as their social relationships, physical status, and family background may be compromised to the integrity verifier or the public.

This information disclosure may help hackers bring disastrous consequences to embedded users. As a result, IoT users are reluctant to disclose any information to third-party auditors during data integrity verification, so the sole job of third-party auditors should be to check the integrity of outsourced storage files. Early integrity verification schemes did not have privacy protection. In response to the challenge, the cloud server generates as responses (are the data blocks that are challenged and are the random Numbers generated by the verifier in the challenge). As each reveals part of the information in the data block, the third-party auditor can obtain the data by repeatedly challenging the data blocks. Later, some integrity verification schemes that support privacy protection are vulnerable to attacks by malicious cloud servers and external attackers, such as offline guessing attacks initiated by third-party auditors. The latest concept of a "zero-knowledge public integrity verification scheme" that can withstand off-line guessing attacks does not give a very critical security model. And their model does not support key update scenarios.

(3) Must support secure key updates

In the PKI system, to use a certification authority (CA), the main role of the CA is to digitally sign and issue certificates to authenticated users. Public key cryptography uses certificates to handle role problems. A certificate is an electronic file that is used to associate the public key of an individual, company, or any other entity for a specified period of validity. The key issue in PKI is how to deal with user key leakage or failure. The expired key indicates that the user's certificate is no longer valid, and the key leak will pose a serious security threat to legitimate users. Therefore, in practical applications, certificate revocation and renewal of reissues are the key links for maintaining the PKI security system of Internet communication security. When the private key leaks, the certificate must be revoked because the owner of the certificate no longer controls the

use of the certificate, or the certificate may generate the wrong signature. If a CA does not have the ability to revoke a certificate, it cannot mark it as an untrusted certificate before the certificate expires, and certificates can take years to expire. Because the original certifier who stored the data in the cloud was no longer valid after the key was updated. In order to solve this problem, a simple solution is to download all data blocks and verification tags from the cloud, recalculate the verification labels of each data block and upload them to the cloud. This model will bring both unacceptable communication costs for IoT users and cloud servers, as well as significant computational costs for IoT users. In order to solve this problem, it is essential to adopt an efficient key update and authentication protocol in the data integrity verification scheme for the practical use of IoT data cloud storage mode.

(4) Must use the lightweight algorithm

Resources such as computing power, storage capacity and bandwidth of IoT devices are often greatly limited. Existing data integrity verification schemes often claims that their solutions are efficient, but still cannot meet the requirements of data cloud storage in Internet of Things security audit efficiency. This is because these schemes rely on the homomorphic authentication technology and random masking code technology to achieve the privacy protection and open integrity verification functions, and the use of aggregate signature technology to complete the data integrity verification batch task processing, but the program in order to complete the homomorphic authentication basic on the use of a computationally expensive bilinear pair computing, in the actual experiment of its communication costs and computational costs are always unsatisfactory.

4.3. Our work

(1) Lightweight algorithms that do not use bilinear pairs an efficient open data integrity verification scheme for cloud storage data is designed [8]. The scheme uses variant Schnorr signature algorithm and homomorphic message authentication code (MACs) technology to reduce storage space for authentication information, calculation cost of audit proof response information and communication cost between verifier and cloud server and using random mask technology to ensure that third-party auditors cannot recover the user's original data blocks through audit information, effectively protecting the privacy of users.

(2) Secure Key Update Protocol

An open cloud storage data integrity verification scheme is designed to support the key update function [9]. The scheme is designed by combination of zero-knowledge proofing system, proxy re-signature technique and linear homomorphic verification tag, which effectively solves the problem of key Updated questions. When the cloud user's key expires and needs to be updated, it is no longer necessary to download his data from the cloud. Instead, a very small authenticator is downloaded from the cloud as an alternative to the entire file. This allows the user to change the validation tag effectively, which greatly reduces the cost of communication and computation. The scheme is also proved to be safe in the stochastic prediction model.

Further, it is found that there is a malicious cloud server in the process of revoke the collusion with the revoked user attacks will cause the user private key leak problem. Therefore, in order to ensure the security of the key renewal process, the scheme [10] improves the key renewal mechanism and defends the attack by using the new user private key to sign the original user public key and form the user renewal key.

Furthermore, after the anti-collusion strategy is added, the verification efficiency of the scheme reduced to a certain extent, and the foregoing scheme does not prevent and revoked user from conspiring with the malicious cloud server to modify its signature data. Therefore, a new one-way proxy re-signature algorithm is adopted. The proxy re-signature key is generated by combining the current user's private key with the revoked user's public key. There is no private key leakage problem and the data ownership can be safely transferred [11].

(3) Efficient support key new data integrity verification scheme

The lightweight integrity algorithm is used to reconstruct the above data integrity scheme that supports key update, and finally a lightweight application that supports public auditing, zero-knowledge privacy protection, key update support (anti-collusion attacks) applicability of data security in the Internet of things.

5. Conclusion and future work

With the development of information technology, the computing power, storage capacity and bandwidth of IoT devices will be greatly improved. However, at the same time, its functional requirements and application scenarios are extended, and the problem of data verification efficiency will not only disappear but will become more and more severe. The introduction of new concepts such as "material computing" provides a new way to calculate efficiency while on the one hand the large amount of data generated faces more security challenges. In another study, we used the third-party proxy computing platform (TPC) of "fully homomorphic encryption" technology to perform operations on encrypted data and decrypted the results to be

consistent with the plaintext operations. It is proposed to adopt this model ensure the data in the network always maintain cipher text status, to resist various attacks from the untrusted network. Finally, it must be pointed out that in the future the Internet of Things will be huge in scale and the data integrity verification center processing model is bound to be replaced by the distributed processing model. "Blockchain" technology offers a very attractive "decentralized" model with strong resource support. The combination of "blockchain" technology and data integrity verification is a predictable hot spot.

References

[1] A. Juels, S. Burton, J. Kaliski. Pors: proofs of retrievability for large files[C]. Proceedings of the 2007 ACM Conference on Computer and Communications Security, CCS2007, Alexandria, Virginia, USA, 2007, 584–597

[2] G. Ateniese, R. C. Burns, R. Curtmola, et al.. Provable data possession at untrusted stores[C]. Proceedings of the 2007 ACM Conference on Computer and Communications Security, CCS2007, Alexandria, Virginia, USA, 2007, 598-609

[3] H. Shacham, B. Waters. Compact proofs of retrievability[C].Advances in Cryptology-ASIACRYPT 2008, Melbourne, Australia, 2008, 90-107

[4] K. Draou, N. Bellakhal, B. G. Cheron, et al. Ensuring Data Storage Security in Cloud Computing.[J]. Materials Research Bulletin, 1998, 33(7):1117–1128

[5] Q. Wang, C. Wang, J. Li, et al.. Enabling Public Verifiability and Data Dynamics for Storage Security in Cloud Computing[C]. 14th European Symposium on Research in Computer SecuritySaint-Malo, France, 2009, 355-370

[6] C. Wang, S. S. M. Chow, Q.Wang, et al.. Privacy-preserving public auditing for secure cloud storage[J]. IEEE Transactions on Computers, 2013, 62(2): 362–375

[7] Qin Zhiguang, Wu Shikun, Xiong Hu. Review of Data Integrity Audit Scheme in Cloud Storage Service [J]. Information Network Security, 2014(7):1-6

[8] Q. Wang, C. Wang, K. Ren, et al. Enabling public auditability and data dynamics for storage security in cloud computing[C]. Proceeding of ESORICS 2009, Saint Malo, France, 2009, 21-25

[9] Information Security and Privacy：Public Cloud Data Auditing with Practical Key Update and Zero Knowledge Privacy，

[10] Xinpeng Zhang, Chunxiang Xu, Xiaojun Zhang, Taizong GuWei, Zhi Geng, Guoping Liu. Efficient dynamic integrity verification for big data supporting users revocability [J]. Information, ID: information-119914.

[11] Zhang Xinpeng, Xu Chunxiang, Zhang Xin Yan, Sai Wei, Han Xing Yang, Liu Guoping. Public Auditing Scheme for Reversible Cloud Storage Data Based on Proxy Resignature. Computer Application Research

Blockchain-based Integrity Protection System for Cloud Storage

Pratima Sharma
Computer Science and Engineering Department
Delhi Technological University
Delhi, India
pratima.sharma1491@gmail.com

Rajni Jindal
Computer Science and Engineering Department
Delhi Technological University
Delhi, India
rajnijindal@dce.ac.in

Malaya Dutta Borah
Computer Science and Engineering Department
National Institute of Technology Silchar
Assam, India
malayaduttaborah@gmail.com

***Abstract*— Data are now an invaluable asset; therefore, in most machine-aided human activities, it guides all organizational decisions. Attacks to data integrity are therefore of paramount importance, as essential organization decisions can be maliciously affected by manipulation of information. This problem is particularly true in cloud computing settings where information owners are unable to monitor basic information elements such as access control and physical data storage. Recently, Blockchain has appeared as an intriguing technology that offers convincing data integrity characteristics among others. In this paper, we design a blockchain-based cloud architecture, focusing the threats of data integrity. We propose a methodology in peer to peer cloud storage to verify integrity based on blockchain, making checking more transparent, open, and publicly available. In this framework, we confirm the data integrity by using merkle trees and design a smart contract for the authentication process. Furthermore, we present an elementary model of an e□icient blockchain structure for cloud computing settings. Thus, this methodology helps in enhancing the security and integrity of the cloud storage system.**

Keywords—blockchain, cloud computing, merkle tree, data integrity.

I. INTRODUCTION

Data is the main asset nowadays. In various distinct areas, from commerce and insurance cover to healthcare, admin and learning department, it is strategic to drive any company choice. As more and more machine-aided human activities rely on information, believing information has now turned into critically important. At the same moment, the important role of data has created it a desirable destination for attacks aimed at violating the basic CIA characteristics (Confidentiality, Integrity, Availability) that information must follow to be assured. Cyber-attacks [1] on CIA assets trigger various trust deficiencies in data depending on the estate undermined. Specifically, disruption of accessibility keeps data from being obtained for a short span of moment only, but activities can be restarted as quickly as the information is again available. Whereas giving up confidentiality reveals sensitive information, and it cannot be reversed, but underlying information remains obtainable and useable, at least to the level permitted by the damage caused (i.e., an organization perpetrator of information exposure can experience economic implications). Instead, manipulating data integrity has always been an extremely harmful assault which opens the door for critical problems to data trust. Indeed, information manipulation may go unnoticed and maliciously operate activities by removing specific records (i.e., removing inconvenient hints) or changing particular parts of information (i.e., changing the behavior of customers information).

Unlike confidentiality and accessibility, there's no route to recover the initial information once integrity is broken, it's wasted indefinitely. As integrity assaults are difficult to recognize and truly effective, we concentrate on data integrity instead of confidentiality or accessibility in this paper. In cloud computing settings, information integrity problems are worsened as information holders hardly regulate where their information is placed, who can effectively tap it, and how [2].

The integrity of data generally protected by using two methods, one is by using cryptographic tools (i.e., message digests, asymmetric encryption) and other by deploying data replication techniques. To protect user's data, cryptographic methods are used to sign consecutive parts of data. Thus, any counterfeited can be identified quickly through signature-based validation techniques. Indeed, an effective attack should need the secret keys violation, therefore, to refresh data signatures and prevent the cryptographic based integrity controls. These assaults are difficult to perform, but they are virtually undetectable once they have been realized. It is therefore strongly recommended that suitable policies for data replication be used to guarantee data integrity anyhow.

Duplicating and spreading data across a collection of nodes is critically hampering information integrity: an intruder compromised all the replicated information without detection. This strategy to replication [3] is commonly accepted in reality, such as in cloud computing settings where distributed storage funds are limited. However, while replication certainly raises the load in a cloud environment for an effective assault, cloud suppliers themselves may make deals with intruders to violate data integrity readily. To prevent these collusion assaults and prevent blind faith in the privacy promises asserted by cloud suppliers, we support advanced use of blockchain technology to develop and execute a centralized, safe blockchain-based repository for cloud computing settings [4].

A blockchain is defined as a distributed ledger spread among thousands of diverse nodes. It was used as a public ledger for Bitcoin operations in its first design. Recently, it has acquired considerable attention for the intriguing characteristics it ensures (e.g., consensus, transparent, verifiable, and non-repudiable data). It is practically tough to alter the information stored in a blockchain ledger is the main focus for data integrity protection. The distribution, permanent, and replication features of blockchains initiate their broad implementation in cloud environments. In this paper, we propose a blockchain-based solution for a cloud storage system for tackling integrity-based threats [14].

Structure of the paper: In Section 2, we first present related work that paves the way to an efficient blockchain-based solution for the cloud storage system. Then, Section 3 identify the significant threats to data integrity to adequately address the feasibility of a proposed solution. In Section 4, we explain the proposed solution. In Section 5, we present our solution tackling identified threats. Finally, we conclude.

978-1-7281-3755-1/19/$31.00 ©2019 IEEE

II. Related Work

According to Christidis and Devetsikiotis [5] a blockchain is a list which is structured to save data in a form that is akin to a database which is distributed and is designed such as to ensure that any arbitrary manipulation of the data becomes complex owing to the fact that participants within the network save and verify the blockchain. Every block within the chain is made up of a structure that comprises of a body as well as a header. Hash values of the existing and previous blocks and nonce are included within the header. Considering that hash values within every peer inside the block are impacted by the values of the preceding blocks, it is very complex to fabricate or modify the data that has already been registered. While there is scope for changing the data but only in the case where 51 percent of peers are attacked simultaneously, however, the scope for an attack is highly difficult from a realistic perspective.

Shetty et al. [6] state that the provenance system within the cloud facilitates the identification of any violation in the security based on technologies related to auditing and logging. Nonetheless, executing or putting such systems into practice within the cloud domain might not be as effective as required owing to numerous layers of hardware and software elements that interoperate and are spread beyond organizational as well as geographical boundaries. Apart from identification of the cause and effect of violations in security within cloud warrants forensics and logs to be accumulated from various and dissimilar sources which can prove to be a task that is difficult to overcome.

A blockchain-based ProvChain architecture was proposed by Liang et al. [7] to facilitate the collection and verification of provenance of cloud data, especially by operating within three stages. These stages would comprise of data validation, data collection, and data storage with the help of Tierion API. The use of Apache JMeter for evaluating the performance substantially exhibits the ability of ProvChain from the aspect of security features which comprises of provenance that cannot be tampered with, reliability and privacy of users with the reduction in overheads for applications within cloud storage. The issue of block suppressing attack was modeled by Tosh et al. [8] which commonly prevails in Proof of Work (PoW) based mining pools to comprehend the strategy of the attacker to take over the rewards of the pool member. Evaluation of the block suppressing attack is facilitated with the help of simulation by taking into account a blockchain that is based on private proof-of-stake. The findings indicated the access of the attacker to additional computational power has the potential to disturb the operation of honest mining within the model of blockchain.

A share Cloud Object Store (COS) architecture was designed by Barger et al. [9] to tackle the problem associated with bloat in the blockchain. The authors augmented the proposed architecture by adding a capability to establish a group of business parties who were open to sustain a data store which was shared within COS, at the same time governance of the plan for data control was with Hyper Ledger Fabric (HLF). For blockchain, many lightweight nodes have been proposed. It has been proposed by Frey et al. [10] that management of Unspent Transaction Output (UTXO) set within Distributed Hash Table (DHT) to overcome costs related to storage of thin nodes such as smart phones. As per their proposal, thin nodes are rendered such that it facilitates verification of (Transactions) TXs at a local level by sharing UTXO set on DHT and entrenching a hash of hash block lists and a shard hash on a data block. In this manner, every node would be in a position to confirm that shards are managed appropriately by others.

With the objective to overcome issues related to size of storage, a scheme for balancing new storage was proposed by Shannigrahi et al. [11] with the help of storage that was distributed on the basis of DHT where the whole blockchain was held on the architecture based on bitcoin nodes within an array of DHT nodes. A simulation utilizing Overlay Weaver helped in assessing the proposed scheme, wherein it was found that nodes had the propensity to behave in the same manner as to complete nodes while resolving the required size for storage. A data logging and integrity management system based on blockchain was proposed by Park et al. [12] where the authors incorporated three layers. One layer was the Common Service Centres (CSC) utilization layer where instance data utilized by CSC was stored in the layer of Cloud Service Provider (CSP) management, while layers pertaining to the blockchain system were utilized for storing data's integrity management. The performance of the proposed system was compared with other blockchain-based cryptocurrencies where it was observed guarantee in data integrity at the time of processing additional transactions than current blockchains that were devoid of permissions.

To resolve security-related challenges, the EU SUNFISH project was developed, which aimed at proposing a platform for democratic cloud federation that was distributed and would ensure the security of the managed data by-design. The proposal [13] was designed as an innovative and novel service that allowed cloud data and services to be created and managed securely. The features of Federation as a Service (FaaS) comprised of high-end services in data security and were based on innovative principles of design which lead to the governance of democratic and distributed cloud. Owing to the need to manage data with high sensitivity by cloud federations, FaaS must extend superior assurances related to complying with member contracts. Apart from enforcing runtime of the contracts, FaaS is also required to ensure the contract's integrity. Also, to make sure about non-repudiable proof of contract enforcement, every interaction that is inter-cloud should be examined, and the logs are to be stored with a robust guarantee with regards to its integrity. Table 1., summarizes the various research work.

TABLE I. Summaization of Related Work

Author	Finding
Christidis, Devetsikiotis [5]	The authors find that the amalgamation of blockchain and IoT is robust and can facilitate major transformations across industries.
Shetty, Red, Kamhoua, Kwiat, Njilla [6]	Provenance system within the cloud facilitates the identification of any violation in the security based on technologies related to auditing and logging.
Liang, Shetty, Tosh, Kamhoua, Kwiat, Njilla [7]	Proposed a ProvChain architecture based on blockchain. The use of Apache JMeter for evaluating the performance substantially exhibits the ability of ProvChain from the aspect of security features which comprises of provenance that cannot be tampered with.
Tosh, Shetty, Liang, Kamhoua, Kwiat, Njilla [8]	The findings indicated the access of the attacker to additional computational power has the potential to disturb the operation of honest mining within the model of blockchain.
Barger, Manevich, Bortnikov, Tock, Factor, Malka [9]	A share cloud object store architecture was designed to tackle the problem associated with bloat in the blockchain. HLF could facilitate governance for data control.

Frey, Makkes, Roman, Taïani, Voulgaris [10]	Thin nodes are rendered such that it facilitates verification of TXs at a local level by sharing UTXO set on DHT and entrenching a hash of hash block lists and a shard hash on a data block.
Shannigrahi, Fan, Papadopoulos [11]	A simulation utilizing Overlay Weaver helped in assessing the proposed scheme, wherein it was found that nodes had the propensity to behave in the same manner as complete nodes while resolving the required size for storage.
Park, Park, Huh [12]	It was observed guarantee in data integrity at the time of processing additional transactions than current blockchains that were devoid of permissions.

III. Threats To Data Integrity

There may be numerous and varied risks to data integrity in the framework of cloud storage. Our concentration is on the user's data storage database, hence information whose corruption has a crucial impact on privacy and safety in an organization. The risks that we believe range from harmful information changes to information releases without informing all the participating participants.

Data deposited in the cloud may be affected by the harm caused by transferring information to/from cloud storage. Due to the outsourcing of information and data processing to a remote server, information integrity must be continuously preserved and verified to demonstrate that information and computation are untouched. The integrity of information implies that information should not be altered unauthorizedly. Any changes to the data must be identified. Data integrity may be helpful when data is wasted or notified when information is manipulated. The following are two instances of how to violate data integrity. More specifically, the following threats are considered:

Threat 1: Loss or Manipulation of Data

Consumers have a lot of customer records. Cloud suppliers, therefore, provide Storage as a Service (SaaS). These documents can be obtained daily or scarcely at times. Therefore, they need to be protected from unauthorized access/manipulation. This need is due to the complexity of cloud services as the information is exported to a distant cloud that is unsafe and useless. Because the cloud is unreliable, unlawful customers may lose or modify the information. Data may be changed deliberately or inadvertently, in many instances. There are also many organizational mistakes which could lead to loss of data like obtaining or returning inaccurate backups. The intruder may use information outsourced by customers as they missed ownership of it.

Threat 2: Untrustworthy Remote Server Carrying Out Computation on Behalf of User

Cloud computing does not only concern storage. Some extensive computations require the authority of cloud processing to fulfil their duties. Users are, therefore outsourcing their processing. Since the cloud provider is not within the security boundary and is not transparent to the task owner, nobody will confirm if the integrity of the computation is intact or not. The cloud provider sometimes acts in such a manner that no one finds a variation from ordinary implementation in computation. The cloud provider could not perform the job properly because the resources have a significance for the cloud provider. Even if the cloud provider is deemed to be safer, many problems arise from the fundamental structures of the cloud provider, sensitive software, or misconfiguration.

IV. Proposed Architecture

In this section, we introduce the proposed architecture for data integrity verification. The architecture consists of two primary components, a smart contract and blockchain-based storage of cloud data. A blockchain-based smart contract is designed for checking the authenticity of the users for an organization. The smart contract consist of various functions is deployed on the blockchain network. Smart contract execution triggers automatically whenever the user initiates a request for accessing the cloud data to check the authenticity of the user. The cloud data is stored in the form of blocks in the peer to peer network. The merkle tree is used for cloud data integrity verification. The overall proposed architecture is shown in Fig. 1. The workflow of the architecture is divided into two steps: the initial step and verification step. There are 4 steps in the initial stage. In the beginning, the user will create two merkle roots, one storing owner related information and another for storing file information by dividing the data into multiple shards, then utilize them to create a merkle trees. In the next step, the client will initiate the request for uploading the information. In the third step, smart contract checks the validity of the user by executing various functions and then complete the uploading process. In the last step, user data will successfully store in the peer to peer network.

In the verification phase, the user will request for the outsourced data to the blockchain network, and then this will automatically execute the smart contract function. Then, after checking the status of the user, a smart contract will request a challenge number from the user. After accepting the challenge number xi smart contract selects requested file to check the integrity by calculating the hash and compares new root with previously stored root. If they are equal, the data integrity is safe otherwise not. In the end, the network will return the result to the user. The detail explanation of the architecture is given in the following subsections.

A. Smart Contract

A smart contract is defined as a digital contract or a computerized protocol used for the transaction to implement contract terms. It is used for the conversion of statutory provisions (attachment, collateral, etc.) impending into the code and lastly embedding them into assets (hardware / software) which could auto-enforce them, that in effect minimizes the need for intermediate trust between transacting sides and the existence of accidental or false events. Smart contract runs on every node in the network independently and automatically, depending on the information to be included in the initiating operations. They enable us to have specific chain-based computations, as shown in Fig 2. Below are some of the findings are listed below,

1) This smart contract will be staying in its state and will safeguard the assets on the blockchain.
2) A proper smart contract should describe all possible outputs of the contract.
3) It can be triggered by transactions or messages sent to its addresses.
4) It is deterministic. That is to say, and the same entry produces the same yield. If an attempt is made to implement a non-deterministic smart contract, the network will be dismissed.

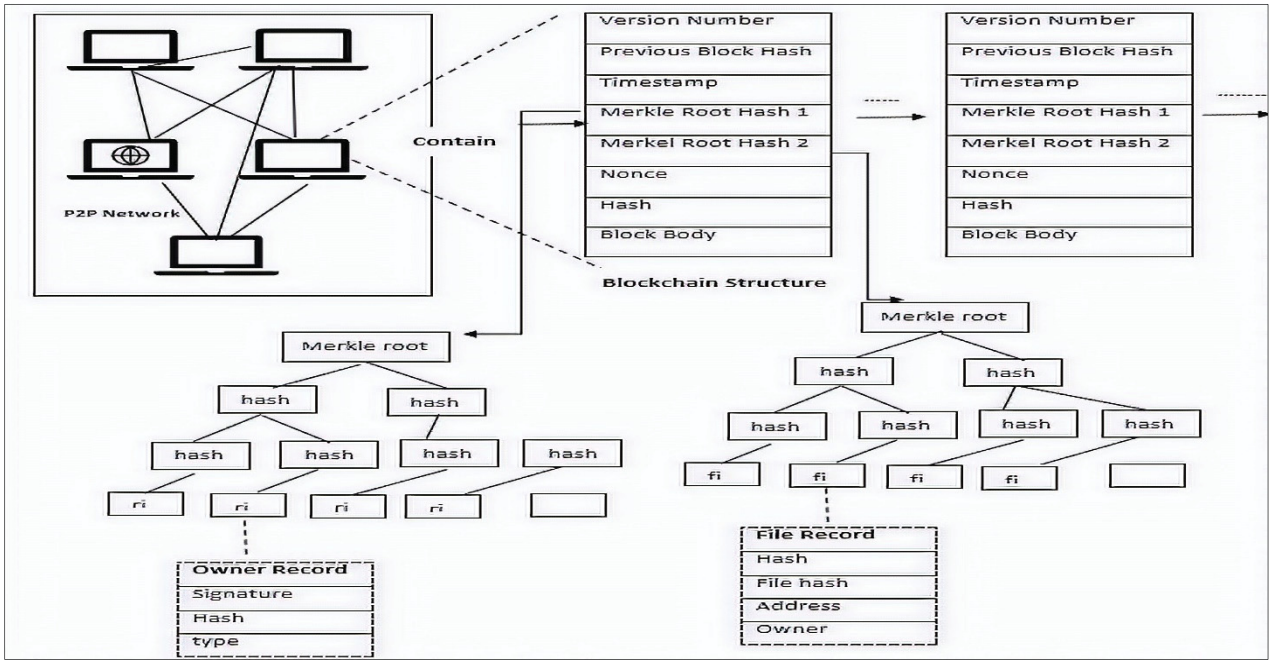


Fig. 1. Blockchain-based Integrity Protection Architecture for Cloud Storage.

As all the interactions occurred through signed messages on a blockchain, then all network participants will get an intimation of verification trace of smart contract's operations. The Smart Contract (SC) is used for user authentication and is released on the blockchain afterward. The SC offers various features to recognize legitimate customers effectively, efficiently, and securely. The smart contract is a safe and effective programmable asset that runs precisely as scheduled. The blockchain also publishes all tasks conducted using the SC. The SC has the following characteristics: 1) allowing organizations to attach user identification and other associated data, 2) allowing organizations to handle and alter data as required transparently.

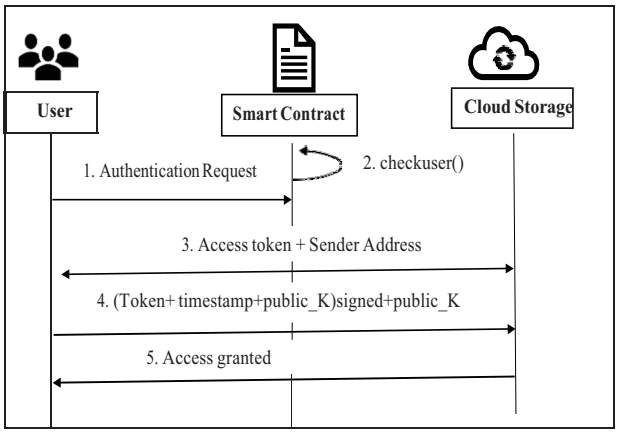


Fig. 2. Smart Contract Authentication Scenario.

The SC is used for the authentication of the organization's users. The smart contract in a proposed architecture is composed of the following functions: 1) adduser (): for adding a new user of the organization. 2) checkuser (): for the checking the validity of the user on the user's authentication request. 3) removeuser (): for removing the requested user's details by the organization. 4) verifyintegrity (): for checking the integrity of the requested information by using challenge generation process. The functions are in such a way that organization users can execute and get access to cloud storage services. As Shown in Fig 2. the smart contract executes the series of steps in order to grant the access to the user for storing data on cloud.

B. Blockchain

The blockchain technology comprises of mathematics, cryptography, algorithm, an economic model that combines peer-to-peer networks and also used for solving database synchronization problem using a distributed algorithm. Blockchain includes key data, parent block hash, present block hash, timestamp, and other information as shown in Fig 3.

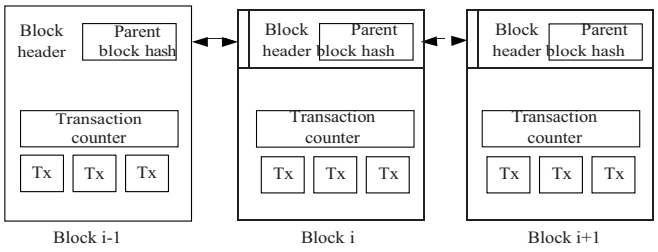


Fig. 3. Blockchain Structure.

Existing blockchain-based security architecture are developed for their own security objective, so block data, data flow, and consensus algorithm implemented in these systems are distinct. Because of a block's limited ability, we keep significant metadata that provides customer information original position. Fig 1., depicts the metadata of the file stored by the user, such as the address of file, file owner, and hash related information.

Our proposed system modifies the original structure of the blockchain transaction to store customer's file operations. As depicted in Fig 1., the structure of the transaction contains the customer's signature fields and transaction hash. The feature section is introduced to indicate the service type of the

customer, and the outcome field is used after user operation to track the hash value of the information. However, if the customer changes new information, the data field cannot be vacant. The information transmitted will be recorded in the data field, and the other kinds of operation will be vacant. The target field maintains the data's hash code and is used to find the file. We have also introduced an array in the file frame to save data about the file metadata. Correspondingly, to check the validity of file data, a merkle root tag is introduced to the file header. Merkle root is a root value of merkle tree in which every leaf node is labelled with the hash of a data block, and every non leaf node is labelled with the cryptographic hash of the labels of its child nodes.

The field fileinfo is used to keep customer data metadata details. It includes the data owner, the filehash used to check data integrity, the real data storage address, and the fileinfo structure hash. We are building two merkel trees in the frame chamber in this scheme. One is used to save usage data, and the other is used to save a file's metadata. If the block data is manipulated, a mistake will be detected while the block information is checked, and then the block will be dismissed. One of the system's key components is to guarantee the information source's precision. If the origin of the information is not reliable, the audit outcome is suspicious. Any procedure on the file may produce a matching transaction file in the client automatically in this scheme. The customer signs these transaction documents with their private key and then broadcasts them to the blockchain. The blockchain section of the miner servers recognizes the file. They first check the validity of the file and then fit the relevant document into a pocket to ensure information safety. The metadata of the files contained in the block frame may be utilized by the smart contract to audit integrity. The file hash value was registered when the file was uploaded. In other phrases, a file's original state was registered. The hash value is recomputed when the file is acquired through the respective URL. If the hash value stored in the block body is not equal, the file will be deemed to be damaged.

V. Adressing Integrity Threats

A threat model is constructed to ensure the accurate determination of the attack surface for the proposed application. Several threats are outlined in section 3. In this section, we explain how the proposed architecture addresses the identified threats. Associated mitigation processes are driven to ensure security. The manipulation of data can be easily identified by the proposed architecture, as explained in the previous section. The remote server unauthorized computation can be tackled by using the authentication process before providing any access. Whenever the threat raised against security, then appropriate mitigations should get implemented. In this proposed method, when a threat raised while authenticating the user, some of the measures are mitigated. 1) Generic error messages are raised, which should be displayed by preventing guessing valid user accounts. 2) If dictionary attacks are raised, then preventive measures like locking the process after n number of failed attempts.

Conclusion

In this paper, we have presented an integrity verification architecture in peer to peer network. It deals with the issue of unreliable traditional verification method by using blockchain technology. It also designs the threat model to analyse the various threats involved in the integrity verification process. Our primary input lies in the rationale for designing a blockchain-based model capable of providing the required data integrity, efficiency, and stabilization. This paper paves the path for possible future works. The route we suggest can be further explored by implementing a functioning model to verify the effectiveness of our alternative in terms of attainable throughput and network performance. In addition, a more detailed and official review of the trade between efficiency and data integrity protection is needed to demonstrate the effectiveness of our layout against recognized risks. The method indeed allows consistency among dispersed replicas to be achieved and the threat handling to be simplified. Finally, research into the effectiveness of achieving more stable blockchains is key to enabling their wide-ranging implementation as secure memory infrastructure, e.g., in cloud computing settings.

References

[1] Castiglione, A., Pop, F., Ficco, M., Palmieri, F.: Cyberspace Safety and Security. Springer International Publishing, Cham (2018).

[2] Wang, R..: Research on Data Security Technology based on Cloud Storage. Procedia Engineering. 174. pp. 1340-1355. (2017).

[3] Kemme, B., Alonso, G.: Database Replication. Proc. VLDB Endow. 3, 5–12 (2010).

[4] Gaetani, E., Aniello, L., Baldoni, R., Lombardi, F.: Blockchain-based Database to Ensure Data Integrity in Cloud Computing Environments. In: Italian Conference on Cybersecurity. pp. 146–155. Venice, Italy (2017).

[5] Christidis, K., Devetsikiotis, M.: Blockchains and Smart Contracts for the Internet of Things. IEEE Access. 4, pp. 2292–2303 (2016).

[6] Shetty, S., Red, V., Kamhoua, C., Kwiat, K., Njilla, L.: Data provenance assurance in the cloud using blockchain. In: Proc. SPIE 10206, Disruptive Technologies in Sensors and Sensor Systems. May 2 (2017).

[7] Liang, X., Shetty, S., Tosh, D., Kamhoua, C., Kwiat, K., Njilla, L.: ProvChain: A Blockchain-Based Data Provenance Architecture in Cloud Environment with Enhanced Privacy and Availability. In: 2017 17th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGRID). pp. 468–477. IEEE (2017).

[8] Tosh, D.K., Shetty, S., Liang, X., Kamhoua, C.A., Kwiat, K.A., Njilla, L.: Security Implications of Blockchain Cloud with Analysis of Block Withholding Attack. In: 2017 17th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGRID). pp. 458–467. IEEE (2017).

[9] Barger, A., Manevich, Y., Bortnikov, V., Tock, Y., Factor, M., Malka, M.: Shared Cloud Object Store, governed by permissioned blockchain. In: Proceedings of the 11th ACM International Systems and Storage Conference on - SYSTOR '18. pp. 114–114. ACM Press, New York, New York, USA (2018).

[10] Frey, D., Makkes, M.X., Roman, P.-L., Taïani, F., Voulgaris, S.: Bringing secure Bitcoin transactions to your smartphone. In: Proceedings of the 15th International Workshop on Adaptive and Reflective Middleware - ARM 2016. pp. 1–6. ACM Press, New York, New York, USA (2016).

[11] Shannigrahi, S., Fan, C., Papadopoulos, C.: SCARI. In: Proceedings of the Asian Internet Engineering Conference on - AINTEC '18. pp. 1–8. ACM Press, New York, New York, USA (2018).

[12] Park, J.H., Park, J.Y., Huh, E.N.: Block Chain Based Data Logging and Integrity Management System for Cloud Forensics. In: Computer Science & Information Technology (CS & IT). pp. 149–159. Academy & Industry Research Collaboration Center (AIRCC) (2017).

[13] Margheri, A., Schiavo, F.P., Vladimiro, S., Nicoletti, L.: FaaS : Federation-as-a-Service (Technical Report), SUNFISH project (EU-Horizon2020). 1–56 (2014).

[14] Madhumohanreddy, C., Raghavendra, G.: Blockchain-Based Database to Ensure Data Integrity in Cloud Forensics. Int. J. Sci. Res. Comput. Sci. Eng. Inf. Technol. © 2018 IJSRCSEIT. 4, 2456–3307 (2018).

Blockchain for Cybersecurity : Working Mechanism, Application areas and Security Challenges

Mini Sharma
Dept. of computer science and engg.
State Institute of Rural Development
Ranchi, Jharkhand, India
mini_kensharma@yahoo.com

***Abstract*—Blockchain is receiving huge attention from researchers and various organizations as it brings forth magical solutions to the classical centralized architecture problems. Blockchain, either private or public, is a kind of distributed ledger for maintenance of integrity among various transactions by the process of ledger decentralization among the participating end users. On the contrary, Internet of Things (IoT) is the internet revolution that facilitates connection of all end user devices over the internet so that they can share their services and applications for improvement of daily living standards. Centralized IoT provides numerous benefits, however, it also brings forth several security challenges. Integrating blockchain with IoT is an effective way to resolve these security challenges. This paper presents a detailed review that highlights this integration process. The paper presents the working mechanism of the blockchain and how it changes the nature of value. Application areas and security challenges are also outlined and discussed in the paper.**

Keywords—Blockchain; cybersecurity; cryptocurrency; bitcoin; security; privacy.

I. INTRODUCTION

In the absence of audit mechanisms or verifications, information systems' trust becomes problematic. This issue becomes severe when sensitive information has to be dealt with. In 2008 [1], Satoshi Nakamoto came up with two essential theories that had great outcomes. First one being Bitcoin, which is a virtual cryptocurrency. Bitcoin upholds its value exclusive of involvement of any financial or centralized specialists [2, 3]. Instead, coin is kept jointly and safely with the aid of a dispersed P2P network of actors that create demonstrable network. Second theory is Blockchain that has become very popular [4, 5].

Blockchain is a method that permits transactions to be tested with the help of a group of untrustworthy actors. It delivers a transparent, safe, auditable, dispersed and absolute record. Blockchain thus can be referred completely and provides access to all transactions occurring in the system. Information is contained in a chain of blocks in the Blockchain protocol. Every block here contains a collection of Bitcoin transactions implemented at some point of time. Each block has a reference to the previous block, hence creating a chain [6, 7].

Blockchain has huge number of advantages that may be helpful to resolve the cybersecurity problem. Firstly, blockchain is a doubtful gadget in which humans' considerations does not have any meaning. It considers any kind of attack may be outsider or insider attack, so it is totally unbiased of human ethics. Secondly, blockchain is unchangeable, so absolutely everyone can keep facts and secure things using distinct cryptographic capabilities such as digital signatures and hashing. As quickly as records shaped as a block inside the blockchain, their deletion and alteration are not possible. Thirdly, blockchain is capable of incorporating several users, so adding or changing a block wishes to be confirmed through the most of users thereby decreasing the probability of attacks on the system. [8]. Integrating the song enterprise with blockchain technology can resolve numerous issues related to ownership and transparency price. The Blockchain may be used to generate a particular disperse database for protecting the information and ledger rights. Also, clever agreement offers a digital and relaxed contract helpful in tracking industries [9, 10].

Numerous governments begin to adopt Blockchain era to aid numerous public offerings to their citizens. The Estonian government is an example of this as they have utilized blockchain technology for permitting the residents to perform numerous tasks using their identification cards which include vote casting, sign up for their agencies, order pay taxes and scientific prescriptions [11]. Similarly, the United Kingdom paintings and pensions branch has begun to undertake the blockchain for the enhancement of payments welfare. Additionally, Sweden had come up with actual estate transactions related to blockchain after carrying out several tests [12].

The remainder of the paper is organized as follows. Section II explores the various security requirements for routing mechanism in IoT. Section III of the paper focusses on how the blockchain brings change in the nature of value. Section IV presents the detailed working mechanism of the blockchain along with highlighting the major differences between federated, public and private blockchain. Various applications areas for blockchain is outlined in section V of this paper. Section VI presents the security issues and challenges in blockchain. Finally, the paper concludes in section VII with several open research challenges.

978-1-7281-0283-2/19/$31.00 ©2019 IEEE

II. SECURITY REQUIREMENTS FOR THE ROUTING MECHANISM IN IOT

Routing protocol for low power and lossy networks (RPL) provides a framework for adapting the requirements of different classes of applications. Here the internal operation of RPL is discussed and then the security mechanisms are discussed to protect communications during routing.

A. Routing in RPL

Routing in RPL must adapt to various requirements of the applications and RFC facilitates the optimization requirements. Requirements for various applications are defined, for urban applications in RFC 5548 [13], for industrial applications in RFC 5673 [14], for home automation in RFC 5826 [15], and for building automations in RFC 5867 [16]. RPL generates Directed Acyclic Graph (DAG) which is decomposed to Destination Oriented DAG (DODAG). A single DODAG having only one single root is the simplest RPL topology. Multiple RPL instances may run concurrently, each having varied optimization objectives, on the same network. RPL supports three basic traffic topologies: Point-To-Multipoint (P2MP), Multipoint-To-Point (MP2P) and Point-To-Point (P2P). RPL supports different control messages: DIO (DODAG Information Object), DAO (DODAG Advertisement Object), DIS (DODAG Information Solicitation), CC (Consistency Check), and DAOACK (DAO Acknowledgement) messages. A node computes their rank by transmitting DIO messages having another nodes information. They compute their rank so that they can join any alive DODAG and select preferred parents among all existing neighbours in that DODAG. DIS and DIO messages are used for upwards route establishment in the RPL tree while DAO messages establish downward paths to the root. CC messages are responsible for counter value synchronization among communicating nodes. Upon reception of DIO messages, DAO messages trigger. Control messages of RPL are encapsulated within ICMPv6 packets.

B. Security in RPL

The security field information denotes the security level and the cryptographic functions employed to deliver the desired security. This field does not include the signature or the Message Integrity Code.

Support for Data Authenticity and Integrity: RPL employs AES/CCM and 128-bit key to support integrity by generating MAC. To support both data authenticity and integrity it generates digital signature by employment of RSA with SHA-256. RFC 6550 uses MAC-64 and MAC-32 authentication codes to provide authenticity, integrity and confidentiality.

Support for Replay Attack Protection: A CC control message (Consistency check), facilitates a sensing node to validate current counter value of other nodes. The *Counter* field provides semantic security against replay attacks.

Support for Confidentiality: RPL control messages supports delay protection and confidentiality. AES/CCM forms the basis for supporting security. These control messages are protected by the use of both an authentication suite and integrated encryption. Both employ different algorithms for authentication and encryption. Since the IPv6 and ICMPv6 header is required for correctly decrypting the packet, these are not encrypted.

Support for Cryptographic Key Management: It is the Key Identifier Mode that determines the way to determine the cryptographic key; either it is done implicitly or explicitly. RFC 6550 supports varied key management approaches by defining different values for KIM field. This field is subdivided further into two subfields: *key index* and *key source* subfields. *Key index* subfield provides unique key identification while the *key source* subfield logically identifies the group key.

C. Security modes in RPL

There are three security modes in the existing RPL specification.

- *Unsecured:* No security to the routing control message is applied in this mode.
- *Preinstalled:* This is employed with the use of preconfigured symmetric key to couple an existing RPL instance. This supports confidentiality, authentication and data integrity for the routing control messages.
- *Authenticated:* This mode is best suited for devices that operate as a router. A device using preconfigured key initially joins the network and then obtain cryptographic key in the *preinstalled mode* from a trusted key authority. Then it may start deport itself as a router.

III. BLOCKCHAIN BRINGS CHANGE IN THE NATURE OF VALUE

Rare gold currency requires efficient power works by labors for its mining in industrial age. Thus, in typical bureaucratic nation, a classic labor unit can be used to measure "labor's worth". In the date nation, the word "labor's worth" demanded to be re-defined since it signifies that worth is built by any labor variety. Indeed, in the plan of typical theory this idea express shuffling value created by human labor in the series of material transformation. Brain work, or the course of data being processed by a single person, was no more treated by the prototype to be beneficial. This honest stand was in beginning invaded by members of historical school directed by F.List, who made an crucial step against initialing the political nation of the data culture. As he brings it, "those who fatten pigs, make bagpipers or prepare pills are productive, but the instructors of youth and of adults, virtuosos, musicians, physicians, judges and administrators are productive in a much higher degree. The former produce values of exchange, and the latter productive powers".

The labor approach of appraisal and the innovative one rest upon the completely different fair production, but are connected by methods. In both inceptions, mean life time is pretended to be the basic principle of worth. The typical labor

thesis cope with the worth is set up in the procedure of costs of typical labor ability, which on average, away from any extra growth, live in the creature of all standard individual. The innovative theory, on the opposite proclaim that worth is builded by innovative labor. H. Bergson even declared that "time is invention on it is nothing at all". As long as the entire extent of community representative breaks into classic and innovative part, there is completely understandable connection betwixt the labor and innovative worth; the considerable is one of them, slighter is the another, and vice versa. Therefore, as the part of innovative labor in the society grows, the realm of goods interchange gets decreased, and the site of value interchange is loaded by innovative value. The difference betwixt the typical presumption of typical bureaucratic nation and the data nation thesis is connected is dissimilar fair and observed bases. During the typical thesis was evolved when human labor proved more powerful, the data theory is proceeding shape in the time when the superiority of innovative labor is becoming more and can be split into easy and creative component, the depiction of actuality stated either of both theories individually is most simplified and can be effortlessly, and justifiably, censure by the opposing school of economics. Example, traditional followers of A. Smith are unlikely to concur with the declaration that in the future manual labor will be disadvantaged of worth and manual way of production would not think to be as capital. As yet, consistent protest could be embossed to the typical thesis in its structure, innovative labor, together with its outcome, is act towards as lacking of worth, and sole innovative ability has reportedly not anything to do with capital. In the era of data, the next significant scarce resources are information created in the way of innovative activity. The worth formation now needs both easy and innovative work, calling for intro of a new component of value. What begins to be a such component is bitcoin formulated as a prize for utilizing electric power and calculate processing work. Embodiment of a multisided manual extent into the new idea of worth indicates multi worth measures, i.e. movement of different currencies at the same time.

IV. WORKING MECHANISM OF BLOCKCHAIN

As earlier said, that the Integrity of transaction is maintained by distributed ledger and decentralization in blockchain. Let's talk about centralized architecture or classical ledger before discussing blockchain works. Ledgers were used for long time as means by governments and bankers to store land possession transactions and other activities like maintaining transaction record. Building and maintaining a trust relationship were major problem between certain transaction parties, government office or bank was used to accomplish required changes as central authority for defining who possess what and transactions contracts. Therefore, central authority is only able to distinguish between fake and genuine transactions. The following table 1 presents the major differences between Federated, public and private blockchain.

TABLE I. DISTINGUISH BETWEEN FEDERATED, PUBLIC AND PRIVATE BLOCKCHAIN.

Item	Federated	Public	Private
Speed	Faster and lighter	Slower	Faster and lighter
Immutability	Could be tampered	Nearly impossible to tamper	Could be tamper
Asset	Any Asset	Native Asset	Any Asset
Access	Write/Read for multiple selected organizations	Write/Read for anyone	Write/Read for a single organization
Consensus process	Known identities and Permissioned	Anonymous and Permission less	Known identities and Permissioned
Security	Voting/multi-party consensus and Pre-approved participants.	Proof of stake, Consensus mechanisms and Proof of work	Voting/multi-party consensus and Pre-approved participants.
Network	Partially decentralized	Decentralized	Partially decentralized
Efficiency	High	Low	High

The required trust is built by the ledger manager (government office or bank), since access to details on the ledger is controlled by the centralized manager people who can buy and sell without worry. Totally centralized ledgers are an organization or third-party person which has complete control over transaction management and trusted by every user. Since for only ledger manager the contents are visible that's why ledgers are black boxed. In terms of maintaining and storing transactions similar functions is provided by blockchain, but there is no requirement of third party. Transaction by ledger decentralization is verified by solving the central authority problem in which every participating user holds an original ledger copy within the blockchain ledger. Request for addition of transaction can be done by any participating user; however, transaction is alone putted to the block and alone in the blockchain network large number of users verifies it. For generating protected and fast ledger an automatic examination is done reliably for every user, which helps in making blocks and transaction significantly tamper-proof [17].

A transaction is linked and added with other transactions in a block if it is verified. The transaction is linked in the ledger with previous blocks through hash function and timestamp. Blockchain is created by these chains of blocks. As soon as the block is created, every participating user inside the blockchain community begin to search for the following block with the aid of seeking to answer the complicated mathematical function along with create an authentic transaction in the form of encrypted block to include it in ledger. This technique is referred to as mining, where every customers (minors) compete to create a novel block. The primary minor to create an authentic block and include it to the ledger. Expenses are carried out to every transaction. As blocks contain a huge quantity of transactions that are delivered time and again, minors may be able to gather more fees.

The ledger kept by using every involved customer inside the community is updated after the singular block is joined. If the newly introduced block is confirmed by means of all taking part customers and every transaction are actual, the block could

be putted and stays permanently within the ledger as a public document. If a warfare is determined then the block is discarded. Distorting a classical ledger desires an assault at the 3rd party (centralized supervisor). At the same time as the blockchain is set, so in case any adversary tries to regulate any deal taking place, this can want multiple repeated PoW computations for the concerned block and every other block in a while. These computations are heavily tough to perform unless majority of the users lying within the blockchain network are adversaries. Also, the opportunity of using a fraud ledger does no longer exist when you consider that all taking part users possess their very own genuine ledger replica to evaluate with [18] suggests the waft system of a standard economic transaction with the help of blockchain while a consumer A desires to give money to person B. The course starts while person A requests to combine a ledger and a block that includes information related to his financial transaction. Once the block increases, it is broadcasted across every participating customer inside the network to get itself certified. While the new block is established by way of every participating customers in the community, the introduction of the block with the ledger having transfer operation might be finished. At ultimate person B can get the money.

V. APPLICATIONS AREAS OF BLOCKCHAINS

There are numerous programs that reap advantages from various talents of Blockchain technology. These programs are detailed in the section below.

A. Music Industry

Because of the development of the internet technology and accessibility to different streaming centers over the net, the track employer has end up one of the programs that can advantage from excellent advantages furnished by means of using blockchain era. The song organization involves entities selection together with artists, publishers, streaming, and carrier businesses. The tune ownership has modified and come to be harder because of the internet boom. There may be a want for transparency within the possession bills for artists and songwriters [19].

B. Education

Education is another field that began to consider blockchain in thrilling and present-day packages alongside control of achievements and records, evidence of studying, control of recognition and management of pupil facts. As a decentralized database we can use blockchain to keep top notch forms of schooling records completely. Thereby the universities can may adopt confirmable certificate and cryptographically-signed on the blockchain which permit every students and employers to access it effortlessly [20]. Blockchain integration for gaining knowledge of societies lead to revolutionary Instructional programs which forma a novel learning model leading to alternate of thoughts and ideas in integration to a tracking device that evaluates the knowledge of consequences. The blockchain may be used within the law of contracts and bills to file academic development and assess learning together with paying tuition expenses by peer-coaching with different students.

C. Pubic Services

Records generated by way of governmental groups are internally opaque and fragmented to residents and corporations. Even as with the use of blockchain generation, information may be created and proven speedy with ensuring transparency and security of information. Blockchain capabilities inclusive of time-stamping and virtual signatures are estimated to offer infinite benefits in public offerings to permit residents to generate debts and address transactions independently, authorities' officers and different third parties.

D. Healthcare

Blockchain can also solve interoperability troubles of the prevailing healthcare systems. It enables healthcare objects with researchers to gain percentage of their Electronic Health Report (EHR) in a protected and secure manner. Also, it permits improvement of medical doctor endorsement along with medical care. Managing the healthcare statistics whether or not via storing or studying isn't a clean operation especially concerning records privacy. Additionally, improving privacy also ensures adoption of private blockchain which permits handiest specific men and women to store or alter the clinical records [21].

E. Cybersecurity

Safety is premiere trouble for every upcoming technology. Popular agencies confronted many security issues. As a site, Cambridge Analytical breached almost 50 million Facebook profiles in order to make them a target for personalized political commercials that affects America citizens in finalizing o the presidential elections. Moreover, during the period of 2016, yahoo, the well-known search engine, confronted a primary attack and compromised around one thousand million yahoo money. While security businesses did their studies approximately not unusual safety vulnerabilities, they determined that sixty-five% of the fact's breaches had been passed off because of susceptible, or stolen passwords. Additionally, they determined phishing emails steal touchy information together with password, financial statistics and username [22].

F. Voting

Vote casting is an essential device for any democratic authorities. It is an ought to system for every person; however, the conventional paper poll gadget of vote casting faces several problems. For instance, the gadget can't be computerized, and those must be present at the venue physically wherein the poll bins are stored which allows them to wait in traces for long instances. Additionally, vote counting takes a long time with the election may be ruptured by laying bogus ballot papers. Also, huge amount of papers is wasted in this operation [23]. With immutability and transparency capabilities of the blockchain, the vote casting system can be tons less complicated leading to simplified balloting system. The voter may create their vote (block), so as soon as the vote is performed, each person must verify the authenticity of the signature.

VI. SECURITY ISSUES AND CHALLENGES IN BLOCKCHAIN

Blockchain is not sincere. In the existing technology it raises many challenges that is to be solved. These challenges are described and summarized as follows.

- Scalability: As mentioned earlier, each transaction is saved in the allotted ledger. These transactions are growing each day. To validate a transaction, every person has to store it at the ledger to take a look at the supply of the existing transaction. Moreover, growing a block faces several problems concerning time and length. As an instance, bitcoin can only create nearly seven transactions according to second, which can't comprehend the processing wishes of billions of transactions in actual-time packages [24]. An important role is played by the transaction in the execution order considering the fact that minors choose to generate blocks with large transaction length as well as high transaction prices. This leads small transactions to have increased latency. A few researches tried to solve the scalability issues. For example, Bruce [25] proposed a garage optimization technique that considers the ledger and deletes antique transaction records. However, to address this challenge we need more research to be done.
- Privacy: Blockchain technology conserves various privacy concerns. The consumer makes use of nameless identification to create and verify transactions the use of their non-public and public key. However, in view that all taking part customers are only able to view transactions, therefore, the blockchain can't guarantee transactions' privacy. Consequently, the privacy trouble of blockchain technology needs greater studies to increase the adoption rate of blockchain in numerous applications.
- Wasted Resources: Until the instant, energy performance is one of the good-sized demanding situations in laptop engineering that wishes to be resolved. Taking the blockchain technology into consideration, the mining manner desires a huge amount of computation to affirm transactions in a comfortable way. However, it is critical to lessen wasted assets within the mining system. Several researchers proposed solutions to mitigate this issue. For instance, Janish [26] has proposed speeding the mining system with the aid of the usage of simultaneous Graphics Processing Units (GPUs) and Primary Processing Units (CPUs) in each machine.
- Data malleability: Data integrity is the major critical components inside the blockchain. Demand of the statistics is to not alter or tamper the transmitted or tested data. Malleability attack on records integrity suggests that the transactions signature confirms the ownership of bitcoin does no longer guarantee signatures integrity. Consequently, an outsider can rebroadcast, seize and adjust a transaction which reasons the transaction's author to consider that the transaction is not proved [27].
- Usability: This feature highlights the fact that blockchain Application Programming Interface (API) is tough to use. The primary target for all emerging technology needs to contain providing smooth and usable to-use interfaces for each customers and builders. Consequently, its miles important to upgrade the blockchain usability via offering the desired gear to permit users to investigate the entire blockchain network [28].
- Bootstrapping: This refers to transferring the existing commercial enterprise files, frameworks or contracts to the novel blockchain based era. This introduces a couple of migration responsibilities which is required to be carried out. As an instance, within the event of a land possession, the present forms require to be formatted and migrated to be equal for the blockchain form, which involves high cost as well as time.

VII. CONCLUSION AND FUTURE WORK

Blockchain received widespread attention in the day-to-day life. It possesses enormous potential and can be a integral part of almost every industry. Blockchain is a kind of tamper-proof distributed ledger that provides a decentralized environment for the transactions to be executed. It is also capable of solving numerous problems related to the traditional centralized model. On the contrary, IoT has came into existence as a new evolution of internet enabling the connection of huge number of devices and objects over the internet. Efficiency as well as human productivity can be improved with the help of actuators or sensors that collects meaningful information from these devices and objects. This leads to the integration of blockchain and IoT as the next level of development. In order to support the claim this paper presents a detailed review of the blockchain technology including the working mechanism, application areas and various security issues and challenges in blockchain.

REFERENCES

[1] Nakamoto, Satoshi. (2009). Bitcoin: A Peer-to-Peer Electronic Cash System. Cryptography Mailing list at https://metzdowd.com.

[2] Hughes, T.: The global financial services industry and the bitcoin. J. Struct. Financ. 23(4), 36–40 (2018). https://doi.org/10.3905/jsf.2018.23.4.036

[3] Koeppl, T., Kronick, J.: Blockchain Technology—What's for Canada's Economy and Financial Markets? C.D. Howe Institute Commentary 468, 2 February 2017. https://doi.org/ 10.2139/ssrn.292781

[4] Bhushan, B., & Sahoo, G. (2019). $$E^{2} SR^{2}$$ E 2 S R 2 : An acknowledgement-based mobile sink routing protocol with rechargeable sensors for wireless sensor networks. *Wireless Networks,25*(5), 2697-2721. doi:10.1007/s11276-019-01988-7

[5] Bhushan, B., & Sahoo, G. (2018). Routing Protocols in Wireless Sensor Networks. *Computational Intelligence in Sensor Networks Studies in Computational Intelligence,*215-248. doi:10.1007/978-3-662-57277-1_10

[6] Bohme, R., Christin, N., Edelman, B., Moore, T.: Bitcoin: economics, technology, and governance. J. Econ. Perspect. 2(2), 213–238 (2015). https://doi.org/10.1257/jep.29.2.213

[7] Davidson, S., De Filippi, P., Potts, J.: Economics of blockchain. In: Proceedings of Public Choice Conference. Fort Lauderdale (2016). https://doi.org/10.2139/ssrn.2744751

[8] K.K. Patel, S.M. Patel, Internet of things-IOT: definition, characteristics, architecture, enabling technologies, application & future challenges, Int. J. Eng. Sci. Comput. 6 (5) (2016) 6122–6131.

[9] Bhushan, B., Sahoo, G., & Rai, A. K. (2017). Man-in-the-middle attack in wireless and computer networking — A review. *2017 3rd International Conference on Advances in Computing, Communication & Automation (ICACCA) (Fall)*. doi:10.1109/icaccaf.2017.8344724

[10] Bhushan, B., & Sahoo, G. (2017). A comprehensive survey of secure and energy efficient routing protocols and data collection approaches in wireless sensor networks. *2017 International Conference on Signal Processing and Communication (ICSPC)*. doi:10.1109/cspc.2017.8305856

[11] M. Crosby, Nachiappan, P. Pattanayak, S. Verma, V. Kalyanaraman, Blockchain Technology Beyond Bitcoin, 2015.

[12] M. Roberto, B. Abyi, R. Domenico, Towards a Definition of the Internet of Things (IoT), IEEE Internet Things, 2015.

[13] M. Dohler, T. Watteyne, T. Winter, and D. Barthel, Routing Requirements for Urban Low-Power and Lossy Networks, RFC 5548, 2009.

[14] Bhushan, B., & Sahoo, G. (2017). Recent Advances in Attacks, Technical Challenges, Vulnerabilities and Their Countermeasures in Wireless Sensor Networks. *Wireless Personal Communications,98*(2), 2037-2077. doi:10.1007/s11277-017-4962-0

[15] A. Brandt, J. Buron, and G. Porcu, Home Automation Routing Requirements in Low-Power and Lossy Networks, RFC 5826, 2010.

[16] J. Martocci, P. De Mil, N. Riou, and W. Vermeylen, Building Automation Routing Requirements in Low-Power and Lossy Networks, RFC 5867, 2010.

[17] C. Decker, R. Wattenhofer, Bitcoin transaction malleability and MtGox, in: Computer Security DESORICS, Lecture Notes in Computer Science, vol. 8713, Springer International Publishing, 2014, pp. 313–326.

[18] J. Yli-huumo, D. Ko, S. Choi, S. Park, K. Smolander, Where is current research on blockchain technology?—a systematic review, PLoS One (2016) 15–27.

[19] H.F. Atlam, R.J. Walters, G.B. Wills, Internet of Things: state-of-the-art, challenges, applications, and open issues, Int. J. Intell. Comput. Res. 9 (3) (2018) 928–938.

[20] M.U. Farooq, M. Waseem, A review on Internet of Things (iot), internet of everything (ioe) and internet of nano things (IoNT), Int. J. Comput. Appl. 113 (1) (2015) 1–7(0975 8887).

[21] H.F. Atlam, R.J. Walters, G.B. Wills, Intelligence of things: opportunities & challenges, in: IEEE 2018 Cloudification of the Internet of Things (CIoT), 2018.

[22] Chakarverti, Mohini and Sharma, Nikhil and Divivedi, Rajiva Ranjan, Prediction Analysis Techniques of Data Mining: A Review (March 11, 2019). Proceedings of 2nd International Conference on Advanced Computing and Software Engineering (ICACSE) 2019. Available at SSRN: https://ssrn.com/abstract=3350303 or http://dx.doi.org/10.2139/ssrn.3350303

[23] H.F. Atlam, A. Alenezi, A. Alharthi, R. Walters, G. Wills, Integration of cloud computing with Internet of Things: challenges and open issues, in: 2017 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData), June, 2017, pp. 670–675.

[24] ITU, Overview of the Internet of things, in: Ser. Y Glob. Inf. Infrastructure, Internet Protoc. Asp. Next-Generation Networks—Fram. Funct. Archit. Model, 2012.

[25] RFC 7452, Architectural Considerations in Smart Object Networking, Computer Networks, 2015.

[26] L. Atzori, A. Iera, G. Morabito, The Internet of Things: a survey, Comput. Netw. 54 (15) (2010) 2787–2805. 36 Hany F. Atlam and Gary B. Wills ARTICLE IN PRESS

[27] H. Ma, Internet of Things: objectives and scientific challenges, J. Comput. Sci. Technol. 26 (2011) 919–924.

[28] J. Gubbi, R. Buyya, S. Marusic, M. Palaniswami, Internet of Things (IoT): a vision, architectural elements, and future directions, Futur. Gener. Comput. Syst. 29 (7) (2013) 1645–1660.

Consensus Protocols for Blockchain-based Data Provenance: Challenges and Opportunities

Deepak K. Tosh[1], Sachin Shetty[2], Xueping Liang[3], Charles Kamhoua[4], Laurent Njilla[5]
[1]Department of Computer Science, Norfolk State University, Norfolk, VA
[2]Virginia Modeling Analysis and Simulation Center, Old Dominion University, Norfolk, VA
[3]College of Engineering, Tennessee State University, Nashville, TN
[4] Army Research Lab, Adelphi, MD
[5]Cyber Assurance Branch, Air Force Research Laboratory, Rome, NY
dktosh@nsu.edu, sshetty@odu.edu, xliang@tnstate.edu, charles.a.kamhoua.civ@mail.mil, laurent.njilla@us.af.mil

***Abstract*—Blockchain has recently attracted tremendous interest due to its ability to enhance security and privacy through a immutable shared distributed ledger. Blockchain's ability to detect integrity violations are particularly key in providing assured data provenance in cloud platform. The practical adoption of blockchain will largely hinge on consensus protocols meeting performance and security guarantees. In this paper, we present the design issues for consensus protocols for blockchain based cloud provenance. We present the blockchain based data provenance framework for cloud. We find that there are performance and security challenges in adopting proof-of-work consensus protocol within this framework. We present unique design challenges and opportunities in developing proof-of-stake for data provenance in cloud platform.**

***Index Terms*—Blockchain, cloud computing; Data provenance; Distributed consensus; Proof of work (PoW); Proof of stake (PoS)**

I. Introduction

Blockchain technology has attracted tremendous interest from a wide range of stakeholders, including finance, healthcare, utilities, real estate, and government agencies. Blockchains are shared, distributed and fault-tolerant database that every participant in the network can share, but no entity can control. Blockchain's distributed database maintains a continuously growing list of records, called blocks, secured from tampering and revision by distributed storage and continuous verification. The blocks contain a temporal listing of transactions that are stored in a public ledger using a persistent, immutable and append-only data structure that is globally accessible by every participant in the underlying peer-to-peer network.The technology is designed to operate in a highly contested environment where, adversarial strategies are nullified by harnessing the computational capabilities of the honest nodes such that information exchanged is resilient to manipulation and destruction. Tampering of blockchains is extremely challenging due to use of a cryptographic data structure and no reliance on secrets.

Blockchain utilizes a distributed consensus algorithm over a decentralized peer-to-peer network for verification of transactions prior to adding blocks to the public ledger. The verification process is determined by users and does not require a centralized administrator. The distributed consensus protocol ensures that the newly added transactions are not at odds with the confirmed transactions in the blockchain and maintains the correct chronological order. The newly added transactions which are waiting to be confirmed are packed in a block and submitted to the blockchain network for validation. In order for the block to be validated, the nodes in the peer-to-peer network solve a crypto-puzzle using computational resources at their disposal. The block is appended to the blockchain once a solution is discovered. This approach is known to be proof of work (PoW) [1] and it turns out to be an energy inefficient consensus approach [2] because the annual estimated electricity consumption of Bitcoin is 15.77 Terawatt hour, which is 0.08% of world's electricity consumption.

With PoW approach, miners opt for various specialized hardware to achieve their computational ability while at the same time they invest in electricity to operate and cool down these hardware. Knowing the eventual goal of the miners is to win the block-adding race so that they can be rewarded, a significant amount of energy is required to do so. The power that is spent to reach consensus using the PoW approach is mostly used in computing the irreversible SHA256 hashing function. Since the value of direct incentives will diminish eventually, the critical question of "how the PoW miners will be motivated to mine?" has to be addressed so as to smoothly run the consensus process. In addition to the energy wastage issue, there exists other security concerns which we describe in the next section. Since blockchain is turning out to be a robust tool to maintain an incorruptible distributed ledger, it has immense usefulness in cloud computing domain, especially in maintaining provenance of data objects across the cloud infrastructure. The existing PoW consensus can have additional overhead to maintain the blockchain in cloud domain. Therefore, it is of utmost importance to design and develop alternate form of consensus particularly for cloud provenance. In this paper, we study the concerns of PoW consensus and the advanced consensus protocols proposed to alleviate issues raised by PoW. We also expressed the need of provenance framework in cloud computing domain and provide an architecture for blockchain enabled provenance system, namely BlockCloud. Finally, we discuss the design

Approved for public release, distribution unlimited 88ABW-2017-4823, dated 28 September 2017.

challenges and opportunities in developing a proof of stake based cloud data provenance framework.

The paper is organized as follows. In Section II we present the PoW challenges and discuss usability of proof of stake consensus model. The Section III expresses the importance of data provenance in cloud systems. The architecture of PoS enabled blockchain cloud is presented in Section IV. Future research directions are briefed in Section V and Section VI concludes the paper.

II. PoW Challenges and Other Consensus Models

A. PoW Attack Concerns

Furthermore, the PoW mechanism's computational procedure for creating the ledger can be exploited by adversaries to impact the integrity of the blockchain. Recently, researchers have listed attacks which exploit the PoW consensus procedure, such as: selfish mining, 51% majority [3] manipulation attack, consensus delay [4] due to distributed denial of service, pollution log, blockchain forking, orphaned blocks, deanonymization, and block ingestion [5]. The impact of the attacks on the PoW mechanism are not felt the same for all blockchain applications. For instance, the 51 % majority manipulation attack is unlikely in cryptocurrency, but more likely in cloud platform, wherein adversaries can create collusion among several virtual machines across federated cloud providers to cause consensus delay.

The following are some blockchain anomalies that can stem for either attacks or random faults or both:

- Selfish mining - The selfish mining attacks are motivated by increasing returns for adversaries and impacting the fairness. The modus operandi for the attack involves adversaries selectively choosing the timing to publish discovered blocks. The intent of the attack is to maximize the chances of generating a longer blockchain than the rest of the network by consistently claiming block rewards for the batch of released blocks. These selfish mining attacks can invalidate blocks of honest nodes and negatively impact the reliability, fairness and robustness of the network.
- 51% majority manipulation - This attack is a coordinated effort by a group of adversaries to manipulate the blockchain network by controlling over a majority of the network's computational power. The major impacts of this attack are: 1) invalidate transactions/blocks at will by denying acceptance in the network; 2) equivocation of transactions and/or blocks by reversing the confirmation; 3) prevent other nodes from adding any blocks for different periods of time.
- Consensus delay - In this attack, adversaries inject false blocks and/or launch distributed denial of service to cause delays in reaching consensus in the blockchain. The impact of this attack on time-critical applications is devastating when consensus needs to be achieved within a short period of time.
- Blockchain fork - Blockchain forks are caused when nodes in the peer-to-peer network have diverging views about the state of the blockchain over long periods of time. These forks can be a result of accidental actions, such as, protocol malfunction and client software upgrades or intentional actions, such as, Trojan nodes that poison the validation process. Adversaries can exploit the presence of blockchain forks to create instability and mistrust among the nodes in the network.
- De-anonymization - The public availability of the blockchain transactions makes it possible to use data analytics techniques to analyze vast amounts of data within them. The analysis could provide valuable information that can sometimes reveal the individual transactions of participants and disclose their identity.

B. Proof of Stake (PoS) Consensus

To circumvent the problem, various consensus mechanisms, such as proof of stake [6], proof of activity [7], variants of Byzantine fault tolerant (BFT) algorithms [8], proof of space [9], are proposed that aim to avoid depletion of computational resources. Among those, the Proof of Stake (PoS) consensus protocol is an interestingly attractive one, which provides the block-inclusion decision making power to those entities that have stakes in the system irrespective of blockchain's length or history of the public ledger. The principal motivation behind this scheme is to place the power of leader-election in blockchain update process into the hands of the stakeholders. This is done to ensure that the security of the system will be maintained while the members stakes are at risk. Roughly speaking, this approach is similar to the PoW consensus except the computational part. Hence, a stakeholder's chances to extend the blockchain by including its own block depends proportionately on the amount of stake it has in the system. We can observe that the PoS consensus mechanism requires the stakes to be pre-distributed at the beginning of the process which was not the case with the PoW approach. However, we see an opportunity of exploiting the PoS based consensus to maintain data provenance blockchain in the cloud computing architecture since most cloud users have pre-acquired their necessary virtual computing resources for their operations.

The initial proof of stake (PoS) design included the age of cryptocurrencies and the total amount to define stake of each miner in the system. To gain the privilege of generating a PoS based block, the miner has to make a special coinstake transaction to himself so as to reset the coin age and prove that its stake is valid. According to their approach, a miner has a chance to extend the blockchain with his block having total unspent output $\mathcal{U}$, given the following condition is satisfied. Here, the unspent output refers to the output of a transaction that is not yet an input of another transaction, which means that the output is not been spent.

$$hash(hash(\mathbb{B}_{prev}), \mathcal{U}, t) \leq d \times balance(\mathcal{U}) \times age(\mathcal{U}) \quad (1)$$

where, $\mathbb{B}_{prev}$ is the previous block on which blockchain is to be extended, $balance(\mathcal{U})$ is the miner's stake amount, $age(\mathcal{U})$ is the aggregated age of the stake, and d is the mining

difficulty, which is of higher value unlike the traditional PoW based consensus. As seen in the Eq. 1, the computed hash value in the left side of inequality depends on the miners stake amount, so a large stakeholder can easily find a hash and hence has higher probability of adding its block in the blockchain. However, it is challenging to adopt this form of PoS in blockchain based cloud data provenance framework because the resources in cloud do not exhibit highly correlated characteristics with the tokens of cryptocurrency domain.

III. Data Provenance in the Cloud

Cloud computing environments are dynamic and heterogeneous and involve several diverse and disparate software and hardware components which are manufactured by different vendors and require interoperation. assurance of the ancestry of the data (where the data came from) is a challenge in cloud environments. Data provenance addresses the ancestry of the data based on detailed derivation of the data object. If true data provenance existed in the cloud for all data stored on cloud storage, distributed data computations, and data exchanges and transactions, then detecting insider attacks, reproducing research results and identifying the exact source of system or network intrusions would be achievable. Unfortunately, the state-of-the art in data provenance in cloud does not provide such assurances and there is a need to develop techniques to address this challenge.

Current state-of-the art provenance systems in the cloud support the above tasks through logging and auditing technologies. To identify the origin, cause and impact of security violations in cloud infrastructures will require collection of forensics and logs from the diverse and disparate sources which is an unduly heavyweight task. At the same time, logs only provide a sequential history of actions related to every application. The provenance data provides the history of the origins of all changes to a data object, list of components that have either forwarded or processed the object and users who have viewed and/or modified the object and has enhanced requirements for assurance. Besides the limited functionality of comparing logs to audit data, today's provenance functions are done in a private manner to establish ownership of digital assets. This, in turn, has a few limitations. First, the cost of provenance is high and prohibitive, in the sense that a provenance assurance should be established for each individual cloud service. Second, the process of provenance assurance, when multiple players are involved as is typical in cloud computing, lacks transparency. As such, moving to a more transparent, open, and public system is desirable.

A. Need of Blockchain in Maintaining Data Provenance

Blockchain technology provides such a capability and resolves many needed functionalities and properties for effective provenance in cloud [10]. In essence, a blockchain is a peer-to-peer ledger system, where information that constitutes provenance for physical, virtual, and application resources can be stored publicly for transparent verifiability and audit. As such, both transparency and cost effectiveness are provided, while access control and privacy for individual users of the ledger are ensured through encryption techniques, where individuals can see only parts of the ledger that is related to them.

Thus, blending the blockchain technology into the cloud environment can lead to achieve the task of data provenance, where the cloud nodes implicitly create a distributed network to record provenance data in the distributed and fault-tolerant ledger that is secured with a strong cryptographic notion. This distributed ledger of the blockchain is to be updated by all the nodes in cloud environment, but this depends on a certain rule that every node agrees upon. Designing such a consensus mechanism that ensures consistency in the blockchain is challenging. The traditional PoW consensus approach may not be applicable especially in the cloud computing domain due to its large computational power requirement. Therefore, it is important to investigate the usefulness of the proof of stake (PoS) based consensus method in this situation.

IV. Architecture and PoS Model for Blockchain Cloud (BlockCloud)

BlockCloud is our proposed data provenance architecture built on top of blockchain technology, which will provide the ability to audit data related operations for cloud users and providers in the federated cloud setup. It aims to achieve the following four objectives.

- Cloud Data Provenance: User operations are monitored in real time to collect provenance data to support access control policy enforcement [11] and intrusion detection.
- Proof-of-Stake Validation - As opposed to our previous PoW based provenance model [12], the consensus process BlockCloud will be driven by the staked resources of virtual machines housed in a federated cloud computing environment. The presence of validating VMs will provide supervisory control over the consensus process.
- Tamper-proof Environment: Data provenance record is collected and then published to the blockchain network which protects the provenance data. All data on the blockchain is shared among blockchain network nodes. BlockCloud builds a public time-stamped log of all user operations on cloud data without the need for a trusted third party. Every provenance entry is assigned a blockchain receipt for future validation.
- Provenance Data Validation: Data provenance record is published globally on blockchain network, where a number of blockchain nodes provide confirmation for every block. ProvChain uses blockchain receipt to validate every provenance data entry.

Our proposed BlockCloud architecture, as depicted in Figure 1, achieves the above objectives by monitoring user activities in real time using hooks and listeners (special classes of event listeners) so that every user operation on files will be collected and recorded for generating provenance data. Each piece of provenance information is referred as transactions that is broadcasted to the core of blockchain network created by a specific set of validating VMs. The validators collect the raw transactions and create their blocks individually, then

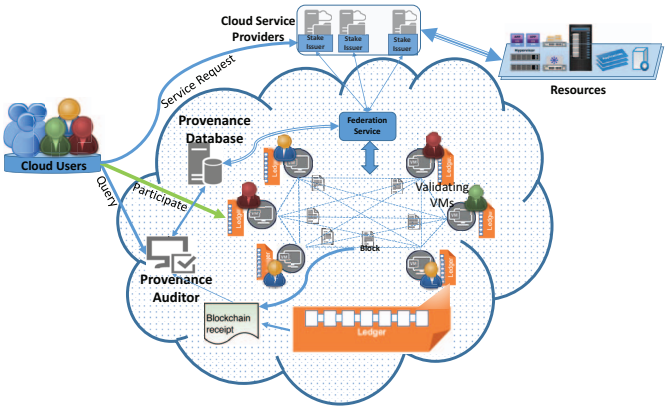


Fig. 1: BlockCloud Data Provenance Architecture

wait for the consensus to converge after which a leader will be selected to extend the blockchain. Every validator node on the blockchain network can verify the data operations or transactions before including into their block to ensure that the provenance data is authentic. We consider the presence of a provenance auditor in our model who collects the confirmation of each transaction block that is successfully added into the blockchain and records in the searchable provenance database. To maintain the privacy of cloud users, double hash of their user ID is used while broadcasting their transactions so that validators and provenance auditor cannot determine the exact user identities involved in those data operations.

In regard to the consensus process, we have considered the proof of stake model, where a selected set of validators among all the cloud users participate. To resolve the issues of initial resource distribution as well as stake validation, we assume a federation service in our model. This service starts at the beginning of consensus process and continuously work alongside the cloud validators to verify their stakes and select leader of every round in the blockchain network. The federation service is also considered to have the authority to decide the reward for successful block extension and at the same time to punish if validators act maliciously. Given the stakes of validators are used to decide leaders in the consensus process, they are not allowed to use the staked resources during the slot and it is monitored by the federation service. If it is found to be acting dishonestly, the staked resources will be forfeited. This will ensure that the cloud validators cannot abuse the proposed provenance system.

Now, we describe the role of BlockCloud's critical components as illustrated in the figure.

- Cloud User: A user in the cloud setup may have ownership over its own data and/or shared relationship on other users' data. They may participate in the PoS based consensus of provenance blockchain to help each other in creating a tamper resistant cloud environment. Participation in provenance may be voluntary or reward-driven but it comes at a cost of devoting some of their resources for achieving consensus.
- Cloud Service Provider (CSP): The cloud service provider offers a cloud storage/compute service and is responsible for user registration. Multiple CSPs form a federated environment that allows the cloud users to dynamically stake the virtual resources allocated to them. It is possible that they can be a part of the provenance system by participating in the blockchain consensus, which will benefit them in avoiding unauthorized data manipulations across the network.
- Provenance Database: The provenance database records all provenance information regarding every data object belonging to one or multiple cloud users. The provenance information is validated by the miners of the blockchain network, and is used for detecting malicious behaviors in the system. All data records are implicitly anonymized so as to protect cloud users' identity and privacy.
- Provenance Auditor (PA): PA can retrieve all the provenance data from the blockchain into the provenance database and validate the blockchain receipts. The PA maintains the provenance database but cannot link the provenance entry to the corresponding data owner. PA also acts as a mediator to query for provenance records from users so that every user can keep only the block headers with them and request the PA to fetch the detailed records whenever necessary.
- Blockchain Network: The blockchain network consists of a set of participating cloud users. This network only serves to run the consensus procedure by communicating transaction blocks with each other. This may not be a fully connected network as exhibited in the diagram, but can have a multi-hop topology at its core.
- Federation Service: This is a resource controlling entity that manages the process of stake allocation and verification. It also serves for deciding leaders in each block addition round based on the amount of staked resources.

Following the above architecture design, the blockchain enabled data provenance service with the proposed Proof of Stake based consensus in the cloud semantics is illustrated in Algorithm 1.

A. Stake in Cloud Computing Environment

From the cryptocurrency sense, the definition of stake is quite understandable, however, there is no prescribed way to correlate the definition of stake in a cloud system. So, we propose a stake model for the cloud system users as they participate in the consensus process. We consider the important resource components, such as CPU power, allocated memory, and network capabilities, as stakes for the cloud users, which are provisioned by the service provider in an on-demand basis.

B. Modeling Stake for a Cloud Instance

Considering the fact that users in a cloud environment are the entities, who occupy several virtual resources for their services and operations, such allocation of resources is tied to the idea of modeling stake for cloud users. For provisioning distributed consensus in the cloud environment, the stake must be defined in terms of the important resources a cloud user

Algorithm 1: Proof of Stake for BlockCloud

```
Data: TXs, Mining Peers
Result: New block generation and validation
initialization;
while true do
    peerNum ← number of peer nodes in the cloud;
    i ← 0;
    while i < peerNum do
        peer(i).stake = computeCloudStake();
        peer(i).possibility =
        computePossibility(peer(i).stake);
    end
    StakeWinner = StakeBiasedRandomChoose(peers);
    newBlock = StakeWinner.generateNewBlock(txs,
    StakeWinner.stake, prevHash);
    newBlock.broadcast();
    if peers.validate(newBlock) == true then
        ledger.append();
    else
        newBlock.discard();
    end
end
```

holds in the system. For simplicity, we consider the above three critical components/resources to define our stake model.

We consider participation in BlockCloud data provenance system to be either voluntary in nature or spurred by the incentive stemming from the possibility of getting rewarded in the future for maintaining the blockchain. To model the stake component, we assume a cloud user i has been allocated with total C_i number of CPU slices, S_i amount of storage in kilobytes, along with a data rate of D_i Kbps to perform its business operations and serve for maintaining the blockchain based data provenance. The above parameters are not the exhaustive list of resources that are needed to model the stake but it is still an open problem to come up with different innovative stake designs using several other cloud resources. The stake $\mathcal{X}_i = f(C_i, S_i, D_i)$ for user i can be defined as a function of above parameters, where, $f(.)$ is a transformation function of different cloud resources to a common scale that represents the stake of a cloud user i that is needed for the other members to compare and verify staked resources. Since the parameters of the function are homogeneous in scale across all the virtual miners in cloud, the function must be increasing in nature with respect to increase in quantity of each resource (C_i or S_i or D_i). Thus, $\frac{\partial \mathcal{X}}{\partial C_i} > 0$, $\frac{\partial \mathcal{X}}{\partial S_i} > 0$, and $\frac{\partial \mathcal{X}}{\partial D_i} > 0$ for each cloud miner i. This property is a necessary condition to satisfy because the cloud users must have a common ground to compare the stakes for understanding who has more stake in the system, so that leader election in consensus process will be easier to achieve. For instance, when two stake amounts from user i and j are compared and found that $\mathcal{X}_i \geq \mathcal{X}_j$, then it is inherent that $(C_i, S_i, D_i) \succcurlyeq (C_j, S_j, D_j)$ is satisfied, where $\succcurlyeq$ represents a component-wise comparison. In other words, user i has staked more resources compared to j.

V. Research Directions

Despite the advantages of PoS based blockchain in ensuring data provenance in intra-cloud and inter-cloud environment, there are several research avenues to be explored in implementing a robust proof of stake consensus model for the cloud computing systems.

A. Role of Cloud Service Provider (CSP)

Since cloud computing platforms are designed to provide quick, on-demand access to private/public resources and aim to meet the service level agreements (SLA) in a consistent manner, the role of a cloud provider is critical in such an environment so as to manage the dynamism of workload and resource utilization. Although an CSP is responsible for providing a secure, reliable and highly available cloud environment to the end users, it is unethical to track the users' internal service data that are used for ensuring provenance. Therefore, the blockchain-enabled data provenance system for the cloud must resolve the challenge of whether to keep the CSP as a part of the blockchain consensus process or leave it out. Since, we consider the provenance service to be an integral component toward security-provisioning of the cloud environment, inclusion of the CSP in the consensus process will maintain the blockchain decentralization intact. Moreover, the nature of the blockchain, whether permissioned or permissionless, to adopt in the cloud provenance still needs to be analyzed. Exclusion of the CSP from the blockchain consensus could make the cloud users insecure because the provider still possesses the control of data flow in the cloud environment. Thus, this trade-off is a crucial. It needs to be resolved while designing the PoS based blockchain for cloud data provenance.

B. Initial Resource Distribution

To begin the blockchain consensus process in a cloud environment, the users first need to have an understanding of the amount of resources each user holds in order to verify their stakes in the system at later stages. In case of crypto-currency, this process involves members purchasing the bitcoins from an external agency and transferring to their wallet before they perform any transactions. However, when a similar mechanism is adopted in the design of BlockCloud, it poses the concern of whether to rely on the external provider or not because it may not be a part of the provenance system. Apart from this, revealing the distribution of resources among the peers (irrespective of inter/intra-cloud users) may disclose their stake-power, which can be of particular interest when a malicious cloud miner is co-resident. Thus, a robust mechanism is required to distribute the initial resources so that cloud users' privacy and secrecy, related to resource utilization, remains unaltered.

C. Amount to Stake

Considering the fact that cloud users have been provisioned resources to use the services, a portion of the provisioned resources must be reserved to enable PoS based consensus for the BlockCloud. For the consensus process operating in time-slotted implementation, it is assumed that the staked resources are hindered from running any user-level services for a particular slot unless consensus is achieved. However, at the end of each slot, cloud users have the option to modify the staked amount prior to proceeding to the next consensus slot. Holding up the staked resources will be helpful for avoiding untruthful behavior of users in the system, so that when any maliciousness is detected, the corresponding user will forfeit its stake. Although higher stakes in the system provides user a high chance to add its block and collect incentives from the transaction fees. There is still an open question: *What amount of resources can a user choose to dedicate for the consensus process so that it balances out the gains from service provisioning of such resources and participation in BlockCloud consensus?*

D. Incentivizing Cloud Stakeholders

Another important challenge in the PoS enabled BlockCloud is the identification of incentive mechanism to motivate the cloud users to participate in the consensus process. Serving in the blockchain consensus requires the users to dedicate some of their resources to serve for the rest of the cloud users. Hence, sufficient reward for the participants needs to be provisioned, else they may leave the framework. The reward component in the PoW based blockchain was to incentivize the winner with a specific amount of coins. However, monetary reward may not be feasible in the cloud computing environment. Thus, it is important to identify ways of rewarding the cloud stakeholders who play a major role in the consensus process. A possible solution to incentivize the cloud users can be the following: A certain percentage of the previously staked resources will be returned back to the user which can be used to run their services. Hence, if a user has higher stakes and truthfully participates in the blockchain consensus, it can effectively augment its computing power for usage toward serving its regular workload. However, for this scenario, an important design issue will be to identify the appropriate percentage of the stake to reward for survival of the BlockCloud because virtual resources cannot be created out of thin air.

E. Transaction Privacy

The blockchain involved in traditional crypto-currency achieves consensus by a set of miners who always compete among each other to extend the chain with their own block. Equivalently, there exists a set of privileged cloud users who are involved in maintaining consensus in the BlockCloud architecture. However, these virtual miners live in a federated cloud environment to perform the validity of cloud data operations by reaching to a common knowledge from the prior transaction details derived out of blockchain. Unlike the traditional blockchain where transactions refer to transfer of coin ownership, the transactions in BlockCloud includes records of various operations on different data objects created/shared by a single/multiple cloud users. Due to this, verification of coin ownership and validity of transactions are easier to perform in the traditional blockchain system, but in the case of BlockCloud, the actual stake and transactions included in a block are different. Hence, it is challenging for the virtual miners in the cloud environment to maintain the privacy of transactions while achieving consensus. So, the PoS consensus in BlockCloud will be negatively impacted unless there is a mechanism in place to derive the transaction history of various cloud data objects in cloud without hampering the privacy of data transactions performed by cloud users.

VI. Conclusions

Preventing malicious activities in the federated cloud environment requires assurance of data provenance so that every operation on data objects can be tracked effectively. Blockchain offers unique set of features to do so, however the underlying PoW consensus is inapplicable while integrating in cloud domain. We discussed the reasons which make PoW unfit for our case. We have proposed blockchain based cloud data provenance framework, namely BlockCloud, that incorporates PoS as consensus engine. Despite the benefits of PoS powered blockchain, we came across several design challenges while integrating the distributed ledger technology in cloud computing domain, which we discussed in details.

References

[1] S. Nakamoto, Bitcoin: A peer-to-peer electronic cash system (2008).

[2] Bitcoin energy consumption index, http://digiconomist.net/bitcoin-energy-consumption.

[3] I. Eyal, E. G. Sirer, Majority is not enough: Bitcoin mining is vulnerable, in: International Conference on Financial Cryptography and Data Security, Springer, 2014, pp. 436–454.

[4] 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.

[5] D. K. Tosh, S. Shetty, X. Liang, C. Kamhoua, K. Kwiat, L. Njilla, Security implications of blockchain cloud with analysis of block withholding attack, in: International Symposium on Cluster, Cloud and Grid Computing, IEEE/ACM, 2017.

[6] S. King, S. Nadal, Ppcoin: Peer-to-peer crypto-currency with proof-of-stake, self-published paper, 2012.

[7] I. Bentov, C. Lee, A. Mizrahi, M. Rosenfeld, Proof of activity: Extending bitcoin's proof of work via proof of stake [extended abstract] y, ACM SIGMETRICS Performance Evaluation Review 42 (3) (2014) 34–37.

[8] M. Castro, B. Liskov, Practical byzantine fault tolerance and proactive recovery, ACM Transactions on Computer Systems (TOCS) 20 (4) (2002) 398–461.

[9] S. Dziembowski, S. Faust, V. Kolmogorov, K. Pietrzak, Proofs of space, in: Annual Cryptology Conference, Springer, 2015, pp. 585–605.

[10] S. Shetty, V. Red, D. Satterfield, C. Kamhoua, K. Kwiat, L. Njilla, Data provenance assurance in cloud using blockchain, in: SPIE Defense + Security Conference, 2017.

[11] J. P. D. Nguyen, R. Sandhu, Dependency path patterns as the foundation of access control in provenance-aware systems, 2012.

[12] X. Liang, S. Shetty, D. Tosh, C. Kamhoua, K. Kwiat, L. Njilla, Provchain: A blockchain-based data provenance architecture in cloud environment with enhanced privacy and availability, in: International Symposium on Cluster, Cloud and Grid Computing, IEEE/ACM, 2017.

Is Blockchain a Suitable Technology for Ensuring the Integrity of Data Shared by Lighting and Other Building Systems?

Alex Vlachokostas, Michael Poplawski, Sri Nikhil Gupta Gourisetti
Pacific Northwest National Laboratory
Richland, USA
{alex.vlachokostas, michael.poplawski, srinikhil.gourisetti}@pnnl.gov

***Abstract*—Increasing amounts of data are available from lighting and other building systems. In commercial buildings, these data can be used to improve system and energy performance, detect and diagnose faults, and facilitate maintenance. Building data are not only of interest to owners and operators of the building systems, however. Owners and operators of similar buildings, manufacturers of building systems, utilities, and city agencies also have interesting use cases. Energy data can, for example, be used to verify the performance of energy-conservation measures, issue renewable-energy certificates, and financially settle grid services. Increased data sharing, however, significantly expands the cyber-attack surface, creating new challenges. In this work, blockchain is explored as an option for ensuring the integrity of data that are shared by lighting and other building systems. Blockchain fundamentals and variants are briefly reviewed, and value propositions relevant to building systems are discussed. A recently developed blockchain applicability framework (BAF) that builds upon and addresses the limitations of previous applicability models is also briefly reviewed. The BAF is used to assess the suitability of blockchain over other technologies or approaches for building-data applications, using emerging connected lighting systems as an example use case.**

Keywords – data integrity, data exchange, connected lighting systems, blockchain, distributed ledger technology

I. INTRODUCTION

Although the U.S. Environmental Protection Agency reports that people in the U.S. spend 90% of their time indoors [1], the operation of building systems, such as those that provide lighting or heating, ventilation, and air conditioning (HVAC) service, has changed little over recent decades. While new technology has delivered significant increases in the energy efficiency of some devices (e.g., LED technology for lighting), other aspects of building systems have not yet seen similar disruption. Building systems are largely specified, installed, controlled, operated, and maintained in the same ways, using the same technologies and practices, as they have been for decades. However, the Internet of Things (IoT) is bringing modern network-communication interfaces, sensors, and intelligence to all sorts of devices, including those that comprise lighting and other building systems. While the computing and mobile-device industries drive performance up and cost down for network technologies and sensors, Moore's law continues to drive processor size and cost down, facilitating the ability to imbue devices with increasing amounts of distributed processing power. As a result, building systems are becoming more capable of producing and sharing data, such as information about their energy consumption, operating and fault conditions (e.g., input voltage and current, out-of-tolerance alerts), and the environments that they serve (e.g., occupancy, temperature, air quality). Such data can be used to optimize performance; identify relationships between devices, the buildings they operate in, the spaces and occupants they serve, and the environment; and much more. In some cases, it may be possible to predict outcomes before an event (e.g., a device fails or a space becomes occupied) takes place.

Lighting systems, in particular, are evolving from merely being able to illuminate dark spaces to also being able to serve as data-collection platforms that can sense, analyze, and act based on environmental inputs. Connected lighting systems (CLS) are an emerging class of lighting infrastructure with integral network-communication capabilities, sensors, and intelligence [2]. They can sense daylight availability, occupancy/vacancy, and perhaps even occupant preferences and real-time pricing for electricity – and can use this information to optimize operating costs as well as energy and lighting performance. Data shared by CLS with other building systems might serve many purposes. For example, occupancy data might be used to implement thermostat setbacks in unoccupied zones. In practice, the energy consumption of thermostatic-controlled loads can be reduced by as much as 1% for each degree of temperature setback, when the setback period is eight hours long [3]. The ability of these emerging smart building systems to produce high-fidelity data facilitates the application of rapidly advancing data-consuming technologies such as machine-learning, for a variety of use cases, such as anomaly prediction and forecasting. However, such techniques require tamperproof data to ensure accurate outcomes.

Some use cases might require the sharing of data between operational technology (OT) devices (e.g., lights and thermostats) and information technology (IT) devices (e.g., personal computers and mobile phones). For example, when a security system or occupancy sensor detects a new building entry, a beacon might be activated to discern whether the mobile device of a known occupant has entered the building. If the device owner enables access to their calendar, the building might turn the lights on in their private office and set them to preferred settings or alert the destination conference room to their impending arrival. These interwoven cyber-physical systems can be the Achilles heel for a cyber threat/attack aimed at a building. OT systems have previously been used as potential Trojan horses to IT systems that contain sensitive business data [4]. Increased connectivity between systems and devices in buildings creates a major cybersecurity challenge that must be addressed if the perceived value of data

This study was conducted at the Pacific Northwest National Laboratory, which is operated for the U. S. Department of Energy by Battelle Memorial Institute under Contract DE-AC05-75RL01830.

sharing is to be realized. Devices are often designed and deployed with functionality, price, and ease of use as higher priorities than cybersecurity, as highlighted by a recent study by HP that noted that 70% of the most common IoT devices contained vulnerabilities, with an average of 25 vulnerabilities per device [5]. This expanded device-and-attack landscape cannot be managed with traditional cybersecurity procedures and mitigations such as secure configuration, whitelisting, patch deployment, and inventory management.

Securing buildings from cyber threats will become increasingly necessary as the confidential information that they contain increases. While many cybersecurity policies, procedures, standards, and risk-management frameworks exist for IT and industrial control systems, an insufficient effort has been made adapting and extending these resources for the building OT environment. The U. S. Government Accountability Office recognized additional gaps in a recent report that noted the government was not "addressing the cyber risk to building and access control systems particularly at the nearly 9,000 federal facilities protected by the Federal Protective Service (FPS) as of October 2014." The report also noted that the government "lacks a strategy that: (1) defines the problem, (2) identifies the roles and responsibilities, (3) analyzes the resources needed, and (4) identifies a methodology for assessing this cyber risk" [6].

Malicious actors in cyberspace can recognize these gaps and actively exploit vulnerabilities, resulting in financial, reputational, and physical damage to private and public organizations [7]. To enhance the cybersecurity of commercial buildings, a comprehensive approach is needed that also takes into account their great diversity in terms of systems, size, complexity, function, occupants, financial resources, and risk sensitivity. A balance needs to be established between building security and functionality, reliability and resilience, opportunity, and cost. That is, the approach to security must cost-effectively support critical building functionality and new high-value data-enabled use cases, while ensuring that the OT and IT systems that the building relies on, and the sensitive data they contain, are secured from threats.

Connecting building OT and IT systems in a single building expands the cyber-attack surface. The attack surface grows further, in some cases exponentially, if and when data are shared between buildings and other entities. However, such sharing can, again, serve many purposes. For example, energy data may be used to verify compliance with city or state energy codes, optimize energy performance, verify the energy savings that result from conservation measures implemented by energy-service companies, issue renewable-energy certificates, verify demand response, and financially settle grid services. Attack surfaces can grow exponentially when the level of data-sharing is unconstrained. Such situations present a number of unique challenges.

For example, online systems commonly require entities that desire access to their services or data to provide specific types of information (e.g., usernames, passwords, last digit of social security number) in order to authenticate themselves (i.e., verify their identity) [8]. This information leakage might comprise or be capable of being linked to personal identifiable information (PII), which might compromise the value of some use cases. For example, a property manager who is required to disclose the monthly energy-consumption data for his buildings in order to receive energy-efficiency rebates might not do so if the risk of disclosing other information about any individual buildings is too high. Notably, the information leakage required by authentication processes is typically not a one-time risk; such information must be provided each and every time system access is required, unless continuous access is feasible or even possible.

Most data-sharing use cases only require the means for many data consumers to access the data of a single data provider. For example, a bank might provide access to financial data for many users, while only requiring or allowing those users to authenticate themselves and supply limited data in return (e.g., address changes). However, when many entities need to share data with many entities, so-called peer-to-peer (P2P) environments are created, which pose unique infrastructure needs. The technology companies behind P2P economies – such as Airbnb for hospitality services, Uber for ride-sharing, and/or TaskRabbit for freelance labor – promote the idea that peers create and share platform value (i.e., benefits and costs) [9]. While the peers may share their portion of the benefits and costs relatively equally, it has been noted that such companies are effectively monopolies that dictate how benefits and costs are accrued and paid [9]. They do not create data but, rather, aggregate data from peers and thereby create an "aggregating economy," from which they extract a relatively large portion of the benefits and pay a relatively small portion of the costs. These companies rely on traditional central/trusted node technologies (e.g., client-server systems) to manage the data that drive their platforms and economies, which are costly to maintain and to secure. Further, if a node is breached, perhaps as a result of insufficient attention being paid to cybersecurity vulnerabilities, then the whole business can be brought to a halt.

Traditional methods for managing cybersecurity threats typically utilize federated architectures to isolate disparate systems and networks. Such approaches can have heavy overhead costs as the number of systems and networks grow, and the inherent single points of failure can be catastrophic. In such cases, approaches that utilize distributed trust mechanisms might offer significant advantages. Distributed ledger technologies (DLT) and blockchain are one example of technologies with these attributes. The remainder of this paper will provide an overview of DLT and blockchain and evaluate their feasibility for a specific building system use case.

II. Distributed Ledgers and Blockchains

Most data systems store information in centralized databases that can be queried and analyzed as necessary. The challenge with such architectures is the inability to distribute administrative privileges to a distributed network of disparate entities. In contrast, DLTs store information in a distributed architecture consisting of "nodes." Such distribution eliminates the single point of failure and also distributes the overall networking and operations costs. The result is a "database" that exists simultaneously in multiple nodes. The node sites can vary from big entities, such as municipalities or utilities, to smaller entities, such as real-estate companies, maintenance providers, or individual buildings. The information that is shared among all these entities is not controlled by a single entity, and therefore there cannot be a single point of failure or loss of data. Some of the most important features of the distributed ledgers are transparency, provenance, integrity, immutability, and non-repudiation for data applications.

When cryptographic signing is incorporated into the interconnected nodes of a distributed ledger, and records in the ledger are linked via a hashing function, that ledger is often referred to as a "blockchain." In a blockchain, consensus-forming nodes verify and approve transactions for a given period and then group them into a timestamped block. The newly created block contains the transactions performed, as well as the cryptographic hash and nonce of the previous block. A hash function is a mathematical algorithm that can transform any type of input, such as a text file, into a fixed-size output string called a hash. A nonce is an arbitrary number that is added to the input prior to hashing. A given nonce is only used once and is often a random or pseudo-randomly generated number. This ability to link, through cryptographic hashing, the present with every instance of the past is a fundamental characteristic of all blockchains.

An ideal hash function does not work in reverse, meaning that the raw data cannot be reconstructed from the hash. However, several commonly used hash functions have been compromised in recent years, meaning that someone was able to establish some means to reconstruct the data from the hash. When that happens, the hash function is deemed to be risky to use from a cybersecurity point of view. Most well-known blockchains currently use the SHA-128, SHA-256, or similar hashing algorithms, which to date are uncompromised and thus considered cybersecure. For example, Figure 1 shows the result of an example lighting transaction in a building with a unique identifier [10] run through an SHA-256 hash algorithm.

Transaction: "Luminaire #002 daily energy data at floor one at building with Unique Building ID 87C4VVMJ+3VR-10-7-11-7"

Hashes to:

4BE177EDDE0646B55E944228306917CB2E6390123BF8C52FC9F921DE37BD7497

Fig. 1. Hash-function example for a lighting-system transaction.

While the use of cryptographic hashing and block-forming process ensures the immutability of transactions that have been added to the blockchain, the integrity (e.g., verification and validation) of each block must be established prior to its insertion into the blockchain. This is accomplished by the process used to form the blocks. When a transaction occurs, such as the creation of a set of energy data for all lighting devices in a building, the integrity of the transaction is checked, or confirmed, by blockchain nodes, using an established consensus mechanism. If the block is approved, then it is added to the blockchain. A blockchain is transparent in that the public-address transactions are available for viewing. For example, someone can use a blockchain explorer and a user's public address to view the transactions that have been performed by that user. This data transparency helps to maintain the integrity of data from the moment that the data is generated [8].

A. Blockchain Variants

While all blockchains use cryptographic hashing to ensure immutability, the same is not true for how they ensure integrity. Different blockchain technologies vary in how nodes are defined and enabled to perform certain actions (e.g., read, write) and form consensus. Blockchains are often characterized at a high level as being public or private. In public blockchains, anyone can join the network, and everyone has the same role and responsibilities. In private blockchains, multiple roles are defined, and participants can only join the network upon receiving an invitation from administrator nodes. Some literature synonymizes "public" with "permissionless" and, similarly, "private" with "permissioned." Such distinction is based in part on the following factors: 1) who (i.e., what nodes) can validate transactions, 2) who can write (or perform transactions), and 3) who can read transactions. However, there have been recent experiments using hybrid public-private ledgers that require an invitation in order to become a node that can validate and write blocks, whereas reading transactions are public.

Blockchains use various kinds of trust or consensus mechanisms. The basic concept behind proof-of-work (PoW) mechanisms is that one party (the prover) presents the result of a computation that is known to be hard to compute but easy to verify. Anyone can then perform a simple computation to be sure that the prover performed a certain amount of computational work to generate the result [11]. A PoW mechanism is typically used for applications where there is no established trust, as is the case for cryptocurrencies (e.g., bitcoin and Ethereum). However, PoW mechanisms are slow and energy-intensive [12], and favor nodes with more mining resources, which can eventually lead to power consolidation with a certain number of resourceful nodes. For this very reason, some researchers argue that, over time, PoW mechanisms unintentionally convert a decentralized blockchain to one that is centered around a few nodes.

PoW mechanisms are often seen as too expensive and slow for some applications, which has led to the development of other consensus mechanisms. In proof-of-stake (PoS) mechanisms, the members of the blockchain network that have the most to lose in the event of a blockchain failure (e.g., nodes representing the building owner) and thus have the largest stake also have the greatest voice in the consensus process. Although PoS mechanisms are less energy-intensive than PoW mechanisms, they are also prone to centralization and power consolidation.

Proof-of-authority (PoA) mechanisms use a set number of predetermined nodes to form the consensus. Therefore, PoA mechanisms are generally permissioned/private blockchains. In a permissioned/private PoA blockchain, the transacting nodes need to be permitted into the network. However, authority nodes (permission-granting and consensus nodes) have no control over the transactions. Therefore, the transacting nodes continue to have the core blockchain features and freedom that are seen in PoW and PoS blockchains. During the consensus-forming process, the consensus nodes only evaluate whether the rules of the network are being followed. Therefore, as long as the transaction abides by the network's rules, it should be approved. Several PoA blockchains have been exploited in recent years, however. As a result, researchers are exploring the incorporation of voting schemes into PoA mechanisms. The most-common means of achieving consensus via voting is a class algorithm called Byzantine Fault Tolerance (BFT), which uses multiple rounds of explicit votes to reach consensus. Many variations of BFT have been developed, such as the Federated Byzantine Agreement (FBA) and Practical Byzantine Fault Tolerance (pBFT), each with its own advantages and disadvantages, which are beyond the scope of this paper. More detail on BFT and other consensus-level algorithms can be found elsewhere [11].

B. Smart Contracts

DLTs present a means of creating trustable entities that can engage with each other via smart contracts. A smart contract is software that executes agreements between participants in a P2P network, such as a blockchain network, without the need for trusted intermediaries. Smart contracts that are hosted on a blockchain can automate and validate workflows that require human resources and time, and thereby reduce the need for bureaucracy and improve the efficiency of many legacy business processes.

Blockchain smart contracts cannot access data outside of their network. If such data are needed to enable a particular use case, then an oracle service triggered by a data request from the blockchain can be used to deliver the data to the smart contract [12]. However, the use of oracles poses a trust issue between a blockchain network and its oracles, as it is at least theoretically possible to perform "man in the middle" attacks standing between the contracts and the oracles. Securing an oracle is currently a challenge for smart contracts [12]. Researchers have been investigating ways to develop middleware agents that can not only facilitate secure data exchange between blockchains and oracles, but also enable data exchange between blockchains. However, most approaches that have been explored to date have not been proven at a large scale.

III. DLTs and Blockchains For Building Data

Buildings that generate and share data with various third parties form a network of distant and separate entities that populate a dynamic landscape. At present, building data typically exist in silos, and, as a result, building owners and operators, government agencies, utilities, manufacturers, and other third parties lack transparency and data provenance, and are typically not able to detect datasets that have been tampered with as a result of human mistakes or actors with malicious intent.

Distributed ledgers can allow this disparate group of network actors to exchange data more seamlessly than can current practices that rely on accessing data through client-server architectures and that often result in multiple copies of data residing in shared drives or portable hard disks. Further, blockchains offer a variety of value propositions to buildings that desire to exchange data in a secure and efficient way. However, DLTs and blockchains represent only one of many approaches that might be taken to address the needs of the building-data sharing use cases presented here. In the following sections, a specific use case that requires the sharing of building data with entities outside of the building is presented in some detail, and a unique framework is used to compare blockchain with alternative approaches and discern which blockchain variant (i.e., public vs. private, PoA vs. PoB vs. PoS vs. PoW consensus mechanism) is best suited to meet the needs of the use case.

A. Demand Response Settlement and Reconciliation

As defined by the U.S. Department of Energy, demand response (DR) "provides an opportunity for consumers to play a significant role in the operation of the electric grid by reducing or shifting their electricity usage during peak periods in response to time-based rates or other forms of financial incentives" [13]. The challenging part of demand response is the measurement and verification (M&V) of demand reduction quantities at the building level, and more specifically at the building system level (e.g., lighting, HVAC). This M&V enables the financial settlement for the realized demand reduction between the market participant (e.g., building, CLS) and the DR market. Typically, if the building or system achieves the DR goal, then it receives payment; whereas if it fails, it pays a penalty [14].

M&V might be done at the building level, using the utility meter, if advanced (*smart*) metering infrastructure (AMI) has been deployed. However, such infrastructure is not ubiquitous. According to the Energy Information Administration, 78.9 million advanced meters were operational in the United States in 2017, out of a total of 152.1 million meters, indicating a 51.9% penetration rate [15]. For buildings operating where AMI has not been deployed, or for which AMI is insufficient for M&V purposes (e.g., the reporting interval is too long or the reporting latency too high), energy sub-meters are typically installed by the building owner and used to verify the determined demand reduction. The collection, storage, and sharing of energy data by such sub-meters typically require manual effort by an energy auditor or manager and are subject to human error, data tampering, and data loss. Emerging CLS have integral energy-metering capabilities and therefore do not require the presence of AMI or the installation of sub-metering M&V infrastructure.

In cases where a utility owns and operates AMI and is able to use its metering data to perform M&V, the risk of compromised data integrity might be considered low or acceptable. However, risks are greater if a vendor or other third party manages the AMI, or energy data are provided by sub-metering or end-use systems (e.g., CLS) owned and operated by the customer or a third party (e.g., equipment manufacturer, cloud service provider). Typically, such data lie within onsite or cloud data silos, and sharing data with other systems or parties requires, in the best case, electronic data transfer facilitated by application programming interface (API) calls generated by custom developed code; and in the worst case, it requires physical data transfer facilitated by hardware storage devices (e.g., USB or hard storage drives).

For this use case, blockchain technology provides a common platform where data from building systems are stored and made accessible to designated systems and parties. These entities are distinct nodes in the blockchain, but they are able to communicate with each other through the use of smart contracts. As a result, demand reduction is recorded directly from building-energy systems and relayed to third-party participants. A conceptual diagram of actors involved in a blockchain-based demand-response data verification between a utility and a connected lighting system is shown in Figure 2.

Blockchain appears to show some promise for addressing notable use-case challenges. As the various systems in the building transact on the blockchain network, an immutable cryptographic connection is developed between the data records over time. Such cryptographic linkage makes it almost impossible to corrupt the data. In addition, a set of administrator nodes can define multi-tier access controls that restrict read, write, and execute actions on the data. Since several nodes are associated with either verifying a block of transaction (i.e., data exchange) or forming a consensus (through proof-based or voting-based consensus mechanisms) on the block of transactions, blockchain-based data collection has the potential to eliminate single points of failure.

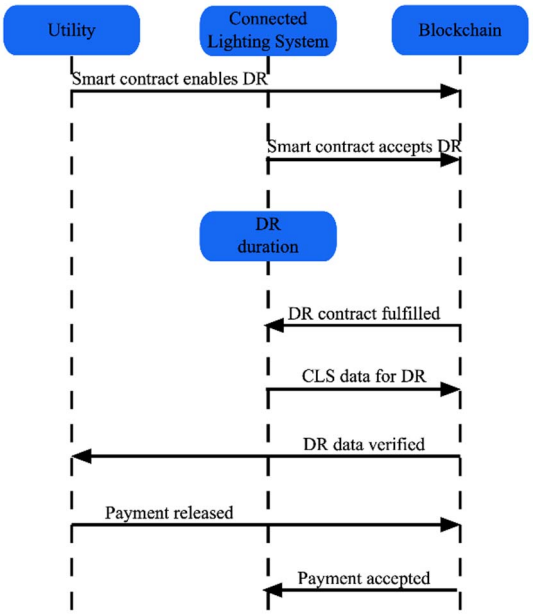


Fig. 2. Data flow depicting the use of blockchain to facilitate demand response settlement and reconciliation between a building with a CLS that can report its own energy consumption, and an electric utility.

Emerging blockchains support both off-chain and on-chain applications. Such networks can be used to securely store the raw data in an off-chain system (such as a relational database), whereas the hashes of the data can be stored on the chain. In such cases, a smart contract can be written to trigger periodic verification or integrity checks of the data. Such process can detect compromised or corrupted data records, and the elimination of such faulty data can improve the data-driven building-performance analysis.

B. *Blockchain Applicability*

Numerous models have been developed to evaluate how applicable blockchain technology is for a given application. In an article that appeared in IEEE Transactions of Engineering Management [16], the authors reviewed many of these models and proposed the blockchain applicability framework (BAF) to build upon and address the limitations of existing models. The BAF ingests use-case based responses to 90+ questions and uses an internal mapping to correlate the user responses to blockchain suitability. Evaluation questions are focused on five distinct domains: data participation, technical attributes, security, trust parameters, and performance and efficiency. Here, we used the BAF questionnaire to evaluate how applicable blockchain might be for ensuring the integrity of energy data shared by lighting and other building systems to support demand response settlement and reconciliation. The BAF results for this targeted use case are shown in Figure 3. The BAF recommends (79%) the use of blockchain for this use case. Further, it suggests a permissioned/private implementation (73%) over a permissionless/public one (27%), and the use of a PoA consensus algorithm (73%) as opposed to one of the other three supported candidates (PoB, PoS, PoW). Note that while the current version of BAF does not support voting-based PoA consensus mechanisms, we suspect that such blockchains might be particularly well-suited for building data and related use cases.

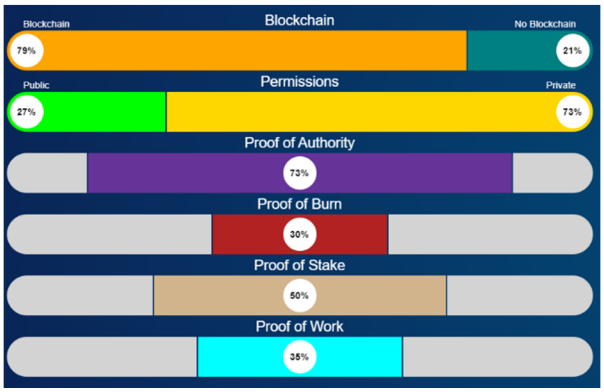


Fig. 3. Blockchain-applicability framework results for lighting-system data.

While a detailed review of the BAF analysis is beyond the limits of this paper, some key observations can shed some light on the results. Blockchains are suitable for this use case in part due to its peer-to-peer requirements. In a typical central authority infrastructure, the cost of networking and operations are associated with the central authority. Blockchains can facilitate distribution of authority and ownership and in turn, distribute and potentially reduce network operation costs.

Private blockchains are better suited than public ones for this use case because they do not expose authentication data. Such blockchains use trusted devices (i.e., network nodes) that share only required information and nothing else. Properly configured and deployed private blockchains have been demonstrated to substantially reduce the risk and cost of verifying the identity of an entity that desires to access data [11]. PoB, PoS, and PoW consensus mechanisms are primarily associated with public blockchains. The preference for PoA for this use case is largely a function of the decision that a private blockchain best meets use case needs. In several private blockchain applications, voting based consensus mechanisms (e.g., PBFT) can substitute for PoA.

C. *Future opportunities*

Building-performance analysis and optimization are recurring processes, and their effectiveness can be dependent on the integrity of the models used to estimate their performance during design and operational analysis, the data used in the analysis, and the model-predictive control framework used to adjust system operating parameters. As models and model-predictive control frameworks are developed and enhanced by in-house experts or third parties, they can be inserted into the blockchain's immutable ledger. Human and autonomous performance analyzers can verify the authenticity of the modeler's enhancements and validate and verify the integrity of the model prior to running analysis.

In the future, it is envisioned that building systems might be capable of optimizing their performance across a broader set of considerations than energy consumption/cost and occupant needs – which are generally the considerations today. Some futurists envision that a building device or system might "roam the Internet with its own wallet and its own capacity to learn and adapt, in pursuit of its goal determined by a creator, purchasing the resources it requires to survive (e.g., energy, computer power), all while selling services to other entities" [12]. Buildings might transact with each other in such a way over a blockchain, executing smart contracts between themselves and different parties. Individual

building systems might even have their own "wallet" to settle their specific service needs (such as lighting maintenance) and financial transactions (such as energy bills). At the end of the year, a building comprised of such systems might pay its city taxes, send energy benchmarking data to local government agencies, and summarize lighting use and air quality through smart contracts that are automatically executed and audited by interested parties such as the Departments of Buildings, Energy, Environment, and Health.

IV. Summary

In this paper, we described the emergence of building systems that can generate and share data with other building systems, and how this richer set of building data might serve use cases outside of the building. These data often reside on a server in the building or in a corporate cloud. While data extraction might be automated by leveraging an applicable API, it often requires a manual process that involves the engineer querying the data in a local database, exporting the data to a suitable file type, and then sharing that file with utilities or energy auditors. This process is cumbersome and subject to human error, data tampering, and data loss.

We then reviewed blockchain technology and its variants, and identified general blockchain features that appear to address existing challenges that arise when building data are shared, most notably the ability to ensure the integrity of the data that are exchanged (or transacted) between various participants in the network. While typical data-sharing techniques are subject to tampering, blockchain creates an immutable and timestamped cryptographic connection between data records over time. Further, distributed networks, where the participants can perform secure data exchanges by trusting the network and reducing the overhead that comes with establishing trust between each participant, appear to have significant potential value for building systems when data needs to be shared with unconstrained and potentially large numbers of users or systems.

Based on this initial assessment of its potential, a question-based evaluation framework was used to determining whether blockchain was well suited to a specific building data use case. This Blockchain Applicability Framework analysis suggested that a private blockchain that utilizes a PoA consensus mechanism was well suited to meet use case needs. This initial positive assessment of blockchain potential was then leveraged to imagine some future scenarios whereby a building blockchain might be used to manage the deployment of new building models or model-predictive control frameworks, or whereby a building or its constituent systems might be enabled with a blockchain wallet and utilize smart contracts to execute and settle transactions on behalf of itself, its owner, and its tenants. Based on the results of this analysis, the authors recommend future research into the deployment of a blockchain network for managing data sharing between building systems, including emerging CLS, and other systems and agencies outside of the building.

References

[1] U.S. Environmental Protection Agency (EPA). 2018. The Inside Story: A Guide to Indoor Air Quality. https://www.epa.gov/indoor-air-quality-iaq/inside-story-guide-indoor-air-quality.

[2] U.S. Department of Energy. 2019. Connected Lighting Systems. https://www.energy.gov/eere/ssl/connected-lighting-systems.

[3] U.S. Department of Energy. 2012. Program your thermostat for fall and winter savings. https://www.energy.gov/energysaver/articles/program-your-thermostat-fall-and-winter-savings.

[4] Mylrea, M., Gourisetti, S.N., Nicholls, A. 2017. An introduction to buildings cybersecurity framework (BCF), in Proc. IEEE Symp. Comput. Intell. Appl. Smart Grid, Honolulu, HI, USA, pp. 1–7.

[5] HP News. 2014. HP Study Reveals 70 Percent of Internet of Things Devices Vulnerable to Attack. https://www8.hp.com/us/en/hp-news/press-release.html?id=1744676#.V41Wm01f3X6.

[6] U.S. Government Accountability Office. 2014. Federal Facility Cybersecurity. https://www.gao.gov/assets/670/667512.pdf.

[7] Hardin, D.B., Corbin, C.D., Stephan, E.G., Widergren, S.E., Wang, W. 2015. Buildings Interoperability Landscape (PNNL-25124). Pacific Northwest National Laboratory, Richland, WA. https://www.pnnl.gov/main/publications/external/technical_reports/PNNL-25124.pdf.

[8] Catalini, C., Gans J.S. 2016. Some Simple Economics of the Blockchain. https://www.nber.org/papers/w22952.

[9] Tapscott, D., Tapscott, A. 2016. Blockchain revolution How the Technology behind bitcoin and other cryptocurrencies is changing the world. Penguin.

[10] Wang, N., Vlachokostas, A., Borkum, M., Bergmann, H., Zaleski, S. 2019. Unique Building Identifier: A natural key for building data matching and its energy applications. Energy Build., 184, pp. 230-241.

[11] Andoni, M., Robu, V., Flynn, D., Abram, S., Geach, D., Jenkins, D., McCallum, P., Peacock, A. 2019. Blockchain technology in the energy sector: A systematic review of challenges and opportunities, Renewable and Sustainable Energy Reviews, 100, pp. 143-174.

[12] Voshmgir, S. 2019. Token Economy: How Blockchains and Smart Contracts Revolutionize the Economy, 1st ed., BlockChainHub Berlin.

[13] U.S. Department of Energy (DOE). 2019. Demand Response https://www.energy.gov/oe/activities/technology-development/grid-modernization-and-smart-grid/demand-response.

[14] Goldberg, M.L, Kennedy Agnew, G. 2013. Measurement and Verification for Demand Response. https://eta-publications.lbl.gov/sites/default/files/napdr-measurement-and-verification.pdf.

[15] EIA, Form EIA-861 Advanced_Meters_2016 data file (re-released Jan. 15, 2019) https://www.eia.gov/electricity/data/eia861/.

[16] Gourisetti, S.N., Mylrea, M., Patangia, H. 2019. Evaluation and Demonstration of Blockchain Applicability Framework, IEEE Transactions of Engineering Management, pp. 1-15.