IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 17 August 2022, accepted 1 September 2022, date of publication 5 September 2022, date of current version 14 September 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3204289

RESEARCH ARTICLE

New Blockchain Based Special Keys Security Model With Path Compression Algorithm for Big Data

CIGDEM BAKIR
Software Engineering Department, Dumlupinar University, 43100 Kutahya, Turkey
e-mail: cigdem.bakir@dpu.edu.tr

ABSTRACT In recent years, following the introduction of the IoT (Internet of Things) into our lives and thanks to the rapidly increasing number of digital applications, data is collected from a wide variety of sources at an astonishing rate, and the amount of data is increasing exponentially. Today, social networks, cloud computing and data analytics make it possible to collect huge amounts of data. The concept of big data has subsequently emerged and is an important topic in many fields. However, it is not only very difficult to store big data and analyze it, but it is also a serious threat to the security of an individual's sensitive information. This study describes the issues surrounding big data security and privacy, and provides a solution involving a new blockchain-based security model. This proposed model is called the Blockchain-based Special Key Security Model (BSKM). BSKM proposes, implements and integrates three elements (confidentiality, integrity and availability) of information security together for big data. With this proposed model, a more practical and flexible structure is established for all operations (read, write, update and delete) performed on a database with real data. In this study performed with a special key, all separate blockchain transactions were used for read, write, update and delete operations, and there was a structure that could ensure both confidentiality and integrity at the same time. By looking at a special key for all the blockchain transaction operations performed on the big data it has been shown what type of authorization and access control can be established between which processes and which users. Thus, in contrast to previous studies seen in published literature, data confidentiality, data integrity and data consistency were guaranteed for all transactions. The results of the proposed BSKM model have also been compared by conducting an experimental study of its application. Moreover, this study has shown the effectiveness and benefits of the path compression algorithm. This result has been shown with experimental studies modeling big data and also shows promise for further studies.

INDEX TERMS Big data, blockchain, path compression, privacy, security.

I. INTRODUCTION

In recent years, technological advancements have brought about the rise of big data and other digital assets. Big data generates huge amounts of data, and following analysis, making use of this huge amount of information in various scientific and engineering domains. Despite many advantages and applications, there are many challenges involved with big data to be tackled for better quality of service, e.g., analytics, management, and privacy and security. Blockchain has the greatest potential to improve big data services and applications. The popularity of blockchain technology, and the huge extent of its application, results in much ongoing research in different practical and scientific areas. Blockchain, with its decentralized structure, transparency, auditability and privacy has attracted attention in many sectors with its immutability and the sense of trust it provides. This technology is used in many industries and organizations to improve performance and security. The most important use of blockchain technology is in the cryptocurrency concept and Bitcoin. In addition,

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

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

it is widely used in many banks and in the financial, health, public, construction, food and energy sectors [1]. The concept of big data is related to large-volume datasets that are from multiple sources, that are complex, and that are growing. In addition to its increased networking, data storage and data collection capacities, this concept is spreading rapidly in all fields of science and engineering by developing fast. While big data has recently become a prominent and fashionable word, studies conducted in this field are being followed as an interrelated research field. This section presents an overview of big data and discusses the related challenges and new opportunities that occur with big data. Although there are frameworks and platforms that have state-of-the-art technology for processing and managing big data, a number of problems arise in its security and confidentiality. Big data addresses a wide range of issues and points to opportunities and research issues that will arise as a result [2], [3], [4].

Big data has brought with it a plethora of opportunities for the improvement of health care, advancement of science, enhancement of education systems, promotion of economic growth and increasing ways of social interaction and entertainment; however big data has its issues. A number of problems arise with big data, such as authorization, confidentiality, integrity, availability and access control [5]. Security and privacy are great issues in big data due to its huge volume, high speed and great variety, like large scale cloud infrastructure, differences in data sources and formats, data acquisition of streaming data, inter-cloud migration and others [1]. Big data applications are a great benefit to organizations, businesses, companies and many other large and small scale industries. Today, the main focus is on security issues that are associated with big data. The diversified concentration of attacks on big data requires new technological methods in addition to privacy techniques. The main challenges include capture of data, compilation of data, storage of data, and transfer and sharing of data. This requires strong security and privacy [6].

The contribution of this scientific study using BSKM was developed to maintain confidentiality, including privacy, with data flow control. This model consists of a user, an object, and a special key. The owners of the objects are users, and they need to share their data objects with others. Users send the data objects and then require blockchain transactions (read, write, update and delete). A special key contains the policy statements of data security issued by each of the owners. Each owner sets their own security and privacy policy independently of other owners. The confidentiality of data in unsecured transport channels is ensured for all the actors in the system by means of special keys while the data are in transit. Data objects are spread and shared securely among users within unsecured environments. In addition, with path compression, the long node chain that is formed while the data objects are passing between the source node and the destination is broken, so that the objects are retrieved speedily, and the cost of access is reduced. This result was shown experimentally by modeling big data for two real datasets (banking and financial data). In addition, the results of the study were carried out on different platforms: Ethereum and Hyperledger. When the path compression algorithm is used, the maximum length and mean length of the chain decreases. Thus, with the path compression algorithm, the long node chain created by the objects is broken, the cost of accessing the objects is reduced, and fast access to the objects is enabled. In short, this study shows that fast access to data can be ensured for big data.

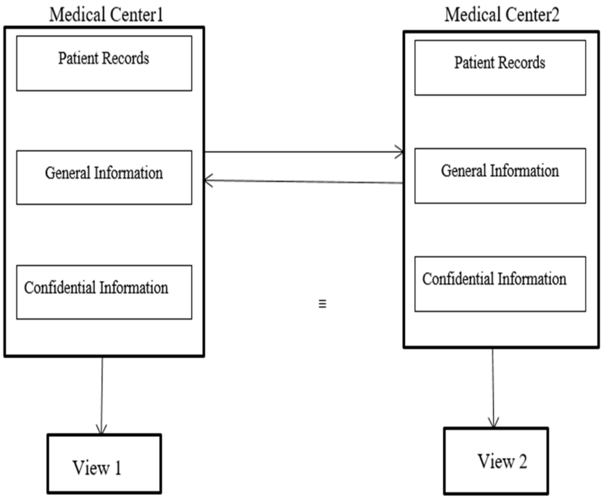


FIGURE 1. Hospital example.

The rest of this paper study is organized as follows.: Section section 2 describes the related works regarding studies in the published literatüreliterature. Moreover, we discuss the blockchain security solutions that are being used in the area of big data are discussed.; Section section 3 presents big data security and access control issues that need to be solved in order to provide secure data management platforms.; Section section 4, defines the traditional security model.; sIn Section 5, defines the blockchain architecture.; Section section 6, describes the study architecture of the proposed system and the its security infrastructure.; Section section 7, presents the experimental results of the proposed model;. Finally, sin Section 8, we presents a performance evaluation and finally, section 9 presents our the conclusions.

A. PROBLEM DEFINITION

As an example, let us consider a scenario that keeps patient records in two medical centers, as in Figure 1. The aim is to share records quickly and securely at both centers. If one center updates the data, the identical patient records should be observed when viewed from either center. In each medical center, a patient's general information is kept, this includes name, surname, TR identity number, date of birth, place of birth, blood group, gender, telephone, address, as well as diagnosis, treatment process, past health findings, medications used, laboratory reports, radiology reports, past surgeries, chronic diseases, infectious diseases, pregnancy status, etc. This includes personal health-related information that must be kept confidential. There is also general and confidential information about patients, as well as medical center registration data. The example given shows a common knowledge object and the users contributing to it. The main users can be seen as patients, doctors and medical center staff. That is, the mentioned users jointly own this information object and can perform read, write, update or delete transactions on this common information according to the access rights granted to them. When access to patient-related information is required between one medical center and another, it receives and sends the information it wants to access in accordance with local security policies. The realization of these policies is ensured by information flow control. The records of the patients should be shown to the authorized persons in both centers and the other users should be blocked. This study answers the following questions about this example:

1) According to which access rights do users determine read, write, update and delete transactions? How are these access rights granted to users?

2) How to prevent any access violation other than users' access rights?

3) How can the access granted to users be revoked after they complete their transactions? How to prevent access to the system?

In short, with the model proposed, the aim is to provide access control and access revocation in a secure manner. In other words, within the scope of this study, the BSKM model that will protect data confidentiality with information flow control for big data is explained. This model was used in a real environment and the effectiveness and success of the algorithm proposed was demonstrated by experimental study. In accessing the data, the cost of access was reduced and the data was accessed quickly.

The difference to other studies is that it targets data privacy with untrusted users and in untrusted environments using the BSKM model. By means of the private keys given to the data, each user can determine their own access authority independently from other users. The issue of reviewing access authorization is one of the laborious studies undertaken under the title of information security. The studies on this subject are time consuming and involve labor costs within institutions. Especially for complex big data, the review of access authorizations cannot always be done in a cost-effective manner; therefore, it may be neglected or not performed in a sufficiently proficient manner. In this current study, access control and authorization were provided by considering information flow control without causing data leakage in accordance with the wishes of the users. Users can create their own security, privacy and integrity policies in a practical and flexible way. At the same time, they can easily change these policies or delete them completely when they are done. This is done during runtime in this study, both statically and dynamically. Unlike other studies, it provides data confidentiality, data integrity and data consistency combined.

II. RELATED WORKS

Due to the development of technology, huge amounts of data are produced in agriculture, finance, banking, business, education, medicine and healthcare. Because diversity in size and format of data is continuously increasing, more flexible data processing tools and platforms are needed to find patterns and useful information in the data [7].

With this large amount of data, information security problems have emerged. These problems are data privacy preservation, identity and access control, data ownership authentication and authorization. Nowadays, the development in technologies gives rise to concerns about the security and protection of data during storage, transmission, processing and access. Blockchain technology is gaining more attention from the security industry, which is looking for effective ways to secure, protect, store and modify data. Blockchain is a distributed ledger that records transactions linked and secured using cryptography. Transactions can be an exchange of an asset, the execution of the terms of a smart contract, or an update to a record.

In recent years, various studies using different techniques for the purposes mentioned have been described in published literature. Lv *et al.* adopted blockchain technology to solve the privacy protection problem of unmanned aerial vehicle (UAV) big data [5]. In that study, a cryptosystem unit for encryption of the blockchain data was used. This proposed model for analysis using blockchain to protect the privacy of drone big data included a user layer, a data layer, a cloud layer and a blockchain layer. In the user layer, users used blockchain technology to track transactions to prevent shared data from being stolen or tampered with. The information that the user wants to protect is stored in the data layer. The data layer uses a decryption algorithm to recover the original data. The cloud layer is a medium for downloading, uploading, writing and reading data. The blockchain layer can provide a powerful abstraction for distributed protocols. The performance evaluation results show a big data privacy protection scheme based on blockchain technology has low computing costs in terms of key production, encryption and decryption. However, there are some difficulties in implementing this in practice. In addition, this study was only recommended for UAV data.

Li *et al.* proposed a blockchain technique to develop a novel public auditing scheme for integrity recognition in big data in cloud storage [8]. In the proposed scheme that involved three participants, the centralized and expensive third party auditor was removed, and the computation and communication involved were reduced. The aim of that study was to use a blockchain technique for security issues to defend against malicious attacks in cloud storage. But, more secure and efficient services are required on blockchain-based public auditing techniques.

Tan *et al.* proposed a blockchain-based access control framework for cyber-physical social system big data called BacCPSS [6]. It is important to preserve privacy of access control in CPSS big data by utilizing blockchain features. In BacCPSS, the account address of the node in the blockchain with access control permission is redefined and stored in the blockchain. Also access control and audit in BacCPSS are designed for authorization. Results showed that BacCPSS was feasible and effective, and could achieve secure access in CPSS while protecting privacy.

Biswas *et al.* presented a solution for e-health systems using a unified blockchain-based model [9]. In that study, a blockchain network connected individual and independent e-health systems. Access to patient data was controlled through patient-centric channels and policy transactions. A patient's digital assets could be transferred from one service provider to another using policy transactions. The proposed solution could interconnect different e-health systems efficiently. The protection of privacy is to prevent sensitive and important personal or corporate data from getting into the hands of individuals who might abuse it. However, the rapid increase in patients and transactions brings with it a scaling problem. Protecting the sensitive data of new users comes with some problems such as authorization, authentication and storage.

Ma *et al.* proposed trusted data sharing with flexible access control based on a blockchain system [2]. Also, new searchable encryption was described. It presented a trusted data sharing framework based on attribute-based encryption (ABE), searchable encryption and blockchain technology with multi-keywords. They transferred the retrieval process from the untrusted cloud to a distributed trusted blockchain. However, with the rapid growth of big data technology, new privacy problems may emerge, such as the leakage of user's sensitive data and user's privacy queries. Also the problem of data updates and access control authorization updates occurred due to insufficient schema usage.

Liu *et al.* proposed blockchain access control in the IoT environment called Fabric-IoT [3]. This work containsed a device contract (DC), a policy contract (PC) and an access contract (AC). Moreover, it combined the blockchain technology with attribute-based access control (ABAC). This improved the distributed performance of the system and more physical devices could be used to test the reliability and the throughput of the system. It could be used to try and improve the scalability of Fabric-IoT and to support more IoT application integration.

Lyu *et al.* proposed information-centric networking (ICN) to meet the increasing demand for efficient content delivery [4]. In this study, they proposed a secure blockchain-based access control framework called Secure blockchain-based access control (SBAC), to provide data security and privacy for content providers. Mounnan *et al.* created a decentralized access control infrastructure based on blockchain technology for big data to publish the policies and provide the identification and authentication processes [10]. The policies were visible to the public. Hence, every user could see the policy paired with a resource. However, this needed to use access control tools and a private blockchain.

In order to solve the security problem of personnel information management in big data, a personnel management system based on blockchain was proposed [11]. It created a prototype system separating and storing data to solve the problem of blockchain information redundancy and insufficient storage space and developed a prototype system to query, add, modify and track personnel information. However, this model was not effective in solving such issues as information leakage and tampering. Moreover, the data storage mode proposed by that study needed a better solution to the problem of large-scale data storage in main blockchain technology.

Zhang *et al.* proposed an attribute-based access control scheme that provides decentralized, flexible and fine-grained authorization for IoT devices using blockchain technology [12]. Sultana *et al.* proposed a blockchain-based smart contracts data sharing and access control system for communication between Internet of Things (IoT) devices [13]. Egala *et al.* proposed a blockchain-based access model that provided a decentralized and smart contract-based service automation without compromising the system security and privacy. This research introduced hybrid computing with a blockchain-based distributed data storage system for an IoT healthcare system [14]. Zhou *et al.* proposed a blockchain-based access control framework for secured data sharing and smart contract technologies in the Industrial Internet of Things [15]. Lopez *et al.* presented a blockchain framework for addressing the privacy and security challenges associated with big data in smart mobility. It sent encrypted data to the blockchain network and could make information transactions with other participants for smart mobility big data [16]. Yang *et al.* proposed a blockchain-based access control framework with privacy protection called AuthPrivacyChain. They defined the access control permission for data in the cloud, which was encrypted and stored in a blockchain, and they designed processes for access control, authorization and authorization revocation in AuthPrivacyChain [17].

III. BIG DATA SECURITY

The most important problem with big data collected from many areas, such as government agencies, health, education and banking sectors, private enterprises, telecommunications, the Internet, databases of large enterprises, Google, Facebook, Yahoo, YouTube and Skype, is security and confidentiality. With the acquisition of information by users and the collection and sharing of information, security problems related to data confidentiality, data integrity and unauthorized access arise. In this case, the protection of sensitive and valuable personal and public data is mandatory. Therefore, in order to solve the problems arising from the seizure of information by unauthorized persons, information leakage and modification, and non-provision of information confidentiality and privacy, some studies have been conducted and are seen in published literature [18].

Big data security consists of three basic elements: confidentiality, integrity and availability. Data confidentiality refers to preventing the transfer of valuable data of companies, institutions or individuals into the hands of unauthorized persons and ensuring that only authorized users can process the data. For example, viewing bank data belonging to customers by unauthorized users is a data confidentiality violation. Data confidentiality is ensured by using different encryption methods in communication security. Data integrity aims not only to prevent unauthorized users from modifying the data, but also to ensure that the data is obtained from up-to-date, error-free and accurate sources. Methods combining different techniques such as cryptography and access control have been used to ensure data confidentiality and integrity. Availability, on the other hand, is the fact that the data can be accessed and used at any time by authorized users [19].

These three basic elements need to be provided together for big data security. In addition to these three elements, the term privacy has emerged, which limits access to data to protect a person's identity in the collection, sharing and storage of users' data, is a more complex concept than data confidentiality. Privacy refers to the ability of users to show the desired amount of sensitive data (patient data, bank data, etc.) to the people that they want and restriction of access to users' other data. As an example, let us say that with an application, users want to get access to a museum system from different places through smartphones. In this type of museum application, in general, data such as users who visit the museum, the museum administration, the works of art exhibited in the museum, and the environmental information of the museum are collected and stored by different services in a database. Suppose that when a user purchases a ticket, some information such as the user's name, phone number, gender, zip code, ticket type and payment type are saved and stored by the museum system. In addition, the level of the surrounding temperature, humidity, particulate matter and noise is detected by various sensors and recorded in the database in order to avoid causing damage to the works of art in the museum when users are visiting. Several violations of privacy occur in the collection, sharing and storage of this museum data. To be more specific, for example, an attacker can access the phone numbers of all museum visitors or find out which people are in which sector (health care worker, soldier, sailor, pilot, civil servant, etc.) according to the type of ticket. In addition, payment information may be provided to various advertising companies, and these companies may send a number of unwanted random messages to users. These can be shown as concrete examples of privacy violations.

All stages of big data security and confidentiality are shown in Table 1. Big data security includes all the steps from data collection to data processing, storage, visualization and programming, and it is important at every step.

Big data security and privacy challenges are examined in four stages: data collection, data management, system programming and data analysis [20].

TABLE 1. Big data security stages.

Data Collection	**Data Management**	**Programming and System**	**Data Analysis**
Data Cleaning	Data Storage	Recursive Programming	Queries
Data Conversion	Distributed Databases	Graph Programming	Semantic Analysis
Data Integration	Distributed File	Memory Programming	Human-Computer Interaction
Data Normalization	Data Hiding	Calculation of Data Flow	Data Visualization

Data Collection: Big data consists of real-time structured, unstructured, or semi-structured data obtained from online services, business processes and media. At this stage, conventional security methods (encryption, authorization, access control, etc.) cannot fully ensure confidentiality and privacy due to the variety of big data (one of the 5V characteristics: velocity, volume, value, variety and veracity), due to the fact that the data consists of different types from different sources. It is not possible to monitor the data traffic that occurs when storing the data because the data increases very quickly. Differences in data format cause security vulnerabilities [21].

Data Management: This stage includes security and confidentiality issues that occur when storing data collected during the data collection process. The main purpose of data storage is to ensure that authorized individuals in an organization can access the data at any time that they want [22]. But due to the increasing volume of big data, the servers of organizations are negatively affected. Conventional data warehouses are not capable of solving this problem. Outsourced data servers such as cloud and distributed systems are used. For example, many necessary or unnecessary data such as customer feedback, e-mails, blogs, social media messages and marketing information can be stored in an e-commerce application. If the processed data is necessary for the organization, it is important to analyze and simulate it. Otherwise, too much data is shared unnecessarily over different units within the organization. This, in turn, leads to cost and time losses. In addition, because a large number of operations are performed in storing the same data due to these problems, data integrity is negatively affected. Integrity, which is one of the three basic elements of data security, is not provided by conventional encryption and security methods in multiple structures. It can give attackers the opportunity to hijack servers by taking advantage of these specified vulnerabilities in such structures. Misuse of any data will cause a data leak.

These problems may cause data breaches, haphazard sharing of data, and data theft. Access control shows which users can perform which operations on the data. There are mandatory, discretionary and role-based access control models. In the mandatory access model, users' access to data is determined based on some rules predetermined by the central authority. In the discretionary access model, users can access data within the limits determined for them, or their access rights can be revoked. Users define their authorization as "there is" or "there is not". In the role-based access control model, roles are assigned to users based on their obligations in the organization in which they work. Users use their privileges according to the assigned roles, and all access operations are determined based on these roles. Due to the differences in users and their permissions, these access control methods cannot be effectively applied in large data storage. For good backup, recovery and improvement, the data that needs to be stored and that needs to be destroyed must be followed up at every stage of big data.

System Programming: For the security and confidentiality of big data, precautions should be taken against attacks and hacking risks from inside or outside. This can also be realized with good systems programming. Against malicious attacks that can exploit the vulnerabilities of a system, secure software that can detect all kinds of threats and provide protection against these threats and risks must be created. For the security and confidentiality of big data consisting of a large amount of data from different sources, a good systems program against attacks, such as identity theft, modification of data, disclosure and Denial of Service (DoS), is required.

Data Analysis: The purpose of this stage is to ensure the secure presentation of the collected sensitive data to users after its use, processing, storage and analysis. With personal products and personal services in an organization, a lot of data is obtained to analyze the interaction between users [23]. Meaningful data is created using various data mining and machine learning techniques. However, data leaks that may compromise big data confidentiality may occur at this stage. Confidentiality issues may arise due to security vulnerabilities at any stage, from collecting data to cleaning, storing, processing and analyzing it. The data may have been damaged during data storage. In this case, data reliability may be compromised. For example, new applications developed without security tests can allow hackers to hijack sensitive information by hiding malicious program code, or by visualization of the data. Users can access all kinds of information with just one click without applying security and confidentiality policies. Therefore, timely and correct decisions should be made to counter security risks that may occur at all stages of big data.

IV. TRADITIONAL SECURITY MODEL

Access control models are an important tool developed for securing modern data systems. Institutions use access control models specifically to define who their employees are, what they can do, which resources they can reach, and which processes they can perform. Then they use them to manage the whole process. Access control is a fundamental concept in security by determining who or what can view or use resources in many places and in many businesses. Authentication and authorization of users and entities is important to minimize the security risk of unauthorized access. The main models of traditional access control are Mandatory access control (MAC), Discretionary access control (DAC), Role-based access control (RBAC) and Attribute-based access control (ABAC).

MAC is a hierarchical model in which access rights are regulated by a central authority by security levels. All users and user groups are allocated a security level. All information is allocated a security label. Users can access or be denied resources that correspond to a security level equal to or lower than theirs in the hierarchy. This system can be quite cumbersome to manage because the administrator must allocate all authorization.

DAC policy is a means of assigning access rights determined by users who have access to their objects. This model is implemented using access control lists by users that can give access authorizations to other users within the limits assigned to them or they can determine the limitations. However, this model increases the risk that data will be made accessible to users that should not necessarily be given access.

RBAC provides access rights based on the roles and privileges of the users. RBAC requires users to be assigned to different roles to get the associated permissions. However, the problems with role explosion limits its use to enterprise systems only. Here, a user may have multiple roles or capacities within a given organization. Thus, when the subject is seeking access to an object, the user must first indicate the role within which the request is being made.

ABAC is an authentication and an authorization model that controls access to objects by evaluating rules against the attributes of entities (subject and object). This model requires the basic principles of logical access control. ABAC is an extension of traditional RBAC and can define permissions based on just about any security relevant characteristics, known as attributes. This model is richer and more expressive because it can be based on any combination of subject, resource and environmental attributes.

V. BLOCKCHAIN ARCHITECTURE

Today, IoT, cloud computing and social networks make it possible to collect huge amounts of data. Big data has arisen from a growing amount of information that organizations are storing, processing and analyzing [24], [25]. Recently, big data has become a important topic for businesses and government organizations, and in various industry sectors such as healthcare, manufacturing, banking, education and transportation [26], [27]. However, it consists of a variety of big data security and privacy challenges. These challenges are data loss, data breach, data leaking and data theft, and

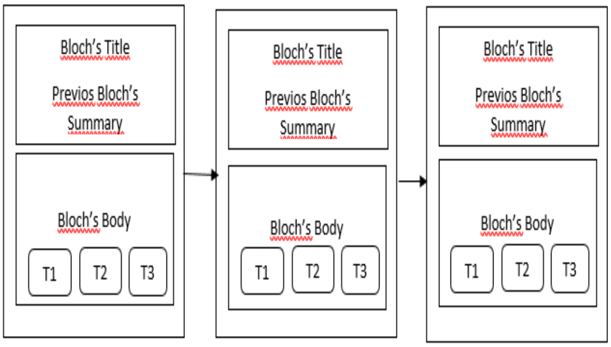


FIGURE 2. Blockchain structure.

they have become critical threats to organization assets. Traditional security access control models are inadequate for solving these problems and are unable to cope with this rapid data explosion. Moreover, these models have failed to cope with the scalability, interoperability and adaptability of big data [28]. Therefore, blockchain-based access controls have been proposed in this current study.

Blockchains are tamper-evident digital ledgers implemented in a distributed fashion and usually without a central authority (i.e., a bank, company or government). Figure 2 illustrates the blockchain structure. Blockchain models consist of the blocks that make up the blockchain and the ledgers that make up these blocks. The blockchain is a distributed ledger that a chain of blocks containing transaction information records in a secure manner. Blockchain ledgers are any content information in which the related blockchain structure is built. This information can be values, such as money transfer and customer records. Ledgers are combined and processed at regular intervals and written into blocks. Each block also contains hash information. It is a distributed database that is encrypted with special algorithms such as hash functions and provides data tracking. Transactions in a blockchain can be an exchange of an asset, the execution of the terms of a smart contract, or an update to a record.

Blockchain technology is not a suitable technology to be applied in every field, but it can offer some advantages specific to the fields in which it will be applied. In this context, it can lead to various improvements due to the advantages of its technical structure in areas where it is applicable. From this point of view, it can present a significant advantage in processes such as confirmation, signature and approval, especially within companies. In the same way, it can contribute to the monitoring of the transactions. In this way, the retrospective determination of the actions of the employees will provide assurance to the employers in terms of internal security. Likewise, the fact that the transactions of the managers can be detected in the transactions made by the employee will provide a guarantee. With a technological infrastructure established in this way, information sharing with stakeholders inside or outside a company will provide significant improvements for processes such as managing the confidentiality of information. From this point of view, it can be said that the use of this technology in the processes where it is applicable has the potential to deliver great benefits.

VI. THE STUDY PROPOSAL

It is of great importance that the technological infrastructure of the company is compatible with blockchain technology. Investment may be needed in this technology during the adaptation process, and costs such as time and training may need to be considered. However, from a security point of view, data in the blockchain structure can be stored on nodes located in different locations in a distributed manner. If the data is stored openly in an open network structure, there is a risk that the data in these nodes may be intercepted.

A transaction is the name given to the process of moving the value of a cryptocurrency from one asset to another in the blockchain network. In this current study, the aim was to ensure data privacy in big data by preventing the capture of valuable data in nodes. The scientific contribution of this study is that data privacy is ensured by defining a private key for each transaction that occurs in the blockchain. This special key has then been applied to real and different large data sets, unlike the studies in published literature. In addition, this private key was created with a bidirectional list data structure. A special path compression algorithm was proposed for data speed while ensuring data security and privacy [29]. This path compression algorithm is implemented for all operations such as adding, deleting, changing and selecting data in the blockchain. The proposed model (BSKM) consists of users, objects and special keys.

A. USER

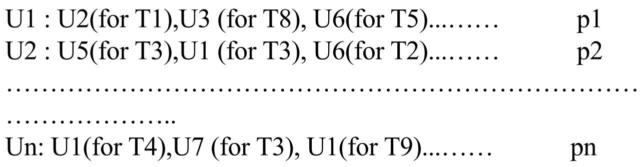
A user is the person involved in the blockchain transaction. Users include data owners and users, or groups of users, who perform operations such as granting and receiving data authorization. Each user labels their data for data confidentiality and integrity. The special key consists of a list of security policies that are provided by the users. Each user labels their data for data privacy. That is, a special key is determined that is paired with a data object. In addition, each user has the right to safely change these security policies separately.

B. OBJECT

This refers to the data that users have in the blockchain, that they share with each other, and that they perform various transactions on.

C. SPECIAL KEY

A special key is a collection of policies that are created for the protection of data. That is, a key is determined that is paired with a data object. This special key contains the encrypted version of the data sent, added, deleted or updated for each transaction that takes place in the blockchain. The special key is encrypted by creating policies. In addition, each user has the right to safely change these security policies separately. This model was developed for unreliable users and



U1 : U2(for T1),U3 (for T8), U6(for T5)......... p1
U2 : U5(for T3),U1 (for T3), U6(for T2)......... p2
....
....
Un: U1(for T4),U7 (for T3), U1(for T9)......... pn

FIGURE 3. Special key structure.

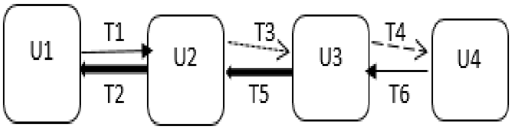


FIGURE 4. A doubly link list S modeling of the special key.

environments. Each user changes their own policy independently of each other. The object consists of data to which authorization is granted or received by users. The user represents a list of security policies issued by users. In addition, each user separately has the authority to safely change these security policies.

U1: U2(for T1),U3 (for T8), U6(for T5).........p1

U2: U5(for T3),U1 (for T3), U6(for T2).........p2

....

....

Un: U1(for T4),U7 (for T3), U1(for T9).........pn

Figure 3 shows the contents of a special key. Here, while U1, U2, ..., Un show the owners of the data object from the users in the system, the terms U2(for T1),U3 (for T8), U6(for T5)...refer to the users to whom authorization has been given for a particular transaction by the data owners: p1, p2, ..., pn i.e., each content definition on the S key shows the security policy of the relevant actor regarding these common data. Each user who owns a data object determines their own policy for the key. Then, one of the users sends these data objects to the other users with the key.

The transaction structure of the proposed blockchain is shown in Figure 4 with a doubly linked list structure. In the doubly linked list structure, the transaction is determined according to the type of arrow. Users of U1, U2, U3 and U4 perform a transaction with each arrow.

T1: Read Transaction (from U1 user to U2 user)

T2: Write Transaction (from U2 user to U1 user)

T3: Update Transaction (from U2 user to U3 user)

T4: Delete Transaction (from U3 user to U2 user)

T5: Write Transaction (from U3 user to U4 user)

T6: Read Transaction (from U4 user to U3 user)

S consists of five parts, namely owner, readers, writers, updaters and deleters. The way the arrows are drawn in the graph show the types of authority needed to access the data. Here, while 'owner' denotes the actors who own the labeled object, 'readers' refers to the users to whom authorization is given to read the owners' data transactions; 'writers' refers to the actors to whom authorization is given to write to the data owners' transactions; 'updaters' refers to the actors to whom authorization is given to update the data owners' transactions and 'deleters' refers to the users to whom authorization is given to delete data owners' transactions. The example special key shown with the doubly linked list can be expressed in the S typing format as follows [27]:

$$S = \{U_1 : U_2, U_4; U_2 : U_3, U_4; U_3 : U_4, U_5; U_4 : U_5; U_5\} \tag{1}$$

The semicolon used when creating a label separates the policies from one another. Accordingly, the S label has five policies: $\{U_1$:U_2,$U_4\}$, $\{U_2$:U_3, $U_4\}$, $\{U_3$:U_4, $U_5\}$, $\{ U_4$:$U_5\}$ and $\{U_5$: $\}$. While U_1, U_2, U_3, and U_4 denote the owners of the data object to which the Slabel belongs, U_2, U_3, U_4 and v_5 represent the actors authorized by the data owners for various object transactions (read, write, update and delete).

Let us assume that the first policy shows the read operation on the object:

The first policy is expressed by $U_1 \rightarrow U_1$, $U_1 \rightarrow U_2$ and $U_1 \rightarrow U_4$ edges. This means that the U_1 user allows the v_1, v_2 and v_4 users to read their data.

Let us assume that the second policy shows the write operation on the object:

The second policy is expressed by $U_2 \rightarrow U_2$, $U_2 \rightarrow U_3$, and $U_2 \rightarrow U_4$ edges. This means that the v_2 user allows the U_2, U_3 and U_4 actors to write to their data.

Let us assume that the third policy shows the update operation on the object:

The third policy is expressed by $U_3 \rightarrow U_3$, $U_3 \rightarrow U_4$, $U_3 \rightarrow U_5$ edges. This means that the U_3 user allows the U_3, U_4 and U_5actors to read their data.

Let us assume that the fourth policy shows the delete operation on the object:

This is expressed by $U_4 \rightarrow U_4$, $U_4 \rightarrow U_5$ edges. This means that the U_4 user allows the U_4 and U_5 users to delete their data.

The last policy is expressed by the $U_5 \rightarrow U_5$ edge. This means that U_5 does not allow anyone other than themselves to perform any transaction on their data.

D. REVOCATION OF ACCESS RIGHT

When a user performs any blockchain transaction (read, write, update, delete), they may want to revoke this access right. In this case, private keys are determined for the object again and the access right granted is revoked. With the re-issued private key, the data is made less restrictive and less stringent.

With the user hierarchy in Figure 5, client_X user allows the lawyers group to read their legal data.

Figure 6 shows the user hierarchy in a law firm. The lawyers user creates a group and all users in this group (lawyer_X, lawyer_Y, lawyer_Z) are members of this group. These lawyers exercise the authority and responsibility for this group. Considering the user hierarchy, the action to re-use 4 different private keys regarding the revocation of the access right is given below:

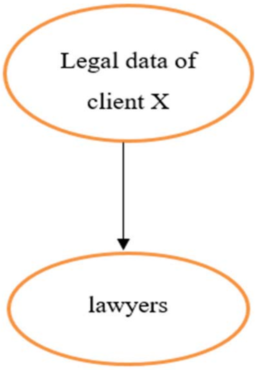


FIGURE 5. Example a user hierarchy.

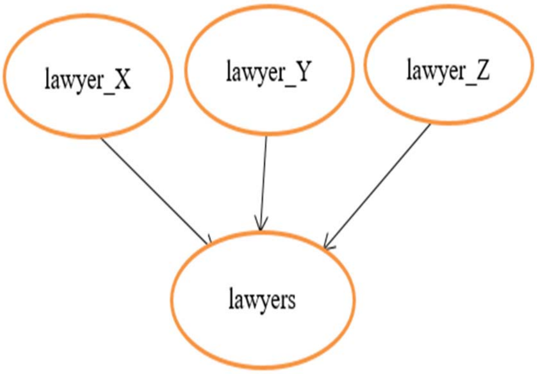


FIGURE 6. User hierarchy examples.

Action 1: Reader access right revocation: The data can be made more restrictive by removing the reader from a private key. For lawyer_X and lawyer_Y attorneys in the lawyers' group to see and evaluate the client_X legal data in the user hierarchy in Figure 5, the data {client_X: lawyer_X, lawyer_Y} and the private key are specified. However, later, lawyer_Y can remove lawyer_Y from the reader list and request only lawyer_X to defend herself. The private key {client_X: lawyer_X} provides this status. In short, users can set their policies and ensure the privacy of their own data through private keys.

Action 2: Cancellation of update access rights: It may be necessary to change the users to whom a power of attorney is given by the owner of the data. For example, the client_X→lawyer_X proxy process is shown in the user hierarchy in Figure 5 and Figure 6. The owner of the data tagged with {client_X: client_X} can be changed to lawyer_X, since the client_X actor is deputizing for the lawyer_X actor. With the new tag { lawyer_X: client_X}, the data can be updated with a different private key.

Action 3: Revocation of adding access rights: Set private keys for a client_X user's legal data as follows:

$$\{\text{client\_}X : \text{lawyer\_}X; \text{lawyer\_}X : \text{lawyer\_}Z\}$$

With this private key, the legal data of client_X is given to lawyer_Z via lawyer_X and lawyer_X. However, when the lawyer_X: lawyer_Z addition needs to be cancelled, the client_X can re-determine the private key of the legal data {client_X: lawyer_X} with the private key. By deleting a new policy from a private key, data becomes more tightly protected.

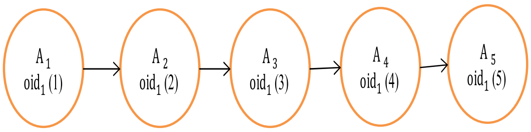


FIGURE 7. Node chain formation.

Action 4: Self-authorization. Let S be the private key {X:Y}. X is the data owner and Y is the reader of this tag. Also, let the user X give the power of attorney to users Z and T. Users given the power of lawyers are added to the data as a reader with their private key {X:Y}. Thus, the data is more restricted with the newly created private key {X:Y,Z,T}. At a law firm, the lawyers authorized by the client can represent the client by adding them as readers to the new label. With the {client_X: lawyer_X} tag, client_X authorizes lawyer_X to conduct their own case. Considering the user hierarchy in Figure 5 and Figure 6, client_X gives the power of lawyers to all lawyers in the lawyers group. By adding lawyer_Y and lawyer_Z as readers to the old private key, the data becomes more restricted with the newly created private key {client_X:lawyer_X, lawyer_Y, lawyer_Z}.

Any transaction can be canceled by the re-switching rules given above because these rules are recreated without breaking the security level. For example, in the multi-level security model, each user includes their own data in one of these classes, including unclassified, classified, secret and top secret, in order to protect their own data, and private keys are determined again. Each user in each security class acts on behalf of its own class, and these users can see all transactions in its class.

The transactions for revoking the access rights granted by a data object from user **ui** to user **uj** are shown below in pseudocode. Meanwhile, in order for the user **uj** to revoke the access right (read, write, update, delete) granted to the user **ui** – a policy in S – which is the private key for this data, it must be included in the data owner or all authorized lists. This is expressed by the following condition:

Revocation of Access Right on Data:

```
if {$1\leq i \leq n$; $\forall_i u_j \in reader_i[S]$, $writer_i[S]$, $updater_i[S]$, $delete_i[S]$} or {$1\leq i \leq n$; $\exists_{i,}$ $u_j \in owner_i[S]$}
  {
        $u_j$ has revoke access right to read,write, update and delete data
  }
  else
  {
        $u_j$ has hasn't revoke access right to read,write, update and delete data
  }
```

Algorithm 1 Path Compression Algorithm

```
Algorithm YolKısalt (Start:Düğüm)
//Start node
X ←Start;
Y ←Start;
if (X=null or next[X]==null) return;
// determine previous node to probe (Y)
while (next[X]!=nll) do
        Y←X;
        X←next[X];
end while
// update display
Z←Start;
while (Z!=X) do
        next[Z]←next[Y];
        Z←next[Z];
end while
return;
```

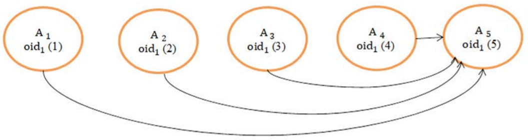


FIGURE 8. The new condition that occurs as a result of the path compression.

E. PROPOSED PATH COMPRESSION ALGORITHM

The *path* is a doubly linked list data structure that shows the transmission of the data objects by the users. The nodes are connected to each other in the form of chains. As an object moves between the users, the object identifier (oid) that shows the identity of the object is updated to include the next address of the object. For example, in Figure 7 an object (oid$_1$) is transferred to the storage nodes A$_1$, A$_2$, A$_3$, A$_4$ and A$_5$, respectively. When the object is transferred, only the address information remains on the previous node and the object is transported to the new node. When the object is moved from the A$_4$ storage node, it is transferred to node A$_5$ and the new address is saved as oid$_1$(4) to node A$_4$. In other nodes where the object is passed, the address (reference) of the object remains (oid$_1$(1)–oid$_1$(2)–oid$_1$(3)–oid$_1$(4)). The object shown with oid$_1$(5) is actually located in node A$_5$. Oid$_1$(5) in A$_5$ shows the new oid value of the object. Because object movements create long chains between nodes in this format, the cost of accessing the object increases. In order to prevent long chains, the *path compression method* is used, which gives the result shown in Figure 8. *Path compression* is the process of updating the reference in each node on the path, which starts from the root node to the node where the object is currently located, with the current location address (See Algorithm 1).

VII. EXPERIMENTAL STUDY

When the proposed Blockchain-based Special Key Model (BSKM) was compared with the traditional security models in published literature, the performance results obtained in terms of accuracy and time are given in the following sections.

A. DATA SETS

The two real datasets used in the study taken from different sectors underwent a preliminary process so that each user and object mentioned in the dataset was classified according to the security transactions. Real classification scales for these institutions were taken as the basis for the classification process. Datasets were labeled as the bank dataset and the financial dataset.

The first dataset is a real dataset obtained from a Turkish bank. Moreover, this dataset consisted of approximately 10 million customers and nearly 100 million transactions such as money transfers and electronic fund transfers. This dataset contained confidential and sensitive information. For this reason, no information is given about the content of the data.

The financial data is what investors use to analyze a company's economic and financial health. It contained a 100 million different transactions. This dataset also contained confidential and sensitive information. For this reason, it was not shared between different environments.

B. ACCURACY

Together with the proposed model, other access control models (MAC, DAC, RBAC, ABAC) have been used on a platform operating a real distributed system, and all models have been separately applied to the two datasets. Accuracy and time results attained for all models applied to each dataset were compared using the ratio of the transaction performed by the user with the authorization permission to all transactions and the working times for a transaction in the system results in use by the application belonging to the sector from which the dataset was taken. This allowed the performance values of the methods to be compared. The proposed model performance was calculated by Equation 2:

$$\begin{aligned}\text{Accuracy} = \; & \text{Number of all transactions performed} \\ & \times \text{correctly by all authorized} \\ & \times \text{users / Number of all transactions} \end{aligned} \tag{2}$$

is calculated. All classes related to being true or false of for all transactions performed are available in the datasets.

The time performance criterion was calculated by Equation3:

$$\text{Time= System uptime for a transaction} \tag{3}$$

In Table 2 and Table 3, the success of the proposed model was compared to published literature methods in terms of

TABLE 2. Traditional security model and Bskm accuracy rate (bank dataset).

Transaction/ Accuracy Rate (%)	MAC	DAC	RBAC	ABAC	BSKM
Read	70.81	71.05	73.09	80.00	95.75
Write	67.34	68.55	69.57	77.71	84.12
Update	65.30	66.77	69.34	73.38	81.09
Delete	71.97	75.42	77.27	82.65	99.82

TABLE 3. Traditional security model and Bskm accuracy rate (financial dataset).

Transaction/ Accuracy Rate (%)	MAC	DAC	RBAC	ABAC	BSKM
Read	82.04	81.72	84.65	87.83	99.67
Write	75.42	78.64	80.55	82.64	92.64
Update	68.42	61.42	73.57	79.64	91.04
Delete	77.18	72.42	80.34	82.60	95.37

TABLE 4. Results for some datasets in the literature.

Methods used in the literatüre/ Datasets	**DAC**	**MAC**	**RBAC**	**ABAC**	**BSKM**
KDD99 cup data set	75.24	78.64	80.01	82.34	98.52
DARPA BSM	69.36	69.24	75.99	80.34	95.34
NSL-KDD dataset	71.67	73.52	77.30	80.65	96.85

accuracy against real data sets, which has been taken respectively from the data of a bank and a financial institution whose classes are obvious. Accuracy rates were calculated using transactions randomly selected from these data sets. In addition, all classes of this data set were specified. Accuracy rates were calculated according to their real class. While measuring the accuracy rate, the classes of the model created for this study were calculated by comparing them with real classes. The success of the proposed model is clearly shown in Figure 9 and Figure 10 for the bank and for the financial data. When the performances of both methods were compared for all operations performed on objects in terms of accuracy rates, the success of the proposed BSKM can be clearly seen. In particular, it gave more successful results in reading and deleting operations. This is because writing and updating operations are more difficult than other operations. In particular, read and delete operations produced more successful results in all tables compared to other operations. This is because updating operations with writing are more difficult than reading and deleting operations. In addition, as the number of users and data decreased, the success rates increased in both methods because less users and more accurate results were produced with the data.

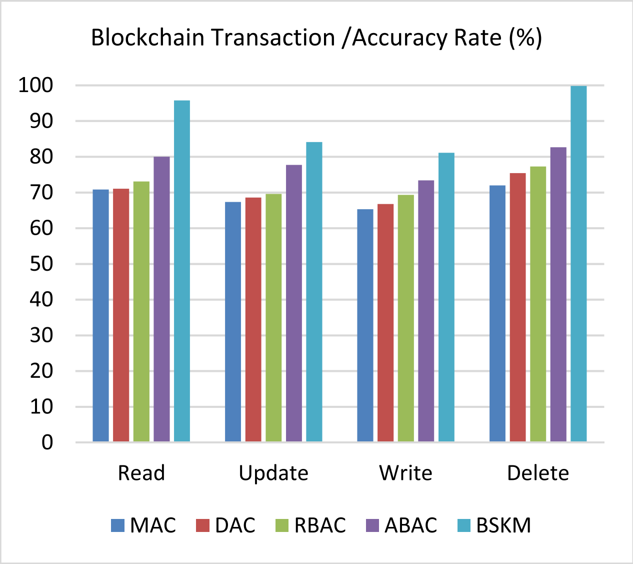


FIGURE 9. Traditional security models and BSKM accuracy rate (Bank dataset).

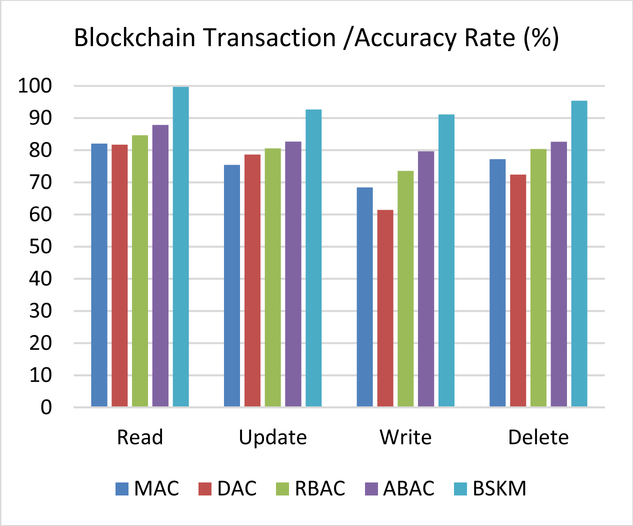


FIGURE 10. Traditional security models and BSKM accuracy rate (Financial dataset).

The proposed model I was compared with some datasets used in published literature in Table 4. The proposed model I gave successful results on the datasets used in the literature. In addition, the success of model I is clearly shown in Figure 11.

This study evaluated the performance of BSKM on the Ethereum and Hyperledger Fabric blockchain platforms. The performance evaluation results of the proposed model were obtained with different numbers of transactions. The experimental infrastructure consisted of four servers, each with 8GB of RAM, a 128G SSD hard drive and running Ubuntu16.04 with four clients to send transactions. The simulation parameter for the current study are shown in Table 5.

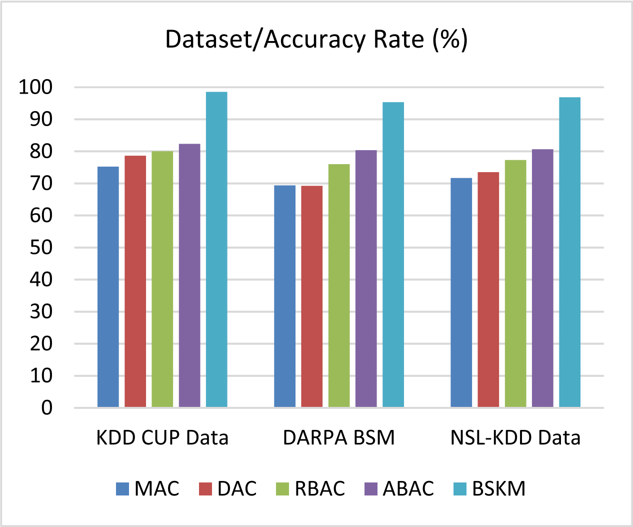


FIGURE 11. Results for some datasets in the literature.

TABLE 5. Simulation Parameter for blochchain platforms.

	Peers Nodes	Orderer	CA
Devices	Virtual Machine	HP Laptop	Virtual Machine
of nodes	4	2	4
CPU	4	8	4
RAM	4GB	8GB	4GB
Ledger	v1.4	v1.4	v1.4

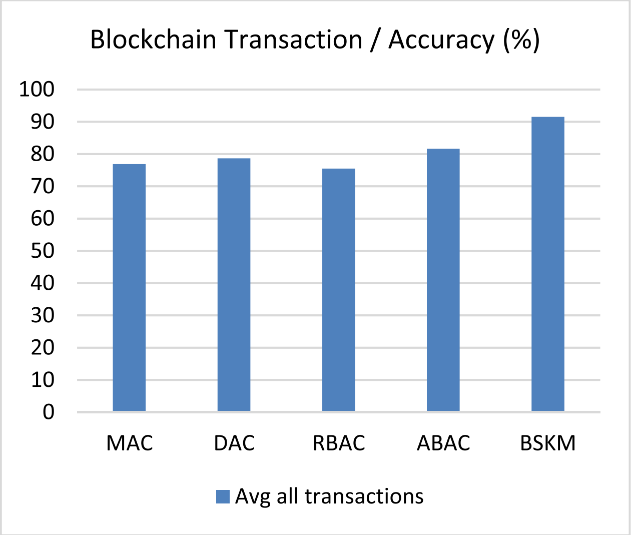


FIGURE 12. Traditional security models and BSKM accuracy rate for bank data in Ethereum platform.

Ethereum is a decentralized and open source blockchain that forms the infrastructure of many cryptocurrencies thanks to the ERC-20 code system. This study shows the results of the proposed model on the Ethereum platform in Figure 12 for bank data and in Figure 13 for financial data. This study built an Etherum blockchain platform, firstly by downloading and installing the latest version of Ethereum (geth1.7.3 [11]) on four servers in the same local area network, deploying four separate nodes, then specifying the same network id and genesis.json for all nodes to ensure that nodes could be properly connected to each other, finally completing the connection between the main node and child node by file configuration.

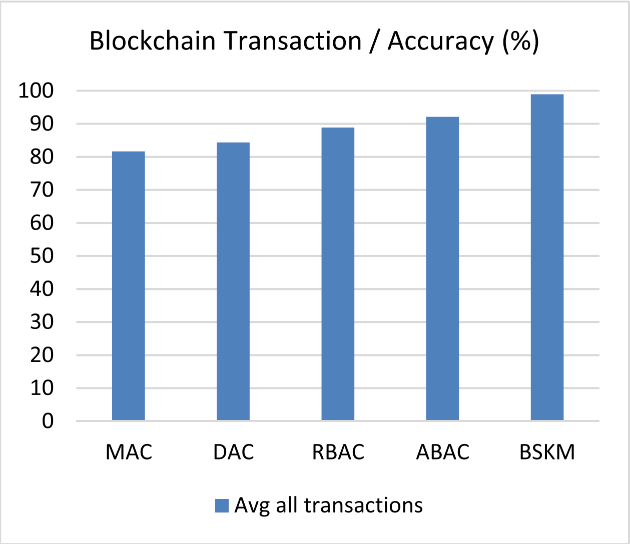


FIGURE 13. Traditional security models and BSKM accuracy rate for financial data in Ethereum platform.

Hyperledger is an open source project created to support the development of blockchain-based distributed ledgers. Hyperledger consists of a collaborative effort to create the necessary frameworks, standards, tools and libraries to build blockchains and related applications. This study built a blockchain platform of Hyperledger Fabric by firstly downloading and installing the Hyperledger Fabric v0.6 on the server, then configuring the fabric environment, and finally deploying the fabric network and testing the chaincode. The results showed that the maximum number of nodes that Fabric v0.6 can have is 40 nodes in the network.

The results of the model for bank data on the Etherum platform are shown in Figure 14. Separately, all transactions for financial data on the Hyperledger platform are shown in Figure 15. When the proposed model was compared with traditional security models, it was observed to give more successful results on both platforms.

C. TIME

In Table 6 the success of the proposed model (BSKM) is compared with published literature in terms of time against the real data set taken from the bank. In Table 7, the success of the proposed model (BSKM) is compared with published literature in terms of time against the real data set taken from the financial data. Accuracy rates were calculated by random

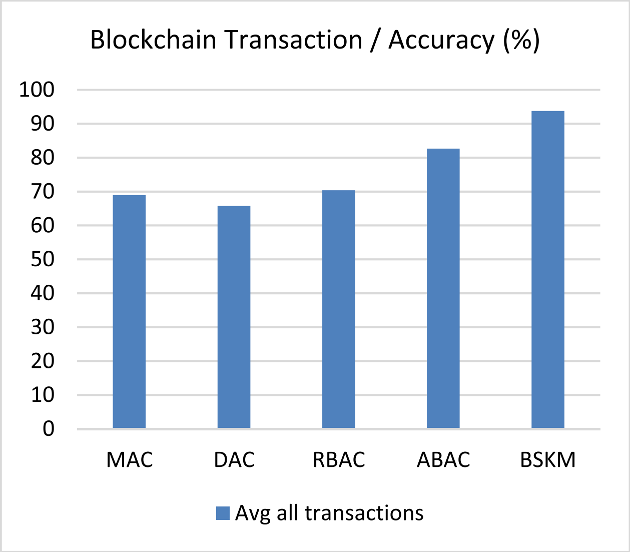


FIGURE 14. Traditional security models and BSKM accuracy rate for bank data in Hyperledger platform.

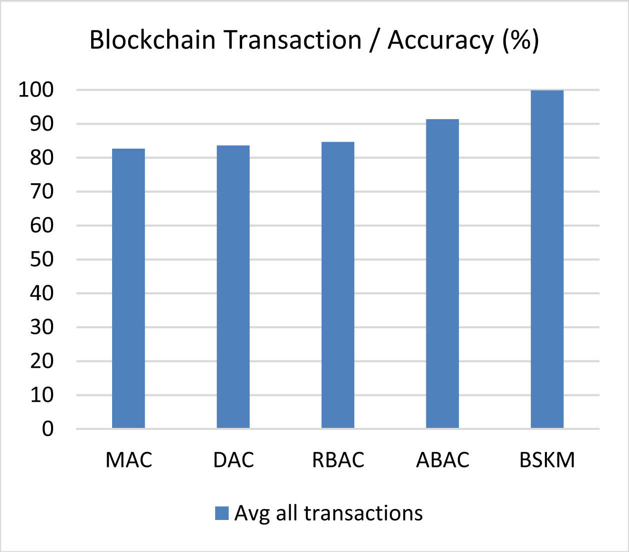


FIGURE 15. Traditional security models and BSKM accuracy rate for financial data in Hyperledger platform.

TABLE 6. Traditional security model and Bskm time comparision for financial data (sec).

Transaction/ Time (Sec)	MAC	DAC	RBAC	ABAC	BSKM
Read	10.92	9.00	14.54	7.93	4.78
Write	15.08	13.54	17.37	12.75	8.24
Update	12.34	11.20	18.99	9.90	6.16
Delete	9.22	8.27	11.74	7.24	2.30

TABLE 7. Traditional security model and Bskm time comparision for financial data (sec).

Transaction/ Time (Sec)	MAC	DAC	RBAC	ABAC	BSKM
Read	13.07	11.92	16.64	8.74	5.24
Write	18.64	15.72	20.98	14.25	7.55
Update	14.08	12.71	18.44	10.48	6.20
Delete	10.83	9.72	12.45	8.92	3.18

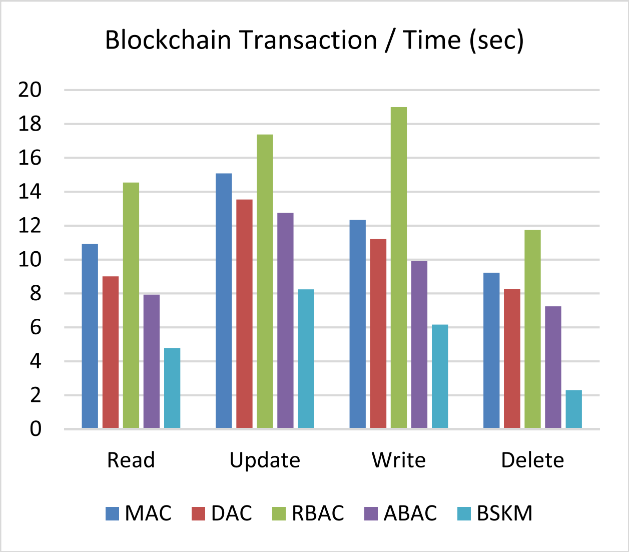


FIGURE 16. Traditional security models and BSKM time comparision for bank data (sec).

selections from these data sets. The success of the proposed model is clearly shown in Figure 16 and in Figure 17 for the bank and the financial data, respectively. In terms of time, it was seen that operations were performed on the data in less time with the proposed model. Writing and updating operations took longer in both methods in terms of time compared to other operations. This is because performing writing and reading operations on the object takes more time. Also, when compared in terms of time, the proposed model gave very successful results for all operations performed on the object. Looked at it in terms of time, it was seen that transactions were performed on the data in less time with the proposed model. Writing and updating transactions take more time than other transactions. This is because it takes more time to write and update the object.

VIII. PERFORMANCE EVALUATION

The developed model being proposed in this study has been applied on a dataset obtained from real life bank data. The performance of the proposed model has been compared with the performances of traditional access control models. When the results obtained were compared, it was observed that object access levels were presented more consistently and quickly with the proposed model. The accuracy results obtained by using the path shortening algorithm with the BSKM model and the traditional recommended access

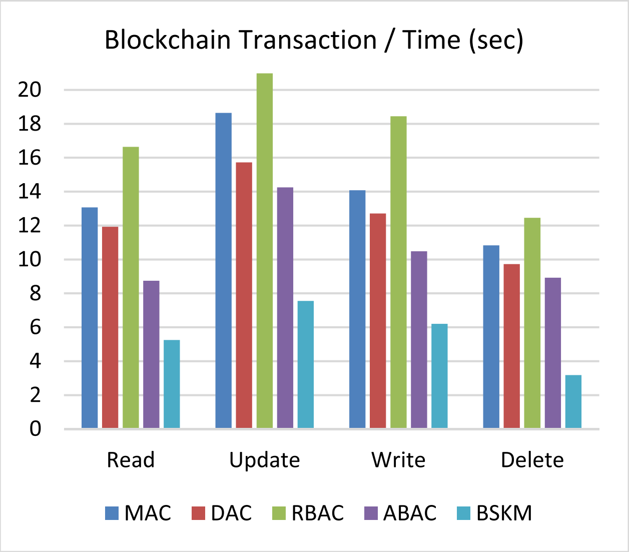


FIGURE 17. Traditional security models and BSKM time comparision for financial data (sec).

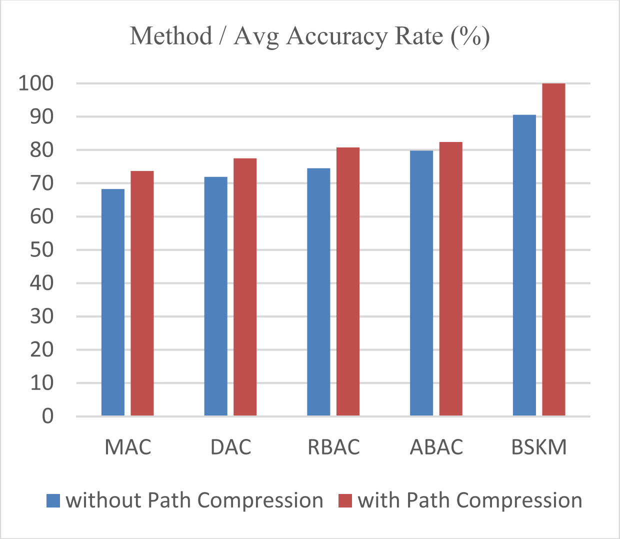


FIGURE 18. Path Compression Evaluation Performance (%).

control models are shown in Table 8. Looking at Figure 18, the path compression algorithm gave more successful results for all methods. When evaluated in terms of performance, the success rate without using the path compression algorithm was lower.

Table 9 shows the comparison of the proposed BSKM model and traditional access control models in terms of the work carried out using the path compression algorithm. The path compression algorithm suggested in Figure 19 gave faster results with all methods. As a result, the proposed path compression algorithm is both faster and more successful. Simulation of the big data was carried out through an experimental study, and in addition to the evaluation of the algorithm, its effectiveness was also demonstrated by the experimental study.

TABLE 8. Path compression performance evaluation.

Method/ Avg Accuracy Rate (%)	without Path Compression	with Path Compression
MAC	68.24	73.66
DAC	71.88	77.48
RBAC	74.49	80.75
ABAC	79.80	82.37
BSKM	90.57	99.99

TABLE 9. Path compression performance evaluation time (sec).

Method/ Avg Time (sec)	without Path Compression	with Path Compression
MAC	12.00	9.64
DAC	9.32	5.35
RBAC	16.87	10.94
ABAC	10.66	6.48
BSKM	6.35	2.07

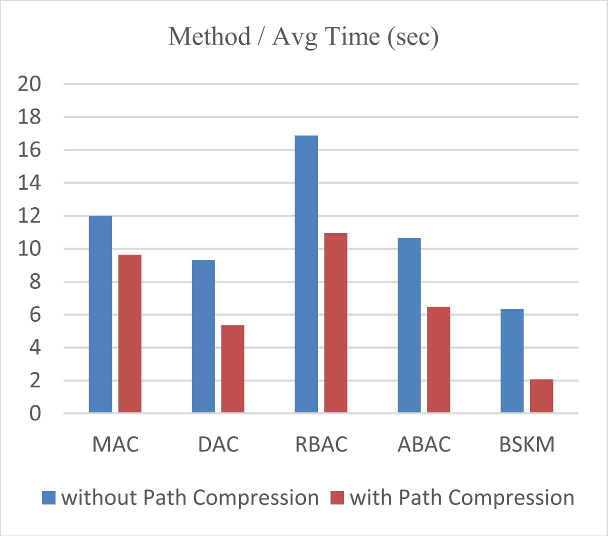


FIGURE 19. Path Compression Evaluation Time (sec).

IX. CONCLUSION AND DISCUSSION

In this study, BSKM was introduced to ensure big data security. In the proposed model, both authorized and unauthorized operations between users were carried out. Also, in the proposed model, there was separate authorization or access control for each blockchain transaction, such as reading, writing, updating and deleting. Access control and authorization operations were performed using special keys. Unlike previous studies, data security was ensured for all operations performed on the big data. Users could take back the authority that they gave at any time, or they can give authority to the user they want. Challenges that occurred during the implementation of security policies on big data were overcome.

In this current study, the problem of data security for big data was addressed. In particular, BSKM related to data flow control was introduced, and examples of applications for its use were shown. The proposed model consisted of users, objects and special keys. In addition, data object flows were modeled in a blockchain-based doubly linked list structure. In the doubly linked list structure, each transaction occurring in the blockchain was determined according to the shape of the arrow. The direction of the arrow only indicates from which user to which other user the transaction will be made. Unlike traditional methods from other studies that were selected, BSKM performed all transactions on data in a fast and reliable way. The results of our study were shown applied to bank data, which was a real data set, by comparing it with traditional methods. The results of the proposed BSKM for all blockchain transactions performed on the data were also shown by the experimental study. It delivered more successful results, especially in reading and deleting operations. In addition, by combining the path compression algorithm with BSKM, both the success and speed of the proposed model with a hybrid model were increased. The proposed model was also compared with the traditional methods used in previous studies in terms of time, and it was seen that it performed all blockchain transaction operations in a shorter time (read, write, update and delete). In this way, data confidentiality, integrity and consistency was ensured. This also shows that the proposed model is flexible.

In this current study, both granting and receiving permissions were performed between users. In this work, no separate authorization or access control was made for each operation, such as reading, writing, updating and deleting. Access control, authorization and revocation of access rights are done through private keys in the blockchain. Unlike previous studies, data security is provided by BSKM for all operations on big data. Users can withdraw the authorization they have given at any time or give the authorization to the users they want. This authorization is done through policies made up of private keys. This shows that the proposed model is flexible.

Preventing information disclosure was performed by monitoring the access of malicious users to data. The results for all operations performed on the data of the proposed BSKM model are also shown in the experimental study. It gave more successful results, especially for reading and deleting operations. It was also compared in terms of time with methods used in previous studies, and it performed operations in a shorter time.

In this study, a novel blockchain-based big data model was introduced to provide advanced security and privacy properties to big data-based bank and financial data. Use of the blockchain technique for big data-based models is not straightforward. Therefore the goal was to eliminate many challenges and improve the security of two real datasets. The proposed model provided reliable data communication over the network and storage of the big data for differential transactions. The concept of BSKM, which provides confidentiality and privacy, was introduced.

For future studies, a prototype application will be created that shows the work of the BSKM, and the model will be enriched by using multi function special keys, which take into account the group of users as well. The main future direction for this work is to implement this system into a testable framework to provide some real world security and efficiency guarantees beyond what has already been established for all the individual components used. It is also hoped to find partners to help bring some of the novel ideas mentioned in this study to become available to the general public.

REFERENCES

[1] E. S. Pagnotta, "Decentralizing money: Bitcoin prices and blockchain security," *Rev. Financial Stud.*, vol. 35, no. 2, pp. 866–907, Jan. 2022, doi: 10.1093/rfs/hhaa149.

[2] X. Ma, C. Wang, and X. Chen, "Trusted data sharing with flexible access control based on blockchain," *Comput. Standards Interfaces*, vol. 78, Oct. 2021, Art. no. 103543, doi: 10.1016/j.csi.2021.103543.

[3] H. Liu, D. Han, and D. Li, "Fabric-IoT: A blockchain-based access control system in IoT," *IEEE Access*, vol. 8, pp. 18207–18218, 2020, doi: 10.1109/ACCESS.2020.2968492.

[4] Q. Lyu, Y. Qi, X. Zhang, and H. Liu, "SBAC: A secure blockchain-based access control framework forinformation-centric networking," *J. Netw. Comput. Appl.*, vol. 149, Jan. 2020, Art. no. 102444, doi: 10.1016/j.jnca.2019.102444.

[5] H. Ghayvat, S. Pandya, P. Bhattacharya, M. Zuhair, M. Rashid, S. Hakak, and K. Dev, "CP-BDHCA: Blockchain-based confidentiality-privacy preserving big data scheme for healthcare clouds and applications," *IEEE J. Biomed. Health Informat.*, vol. 26, no. 5, pp. 1937–1948, May 2022, doi: 10.1109/JBHI.2021.3097237.

[6] L. Tan, N. Shi, C. Yang, and K. Yu, "A blockchain-based access control framework for cyber-physical-social system big data," *IEEE Access*, vol. 8, pp. 77215–77226, 2020, doi: 10.1109/ACCESS.2020.2988951.

[7] U. U. Ugobame, K. A. Schneider, S. H. Kassani, and R. Deters, "Blockchain access control ecosystem for big data security," in *Proc. IEEE Confs Internet Things, Green Comput. Commun., Cyber, Phys. Social Comput., Smart Data, Blockchain, Comput. Inf. Technol., Congr. Cybermatics*, Jul. 2018, pp. 1373–1378.

[8] J. Li, J. Wu, G. Jiang, and T. Srikanthan, "Blockchain-based public auditing for big data in cloud storage," *Inf. Process. Manag.*, vol. 57, no. 6, Nov. 2020, Art. no. 102382, doi: 10.1016/j.ipm.2020.102382.

[9] S. Biswas, K. Sharif, F. Li, I. Alam, and S. Mohanty, "DAAC: Digital asset access control in a unified blockchain based E-health system," *IEEE Trans. Big Data*, vol. 8, no. 5, pp. 1273–1287, Oct. 2022, doi: 10.1109/TBDATA.2020.3037914.

[10] O. Mounnan, A. A. E. Kalam, and L. El Haourani, "Decentralized access control infrastructure using blockchain for big data," in *Proc. IEEE/ACS 16th Int. Conf. Comput. Syst. Appl. (AICCSA)*, Abu Dhabi, United Arab Emirates, Nov. 2019, pp. 1–8, doi: 10.1109/AICCSA47632.2019.9035221.

[11] J. Chen, Z. Lv, and H. Song, "Design of personnel big data management system based on blockchain," *Future Gener. Comput. Syst.*, vol. 101, pp. 1122–1129, Dec. 2019, doi: 10.1016/j.future.2019.07.037.

[12] Y. Zhang, B. Li, B. Liu, J. Wu, Y. Wang, and X. Yang, "An attribute-based collaborative access control scheme using blockchain for IoT devices," *Electronics*, vol. 9, no. 2, p. 285, 2020, doi: 10.3390/electronics9020285.

[13] T. Sultana, A. Almogren, M. Akbar, M. Zuair, I. Ullah, and N. Javaid, "Data sharing system integrating access control mechanism using blockchain-based smart contracts for IoT devices," *Appl. Sci.*, vol. 10, no. 2, p. 488, 2020, doi: 10.3390/app10020488.

[14] B. S. Egala, A. K. Pradhan, V. Badarla, and S. P. Mohanty, "Fortified-chain: A blockchain-based framework for security and privacy-assured internet of medical things with effective access control," *IEEE Internet Things J.*, vol. 8, no. 14, pp. 11717–11731, Jul. 2021, doi: 10.1109/JIOT.2021.3058946.

[15] W. Zhou and J. Jin, "A blockchain-based access control framework for secured data sharing in industrial internet," in *Proc. 8th Int. Conf. Adv. Cloud Big Data (CBD)*, Taiyuan, China, Dec. 2020, pp. 231–236, doi: 10.1109/CBD51900.2020.00049.

[16] D. Lopez and B. Farooq, "A blockchain framework for smart mobility," in *Proc. IEEE Int. Smart Cities Conf. (ISC)*, Kansas City, MO, USA, Sep. 2018, pp. 1–7, doi: 10.1109/ISC2.2018.8656927.

[17] C. Yang, L. Tan, N. Shi, B. Xu, Y. Cao, and K. Yu, "AuthPrivacy-Chain: A blockchain-based access control framework with privacy protection in cloud," *IEEE Access*, vol. 8, pp. 70604–70615, 2020, doi: 10.1109/ACCESS.2020.2985762.

[18] N. Deepa, Q.-V. Pham, D. C. Nguyen, S. Bhattacharya, B. Prabadevi, T. R. Gadekallu, P. K. R. Maddikunta, F. Fang, and P. N. Pathirana, "A survey on blockchain for big data: Approaches, opportunities, and future directions," *Future Gener. Comput. Syst.*, vol. 131, pp. 209–226, Jun. 2022, doi: 10.1016/j.future.2022.01.017.

[19] N. J. Ogbuke, Y. Y. Yusuf, K. Dharma, and B. A. Mercangoz, "Big data supply chain analytics: Ethical, privacy and security challenges posed to business, industries and society," *Prod. Planning Control*, vol. 33, nos. 2–3, pp. 123–137, Feb. 2022, doi: 10.1080/09537287.2020.1810764.

[20] S. Venkatraman and R. Venkatraman, "Big data security challenges and strategies," *AIMS Math.*, vol. 4, no. 3, pp. 860–879, 2019, doi: 10.3934/math.2019.3.860.

[21] R. Toshniwal, K. G. Dastidar, and A. Nath, "Big data security issues and challenges," *Int. J. Innov. Res. Adv. Eng. (IJIRAE)*, vol. 2, no. 2, pp. 15–20, 2015.

[22] S. L. Garfinkel, *De-Identification of Personal Information*. Gaithersburg, MD, USA: National Institute Standards Technology, 2015.

[23] N. Johnson, K. Dharma, and B. Mercangoz, "Big data supply chain analytics: Ethical, privacy and security challenges posed to business, industries and society," *Prod. Planning Control*, 2020.

[24] J. R. Saura, D. Ribeiro-Soriano, and D. Palacios-Marqués, "Setting privacy 'by Defaul' in social IoT: Theorizing the challenges and directions in big data research," *Big Data Res.*, vol. 25, Jul. 2021, Art. no. 100245, doi: 10.1016/j.bdr.2021.100245.

[25] A. Cuzzocrea, C. K. Leung, A. M. Olawoyin, and E. Fadda, "Supporting privacy-preserving big data analytics on temporal open big data," *Proc. Comput. Sci.*, vol. 198, pp. 112–121, Jan. 2022, doi: 10.1016/j.procs.2021.12.217.

[26] H. Rafik, A. Hassan, A. Ali, M. B. Bashir, S. M. Alqhtani, T. M. Tawfeeg, and A. Yousif, "Towards secure big data analysis via fully homomorphic encryption algorithms," *Entropy*, vol. 24, no. 4, pp. 519–520, 2022, doi: 10.3390/e24040519.

[27] A. D. Dwivedi, "Security analysis of lightweight IoT cipher: Chaskey," *Cryptography*, vol. 4, no. 3, p. 22, Aug. 2020.

[28] G. Srivastava, J. Crichigno, and S. Dhar, "A light and secure healthcare blockchain for IoT medical devices," in *Proc. IEEE Can. Conf. Electr. Comput. Eng. (CCECE)*, May 2019, pp. 1–5.

[29] C. Bakir, "A single-label model to ensure data consistency in information security," *Sci. Program.*, vol. 2021, pp. 1–8, Apr. 2021, doi: 10.1155/2021/9913645.

CIGDEM BAKIR received the B.S. degree in computer engineering from the University of Sakarya, in 2010, and the M.S. and Ph.D. degrees in computer engineering from Yildiz Technical University, Istanbul. She is currently pursuing the doctorate degree in computer science with the University of Yildiz Technical, Istanbul. She was a Research Assistant at Yildiz Technical University and Igdir University. She was an Instructor at Erzincan Binali Yildirim, from 2020 to 2021. She has been an Assistant Professor with the Software Engineering Department, Dumlupinar University, since 2021. Her research interests include information security, distributed database, big data, blockchain technology, cloud computing, and computer networks.